

ROBÓTICA MÉDICA

notas prácticas para el aprendizaje de la robótica en
bioingeniería

Autores:

Consorcio OPENSURG

*Proyecto iberoamericano para la docencia e investigación en
robótica médica utilizando recursos de código abierto*

Acción CYTED 509AC0372

Editor:

CYTED

Elche, 2013

Prefacio

En una sociedad plenamente comprometida con la tecnología el esclarecimiento de sus potencialidades y la demostración de su utilidad para resolver problemas concretos de la población resultan las mejores herramientas para que la sociedad perciba su utilidad y pueda distinguir entre las quimeras de la sociedad de consumo y los reales avances de la ciencia para el mejoramiento de la calidad de vida. Este libro presenta la robótica médica tomando como base las experiencias reales de los autores, analizando paso a paso los conceptos y los mecanismos que permiten la formación de profesionales e investigadores en una disciplina de gran complejidad y amplitud temática, como es la robótica médica en sus múltiples aspectos.

El CYTED ve con satisfacción que el esfuerzo de Universidades e investigadores que reunidos bajo la denominación de Consorcio OPENSURG, han desarrollaron un proyecto de alcance iberoamericano que fortaleció plenamente a la docencia e investigación utilizando recursos de código abierto.

Sin duda este texto refleja plenamente los resultados esperados de la intervención del CYTED en el área de las Tecnologías de la Información y la Comunicación esto es: el esfuerzo de los mejores de cada temática, con aplicación inmediata en la región, basándose en la formación de recursos humanos y con una clara orientación al mejoramiento de nuestra sociedad y de su calidad de vida.

No escapa a nosotros que detrás de un trabajo de esta envergadura hay mucho esfuerzo de los autores y voluntad de acción, de llegar a resultados y de compartir sin egoísmos conocimientos y experiencias, a ellos saludamos con orgullo frente a este importante trabajo.

Ing. Daniel Lupi
Gestor Área TICs
CYTED

Dr. Óscar Pastor López
Vocal Área TICs
CYTED

Autores

Los miembros del consorcio OpenSurg que han contribuido a la creación de este material han sido:

- **Dr. José María Sabater-Navarro** *Universidad Miguel Hernández de Elche y coordinador de la acción OpenSurg - Cytel 509ac0372*
- Dr. Nicolás García Aracil *Universidad Miguel Hernández de Elche*
- Dr. Carlos Pérez Vidal *Universidad Miguel Hernández de Elche*
- Dr. José María Azorin Poveda *Universidad Miguel Hernández de Elche*
- Ing. Luis Daniel López Lledó *Universidad Miguel Hernández de Elche*
- Ing. Ricardo Morales Vidal *Universidad Miguel Hernández de Elche*
- Ing. Francisco Javier Badesa Clemente *Universidad Miguel Hernández de Elche*
- Dr. Eduardo Iañez *Universidad Miguel Hernández de Elche*
- Ing. Andrés Úbeda *Universidad Miguel Hernández de Elche*
- Dr. Roque Saltarén Pazmiño *Universidad Politécnica de Madrid*
- Mag. Elena Muñoz España *Universidad del Cauca*
- Mag. Victor Hugo Mosquera Leyton *Universidad del Cauca*
- Dr. Carlos Felipe Rengifo Rodas *Universidad del Cauca*
- Dr. Carlos Alberto Gaviria Lopez *Universidad del Cauca*
- Dr. Oscar Andrés Vivas Alban *Universidad del Cauca*
- Dr. Luis Eduardo Rodriguez-Cheu *Escuela Colombiana de Ingeniería J. Garavito*
- Dr. Juan Manuel Ibarra Zannatha *Cinvestav*
- Ing. Julio Politti *Universidad Nacional de Tucumán*
- Dr. Fernando Farfán *Universidad Nacional de Tucumán*
- Ing. Lisandro Puglisi *Universidad Nacional de Tucumán*
- Dr. Angel Ernesto Rubio Gonzalez *Universidad Central de las Villas*
- Ing. Fidel González Vazquez *Universidad Central de las Villas*
- Ing. José Ignacio Guzmán Montoto *Universidad Central de las Villas*

Contenido del libro

El libro que empieza a leer pretende resumir la experiencia de los autores en su andadura por el mundo de la docencia en robótica médica; combinando los aspectos teóricos y prácticos necesarios para formar a un técnico especialista en bioingeniería. El libro está compuesto principalmente por las notas de clase que los autores utilizan en sus cursos de Grado y Máster en las diferentes Universidades del consorcio [OpenSurg](#), y es fruto del trabajo realizado en el desarrollo de la acción [509AC0372](#) del PROGRAMA IBEROAMERICANO DE CIENCIA Y TECNOLOGIA PARA EL DESARROLLO - [CYTED](#).

El objetivo de este material es servir como libro base para los diferentes cursos que se imparten en las universidades y escuelas técnicas. Para ello se ha tratado de dotar de un **aspecto teórico**, pero también de un importante **aspecto práctico**, que capacite al estudiante a desarrollar futuros proyectos de robótica médica fundamentados en lo aquí aprendido.

El libro se ha escrito de forma que pueda ser utilizado como libro de apoyo tanto en un curso básico de robótica médica como en un curso más avanzado perteneciente a algún máster relacionado con bioingeniería. Para ello, todos los capítulos cuentan con una teoría básica y con unos ejercicios de ampliación donde se muestran ejemplos reales de estudios realizados por los autores. Todos los capítulos tienen un apartado, denominado **ROBOTICA MEDICA APLICADA**, donde se incluyen estos ejercicios prácticos. Para la comprensión y seguimiento de este apartado se presupone un conocimiento básico (y en algunos casos avanzado) de programación y del programa Matlab[®]. Concretamente, en el Bloque I se requieren conocimientos de Matlab[®] y el programa de imagen médica [3DSlicer](#), mientras que para trabajar el Bloque II son necesarios conocimientos de programación avanzada orientada a objetos. En caso contrario, se recomienda consultar la bibliografía indicada.

Además, el libro contiene referencias a los trabajos desarrollados durante los años de la acción [OpenSurg](#). De esta forma, el lector puede aplicar los conocimientos adquiridos utilizando para ello las herramientas de software libre disponibles en la web.

Siguiendo esta filosofía, el libro se ha distribuido en dos bloques:

BLOQUE I En este bloque, que comprende los capítulos **1** a **7**, presentan los aspectos teóricos necesarios para la realización de los ejercicios propuestos como *básicos* para un curso de Grado.

Los capítulos **1** y **2** presentan una revisión del estado del arte de la robótica médica y de las herramientas matemáticas para el análisis de robots que serán utilizadas en el libro.

Los capítulos **3** y **4** también presentan conocimientos transversales que se utilizan en las diferentes áreas de la robótica médica. El capítulo **3** está dedicado al tratamiento de las imágenes médicas y a las diferentes tecnologías existentes, y el capítulo **4** al tema de las señales biomédicas.

Los capítulos 5, 6 y 7 tratan de forma particular cada uno aspectos relacionados con las tres áreas de trabajo en que hemos agrupado a los robots médicos: robótica quirúrgica, robótica de rehabilitación y robótica asistencial.

En todos los capítulos, el apartado ROBOTICA MEDICA APLICADA muestra algunos de los desarrollos llevados a cabo por los autores, y permite una ampliación de los conceptos aprendidos.

BLOQUE II En este bloque se presentan los resultados del trabajo realizado dentro de la acción OpenSurg. Se muestran resumidos los guiones y tutoriales desarrollados durante los años del proyecto. Las versiones más completas de estos tutoriales y ejercicios y más información sobre este bloque puede encontrarse en la página web del proyecto (<http://opensurg.umh.es>) y en el cd adjunto al libro.

El capítulo 8 presenta los guiones básicos desarrollados durante la primera anualidad de OpenSurg, y que capacitan al lector para trabajar con las herramientas de software libre sobre las que se construyen los desarrollos posteriores.

El capítulo 9 se centra en una breve descripción de los simuladores médicos que se encuentran en el cd adjunto.

El capítulo 10 presenta algunas aplicaciones más complejas y que han sido desarrolladas por miembros del consorcio OpenSurg utilizando las herramientas de software libre anteriores.

Adicionalmente al texto, se presenta una distribución live OpenSurg con un sistema operativo basado en la distribución linux Ubuntu www.ubuntu.com y con los ejercicios y documentación necesarios para la docencia de la robótica médica.

Este texto y sus futuras versiones se encuentran disponibles en el portal web roboticamedica.umh.es.

Los autores quieren agradecer enormemente a CYTED (www.cyted.org) la financiación y confianza puesta en el consorcio para haber podido desarrollar este trabajo.

Índice general

Índice de Figuras	VII
Índice de Tablas	XIII
I Robótica médica	1
1. Introducción a la robótica médica	3
1.1. Introducción	4
1.2. Hitos y retos	5
1.2.1. Hitos de la robótica médica	5
1.2.2. Retos de la robótica médica	9
1.3. Áreas de aplicación de la robótica médica	10
1.3.1. Robótica quirúrgica	11
1.3.2. Robótica de rehabilitación	13
1.3.3. Robótica asistencial	14
1.4. Tecnología para la robótica médica	14
1.4.1. Actuadores para interacción hombre-máquina. (<i>Soft robotics</i>)	15
1.4.1.1. Disposición antagonista de dos resortes no lineales	17
1.4.1.2. Control de la rigidez estructural	21
1.4.1.3. Control mecánico de la rigidez	22
1.4.2. Sensores para interacción hombre-máquina.	22
1.4.3. Sistemas de control en la interacción hombre-máquina.	25
1.4.3.1. Hybrid Position/Force Control	25
1.4.3.2. Compliance Control	31
1.5. ROBOTICA MEDICA APLICADA	35
1.5.1. Ejemplo de cálculo del espacio de trabajo de un robot de rehabilitación con Blender [©]	35
1.5.2. Ejemplo de control de impedancia con Matlab [®]	37
1.6. Resumen del capítulo	40
Bibliografía para ampliar	40

2. Fundamentos de robótica aplicada	43
2.1. Introducción	44
2.2. Herramientas para el análisis de robots	44
2.2.1. Localización de un cuerpo rígido en \mathbb{R}^3	44
2.2.2. Representación de la posición en \mathbb{R}^3	44
2.2.3. Representación de la orientación en \mathbb{R}^3	45
2.2.3.1. Matriz de rotación	46
2.2.3.2. Obtención de la Matriz de rotación como matriz de cose- nos directores	46
2.2.3.3. Obtención de la Matriz de rotación como par de rotación. Fórmula de Rodrigues	49
2.2.3.4. Composición de matrices de rotación	51
2.2.3.5. Propiedades de la matriz de rotación	52
2.2.3.6. Otras representaciones: ángulos de Euler, roll-pitch-yaw	52
2.2.4. Representación conjunta de la localización de un cuerpo en \mathbb{R}^3	54
2.2.4.1. Coordenadas homogéneas	54
2.2.4.2. Matrices de transformación homogénea	55
2.2.5. Convenio de Denavit-Hartenberg	57
2.3. Cinemática de robots	60
2.3.1. Cinemática directa	61
2.3.2. Cinemática inversa	62
2.3.3. Cinemática diferencial	62
2.3.3.1. Análisis de singularidades	63
2.3.3.2. Mapeado de fuerzas	63
2.4. Dinámica de robots	64
2.5. ROBOTICA MEDICA APLICADA	66
2.5.1. Ejemplo de análisis de la cinemática un robot serial de 6 GDL	66
2.5.2. Ejemplo de análisis de la dinámica de un robot 6UPS con SimMe- chanics	69
2.5.2.1. Modelo Dinámico en SimMechanics	71
2.5.2.2. Dinámica Inversa del Mecanismo 6UPS	72
2.6. Resumen del capítulo	74
2.7. Ejercicios	76
Bibliografía para ampliar	77
3. Imagen médica	79
3.1. Introducción a la imagen médica	80
3.1.1. Principales hitos en la imagen médica	81
3.2. Uso de imágenes médicas	82
3.3. Tipos de imágenes médicas	83
3.3.1. Imágenes de radiografía	83
3.3.2. Imágenes por ultrasonidos	85
3.3.3. Imágenes de Tomografía Axial Computerizada	86
3.3.3.1. Principio de funcionamiento	86
3.3.4. Imagen nuclear	88
3.3.4.1. SPECT	89

3.3.4.2. PET	89
3.3.5. Resonancia magnética	90
3.3.5.1. Resonancia magnética funcional	91
3.3.5.2. Imagen de difusión	93
3.4. Análisis de imagen médica	94
3.5. ROBOTICA MEDICA APLICADA	94
3.5.1. Ejemplo de lectura de imágenes DICOM con Matlab®	94
3.5.2. Ejemplo de segmentación y reconstrucción 3D de la arteria aorta	100
3.6. Resumen del capítulo	105
3.7. Ejercicios	106
Bibliografía para ampliar	106
4. Señales biomédicas	109
4.1. Introducción	110
4.2. Clasificación	111
4.2.1. Electromiograma. (EMG)	111
4.2.2. Electroencefalograma. (EEG)	112
4.2.3. Electrocardiograma (ECG)	115
4.2.4. Electrooculograma (EOG)	118
4.2.5. Electrorretinograma (ERG)	119
4.3. Análisis de señales fisiológicas	121
4.3.1. Conceptos generales de filtrado de señales	121
4.4. Hardware y Software para la adquisición y procesado de señales fisiológicas	122
4.5. ROBOTICA MEDICA APLICADA	125
4.5.1. Procesamiento de una señal EOG usando Matlab	125
4.5.2. Filtrado de señal EMG usando Matlab	128
4.6. Resumen del capítulo	130
Bibliografía para ampliar	130
5. Robótica Quirúrgica	131
5.1. Introducción	132
5.2. El proceso global de cirugía asistida por computador (CAS)	133
5.2.1. El problema de la incertidumbre	135
5.2.2. Limitaciones de la robótica quirúrgica	135
5.3. Clasificación de dispositivos robóticos para cirugía	136
5.4. Ejemplos de dispositivos robóticos para cirugía	138
5.4.1. Sistema de radiocirugía con fuente externa. Cyberknife.	138
5.4.2. Sistema HIFU con fuente de energía interna	140
5.4.3. Sistema de telecirugía. DaVinci	141
5.4.4. Sistema de cirugía ortopédica. Robodoc	143
5.4.5. Sistema de robot montado en paciente. SpineAssist	145
5.4.6. Sistema de robots colaborativos	147
5.5. Simuladores quirúrgicos	148
5.5.1. Requerimientos de los simuladores quirúrgicos	148
5.5.2. Simuladores comerciales y de investigación	150
5.6. Navegación quirúrgica	151

5.6.1.	Tecnologías de los sistemas de navegación	152
5.6.1.1.	Ópticos	152
5.6.1.2.	Electromagnéticos	154
5.6.2.	Tipos de sistemas de navegación	154
5.6.2.1.	Basados en información CT	154
5.6.2.2.	Basados en información de Fluoroscopia	156
5.6.2.3.	Imageless - Sistemas sin imagen	157
5.7.	ROBOTICA MEDICA APLICADA	157
5.7.1.	Análisis del espacio de trabajo de una muñeca paralela de uso quirúrgico.	157
5.7.1.1.	Metodología de análisis	159
5.7.1.2.	Análisis del espacio de trabajo con teoría de screws	160
5.8.	Resumen del capítulo	164
5.9.	Ejercicios	165
	Bibliografía para ampliar	165
6.	Robótica de rehabilitación	167
6.1.	Introducción a la rehabilitación robótica	168
6.1.1.	Revisión de las técnicas de rehabilitación en ACV	169
6.2.	Sistemas de rehabilitación robótica	170
6.2.1.	Sistemas de rehabilitación de miembro superior	170
6.2.2.	Sistemas de rehabilitación de miembro inferior	178
6.2.3.	Sistemas de rehabilitación de extremidades distales; mano y pie	185
6.3.	La realidad virtual como ayuda a la rehabilitación física	191
6.4.	ROBOTICA MEDICA APLICADA	193
6.4.1.	Análisis de un dispositivo de rehabilitación de miembro superior (PuParm)	193
6.4.2.	Análisis del gait en un dispositivo de rehabilitación de miembro inferior (GBO)	194
6.4.2.1.	Simulación de un sistema de rehabilitación para miembro inferior	196
6.5.	Resumen del capítulo	199
	Bibliografía para ampliar	199
7.	Robótica asistencial	201
7.1.	Introducción a la robótica asistencial	202
7.2.	Clasificación de robots asistenciales para personas con movilidad reducida	202
7.2.1.	Sistemas fijos dedicados	202
7.2.2.	Manipuladores montados sobre sillas de ruedas	204
7.2.3.	Sistemas móviles autónomos	205
7.2.4.	Sillas de rueda con guiado autónomo	205
7.2.5.	Robots asistenciales para la navegación	206
7.3.	Prótesis y órtesis mecatrónicas	207
7.3.1.	Introducción	207
7.3.2.	Justificación de la robotización	207
7.3.3.	Dificultades y desafíos	208

7.3.4.	Clasificación de las diferentes actividades	209
7.3.4.1.	Prótesis no robotizadas	210
7.3.4.2.	Prótesis robotizadas	211
7.4.	Dispositivos interfaz hombre-máquina para comunicación de pacientes dis- capacitados	215
7.4.1.	Interfases Cerebro Computadora (Brain Computer Interface - BCI)	216
7.4.2.	Interfases basadas en la interacción mioeléctrica	217
7.4.3.	Interfases basadas en señales de electrooculografía	218
7.5.	ROBOTICA MEDICA APLICADA	219
7.5.1.	Evaluación de los modelos para una prótesis de mano	219
7.5.2.	Diseño de una prótesis de mano	223
7.5.2.1.	Aspectos preliminares	223
7.5.2.2.	Metodología	223
7.5.2.3.	Selección de la arquitectura de la mano	224
7.5.2.4.	Parámetros geométricos	224
7.5.2.5.	Modelo geométrico directo de la mano	226
7.5.2.6.	Modelo dinámico de la mano	227
	Bibliografía para ampliar	229

II OPENSURG 231

8. OPENSURG - guiones básicos 233

8.1.	Introducción al proyecto OpenSurg	234
8.2.	Instalación básica de Ubuntu 8.10 Intrepid Ibex para OpenSurg	234
8.2.1.	Prerrequisitos	234
8.2.2.	Instalación	234
8.3.	Tutorial básico de operaciones con ITK	237
8.3.1.	Prerrequisitos	237
8.3.2.	Ejercicio 1. Operaciones básicas	237
8.3.3.	Ejercicio 2. Filtrado de imágenes	238
8.3.4.	Ejercicio 3. Detección de bordes	239
8.4.	Tutorial básico de uso de VTK	240
8.4.1.	Ejercicio 1. Hola mundo. El cono de VTK	240
8.4.2.	Ejercicio 2. Ejemplo básico de reconstrucción 3D con imagen médica	241
8.4.3.	Ejercicio 3. Ejemplo de integración VTK-Qt	241
8.5.	Tutorial básico de uso de IgsTK	242
8.6.	Ejercicio para la detección y medición de calcificaciones en mamografía 2D	244
8.6.1.	Indicaciones para la solución	245
	Bibliografía para ampliar	246

9. OPENSURG - Simuladores médicos 249

9.1.	Introducción	249
9.2.	Herramientas para la simulación	250
9.2.1.	OGRE	250
9.2.1.1.	Gestión de escenario	251

9.2.1.2.	Gestión de recursos	252
9.2.1.3.	Renderización	252
9.2.2.	PhysX de NVIDIA	253
9.2.3.	NxOgre	255
9.2.3.1.	Critter	255
9.2.3.2.	Flour	255
9.2.4.	Blender	256
9.3.	Entrenador de técnicas MIS	256
9.4.	Simulador de un robot para tareas de colicestomía	257
9.4.1.	Descripción de las trayectorias de colicestomía laparoscópica	257
9.4.2.	Estructura funcional del robot diseñado para realizar operaciones de laparoscopia	257
9.4.3.	Trayectorias para simulación	259
9.4.3.1.	Interfaz Gráfica de Usuario (GUI) y Ambiente 3D	261
9.5.	Simulador de traumatología. Operación de reemplazo total de rodilla	263
10.	OPENSURG - Aplicaciones	267
10.1.	Introducción	267
10.2.	Sistema de bajo coste para rehabilitación de la mano	268
10.3.	Guante haptico sensorizado	270
10.4.	Robot de 2 grados de libertad para el fresado maxilofacial	271

Índice de figuras

1.1. Robot industrial vs. robot médico	5
1.2. Algunos hitos de la robotica médica	6
1.3. Robot neumático para rehabilitación de miembro superior	8
1.4. Escenarios futuros vistos en ciencia ficción	11
1.5. SpineAssist - Robot paralelo para operar en la colocación de fijaciones mecánicas de prótesis en la columna vertebral. Cortesía de Mazor Inc.	12
1.6. Imhotep - Muñeca esférica paralela para operaciones MIS. Video youtube de demostración	12
1.7. Robot 2UPS1U para cirugía maxilofacial.	13
1.8. Robot HAL exo-esquelético. Extraído de la webpage de Cyberdyne.video	14
1.9. Robot asistente desarrollado en la Universidad Carlos III de Madrid. Cortesía del Robotics Lab la UC3M	15
1.10. Ejemplos CAD de <i>Series Elastic Actuator</i> con actuadores hidráulicos y eléctrico. Extraído de [34]	17
1.11. Configuración antagonista de biceps y triceps en el brazo humano. (Extraído de [34])	18
1.12. Explicación de la necesidad de resortes no lineales. (Extraído de [34])	18
1.13. Esquema de sistema antagonista con resorte cuadrático. (Extraído de [34])	20
1.14. Configuraciones antagonistas con músculos neumáticos. (Extraído de [34])	21
1.15. Variación del momento de inercia por rotación de la lámina. (Extraído de [34])	22
1.16. Bases del control mecánico de la rigidez. (Extraído de [34])	23
1.17. Detector de mentiras o polígrafo. Fuente: U.S. Patent 4,333,084. Donald A. Peterson (1980).	23
1.18. Sensores de la compañía g.tec. Extraído de la webpage	25
1.19. Tarea de lápiz sobre hoja de papel	26
1.20. Esquema con dos lazos de control	26
1.21. Ilustración del problema de <i>peg-in-hole</i>	27
1.22. Diagrama de bloques de un control híbrido posición/fuerza	30
1.23. Trayectoria de apertura de puerta.	32
1.24. Esquema explicativo de <i>Compliance Control</i>	33
1.25. Sistema robótico de dos brazos para rehabilitación de miembro superior	36
1.26. Diagonal D1 en movimientos de flexión y extensión	36
1.27. Imagen del fichero ejercicio1.blender de solución del ejercicio	37

1.28. Croquis del Brazo Robótico de 2 GDL	38
1.29. Bloques SimMechanics del Brazo Robótico de 2 GDL	38
1.30. Modelo del entorno	40
1.31. Comparativa Control de Impedancia vs Sin Control de Impedancia	40
1.32. Comparativa en Magnitud. Control de Impedancia vs Sin Control de Impedancia	41
2.1. Representación de la posición en \mathbb{R}^3	45
2.2. Orientación de un sistema de referencia	46
2.3. Rotación básica en el plano	47
2.4. Rotación básica alrededor del vector OX	47
2.5. Rotación básica alrededor del vector OY	48
2.6. Rotación básica alrededor del vector OZ	49
2.7. Representación de orientación mediante el par de rotación	49
2.8. Rotación finita de un punto P'	50
2.9. Representación de orientación mediante los ángulos 313 de Euler	53
2.10. Representación de orientación mediante Roll-Pitch-Yaw	54
2.11. Movimiento de un cuerpo rígido	56
2.12. Tres cuerpos rígidos con tres sistemas de referencia $\{A\}, \{B\}, \{C\}$	57
2.13. Transformaciones básicas de Denavit-Hartenberg.	58
2.14. Definición de los parámetros de Denavit-Hartenberg de un eslabón	59
2.15. Ubicación de los ejes x_i en los sistemas de Denavit-Hartenberg	60
2.16. Relación entre cinemática directa e inversa.	61
2.17. Parámetros de Denavit-Hartenberg de un robot rotacional 6 GDL.	66
2.18. Esquema dinámico del mecanismo.	69
2.19. Modelo en SimMechanics de la cadena UPS.	71
2.20. Modelo de SimMechanics del Mecanismo 6UPS	72
2.21. Modelo dinámico inverso completo del mecanismo 6UPS	73
2.22. Visualización de la simulación del mecanismo 6UPS	74
3.1. Tipos de errores de proyección	84
3.2. Imágenes de rayos X	84
3.3. Imagen de angiografía	85
3.4. imagen US ecográfica	86
3.5. Principio de funcionamiento de un Scanner CT de 3 ^a generación. Extraído de [47]	87
3.6. Principio de funcionamiento del TAC	88
3.7. Equipo e imagen TAC	89
3.8. Equipo e imagen SPECT	90
3.9. Equipo e imagen PET	90
3.10. Equipo e imagen MRI	92
3.11. Equipo e imagen fMRI	92
3.12. Imágenes de difusión magnética	93
3.13. Ejemplo de estándar DICOM	95
3.14. Ejemplo de Dataset de la cavidad abdominal	96
3.15. Herramientas imtool e imcontrast de Matlab [®]	96

3.16. Lectura de la cabecera DICOM con imtool	97
3.17. Comandos en línea para tratamiento de imágenes DICOM	97
3.18. Ejecución del script <i>test1.m</i> de Matlab®	98
3.19. Ejecución del script <i>test3.m</i> de Matlab®	100
3.20. Modulo <i>SlicerWelcome</i>	100
3.21. Visualización de las imágenes DICOM	101
3.22. Creación del ROI	102
3.23. Extracción de la aorta con objeto ROI	102
3.24. Modulo de Fast Marching Segmentation	102
3.25. Colocación de la primera semilla	103
3.26. segmentación de la aorta	104
3.27. Adaptación del contorno de segmentación	104
3.28. Representación 3D de la arteria aorta y sus ramificaciones	105
3.29. Representación 3D de la pelvis y estructura ósea	106
4.1. Señal EMG y su caracterización energética www.delsys.com	111
4.2. Adquisición de señal EMG, músculo bíceps www.delsys.com	112
4.3. Colocación de electrodos para investigación, mapeo EEG. Gentileza de Sabbatini, R (2006). Universidad de Sao Paolo-Brasil. Tomado de la URL: http://www.cerebromente.org.br	113
4.4. Ondas ritmo beta.Fuente Baztarrica (2002)	114
4.5. Sistema de estimulación eléctrica del corazón donde se muestran los nodos A-V. Extraído de web	116
4.6. La generación de biopotenciales en los diferentes nodos del corazón	117
4.7. Bipotenciales y su relación con los biopotenciales en regiones de actividad eléctrica del corazón	117
4.8. Electrocardiograma de un latido, derivación I. Eje vertical: biopotencial; eje horizontal: tiempo	118
4.9. Registro de biopotenciales EOG en un estudio del sueño	119
4.10. Registro de biopotenciales EOG en un estudio del movimiento ocular sacádico	120
4.11. Registro de biopotenciales Electrorretinograma escotópico	120
4.12. Representación gráfica de señales continua a), y discreta en el caso b)	121
4.13. Señal EMG rectificadas y extraídas del músculo bíceps de un amputado	122
4.14. Circuito electrónico para la adquisición de biopotenciales y filtros analógicos	123
4.15. Ejemplos de filtrado digital de una señal ECG y eliminación de ruido entre 50 y 100 Hz.	124
4.16. Algoritmo de procesamiento electrooculográfico	126
4.17. Procesamiento paso a paso de la señal para cada canal	128
5.1. Proceso CAS global	134
5.2. Sistema Cyberknife. Extraído de www.cyberknife.com	139
5.3. Sistema Xsight de Accuray. Extraído de www.cyberknife.com	139
5.4. Principio de funcionamiento de los sistemas HIFU	140
5.5. Sistema HIFU de Ablatherm®. Extraído de hifu-planet	140

5.6. Primera generación del sistema DaVinci (1999). Extraído de Intuitive Surgical	141
5.7. Esquema del RRC del DaVinci	141
5.8. Ejemplos de instrumentos EndoWrist®. Extraído de Intuitive Surgical	142
5.9. Sistema DaVinci Si HD (2009). Extraído de Intuitive Surgical	143
5.10. Primeras versiones de Robodoc. Cortesía de André Bauer, MD.	144
5.11. Historia de Robodoc®. Cortesía de André Bauer, MD.	144
5.12. Concepto de robot montado en paciente. Mars y Dr. Moshe Shoham	145
5.13. Prototipo del Mars. Plataforma de Stewart-Gough	146
5.14. Aplicación del SpineAssist®	146
5.15. Robots cooperativos quirúrgicos. Extraído de [53]	147
5.16. Proceso de generación de un display visual	149
5.17. Simuladores comerciales de MIS	150
5.18. Limitaciones de la cirugía sin navegación	151
5.19. Sistema de navegación quirúrgica	152
5.20. Modelo estereoscópico de ejes convergentes	153
5.21. Sistema electromagnético de tracking	154
5.22. Navegación basada en información CT	155
5.23. Navegación basada en fluoroscopia	156
5.24. Restricciones cinemáticas en MIS	157
5.25. Espacio de trabajo en forma cónica en cirugía MIS abdominal	158
5.26. Muñeca laparoscópica	158
5.27. Explicación de la representación de las orientaciones en cuaternios	159
5.28. Representación del espacio de trabajo del cono laparoscópico	161
5.29. Análisis del espacio de trabajo	164
6.1. Idea de robot basado en efector final y exoesqueleto. Extraído de [15]	169
6.2. GENTLE/S	172
6.3. Esquema de control utilizado en GENTLE/S	172
6.4. MIT-MANUS	173
6.5. ARMIN	173
6.6. Estructura mecánica de ARMIN	174
6.7. iPAM	175
6.8. ARMEO	175
6.9. REOGO	176
6.10. Realidad virtual para el ejercicio de beber agua	177
6.11. Sistema de investigación PuParm y comercial RoboTherapist2D	178
6.12. GBO	180
6.13. Principio de funcionamiento de GBO. Las líneas en color verde representan los resortes que realizan la compensación de fuerzas gravitacionales.	180
6.14. ALEX	181
6.15. Esquema de control articulaciones ALEX	181
6.16. HapticWALKER	182
6.17. Esquema de control de HapticWALKER	183
6.18. LOKOMAT	183
6.19. Esquemas de control en LOKOMAT	184

6.20. REOAMBULATOR	185
6.21. CAREN	185
6.22. Representación del homínulo de Penfield. Extraído de la web	186
6.23. Primera versión del VPL Data Glove	187
6.24. Rutgers Master II	187
6.25. MIT Hand Robot Alpha Prototype II	188
6.26. Hand Mentor	189
6.27. Cyberglove y Cybergrasp	189
6.28. Primer prototipo de i-phalanx	190
6.29. Primer prototipo de guante sensorizado haptico	190
6.30. Sistema Kineci para rehabilitación con realidad virtual	192
6.31. Clase PuParm creada en Matlab	193
6.32. Espacio de trabajo del dispositivo PuParm	194
6.33. Librería de Simulink del dispositivo PuParm	194
6.34. Evolución del centro de gravedad del mecanismo	197
6.35. Posición del centro de gravedad para una configuración singular	198
6.36. Implementación en <i>Simulink</i> de la ecuación (6.8)	198
7.1. (a) Un estudiante con cuadriplejía de alto nivel comanda un sistema De- VAR para insertar un videojuego, (b) Sistema RAID desarrollado en Lund University Sweden	203
7.2. (a) y (b) Asistente Neater Eater, (c) Asistente Myspoon	203
7.3. (a) silla de ruedas con el brazo robótico MANUS en una acción real, (b) Robot RAPTOR	204
7.4. (a) Robot móvil MOVAID, (b) Tres diferentes estructuras del CARE-O- BOT I en el Museum für Kommunikation de Berlin, (c) Care-O-Bot II ofreciendo una bebida	205
7.5. Primeras prótesis encontradas	208
7.6. Prótesis no articuladas Otto Bock: (a) Gancho, (b) Sistema de articulación de cadera HELIX3D, (c) Articulación modular de rodilla, (d) Prótesis de pie y tobillo.	210
7.7. Prótesis robotizadas: (a) Prótesis Myohand, (b) Mano eléctrica, (c) Próte- sis Codo-Antebrazo	212
7.8. (a) Elementos de un dedo, (b) Mano i-Limb	212
7.9. (a) UTAH arm, (b) Brazo de Luke	213
7.10. PowerKnee instalado	214
7.11. (a) Propior Foot, (b) Propio Foot instalado, (c) Prototipo de tobillo	215
7.12. (a) Procesador de texto basado en BCI, (b) Brazo robótico controlado por P300	217
7.13. Ejemplos de interfases mioeléctricas (a) Prótesis de brazo y mano, (b) dedos y codo, (c) Prótesis de Todd Kulken	218
7.14. Ejemplos de interfases basadas en EoG (a) Interfase desarrollada por el Wearable Computing Lab de la ETH Zurich, (b) control de video juegos con la mirada, (c) Control de la navegación aerea	218
7.15. Uso de <code>manofunc.m</code> , $d3 = 1,93, R = 5$ cm.	220
7.16. Uso de <code>manofunc1.m</code> , $d3 = 1,93, R = 8$ cm.	220

7.17. Evaluación de agarre tipo gancho del modelo <code>mano3.wrl</code>	221
7.18. Esquema de control de posición	222
7.19. Arquitectura de la mano robótica	225
8.1. Imágenes de entrada y salida de la práctica <code>g2p1</code>	238
8.2. Imágenes de entrada y salida de la práctica <code>g2p2</code>	239
8.3. Imágenes de entrada y salida de la práctica <code>g2p3</code> . Original, Sobel y Canny	240
8.4. Hola Mundo de VTK	241
8.5. Ejemplo básico de reconstrucción 3D con VTK	242
8.6. Ejemplo básico de integración Qt-VTK	242
8.7. Ejemplos de ejecución IgsTK	242
8.8. Ejemplo de aplicación de IgsTK	245
8.9. Aspecto de la solución del ejercicio	246
9.1. Esquema general de componentes Ogre.	251
9.2. Esquema general de componentes PhysX.	253
9.3. Entorno de simulación, instrumentos y ejercicio para el desarrollo de la coordinación psicomotora	256
9.4. Estructura cinemática del robot LapBot.	258
9.5. Entorno de trabajo en Blender, plano XZ	259
9.6. Trayectoria del instrumento 1 para la disección del triángulo de Calot.	259
9.7. Trayectoria del instrumento 2 para la disección del triángulo de Calot.	260
9.8. Trayectoria del instrumento 1 para el corte del conducto cístico.	260
9.9. Trayectoria del instrumento 1 para el corte de la arteria cística.	261
9.10. Trayectoria del instrumento 1 para la disección de la vesícula del lecho hepático.	261
9.11. Ambiente 3D con dos LapBot y la interfaz de usuario.	262
9.12. Seguimiento 3D de la trayectoria Disección de Calot.	263
9.13. Componentes del sistema Sitkhr	264
9.14. Esquema general de los módulos del sistema Sitkhr	265
9.15. Instantánea del sistema Sitkhr	265
10.1. Maquetas de experimentación con alambres y muelles de Nitinol	268
10.2. Primer prototipo de i-phalanx	269
10.3. Comparación entre el mecanismo real y el modelo CAD	269
10.4. Arduino IDE con ROS	270
10.5. Primer prototipo de guante sensorizado haptico	271
10.6. Robot 2UPS1U para cirugía maxilofacial.	272
10.7. Implementación real del robot y simulación del mismo	272

índice de tablas

1.1. Magnitudes para la estimación del estado del sujeto	24
1.2. Restricciones naturales y artificiales en el problema del <i>peg-in-hole</i>	27
1.3. Restricciones naturales y artificiales generales	30
2.1. Parámetros de Denavit-Hartenberg para el robot rotacional de la figura 2.17	67
5.1. Sistemas robóticos de cirugía maxilofacial	136
5.2. Sistemas robóticos de “Imaging” en cirugía	137
5.3. Sistemas robóticos de cirugía abdominal y torácica	137
5.4. Sistemas robóticos de radiocirugía	137
5.5. Sistemas robóticos de cirugía neurocirugía	138
5.6. Sistemas robóticos de cirugía ortopédica	138
6.1. Algunos sistemas para rehabilitación de miembro superior	171
6.2. Algunos sistemas para rehabilitación de miembro inferior	179
7.1. Parámetros geométricos de la mano	225
7.2. Parámetros dinámicos de base	228

Parte I

Robótica médica

Capítulo 1

Introducción a la robótica médica

“Nunca me he encontrado con alguien tan ignorante de quien no pudiese aprender algo”
Galileo Galilei

Indice

1.1. Introducción	4
1.2. Hitos y retos	5
1.2.1. Hitos de la robótica médica	5
1.2.2. Retos de la robótica médica	9
1.3. Áreas de aplicación de la robótica médica	10
1.3.1. Robótica quirúrgica	11
1.3.2. Robótica de rehabilitación	13
1.3.3. Robótica asistencial	14
1.4. Tecnología para la robótica médica	14
1.4.1. Actuadores para interacción hombre-máquina. (<i>Soft robotics</i>)	15
1.4.2. Sensores para interacción hombre-máquina.	22
1.4.3. Sistemas de control en la interacción hombre-máquina.	25
1.5. ROBOTICA MEDICA APLICADA	35
1.5.1. Ejemplo de cálculo del espacio de trabajo de un robot de rehabilitación con Blender [©]	35
1.5.2. Ejemplo de control de impedancia con Matlab [®]	37
1.6. Resumen del capítulo	40
Bibliografía para ampliar	40

1.1. Introducción

Estamos acostumbrados a que cuando entramos en una fábrica de automóviles o vemos un reportaje de “*Cómo está hecho...*” nos aparezcan imágenes de robots realizando diferentes tareas. Los robots han llegado a ser habituales en muchos de los procesos de fabricación existentes, los podemos encontrar soldando, pintando, manipulando objetos, etc. En todos estos casos, el robot es una entidad aislada que normalmente se encuentra encerrada en su celda de trabajo, y protegido para que las personas no lo toquen cuando está en funcionamiento. Podemos afirmar, que los robots industriales ya forman parte de nuestra visión del mundo que nos rodea.

Sin embargo, todavía encontramos novedoso que un robot coopere con una persona humana para la realización de una tarea. La imagen que nos llega de la ciencia ficción de robots humanoides o robots de servicio que trabajen codo con codo con una persona humana todavía provoca cierta sorpresa e inseguridad al realizar la tarea conjunta. Esta percepción está quizás influenciada por la imagen del robot industrial trabajando solo encerrado dentro de la celda de fabricación. El caso de los robots para aplicaciones médicas es especialmente llamativo, debido al aspecto de las relaciones humanas en los temas de salud. Podemos confiar en que el robot apriete bien el tornillo, pero tenemos reparos a que un robot participe en nuestra salud. Uno de las posibles razones para este rechazo puede estar basada en la influencia de esa imagen de la ciencia ficción; pensamos en el robot como un ente puramente autónomo que toma decisiones sobre parámetros vitales que afectan a nuestra salud. Quizás sería más correcto ver los robots médicos como herramientas avanzadas a disposición de un profesional de la salud (cirujano, fisioterapeuta, asistente social...) que decide utilizar las nuevas posibilidades de estas herramientas avanzadas para procurar una mejor asistencia a la persona.

Para la definición de estos “*medical and health robots*” hemos de fijarnos primeramente en las tareas y entornos en las que los robots médicos realizan sus trabajos. El área de trabajo de los robots médicos es muy amplia, y se extiende desde el microrobot capsula que navega por el interior del cuerpo humano hasta el robot asistente que permite a personas con discapacidades realizar tareas de la vida diaria que no podría realizar sin la ayuda de ese robot, pasando por robots para el quirófano, robots enfermeros, robots para rehabilitación, o robots prótesis u órtesis biomecánicas.

De forma resumida, y en un sentido amplio, podemos decir que son robots médicos aquellos robots que desarrollan su trabajo en cualquiera de estos escenarios; un ambiente **quirúrgico**, un escenario de **rehabilitación** o un ambiente doméstico donde se realizan tareas de **asistencia**. La característica común que encontramos en estos robots es que, aparte de colaborar para realizar una determinada tarea, estos robots médicos tienen una **relación física directa con la persona humana**, es decir hay contacto físico entre el robot y la persona (figura 1.1). Obviamente esta característica condiciona la interfaz hombre-máquina de estos robots, y los hace diferentes de los robots industriales anteriores.

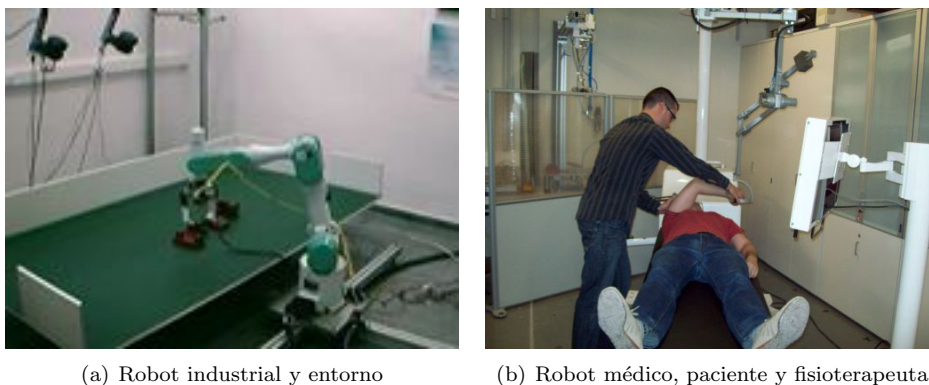


Figura 1.1: (a) Robot industrial vs. (b) robot médico de rehabilitación

1.2. Hitos y retos

1.2.1. Hitos de la robótica médica

Como sabemos, la historia de los automatismos, y por ende de la robótica se remonta a los primeros automatismos de Herón de Alejandría (85 a.C.), y continúa desarrollándose durante la Edad Media y el Renacimiento con los trabajos de, entre otros, Leonardo da Vinci (1452-1519), hasta llegar a la aparición de la palabra *robot* en la novela *R. U. R. - Rossum Universal Robot* del escritor Karel Capek y a las obras de Isaac Asimov (1920-1992). Sin embargo, el inicio de lo que conocemos como robótica moderna, lo podemos situar en los trabajos de telemanipulación eléctrica que se realizaron en 1948 en el Argonne National Laboratory por Raymond Goertz. Posteriormente, en 1956 aparece el primer robot industrial comercial de la empresa Unimation, fundada por Devol y Engelberger. (Una mayor información sobre la historia de la robótica se puede encontrar en *Robótica Aplicada* de R. Saltaren, J.M. Sabater y otros [70] o en *Fundamentos de robótica 2ª Ed* [11] de A. Barrientos y otros).

Por otra parte, la historia de la medicina es la propia historia de la humanidad. Desde las primeras civilizaciones y culturas se ha tratado de curar las enfermedades, bien basándose en una medicina con influencia mágico-religiosa o bien en razonamientos empíricos. Es fácil encontrar resúmenes de la historia de la medicina que recorren la historia desde la prehistoria hasta el siglo XXI, pasando por las diferentes civilizaciones de Mesopotamia, el Antiguo Egipto, Grecia, Roma, China o la América Precolombina (ver por ejemplo www.es.wikipedia.org/wiki/...).

El encuentro entre las dos ramas de la ciencia se produce en 1980, cuando se realizan las primeras experiencias quirúrgicas con robots en neurocirugía y ortopedia, aunque fue en 1985 cuando se utilizó por primera vez un robot PUMA 560 para el guiado de una aguja en neurocirugía. La figura 1.2 resume algunos de los hitos más importantes citando dispositivos pioneros o que han tenido mayor repercusión entre el sector médico.

Dentro del **ambiente quirúrgico**, los hitos más importantes empiezan cuando en 1988 el robot *World First* fue el primero en extraer tejido humano durante una prosta-

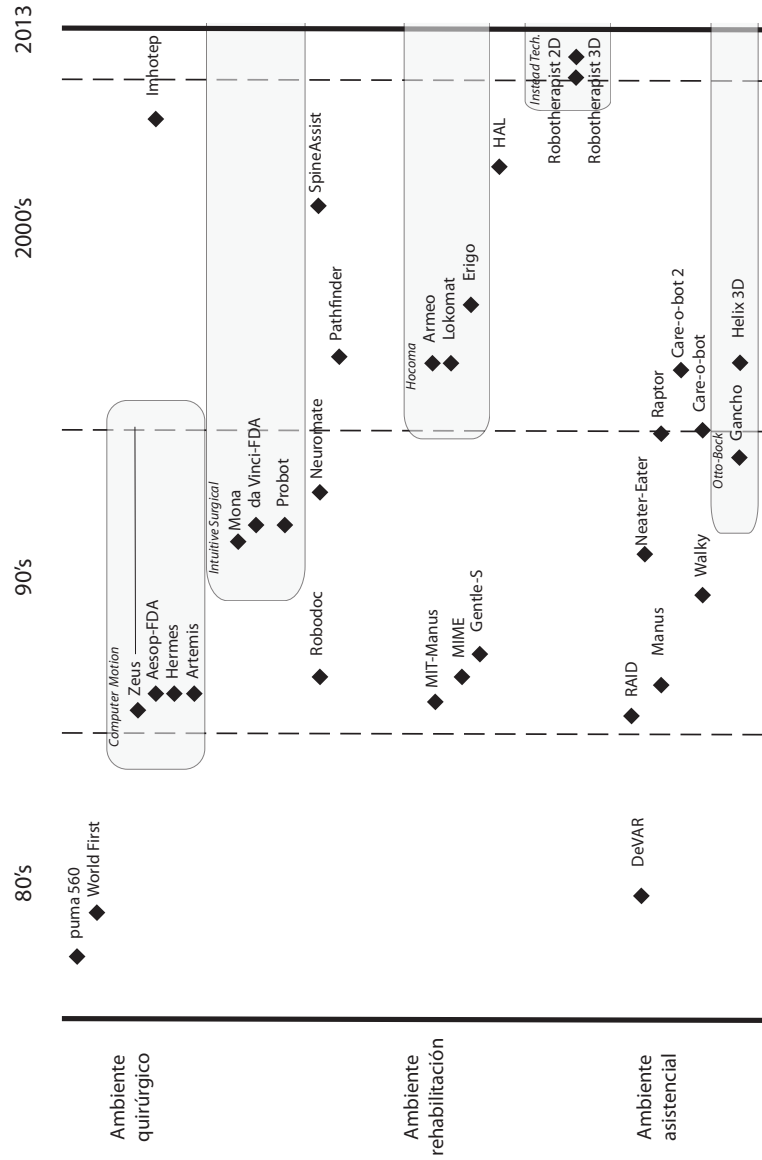


Figura 1.2: Algunos hitos de la robotica médica

tectomía y participar de esa forma en la operación quirúrgica. En 1989 se funda Computer Motion Inc. (Goleta, CA, USA) y se empieza con los desarrollos de los robots quirúrgicos AESOP, Hermes y Zeus. Ya en 1993, se da a conocer el robot Zeus, que en 1996 fue probado en un experimento de cirugía animal. El sistema Zeus se diseñó dentro de un programa militar para la teleoperación a distancia. El mayor hito de este sistema se logró en septiembre de 2001, gracias a la colaboración entre Computer Motion Inc. y France Telecom, cuando se utilizó en una operación de telecirugía con la primera colecistectomía transoceánica entre Estrasburgo (Hospital Civil del Este de Francia), donde estaba la paciente de 68 años, y Nueva York (Mount Sinai Medical Center) a 7.000 Km de distancia, donde el equipo de cirujanos controlaba los mandos de la consola. El proyecto AESOP (Automated Endoscopic System for Optical Positioning), también de Computer Motion Inc., con su modelo 1000 fue el primer robot del mundo aprobado por la FDA. En 1996 ya se comercializaba el modelo 4000. AESOP es un brazo robot operado con comandos de voz desde la interfaz Hermes. Al final de la década de los 90 se presentaron los resultados con el robot ARTEMIS un robot asistente con imagen bidimensional y dos brazos mecánicos que no ha llegado a tener aplicación clínica. En 1995 se constituye la empresa [Intuitive Surgical Inc](#) (Sunnyvale, CA, USA), empresa que acabará adquiriendo las patentes de Computer Motion Inc en 2003. En 1997 ya había iniciado sus experiencias con un sistema de cirugía robotizada mediante un prototipo llamado MONA, un robot precursor del actual DA VINCI, que utilizaba ya un sistema maestro-esclavo con consola de mandos y brazos independientes para que un cirujano, situado fuera del campo quirúrgico, controlara los brazos robóticos colocados, mediante laparoscopia, en el interior del abdomen del paciente. Con esta técnica se realizaron en Bélgica una colecistectomía (1997) y una cirugía bariátrica (1998), la primera en el mundo realizada por un robot dirigido a distancia. En 1999 surge el robot quirúrgico más avanzado conocido hasta la actualidad, el DA VINCI SURGICAL SYSTEM, que en el 2000 obtiene la validación de la FDA para realizar procedimientos de cirugía abdominal vía laparoscópica. Actualmente el sistema DA VINCI-Si es el sistema más avanzado disponible, y se están continuamente añadiendo nuevos procedimientos quirúrgicos validados para este sistema, tanto en cirugía abdominal (funduplicatura de Nissen-Rosetti, miotomía de Heller, anastomosis gastrointestinal tipo Y de Roux, etc) como en otras cirugías. Las intervenciones con da Vinci han pasado de 1.500 procedimientos en el año 2000 a 20.000 en el 2005. Por ejemplo, en España, en 2012 hay 22 sistemas DA VINCI, 12 de ellos en el sistema público de salud, y 10 más de gestión privada. Otros desarrollos importantes han sido el robot PROBOT, del Imperial College of London, que fue usado en el bienio 1997-98 para realizar operaciones endoscópicas de próstata en Inglaterra, o el sistema ROBODOC para operaciones de ortopedia, principalmente de rodilla o cadera. Recientemente están apareciendo nuevos dispositivos con conceptos distintos y aplicables a diferentes quirúrgicas, como el ([Cyberknife](#)) para radioterapia, los sistemas HIFU (high intensity focused ultrasound) para ablación, como el de [Ablaterm®](#) o robots quirúrgicos en miniatura.

Dentro del **ambiente de rehabilitación** hay que destacar algunos sistemas pioneros como el dispositivo MIT-Manus (hoy comercializado como [INMOTION ARM](#)). Este sistema para rehabilitación de miembro superior es un manipulador robótico planar construido mediante un mecanismo SCARA que permite dos grados de libertad traslacionales y que fue diseñado para rehabilitación de pacientes con apoplejía. El MIT-Manus puede

ayudar u oponerse a los movimientos de hombro y codo del paciente. Este robot usa diferentes sensores para medir la posición, velocidad y fuerzas ejercidas sobre el efector final. Esta información es enviada a un PC donde se actualiza una interfaz gráfica de usuario. Esta interfaz consiste de varios juegos diseñados para motivar al paciente en su rehabilitación. Si el paciente no responde a las indicaciones del juego, el robot comienza a ayudar al paciente a realizar el ejercicio. Otro dispositivo pionero fue el dispositivo MIME, que fue creado para rehabilitación de extremidades superiores. Este dispositivo fue diseñado utilizando un robot PUMA-560 para el brazo hemipléjico y un soporte 3D para el brazo no afectado. Este dispositivo permite al paciente trabajar en un espacio 3D, de forma que el robot PUMA es capaz de sujetar el brazo durante movimientos 3D. El robot está conectado a un sensor de fuerzas para monitorizar la fuerza del paciente. MIME utiliza 3 modos unilaterales y una bilateral de funcionamiento. En el modo bilateral, el brazo no afectado controla el movimiento del robot. Los modos unilaterales, utilizan solo el robot PUMA, y son los modos pasivo, activo-asistido, activo-resistencia, y auto-guiado. El sistema GENTLE/S es un sistema robótico financiado por la Comisión Europea formado principalmente por el brazo robot Haptic Master de 3 GDL y un ordenador en el que se muestra un entorno virtual interactivo. Se utiliza una articulación pasiva de 3 GDL para sujetar el brazo del paciente al robot. Además, se emplean una cuerdas enlazadas a una estructura para soportar al antebrazo y el brazo mediante una órtesis. Este sistema permite movimientos 3D del brazo. La figura 1.3 muestra el sistema SINU (precursor del sistema comercial [RoboTherapist 3D](#)), un robot neumático desarrollado por la Universidad Miguel Hernández para la rehabilitación de miembro superior. Se trata de un sistema con dos brazos neumáticos de 3 grados de libertad activos cada uno.

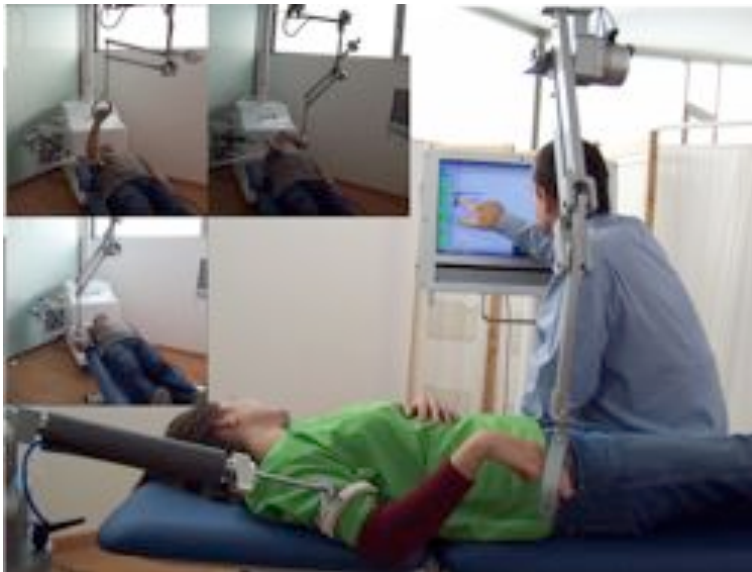


Figura 1.3: Robot neumático para rehabilitación de miembro superior

Desde el punto de vista comercial, existen varias empresas con productos para la rehabilitación robótica. La empresa [HOCOMA](#) ha logrado desarrollar diferentes dispositivos comerciales, tanto para miembro superior (Armeo) como para miembros inferiores (Lokomat) o movilidad del paciente (Erigo). La empresa [Instead Technologies, Ltd.](#) comercializa robots de rehabilitación, tanto neumáticos (robotherapist2D y robotherapist3D) como eléctricos (robin1).

Dentro del **ambiente de robótica asistiva** podemos encontrar diferentes dispositivos asistenciales, tanto para personas con movilidad reducida como prótesis u órtesis cuya finalidad es mantener y/o aumentar las capacidades funcionales de la misma. Entre los dispositivos para movilidad reducida, hay que destacar los sistemas DeVAR (Desktop Vocational Assistant Robot) y RAID (Robot to assist the integration of the disabled), desarrollados en los años 90. También dispositivos actuales con éxito comercial como el Neater Eater de la empresa Neater Co. El sistema de silla de ruedas RAPTOR que portaba un brazo robótico MANUS fue el precursor de otros sistemas actuales como el Asibot de la figura 1.9. En cuanto a los sistemas móviles asistentes, el primer prototipo Care-o-bot construido por el Fraunhofer IPA en 1998, y que tiene su continuación en el actual prototipo Care-o-bot III presenta soluciones para resolver los problemas de navegación autónoma por el interior de una vivienda. Además, también hay que destacar en este área los avances realizados en el desarrollo de prótesis u órtesis actuadas. Desde los primeros trabajos de prótesis realizados en la civilización egipcia hasta los actuales desarrollos de la mano cyberhand, resolver el problema de las amputaciones o movilidads reducidas ha sido un reto importante para la comunidad científica. Empresas como [Otto-Bock](#) ofrecen sus productos protésicos con diferentes grados de movilidad y autonomía.

Una descripción más detallada de los hitos logrados en robótica médica se puede encontrar en los capítulos 52, 53 y 54 del libro “Handbook of robotics” de Springer [76].

1.2.2. Retos de la robótica médica

Los requerimientos de la robótica médica son por tanto diferentes de los requerimientos de la robótica industrial. Esto hace que los retos tecnológicos que deben resolverse también sean distintos. Como ejemplo sirva el cambio de paradigma que se produce en el diseño de robots intrínsecamente seguros ¹:

- En el paradigma de diseño actual, la tecnología ha puesto su contribución en hacerlos lo más precisos posibles, mejores actuadores, mejores reductoras, mejores transmisiones,..., y posteriormente se trata de hacer estos robots precisos lo más seguros posibles gracias a diferentes estrategias de control y diferentes sistemas de seguridad.
- Sin embargo, en el nuevo paradigma de diseño de robots intrínsecamente seguros, la tecnología pone su contribución en hacerlos seguros, nuevos actuadores flexibles, nuevos diseños mecánicos, nuevos sistemas,..., y posteriormente se trata de hacer estos robots seguros lo más precisos posible, de forma que sean capaces de resolver la tarea a la que se asignan.

¹Robots diseñados bajo criterios de seguridad intrínseca, de forma que ante cualquier error o fallo de control, el robot no produce ningún daño.

Los retos que actualmente se trabajan en robótica médica se pueden agrupar en:

Seguridad. Es inmediato que la seguridad es un aspecto vital en el desarrollo de herramientas médicas. Tanto en el ámbito quirúrgico, como sanitario o de rehabilitación, la “seguridad”, y la “seguridad percibida” son dos conceptos que deben potenciarse y mejorarse. Es necesario que se apliquen nuevos diseños basados en la seguridad intrínseca, y que además esta seguridad sea percibida por el usuario del robot.

Fiabilidad. Ligado a la seguridad, es necesario que los robots tengan una fiabilidad alta, de forma que además de ser seguros, realicen la tarea asignada de forma fiable, asegurando así un buen funcionamiento del robot. Por ejemplo, en un robot de rehabilitación, no basta con que sea seguro, sino que debe realizar las tareas asignadas correctamente, de forma que se asegure que la rehabilitación está siendo efectiva.

Facilidad de uso. El aspecto de la interfaz entre el robot médico y el usuario también es un aspecto muy importante. En la mayoría de los casos, el usuario final del sistema robótico será personal médico y un paciente, por lo que se requiere que el uso de este sistema sea sencillo, fácil e intuitivo. Se está avanzando mucho en el diseño de nuevas interfaces, como por ejemplo interfaces hapticas sencillas para robots quirúrgicos teleoperados, interfaces mioeléctricos para prótesis o interfaces BCI (Brain Computer Interface) para control de dispositivos de comunicaciones.

Todo ello debe hacer que podamos encontrar soluciones estratégicas en campos como:

- La cirugía robotizada y la radiología intervencional.
- El reemplazo robótico de miembros amputados (prótesis) o con funcionalidad reducida (órtesis).
- La rehabilitación robótica y la asistencia a personas con necesidades especiales.
- El cuidado de personas con necesidades de asistencia social.

El trabajo en estos retos nos llevará a conseguir escenarios como los que ya se han visualizado en diferentes proyectos, libros o películas, como el sistema de mantenimiento vital a distancia TraumaPod (www.sri.com), la navegación por el interior del cuerpo humano del viaje fantástico de I. Asimov, ([fantastic voyage](#)) o la mano protésica de Luke Skywalker en Star Wars (www.starwars.com).

1.3. Áreas de aplicación de la robótica médica

En aras de una clasificación de los diferentes sistemas de robótica médica, en este libro se ha optado por la clasificación según áreas de aplicación. Obviamente existen otras muchas posibilidades para clasificar los robots médicos, bien atendiendo a su morfología (robots seriales, robots paralelos, robots mixtos serie-paralelo, robots móviles), bien atendiendo a su esquema de operación (robots autónomos, robots teleoperados, robots semi-autónomos), bien a su disposición sobre el paciente (robots montados en camilla,

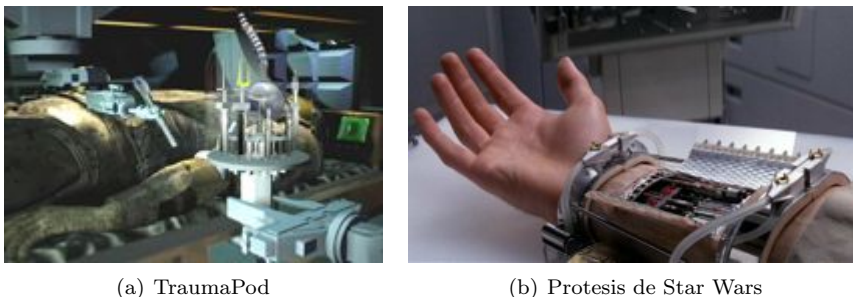


Figura 1.4: Escenarios futuros vistos en ciencia ficción

robots montados en paciente, robots portables (wearable robots)). Se ha elegido la clasificación según áreas de aplicación con el fin de remarcar las particularidades de cada entorno, por ejemplo, no es necesaria la misma precisión en el posicionamiento de un robot en un ambiente quirúrgico que en un ambiente de rehabilitación física. Dentro de cada área se verán en los capítulos correspondientes (5, 6 y 7) ejemplos de diferentes tipos de robots trabajando en dicho entorno.

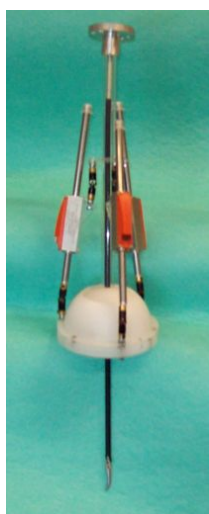
1.3.1. Robótica quirúrgica

En cuanto a la robótica quirúrgica, ésta representa un cambio radical en el trabajo del quirófano, convirtiendo al robot en un integrante más del equipo quirúrgico. En palabras del doctor Julio Mayol, cirujano y coordinador de la Unidad de Cirugía Guiada por la Imagen del hospital Clínico San Carlos de Madrid, *“este nuevo enfoque representa una visión de futuro y requiere un esfuerzo de cambio conceptual del abordaje quirúrgico. Todo ello la convierte en una cirugía integra, intuitiva e inteligente”*. Según Russell Taylor, uno de los pioneros en el campo de la robótica quirúrgica, ésta pretende crear una sociedad entre el cirujano y el robot de forma que se aumenten las capacidades del cirujano y se puedan realizar los procedimientos de forma más eficiente que podrían hacerlos por separado cirujano o robot. Robots como el [DaVinci](#) o [Robodoc](#) son pioneros en la integración de robots en un quirófano, y están convirtiéndose en compañeros del cirujano. Sin embargo, y como pioneros que son, están en continua evolución para llegar a disponer de robots más eficientes, más ergonómicos y sobre todo que permitan realizar nuevos procedimientos quirúrgicos que aumenten las posibilidades y destrezas que actualmente dispone el cirujano. En la actualidad existen muchos debates abiertos alrededor de la aplicación de robots en el quirófano. Algunos de estos debates hacen referencia al uso de esquemas de control de robots autónomos o de robots teleoperados. Otros discuten acerca de la morfología de los robots, que pueden ser de grandes dimensiones y estar fijos a la propia estructura del quirófano (table-mounted robots), como el [Cyberknife](#) o el [daVinci](#), o pequeños y estar fijados a la propia estructura ósea del paciente (patient-mounted robots), como el [SpineAssist](#) (figura 1.5), evitando desplazamientos relativos entre la base del robot y el paciente.

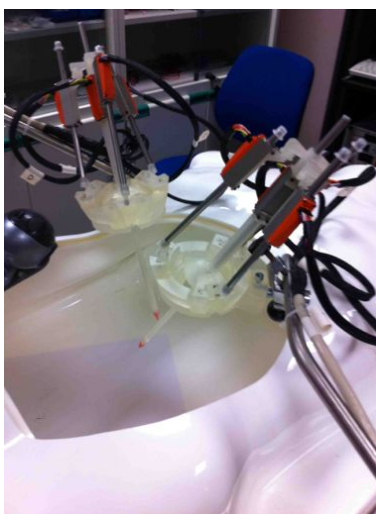
A nivel de investigación, están apareciendo nuevos dispositivos, como el robot [Imhotep](#)



Figura 1.5: [SpineAssist](#) - Robot paralelo para operar en la colocación de fijaciones mecánicas de prótesis en la columna vertebral. Cortesía de Mazor Inc.



(a) versión 1 del prototipo



(b) versión 2 del prototipo

Figura 1.6: [Imhotep](#) - Muñeca esférica paralela para operaciones MIS. [Video youtube de demostración](#)

(figura 1.3.1) o el dispositivo para cirugía maxilofacial (figura 10.6) que presentan avances sobre los actuales robots, como son un menor tamaño, una mejor adaptación al paciente o un mayor nivel de sensorización.

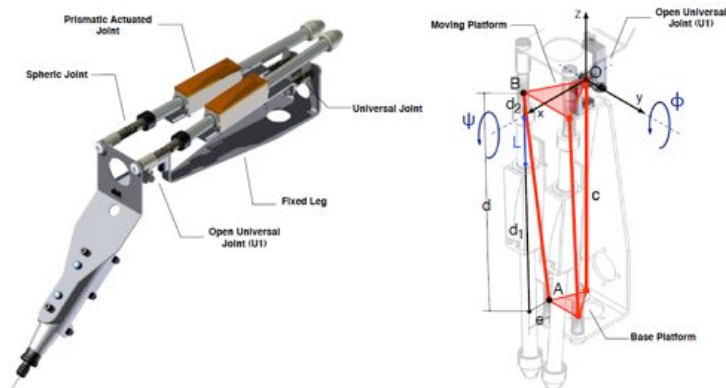


Figura 1.7: Robot 2UPS1U para cirugía maxilofacial.

También aparecen los robots móviles, capaces de navegar por el interior del cuerpo humano, y por extensión, los robots colaborativos, pequeños robots de capacidades limitadas que trabajan conjuntamente para realizar una determinada tarea quirúrgica.

1.3.2. Robótica de rehabilitación

En los robots de rehabilitación la interacción hombre-máquina alcanza su mayor exponente. Las tareas de rehabilitación suelen centrarse en movimientos repetitivos realizados con la ayuda del fisioterapeuta. De este modo el robot permitiría que el paciente realizara los movimientos repetitivos, y únicamente el fisioterapeuta debería indicar inicialmente el movimiento que posteriormente deberá repetir el robot. El uso de robots en rehabilitación tiene dos aplicaciones fundamentales:

- Durante los movimientos activos del paciente permiten almacenar información de la realización del movimiento (posición, velocidad, fuerza), ayudando de esta forma a evaluar la evolución de la terapia sobre el paciente.
- Por otra parte, también pueden provocar movimientos pasivos de la extremidad del paciente sujeta por el robot, sustituyendo en este caso a los movimientos que realiza el fisioterapeuta.

Los robots “wearables”, como el actual HAL (Hybrid Assistive Limb) (figura 1.8), que esperamos en un futuro podrían vestirse como una prenda más. El HAL multiplica de 2 a 10 veces la fuerza del usuario, con un mínimo esfuerzo. Este traje robótico, dice el sitio oficial: “lee las señales nerviosas que son enviadas al cerebro por los músculos vía una motoneurona, que mueve el sistema músculo esquelético como consecuencia. En ese momento, débiles bioseñales pueden ser detectadas en la superficie de la piel. HAL agarra esas señales a través de un sensor adjunto en la piel del usuario. Basado en las señales que obtiene, la unidad de poder responde a las actividades diarias del usuario”. Sin duda el

conocimiento que incorporan estos robots en una parte importante se debe a los avances en los robots humanoides, tales como la mecánica de la marcha, la dinámica y el control en tiempo real de las articulaciones. De otra parte los avances de la tecnología, como son los nuevos materiales plásticos, los avances en los sistemas de potencia en baterías y los avances en la fabricación facilitan que los robots portables cada vez sean más livianos y fáciles de integrar a las personas.



Figura 1.8: Robot HAL exo-esquelético. Extraído de la [webpage](#) de Cyberdyne.[video](#)

1.3.3. Robótica asistencial

Los robots asistentes, como el [Asibot](#) mostrado en la figura 1.9, además de las características morfológicas compatibles con el tipo de asistencia humana, son dispositivos que deben ajustarse a una serie de condiciones de seguridad, robustez y fiabilidad para que su trato diario se ajuste a los requerimientos del usuario en función de su grado de discapacidad. Este ajuste proviene de la información médica y directamente intervienen en el control del movimiento del robot. Aparte de los aspectos cinemáticos estos dispositivos cuentan con una importante integración sensorial para medir las fuerzas y pares que resultan de la interacción del paciente-máquina para que esta sea segura. Esta interacción al final se refleja en la implementación de algoritmos de control que consideran las fuerzas y las posiciones.

1.4. Tecnología para la robótica médica

Debido al carácter multidisciplinar de la robótica, la tecnología que se estudia en un curso básico de robótica incluye varios aspectos:

- En el ámbito mecánico, se estudian las articulaciones del robot, las transmisiones de movimiento, las reductoras, ...



Figura 1.9: Robot asistente desarrollado en la Universidad Carlos III de Madrid. Cortesía del [Robotics Lab](#) la UC3M

- En el ámbito de la electrónica de sensores, se estudian los sensores de posición (encoder, resolver, syncro,...), los sensores de velocidad y aceleración, los sensores de presencia, sensores de fuerza, etc.
- En el ámbito del servocontrol se estudian los controles clásicos de posición, velocidad y fuerza, y se aprende a implementarlos en electrónicas como tarjetas multitejes, controladores dedicados o sistemas embebidos de tiempo real.
- En el ámbito de la interfaz de usuario, se aprende a programar trayectorias cartesianas.

Toda esta tecnología es aplicable y necesaria para seguir este curso de robótica médica, y se recomienda su revisión en cualquier texto de robótica básica, como *Robotica Aplicada* de R. Saltaren, J.M. Sabater y otros [70] o el texto *Diseño y control de robots industriales: Teoría y práctica* de A. Vivas [84].

En esta sección se van a revisar brevemente los conceptos menos clásicos, como son las investigaciones sobre *soft robotics*, o robots “blandos”, debido a su aplicación en la interacción segura hombre-máquina y a su aplicación a la robótica médica. Es obvio que para conseguir nuevos avances significativos en robótica se han de conseguir mejores actuadores, con dinámicas y características más adecuadas para las aplicaciones médicas. También se revisan algunos aspectos de los sensores para interacción hombre-máquina, control o sensores para la navegación quirúrgica.

1.4.1. Actuadores para interacción hombre-máquina. (*Soft robotics*)

Para explicar qué es un actuador flexible (*Compliant actuators* en inglés), es bastante útil hacer primero una definición de lo que es un actuador rígido (*Stiffness actuators*). Un actuador es un dispositivo que es capaz de moverse a una determinada posición o seguir una trayectoria predefinida. Una vez alcanzada la posición deseada, un actuador rígido se mantendrá en esa posición independientemente de las fuerzas externas que actúen sobre él (siempre y cuando estas fuerzas externas no sobrepasen los límites del propio actuador). Los actuadores rígidos se pueden encontrar en la mayoría de aplicaciones de

control de posición. La mayoría de sistemas de actuación eléctrica son rígidos, si bien un motor eléctrico en sí no es rígido debido al campo magnético; sin embargo el uso de un reductor, que normalmente se usa para disminuir la velocidad e incrementar el par motor, se refleja en una mayor inercia. Un reductor con un factor de reducción alto es incluso autoblocante debido a la fricción, lo que significa que, cualquier par de rotación aplicado al eje de salida (dentro de los límites del sistema), no produce ningún movimiento.

Por otro lado, un actuador flexible permite una cierta desviación de la posición de equilibrio, dependiente de la fuerza externa que actúa sobre éste. La posición de equilibrio de un actuador flexible se define como la posición del actuador cuando no genera ninguna fuerza o par. Esta posición puede ser controlada directamente, mientras que la posición real depende también de la rigidez del actuador y de las fuerzas externas que actúan sobre él.

La rigidez de un actuador es comparable con la de un muelle lineal. La variación de la longitud de un muelle lineal depende de las fuerzas que actúan sobre éste de acuerdo con la Ley de Hooke:

$$F = -k(x - x_0)$$

Esto significa que un muelle con una longitud en reposo x_0 y una longitud actual x genera una fuerza F . Si ambas longitudes son iguales no se genera ninguna fuerza, como pasa en un actuador flexible cuando se encuentra en su posición de equilibrio. Cuando se utiliza un muelle con una relación fuerza-desplazamiento lineal, la rigidez se puede definir de la siguiente manera:

$$k = -\frac{F}{x}$$

Cuando esta relación no es lineal, y la rigidez varía con la posición, la rigidez local se puede definir como la variación de la fuerza en función de la posición:

$$k = -\frac{dF}{dx}$$

En el caso de un muelle la posición de equilibrio es fija, mientras que un actuador flexible puede cambiar esta posición, por ejemplo, cambiando el punto de unión del muelle.

En cuanto a la terminología que se utiliza para este tipo de actuadores, hay que hacer algunas observaciones. Mientras que flexibilidad (*Compliance*) y rigidez (*Stiffnes*) son términos opuestos, ambos pueden usarse para describir el comportamiento flexible de un actuador. Para describir un actuador con rigidez variable se usa también el término flexibilidad ajustable, así como flexibilidad variable, rigidez ajustable y rigidez controlada. Con estos ejemplos podemos observar como no hay una terminología estándar para describir el tipo de actuador. Por ello, en los ejemplos de diseños que se describen en este apartado se utiliza la terminología utilizada por los inventores.

Un ejemplo de actuador flexible es el *Series Elastic Actuator (SEA)* (figura 1.10), que es básicamente un muelle lineal en serie con un actuador rígido. La flexibilidad de este actuador viene dada por el muelle elegido, y ésta no puede ser modificada durante la operación.

En cuanto a actuadores con flexibilidad adaptable, se han desarrollado una gran cantidad de diseños. Se pueden distinguir en tres grandes grupos:



Figura 1.10: Ejemplos CAD de *Series Elastic Actuator* con actuadores hidráulicos y eléctrico. Extraído de [34]

- Disposición antagonista de dos resortes no lineales.

La idea es que dos actuadores con una flexibilidad no adaptable, con una relación fuerza-desplazamiento no lineal, estén dispuestas de forma antagonista, esto es, uno trabaja en oposición del otro. Así, controlando ambos actuadores, se puede seleccionar la flexibilidad y la posición de equilibrio de esta configuración antagonista.

- Control de la rigidez estructural.

Modificando las propiedades dimensionales de un resorte, como la longitud o el momento de inercia, se puede adaptar la constante de rigidez del resorte. Por ejemplo, cuando una barra elástica está fijada por uno de sus lados, el otro lado tendrá un comportamiento flexible, es decir, cuando se ejerza una fuerza ortogonal a la barra aparecerá una deformación en ésta. La deformación para una determinada fuerza dependerá de las propiedades del material, de la longitud y del momento de inercia del elemento.

- Control mecánico de la rigidez.

Modificando el punto donde un elemento flexible se fija a la estructura de la articulación, el comportamiento flexible de la articulación cambiará.

1.4.1.1. Disposición antagonista de dos resortes no lineales

El ejemplo más conocido de configuración antagonista es la combinación de bíceps y tríceps en el brazo humano. Cuando el bíceps se contrae y el tríceps se relaja, el brazo se flexiona. Cuando el tríceps se contrae y el bíceps se relaja, el brazo se extiende. Una de las razones por las que se requiere una configuración antagónica es el hecho de que los músculos sólo pueden tirar, y no empujar. Sin embargo, aún se puede conseguir otro efecto con esta configuración: cuando ambos músculos se contraen, el codo se pone rígido, y cuando ambos se relajan el codo se pone muy flexible y el brazo cuelga libremente. Lógicamente, en esta explicación se han mencionado los casos extremos, en realidad los músculos del brazo están controlados continuamente y, por lo tanto, pueden cubrir una

gran número de posiciones y niveles de rigidez. Esta configuración, inspirada en la configuración antagonista biológica (figura 1.11), se usa con distintos actuadores mecánicos para obtener una flexibilidad adaptable.

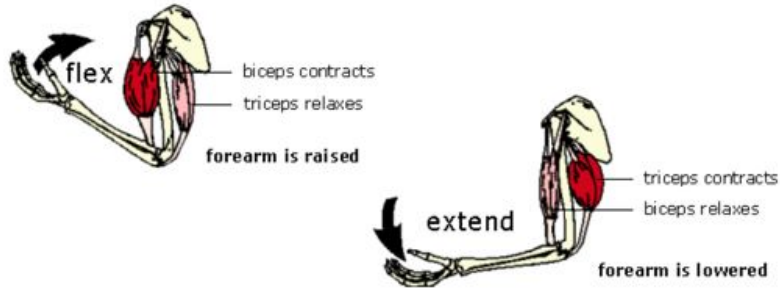


Figura 1.11: Configuración antagonista de bíceps y tríceps en el brazo humano. (Extraído de [34])

Hay una buena razón por la que esta sección se denomina *Disposición antagonista de dos resortes no lineales*, y es que la no linealidad del resorte es esencial para obtener la flexibilidad adaptable. Para explicar esto se va a usar un ejemplo sencillo de configuración antagonista lineal (extraído de [34]). En ella disponemos de dos resortes lineales con la misma constante de rigidez. En la *Figura 1.12*, x_{0A} y x_{0B} son las posiciones controladas cuando no actúa ninguna fuerza externa sobre los resortes. Estas posiciones pueden ser controladas por dos actuadores rígidos. La fuerza sobre el bloque central, aplicada por ambos resortes, es la suma de las fuerzas debidas a estos:

$$F = -k(x - x_{0A}) + k(x_{0B} - x) = -2kx + k(x_{0A} - x_{0B})$$

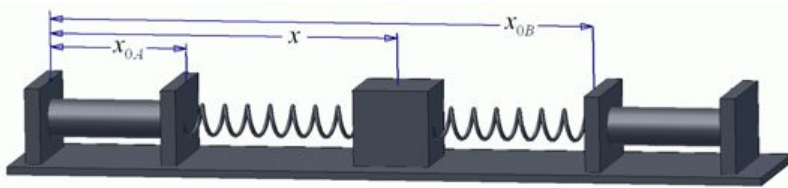


Figura 1.12: Explicación de la necesidad de resortes no lineales. (Extraído de [34])

Para calcular la rigidez, se puede usar la siguiente ecuación:

$$k = -dF = \frac{2k}{dx}$$

Como puede observarse el resultado es independiente de los parámetros controlables x_{0A} y x_{0B} . Por lo que en el caso de una configuración antagonista con dos resortes lineales, la flexibilidad es fija y no puede cambiarse durante la operación.

En cambio, cuando se usan dos resortes con una respuesta cuadrática, la fuerza resultante es:

$$F = -k(x - x_{0A})^2 + k(x_{0B} - x)^2 = 2kx(x_{0A} - x_{0B}) + k(x_{0B}^2 - x_{0A}^2)$$

La rigidez local sería en este caso:

$$k = -\frac{dF}{dx} = 2k(x_{0B} - x_{0A})$$

Como puede observarse, la rigidez es una función lineal de la diferencia entre los parámetros controlables. La posición de equilibrio es la posición donde no se generan fuerzas:

$$\begin{aligned} 2kx(x_{0A} - x_{0B}) + k(x_{0B}^2 - x_{0A}^2) &= 0 \\ \Rightarrow x &= \frac{x_{0A}^2 - x_{0B}^2}{2(x_{0A} - x_{0B})} = \frac{x_{0A} + x_{0B}}{2} \end{aligned}$$

Esta posición es la media de las distancias x_{0A} y x_{0B} . Por lo tanto, controlando estas dos posiciones, podemos configurar tanto la flexibilidad como la posición de equilibrio. Este principio es el utilizado en varios diseños que veremos a continuación.

Ejemplo 1. Control de rigidez de la articulación (inspiración biológica)

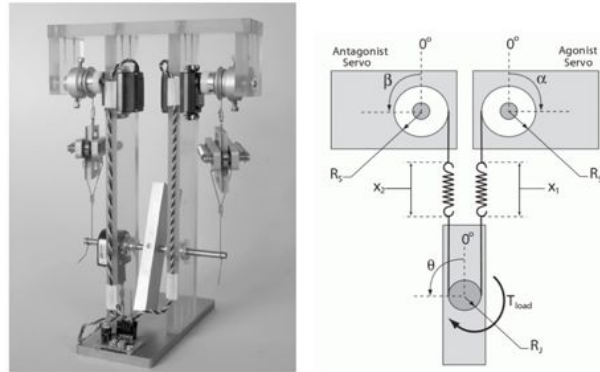
El dispositivo que se va a describir está basado en una configuración antagonista de dos resortes no lineales para formar una articulación rotacional, accionada por dos *Series Elastic Actuator*. En la Figura 1.13(a) podemos ver una imagen de la configuración experimental y de un dibujo esquemático del principio de actuación. En el esquema se muestra la posición de los dos servomotores, y se puede intuir que su labor es modificar la posición de los resortes.

Como ya se ha comentado, para que el sistema pueda tener una flexibilidad adaptable, los resortes tienen que ser no lineales. Para obtener esa no linealidad en la relación angular, se utilizan resortes cuadráticos. En este sistema en concreto, la relación lineal de un resorte la transforma en una relación cuadrática mediante el uso de piezas con formas especiales, sobre las que ruedan dos ruedas, como se muestra en la imagen 1.13(b). Los centros de las ruedas están interconectadas mediante un resorte lineal.

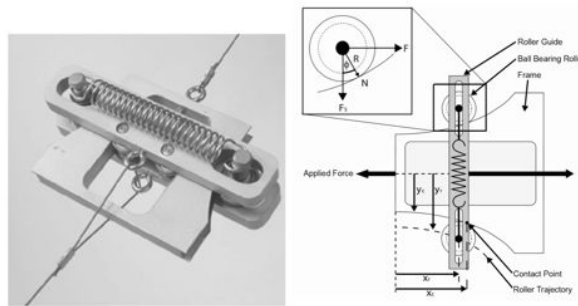
La ventaja de este diseño es que la relación fuerza-elongación del resorte puede elegirse durante el diseño, y por lo tanto la flexibilidad de todo el sistema. Los inconvenientes son el tamaño, la complejidad y la fricción adicional de los mecanismos para hacer cuadrática la respuesta del resorte.

Ejemplo 2. Músculos neumáticos

Un Músculo Neumático Artificial es un dispositivo que, cuando se incrementa la presión del aire de su interior, se contrae según su eje axial mientras se expande según su eje radial. La compresibilidad del aire lo hace intrínsecamente flexible, es decir, actúa como un resorte. La relación fuerza-contracción de un músculo neumático es fuertemente



(a) Imagen y esquema del sistema antagonista con dos SEA



(b) Imagen y esquema del dispositivo resorte cuadrático

Figura 1.13: Esquema de sistema antagonista con resorte cuadrático. (Extraído de [34])

no lineal, por lo que se pueden utilizar para conseguir una flexibilidad adaptable mediante una configuración antagonista.

En la figura 1.14 podemos ver dos configuraciones antagonistas distintas para músculos neumáticos: la configuración de la izquierda es lineal; mientras que la configuración de la derecha es rotacional.

Existen un gran número de tipos diferentes de músculos neumáticos, pero todos basados en el mismo principio. Cuando el volumen diseñado aumenta de presión, el músculo se expande en una dirección y se contrae en la otra. Uno de los inconvenientes de los músculos neumáticos es la histéresis introducida por la fricción, además de un importante umbral de presión hasta que se empieza a generar fuerza.

Un enfoque para solucionar estos problemas es el propuesto por la [Universidad Vrije Universiteit Brussel](#), que ha diseñado un tipo de músculo neumático con pliegues (*Pleated Pneumatic Artificial Muscle*) que reduce drásticamente la histéresis además de bajar el umbral de presión hasta el que empieza a generar fuerza. Otra posibilidad es acudir a técnicas de control no lineal como el “dither” [33] para solucionar estos problemas.

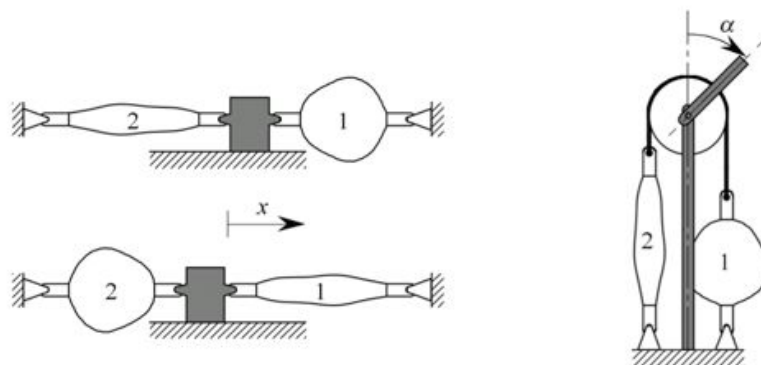


Figura 1.14: Configuraciones antagonistas con músculos neumáticos. (Extraído de [34])

1.4.1.2. Control de la rigidez estructural

Una alternativa a la configuración antagonista, es la variación en la rigidez mediante la manipulación de la estructura de un resorte (*Structured Controlled Stiffness* o SCS). Para entender los conceptos básicos de un actuador SCS, considérese la siguiente ecuación:

$$M = \left(\frac{EI}{L} \right) \theta$$

Donde M es el momento flector, E es el módulo elástico del material, I es el momento de inercia, L la longitud de la barra elástica, y θ es el ángulo de doblado. En esta representación de la doblez, el término EI/L representa la rigidez. Para poder controlar la rigidez de la estructura, sólo disponemos de tres parámetros para manipular.

El parámetro E es una propiedad del material, por lo que no puede ser controlado por un cambio estructural; aunque para algunos materiales podría cambiarse, por ejemplo, cambiando su temperatura. Un cambio en la temperatura puede no ser suficientemente rápido para crear actuadores de flexibilidad variable en aplicaciones de robótica. A continuación se muestran algunos ejemplos de mecanismos donde se varía el momento de inercia (I) y la longitud del elemento elástico (L).

Ejemplo 1. Impedancia mecánica variable

Un SEA (*Series Elastic Actuator*) puede ser construido con un elemento pasivo con impedancia mecánica variable, dando como resultado un actuador con flexibilidad variable. Como resorte, se puede usar una lámina flexible de sección rectangular, ya que cuando las dos dimensiones de la sección difieren bastante, la flexibilidad puede modificarse simplemente rotando la lámina. Un prototipo de este tipo se puede observar en la imagen 1.15. La función del muelle helicoidal es reducir los efectos de pandeo.

Cuando se calcula el momento de inercia con la siguiente fórmula, se puede observar



Figura 1.15: Variación del momento de inercia por rotación de la lámina. (Extraído de [34])

que éste varía dependiendo de la relación anchura-espesor de la lámina.

$$I_{stiffness} = \frac{thickness \cdot width^3}{12}$$

$$I_{compliant} = \frac{width \cdot thickness^3}{12}$$

Ésta es una manera fácil de obtener un elemento flexible con dos configuraciones predefinidas de flexibilidad. Para poder utilizar configuraciones intermedias, habría que estudiar soluciones para el pandeo.

1.4.1.3. Control mecánico de la rigidez

Otro concepto para ajustar la flexibilidad puede observarse en el esquema de la Figura 1.16. El elemento de la izquierda se considera inmóvil. El elemento de la derecha es el brazo actuado rotacionalmente por la articulación. El tercer elemento, es un elemento de palanca, de la que el ángulo φ relativo al primer elemento puede ser controlado por un motor. Entre un punto del brazo y otro del elemento palanca se sitúa un resorte. En caso de tener un resorte de longitud de reposo cero, el resorte tiende a alinear el brazo con la palanca. Mientras no haya ningún par de torsión aplicado al brazo, cuando éste está alineado con el elemento palanca, el elemento palanca es el que determina la posición de equilibrio de la articulación. Si se varía la posición de los puntos donde se sitúa el resorte (variando las longitudes B y C), se puede ajustar la rigidez de la articulación.

1.4.2. Sensores para interacción hombre-máquina.

Muchas investigaciones se centran actualmente en la identificación de los sentimientos y emociones de las personas. Para lograr una efectiva interacción hombre-máquina, se hace necesario, además de disponer de los sensores necesarios para conocer el estado de la máquina, y que ya son conocidos por nosotros (encoders, resolvers, sensores de presencia, sensores de fuerza,...) conocer también el estado del hombre. Esta idea no es

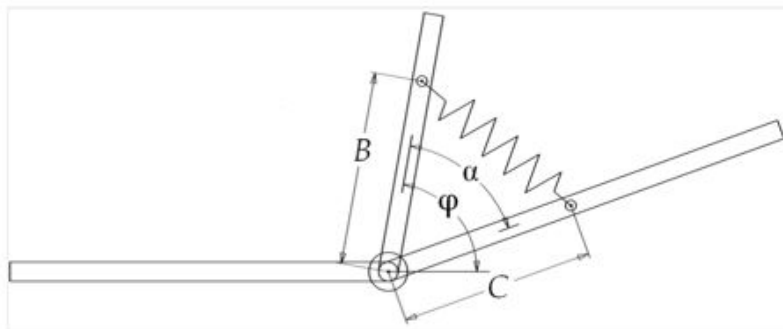


Figura 1.16: Bases del control mecánico de la rigidez. (Extraído de [34])

nueva; consideremos el invento del detector de mentiras. Su inventor Leonard Keeler lo llamó “*cardio-pneumo-psychograph*”, y lo probó en 1938 en Berkeley, California como una máquina que de alguna forma entraba en los sentimientos de los pacientes. Los periodicos de la época lo bautizaron rápidamente como “detector de mentiras” o “polígrafo” (figura 1.17).

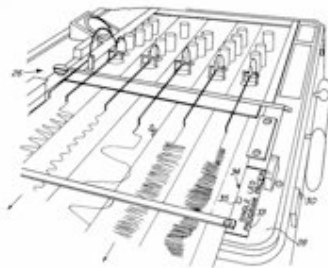


Figura 1.17: Detector de mentiras o polígrafo. Fuente: U.S. Patent 4,333,084. Donald A. Peterson (1980).

En nuestro caso, la adquisición de diferentes bioseñales ligadas al estado físico, neurológico o emocional, es muy importante. En nuestra vida real, nuestro cuerpo reacciona ante los estímulos, tanto externos como internos, que se producen. Algunos de estas reacciones son reflejos inconscientes, reacciones emocionales o cambios en algunos parámetros vitales o en posturas físicas. La identificación y adquisición de estos cambios para estimar las emociones y sentimientos son la base para las técnicas conocidas como *interacción hombre-máquina* (HCI - Human Computer Interface).

La tabla 1.4.2 muestra una serie de magnitudes que pueden ser utilizadas para conocer el estado afectivo, la motivación o el esfuerzo físico de un sujeto.

Como ejemplo, la figura 1.4.2 muestra diferentes sensores de la compañía *g.tec*, como son un sensor de temperatura de la piel (*g.TEMP*sensor), un sensor del esfuerzo respira-

Magnitud	Descripción	Utilidad
sonido voz	Grabar la señal sonora de un sujeto	sistemas de reconocimiento y/o evaluación de emociones. Algunos parámetros son frecuencia, ritmo, melodía, velocidad, etc.
presión sanguínea	Medida de la presión arterial sistólica y diastólica	Es la medida más común para la evaluación de la salud cardíaca. También se relaciona con un incremento de la actividad física.
Electro cardiografía ECG	Grabar la actividad eléctrica del corazón	Se analiza la forma de la señal
Electro encefalografía EEG	Grabar la actividad eléctrica de las estructuras neuronales	Medir los estados de actividad, la influencia de fármacos o el estado del sueño.
Electro miografía EMG	Grabar la actividad eléctrica de los músculos	Medir la actividad de los músculos
Electro oculografía EOG	Grabar los cambios de potencial debidos a movimientos oculares con electrodos de superficie	Detectar movimientos en sujetos con discapacidad severa
Magneto encefalografía MEG	Grabar los campos magnéticos producidos por la activación de las neuronas mediante espiras magnéticas	en experimentación
Pulso	Medida del pulso, usualmente en un dedo o en la oreja	medicina deportiva
Respiración	Medida de la frecuencia y temperatura. Durante la respiración, la temperatura del aire que entra a los pulmones es distinta del aire que sale de los pulmones	medicina deportiva, apnea del sueño
Termografía	Medida de la imagen infraroja del calor radiante	la temperatura corporal se relaciona con cambios en el caudal sanguíneo, actividad física o emocional

Tabla 1.1: Magnitudes para la estimación del estado del sujeto

torio (g.RESPsensor) o un sensor para el pulso cardiaco (g.PULSEsensor). Estos sensores se conectan a un amplificador como el g.USBamp de la misma compañía.



Figura 1.18: Sensores de la compañía [g.tec](#). Extraído de la webpage

1.4.3. Sistemas de control en la interacción hombre-máquina.

Como se comentaba al inicio de este capítulo, los robots han venido realizando gran cantidad de tareas automatizadas que nos han ayudado en nuestro quehacer diario. Para ello los robots debían seguir una determinada trayectoria y en ocasiones realizar una mínima interacción con el entorno. En estas circunstancias, se han venido utilizando clásicamente técnicas de control de posición o de velocidad, o bien un control de fuerzas y una serie de sensores que permitían esa mínima interacción con el entorno (sensores de presencia, de posición articular, de velocidad o de fuerzas). Todo esto ya se conoce de los cursos básicos de robótica. En este apartado vamos a presentar dos técnicas de control más enfocadas hacia la interacción hombre-máquina. En primer lugar se presenta el conocido *Hybrid Position/Force Control*, un principio básico de la planificación de tareas para hacer frente a restricciones geométricas impuestas por el entorno de la tarea. Posteriormente se presentará también un enfoque alternativo para este tipo de interacciones, basada en control de la rigidez (*compliance control*). Este apartado se presenta con mayor profundidad en el ejercicio 2 de este tema y en el anexo correspondiente.

1.4.3.1. Hybrid Position/Force Control

Para comenzar, considérese la tarea diaria mostrada en la *Figura 1.19*, donde se ilustra un robot dibujando una línea con un lápiz sobre una hoja de papel. A pesar de que los seres humanos somos capaces de realizar este tipo de tareas sin tener en cuenta cualquier detalle de control de la mano, el robot necesita comandos de control específicos y una estrategia de control efectiva. Para dibujar la letra *A*, por ejemplo, primero se tiene en cuenta la trayectoria de la punta del lápiz, y se controla la mano para seguir la trayectoria deseada. Al mismo tiempo, hay que adecuar la presión que ejerce el lápiz sobre la hoja de papel. Considerese $o - xyz$ como el sistema de coordenadas, con el eje z perpendicular a la hoja de papel. a lo largo de los ejes x e y , se proporciona comandos de posición para el sistema de control de la mano. Sobre el eje z , se especifica la fuerza a aplicar. En otras palabras, las variables a controlar son distintas entre las direcciones horizontal y vertical. Es decir, se combinan dos tipos de lazos en el sistema de control de la mano, como se muestra en la *Figura 1.20*.

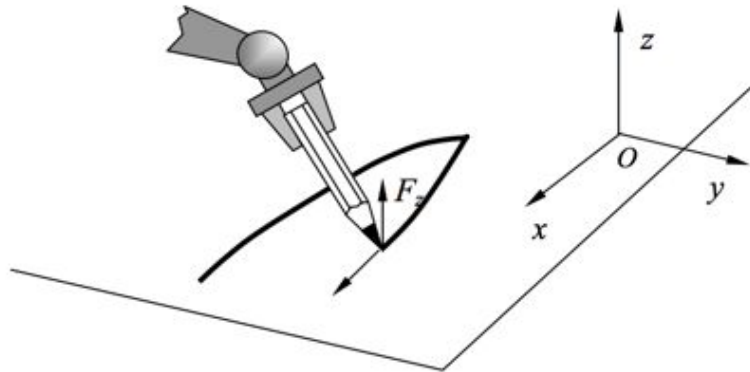


Figura 1.19: Tarea de lápiz sobre hoja de papel

El ejemplo anterior es una de las tareas más simples que ilustran la necesidad de integrar diferentes lazos de control de tal manera que el modo de control es consecuente con las restricciones geométricas impuestas al sistema robótico. Cuando la restricción de la geometría se hace más compleja y el objetivo de la tarea es más complicado, un método intuitivo puede no ser suficiente. A continuación se obtiene un principio general que ayudará a encontrar modos adecuados de control coherente tanto con las restricciones geométricas como con los objetivos de la tarea. Para ello, se considerará una tarea que implique movimiento y contactos en seis dimensiones.

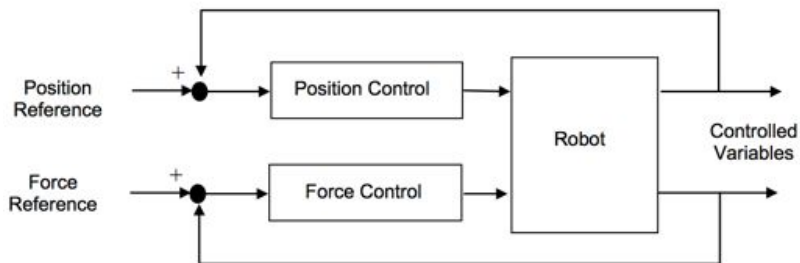
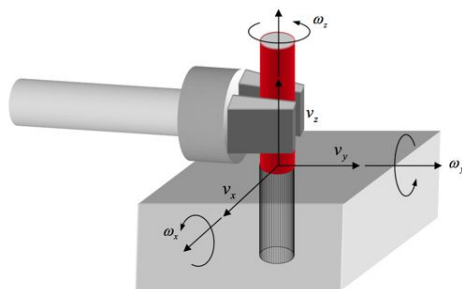


Figura 1.20: Esquema con dos lazos de control

A continuación se muestra una tarea de sacar un palo de un agujero (ver *Figura 1.21*). Supongamos que el palo puede moverse en la dirección vertical sin fricción cuando sale del agujero. Asumamos también que el proceso de la tarea es cuasi-estático, en cuanto a que se desprecia cualquier fuerza de inercia. El sistema de coordenadas $o-xyz$, está referido al espacio de la tarea, como se muestra en la figura. El problema es encontrar un modo de control apropiado para cada uno de los ejes: tres de traslación, y tres de rotación.

Figura 1.21: Ilustración del problema de *peg-in-hole*

La pregunta clave es cómo asignar un modo de control, de posición y de fuerza, a cada uno de los ejes del ejemplo de tal manera que la acción de control no entre en conflicto con las restricciones de geometría ni con la física. Para ello se distinguen entre *Restricciones Naturales* y *Restricciones Artificiales*. Estas restricciones se pueden ver en la *Tabla 1.2*.

	Cinemática	Estática
Restricciones Naturales	$v_x = 0$ $v_y = 0$ $\omega_x = 0$ $\omega_y = 0$	$f_z = 0$ $\tau_z = 0$
Restricciones Artificiales	$v_z = V > 0$ $\omega_z = 0$	$f_x = 0$ $f_y = 0$ $\tau_x = 0$ $\tau_y = 0$

Tabla 1.2: Restricciones naturales y artificiales en el problema del *peg-in-hole*

De la *Figura 1.21* se puede observar claramente que el palo no puede moverse en las direcciones x e y debido a las restricciones geométricas. Por lo tanto, las velocidades en estas direcciones deben ser cero: $v_x = 0$ y $v_y = 0$. Del mismo modo, el palo no puede rotar sobre los ejes x e y , por lo que las velocidades angulares tienen que ser cero: $\omega_x = 0$ y $\omega_y = 0$. Estas condiciones constituyen las restricciones naturales en el dominio cinemático. Las direcciones restantes, son la lineal y angular respecto al eje z . Las velocidades a lo largo de estas dos direcciones pueden ser asignadas arbitrariamente, y pueden controlarse mediante un modo de control en posición. Las referencias de posición de estos lazos de control deberán determinarse de forma que puedan satisfacerse los objetivos de la tarea. Dado que la tarea es para sacar el palo, la velocidad lineal debe dar: $v_z = V > 0$. La orientación del palo sobre el eje z no debe de cambiar. Por lo tanto, la velocidad angular debe dar cero: $\omega_z = 0$. Estas referencias constituyen las restricciones artificiales en el dominio cinemático.

En el ámbito de la estática, las fuerzas y momentos se especifican de tal manera que se satisface la condición cuasi-estática. Esto significa que el movimiento del palo no debe acelerarse con ninguna fuerza no equilibrada, es decir, con ninguna fuerza inercial. Como ya se comentó, se ha asumido que no hay fuerzas de fricción, por lo que no actúa ninguna fuerza resistiva sobre el palo en la dirección que no está restringida por la geometría. Por lo tanto, la fuerza lineal en la dirección del eje z debe ser cero: $f_z = 0$. La rotación sobre el eje z tampoco está restringida, por lo que el par sobre el eje z debe ser cero también: $\tau_z = 0$. Estas condiciones son dictadas por la física, y se denominan como restricciones naturales en el dominio de la estática. Las direcciones restantes están geoméricamente restringidas. En estas direcciones, las fuerzas y pares pueden ser asignados arbitrariamente, y pueden ser controladas con un modo de control de fuerzas. Las referencias para estos lazos de control deben determinarse con el fin de alcanzar los objetivos de la tarea. Para esta tarea, no es necesario empujar el palo en la pared del agujero, ni retorcerlo. Por lo tanto, las referencias se fijan a cero: $f_x = 0$, $f_y = 0$, $\tau_x = 0$, $\tau_y = 0$. Estas restricciones constituyen las restricciones artificiales en el dominio de la estática.

En el ejemplo anterior, está claro que los ejes involucrados en las restricciones naturales y artificiales son ortogonales entre ellos, tanto en los dominios de la cinemática como de la estática. Por otra parte, los ejes involucrados en las restricciones naturales de cinemática y las restricciones artificiales de la estática son las mismas, y que las restricciones naturales de estática y las artificiales de cinemática también son las mismas. Estas relaciones son bastante evidentes en el ejemplo anterior, donde las direcciones de cada eje del sistema de coordenadas está alineado con la dirección a lo largo de la cual está asignado cada modo de control (de posición y de fuerza). Estas propiedades de ortogonalidad son simplemente la consecuencia de los siguientes supuestos:

- Cada eje del sistema de coordenadas debe tener un solo modo de control, ya sea de posición o de fuerza
- El proceso es cuasi-estático y sin fricción
- El movimiento del robot debe ajustarse a las restricciones geométricas

En general, los ejes del sistema de coordenadas no tienen necesariamente coincidentes con la dirección de los modos de control por separado. Sin embargo, las propiedades de ortogonalidad se mantienen, como se verá a continuación.

Sea V^6 un vector espacial de seis dimensiones, y $V_a \subset V^6$ un espacio de movimiento admisible, que es la colección completa de movimientos admisibles conforme a las restricciones geométricas involucradas en la tarea. Sea V_c una restricción espacial, que es el complemento ortogonal al espacio de movimiento admisible:

$$V_c = V_a^\top$$

Sea $F \in V^6$ una fuerza puntual de seis dimensiones que actúa sobre el efector final, y $\Delta p \in V^6$ un desplazamiento infinitesimal del efector final. El trabajo realizado en el efector final viene dado por:

$$\Delta Work = F^T \Delta p$$

Descomponiendo cada vector en uno del espacio de movimiento admisible y en otro del espacio de restricciones se obtiene:

$$\begin{aligned} F &= F_a + F_c & F_a &\in V_a, F_c \in V_c \\ \Delta p &= \Delta p_a + \Delta p_c & \Delta p_a &\in V_a, \Delta p_c \in V_c \end{aligned}$$

y sustituyendo en la ecuación anterior de trabajo:

$$\begin{aligned} \Delta Work &= (F_a + F_c)^T (\Delta p_a + \Delta p_c) = F_a^T \Delta p_a + F_a^T \Delta p_c + F_c^T \Delta p_a + F_c^T \Delta p_c \\ \Delta Work &= F_a^T \Delta p_a + F_c^T \Delta p_c \end{aligned}$$

Siendo por definición $F_a \perp \Delta p_c$, y $F_c \perp \Delta p_a$. Para que el desplazamiento infinitesimal Δp pase a ser un desplazamiento virtual δp , su componente en el espacio de restricciones debe ser cero: $\Delta p_c = 0$. Entonces, $\Delta p_a = \delta p$ pasa a ser un desplazamiento virtual, y la ecuación de trabajo se reduce a un trabajo virtual. Estando el sistema en un estado de equilibrio estático, el trabajo virtual debe desaparecer para un desplazamiento virtual δp_a .

$$\delta Work = F_a^T \delta p_a = 0, \quad \forall \delta p_a$$

Esto implica que cualquier fuerza o momento en el espacio de movimientos admisibles deben ser cero, es decir, las restricciones estáticas naturales:

$$0 = F_a \in V_a$$

Por otra parte, las propiedades de las restricciones estáticas artificiales se pueden derivar de las ecuaciones anteriores de trabajo y trabajo virtual. Para $\Delta p_c = 0$ en la ecuación de trabajo, la condición de equilibrio estático se mantiene, aunque $F_c \in V_c$ tenga un valor arbitrario. Esto implica que para alcanzar el objetivo de la tarea se pueden asignar valores arbitrarios de fuerza y momento en el espacio de restricciones, es decir, en las restricciones estáticas artificiales.

$$\text{arbitrary} : F_c \in V_c$$

Convirtiendo los desplazamientos infinitesimales a velocidades, \dot{p}_a y \dot{p}_c , se puede obtener las restricciones cinemáticas artificiales:

$$\text{arbitrary} : \dot{p}_a \in V_a$$

$$0 = \dot{p}_c \in V_c$$

Arquitectura del sistema de control híbrido posición/fuerza

Se puede construir un sistema de control híbrido posición/fuerza de tal manera que el sistema de control del robot pueda no entrar en conflicto con las restricciones naturales del proceso de la tarea, mientras realiza la tarea para alcanzar su objetivo. En la [Figura 1.22](#) se muestra el diagrama de bloques de un sistema de control híbrido posición/fuerza.

	Cinemática	Estática
Restricciones Naturales	$0 = \dot{p}_c \in V_c$	$0 = F_a \in V_a$
Restricciones Artificiales	<i>arbitrary</i> : $\dot{p}_a \in V_a$	<i>arbitrary</i> : $F_c \in V_c$

Tabla 1.3: Restricciones naturales y artificiales generales

La mitad superior del diagrama es un lazo de control en posición, donde las restricciones cinemáticas artificiales se proporcionan como señales de referencia para el sistema y se comparan con la posición actual del efector final. La mitad inferior del diagrama es el lazo de control de fuerza, donde las restricciones estáticas artificiales se proporcionan como señales de referencia para el lazo de realimentación y se compara con las fuerzas y momentos actuales en el efector final. Nótese que las señales de realimentación están descritas en un sistema de referencia acoplado al efector final.

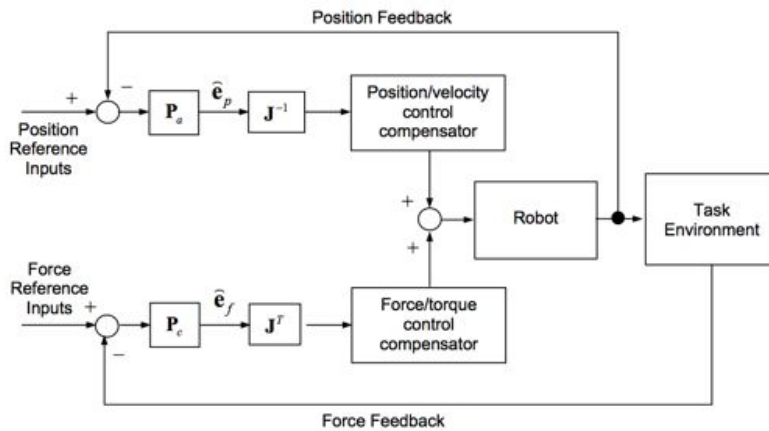


Figura 1.22: Diagrama de bloques de un control híbrido posición/fuerza

Si las señales de realimentación no tienen ruido y el sistema de coordenadas de la tarea está perfectamente alineado con el proceso actual de la tarea, la señal de posición del lazo de realimentación debe estar en el espacio de movimiento admisible, y la fuerza de realimentación debe estar en el espacio de restricciones. Sin embargo, las señales de realimentación están normalmente dañadas por ruidos en el sensor y el sistema de coordenadas puede no estar alineado. Por lo tanto, la señal de posición puede contener algún componente del espacio de restricciones, y alguna fracción de la señal de fuerza puede estar en el espacio de movimiento admisible. Estas componentes se contradicen con las restricciones naturales, y por lo tanto no deben realimentarse a los controles individuales de posición y fuerza. Para filtrar las componentes erróneas, los errores de realimentación se proyectan a su propio subespacio, es decir, el error de posición, e_p , se

asigna al espacio de movimiento admisible, V_a , y la realimentación de error de fuerza, e_f , se asigna al espacio de restricciones, V_c . En el diagrama de bloques se muestran estos filtros como las matrices de proyección P_a y P_c :

$$\hat{e}_p = P_a e_p, \quad \hat{e}_f = P_c e_f$$

Cuando los ejes del sistema de coordenadas de la tarea no están alineadas, individualmente, con las direcciones de los lazos de control de posición y fuerza, las matrices de proyección son diagonales, teniendo como valores 0 y 1 en los componentes de la diagonal. En el caso del problema “*peg-in-hole*” se tiene:

$$P_a = \text{diag}(110110), \quad P_c = \text{diag}(001001)$$

Estos errores de realimentación, \hat{e}_p y \hat{e}_f , están en el sistema de coordenadas general de la tarea, y tienen que convertirse al espacio de la articulación para poder generar comandos de control al actuador. Asumiendo que el vector de error de posición es pequeño y que el robot no está en una configuración singular, la realimentación del error de posición en coordenadas de la articulación viene dado por:

$$e_q = J^{-1} \hat{e}_p$$

donde J es el jacobiano que relaciona las velocidades del efector final en el sistema de coordenadas de la tarea con las velocidades de la articulación. Por otra parte, el error de realimentación de fuerza en las coordenadas de la articulación se obtiene basándose en el principio de dualidad:

$$e_f = J^T \hat{e}_f$$

Estas dos señales de error en las coordenadas de la articulación se combinan después de pasar por una compensación dinámica en los controles individuales de la articulación.

1.4.3.2. Compliance Control

Estrategia de la tarea

En el control híbrido de posición/fuerza, la separación entre los lazos de control de posición y fuerza a través de proyecciones de las señales de realimentación a un espacio de movimiento admisible y a otro de restricciones es explícita. Una alternativa a la arquitectura de separación de espacios es controlar una relación entre posición y fuerza en el espacio de la tarea. *Compliance Control* es una ley de control básica que relaciona el desplazamiento del efector final con la fuerza y momentos que actúan sobre él. En lugar de separar totalmente el espacio de la tarea en dos subespacios de control de posición o fuerza, el *compliance control* reacciona a la fuerza ejercida en el efector final de tal manera que se obtiene una relación funcional (normalmente una relación lineal) entre la fuerza y el desplazamiento:

$$\Delta p = CF$$

donde C es una matriz *Compliance* de $m \times m$, y Δp y F son el desplazamiento y fuerza del efector final representado en un sistema de coordenadas m -dimensional de la tarea. Nótese que la inversa de la matriz C es una matriz de rigidez:

$$K = C^{-1}$$

siempre y cuando la inversa exista.

Los componentes de la matriz *Compliance*, o de la matriz de rigidez, son parámetros de diseños que hay que calcular para cumplir con los objetivos y restricciones de la tarea. Abrir una puerta, por ejemplo, se puede realizar como se ilustra en la *Figura 1.23*. La trayectoria de la manilla de la puerta está geoméricamente retringida por un círculo de radio R centrado en la bisagra de la puerta. El movimiento de la mano del robot debe cumplir con la restricción de la trayectoria de la manecilla, aunque la trayectoria no se sepa con exactitud. El robot no debe romper la manecilla, aunque la trayectoria concebida sea diferente de la trayectoria actual. Este requisito de la tarea se puede cumplir asignando una menor rigidez, es decir, un mayor *Compliance*, en la dirección radial perpendicular a la trayectoria. Como se muestra en la *Figura 1.23*, un pequeño resorte constante genera sólo una pequeña fuerza en respuesta a la discrepancia entre la trayectoria actual de la manecilla y la trayectoria de referencia de la mano del robot. A lo largo de la dirección tangente a la trayectoria de la manecilla, se asigna una gran rigidez. Esto es para forzar a la manecilla a moverse sobre la trayectoria a pesar de la fricción y otras fuerzas resistivas. Por lo tanto la matriz de rigidez viene dada por:

$$K = \begin{pmatrix} k_x & 0 \\ 0 & k_y \end{pmatrix}; \quad k_x \ll 1, k_y \gg 1$$

con referencia al sistema de coordenadas $O-xy$ de la tarea. Usando esta rigidez con la que se sostiene la manecilla, el robot puede abrir la puerta con suavidad y destreza, a pesar de que la trayectoria exacta de la manilla de la puerta no se conoce.

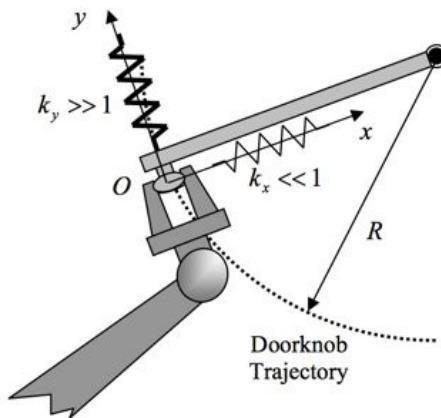


Figura 1.23: Trayectoria de apertura de puerta.

Compliance control

Una vez dada la *Compliance* deseada, consideremos el método para generar la *Compliance* deseada. Hay distintas maneras de sintetizar un sistema de *Compliance control*. El método más sencillo es acomodar las ganancias de realimentación proporcionales de los controles de cada articulación para que se generen las fuerzas deseadas en proporción a las diferencias entre la referencia y la posición actual de los ángulos de las articulaciones. Como se muestra en la *Figura 1.24*, una señal de error de control e_i se genera cuando una fuerza o par perturbadora actúa sobre una articulación. En estado estacionario, hay un balance estático, tal que el par del actuador, τ_i , sea proporcional al error de control e_i para que cancele el par perturbador. La constante de proporcionalidad se calcula mediante la ganancia del lazo de posición k_i cuando la fricción se descuida. Por lo tanto, una rigidez deseada se puede obtener mediante el ajuste de la ganancia del lazo de control.

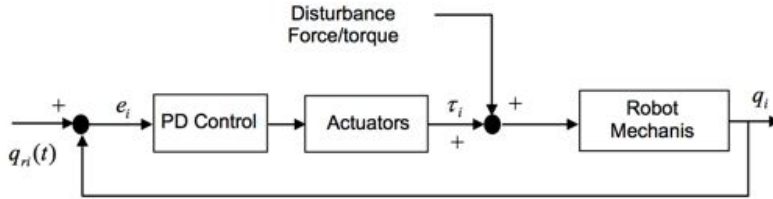


Figura 1.24: Esquema explicativo de *Compliance Control*

La síntesis de *Compliance* es trivial para sistemas de control de una sola articulación. Para robots de n grados de libertad, se deben coordinar varios lazos de realimentación. Ahora, consideremos cómo generar una matriz $m \times m$ de *Compliance* o de rigidez especificada en el efector final modificando las ganancias de las realimentaciones.

Teorema 1.4.1 *Teorema*

Sea J la Jacobiana que relaciona la velocidad del efector final $\dot{p} \in R^{m \times 1}$ con las velocidades de las articulaciones $\dot{q} \in R^{n \times 1}$; y que $\tau \in R^{n \times 1}$ los pares de las articulaciones asociadas con las coordenadas articulares q . Sea $\Delta p \in R^{m \times 1}$ un vector $m \times 1$ del desplazamiento del efector final medida de una posición nominal \bar{p} , y que $F \in R^{m \times 1}$ sea la fuerza del efector final asociada a su desplazamiento Δp . Sea K_p la matriz de rigidez del efector final deseada definida como:

$$F = K_p \Delta p$$

La condición necesaria para la ganancia de realimentación de cada articulación K_q para generar la matriz de rigidez del efector final K_p viene dada por:

$$K_q = J^T K_p J$$

asumiendo que no hay fricción en las articulaciones ni en los mecanismos de enlace.

Prueba Usando la Jacobiana y el principio de dualidad:

$$\tau = J^T F = J^T K_p \Delta p = J^T K_p J \Delta q$$

Usando la ecuación anterior de K_q , la relación se reduce a:

$$\tau = K_q \Delta q$$

Esto implica que K_q es la matriz de ganancias de realimentación de las articulaciones.

Como ejemplo, considérese un robot planar de dos brazos, como se muestra en la *Figura ??*. Obteniendo la matriz de ganancias de realimentación de las articulaciones K_p :

$$K_p = \begin{pmatrix} k_1 & 0 \\ 0 & k_2 \end{pmatrix}$$

Asumiendo que la longitud de los eslabones es 1 para ambos, la Jacobiana viene dada por:

$$J = \begin{pmatrix} -s_1 & -s_2 \\ c_1 & c_2 \end{pmatrix}$$

Teniendo en cuenta la ecuación anterior de K_q :

$$K_q = J^T K_p J = \begin{pmatrix} k_{q1} & k_{q3} \\ k_{q3} & k_{q2} \end{pmatrix}$$

donde:

$$k_{q1} = k_1 s_1^2 + k_2 c_1^2$$

$$k_{q2} = k_1 s_2^2 + k_2 c_2^2$$

$$k_{q3} = k_1 s_1 s_2 + k_2 c_1 c_2$$

Nótese que la matriz de ganancias de realimentación articular K_q es simétrica y que la matriz K_q degenera cuando el robot está en una configuración singular. Si está en cualquier otra configuración, entonces:

$$\Delta p = J \Delta q = J K_q^{-1} \tau = J K_q^{-1} J^T F = J (J^T K_p J)^{-1} J^T F$$

$$\Delta p = K_p^{-1} F = C F$$

Por lo tanto, la ganancia de realimentación articular proporciona la rigidez requerida en la ecuación anterior de K_p .

1.5. ROBOTICA MEDICA APLICADA

En esta primera sección de ROBOTICA MEDICA APLICADA se presentan dos ejemplos sencillos; en el primero de ellos se utiliza el software de renderizado gráfico 3D Blender[©] (www.blender.org) para el diseño de un robot de rehabilitación de miembro superior, y en el segundo se muestra un ejemplo con Matlab[®] para la implementación de un control de impedancia en un robot de 2 grados de libertad. Ambos ejemplos se presentan aquí de forma resumida, mientras que una versión más extensa y los ficheros necesarios para su solución se encuentran en el dvd-live adjunto al libro.

1.5.1. Ejemplo de cálculo del espacio de trabajo de un robot de rehabilitación con Blender[©]

En este ejemplo se hace uso del software de renderizado 3D Blender (www.blender.org) para realizar los primeros pasos en el diseño de un robot para rehabilitación de miembro superior.

Como material adicional para este ejercicio se necesita el software libre Blender (www.blender.org) y se recomienda la lectura del manual básico de uso de BLENDER. Para ello se ha hecho referencia al excelente trabajo realizado por Dr. Carlos González, y que se encuentra en el libro “[Aprende en 24 horas Blender & Yafaray](#)” y en el libro “[Fundamentos de síntesis de imagen 3D](#)”. Aunque se recomienda realizar todos los ejercicios presentes en el libro, es altamente aconsejable realizar y comprender los ejercicios 8 “Animación basada en esqueletos” y 9. “Esqueletos avanzados”. Estas prácticas nos servirán para modelar los problemas cinemáticos en los simuladores que se desarrollen.

Este ejercicio resuelto plantea la siguiente problemática:

En robótica de rehabilitación, el trabajo cooperativo entre el robot y el paciente implica un uso compartido del espacio de trabajo de ambos. Imaginemos un sistema robótico compuesto por dos brazos seriales que trabajan conjuntamente en el rehabilitación del miembro superior de un paciente (ver figura 1.3). Uno de los brazos (A) sujeta la mano del paciente. Se trata de un brazo serial de configuración SCARA y con 3 grados de libertad. El otro brazo (B) tiene también 3 grados de libertad y sujeta el codo del paciente. El sistema está pensado para ser capaz de guiar al paciente a realizar una diagonal (D1) del método Kabat de rehabilitación. Brevemente se trata de guiar el brazo del paciente desde una posición en reposo lateral al torso hasta cruzarlo por encima del hombro contrario. Una más extensa explicación de este método de rehabilitación se encuentra en el artículo anexo a esta práctica. Uno de los problemas que plantea el diseño de una solución robótica para este problema es las posibles colisiones que se pueden producir durante el movimiento, tanto entre los propios brazos del sistema como entre el robot y el paciente. (En este tutorial no se ha considerado la ubicación del fisioterapeuta).

Las siguientes imágenes 1.25 y 1.26 ilustran el problema planteado:

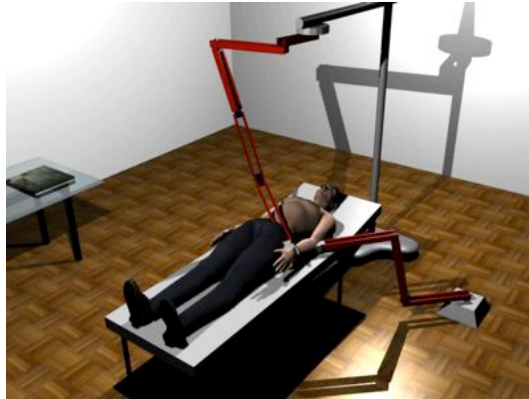


Figura 1.25: Sistema robótico de dos brazos para rehabilitación de miembro superior

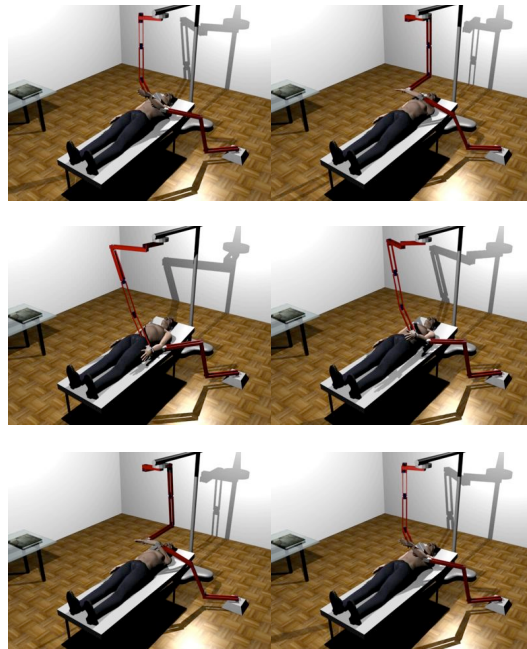


Figura 1.26: Diagonal D1 en movimientos de flexión y extensión

Disponer de una herramienta de visualización 3D con capacidades cinemáticas como BLENDER puede ser de gran ayuda para el diseño y optimización del problema planteado.

En el fichero *ejercicio1.blend* se encuentra una solución al problema planteado resuelta con BLENDER. En esta solución anexa (figura 1.27), se puede observar que en los

frames 1-40 se ha modelado el primer movimiento del brazo del paciente, mientras que en los frames 41-80 se ha modelado el movimiento de retorno. Una vez se tiene el movimiento del brazo del paciente, se han utilizado las herramientas de cinemática inversa de BLENDER para solucionar cada una de las cadenas cinemáticas del sistema robótico. De esta forma se puede observar gráficamente las posibles colisiones o interferencias entre los distintos actores. La capacidad de un diseño paramétrico de BLENDER para definir las dimensiones de los eslabones permite la programación de un script para encontrar un diseño óptimo del sistema.

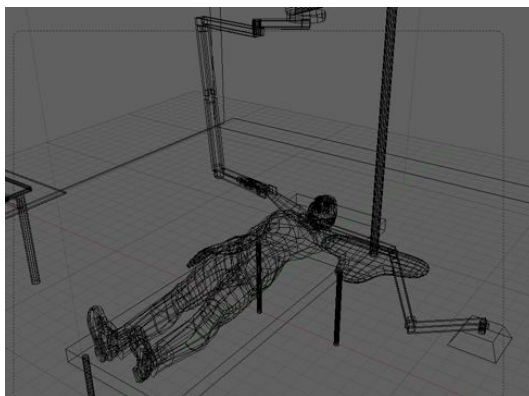


Figura 1.27: Imagen del fichero ejercicio1.blender de solución del ejercicio

El fichero *ejercicio1.blend* con la solución se encuentra en el cd-live adjunto.

1.5.2. Ejemplo de control de impedancia con Matlab[®]

En esta sección se implementará un control de impedancia para controlar las fuerzas de interacción entre el robot y el entorno. El desarrollo más completo del ejercicio con detalles de los pasos dados se encuentra en el documento *ejercicio2.pdf* adjunto del cd-live.

El modelo del robot utilizado será un robot planar de dos grados de libertad. Para simular la interacción con el entorno se implementará el choque del extremo del robot contra un obstáculo rígido. Para evaluar el control de impedancia y compararlo con un control en posición se medirá la distancia de penetración del extremo del robot en el obstáculo con un modelo masa muelle.

De forma resumida, los pasos a dar para la realización de este ejercicio son:

1. Creación del modelo mecánico de un robot planar de 2 grados de libertad con Simmechanics. Las figuras 1.28 y 1.29 muestran el modelo utilizado. En el cd adjunto se pueden probar los modelos de 1 y 2 eslabones que se encuentran en los ficheros *UnEslabon.mdl* y *DosEslabones.mdl*. Al ejecutar la simulación de estos ficheros debemos observar a los eslabones en caída libre sin fricción, y por tanto con un movimiento infinito, ya que no se ha modelado la fricción en las articulaciones.

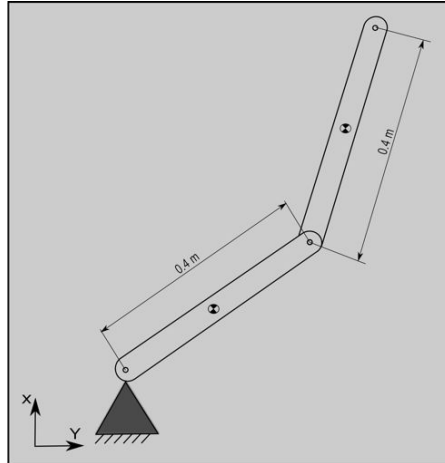


Figura 1.28: Croquis del Brazo Robótico de 2 GDL

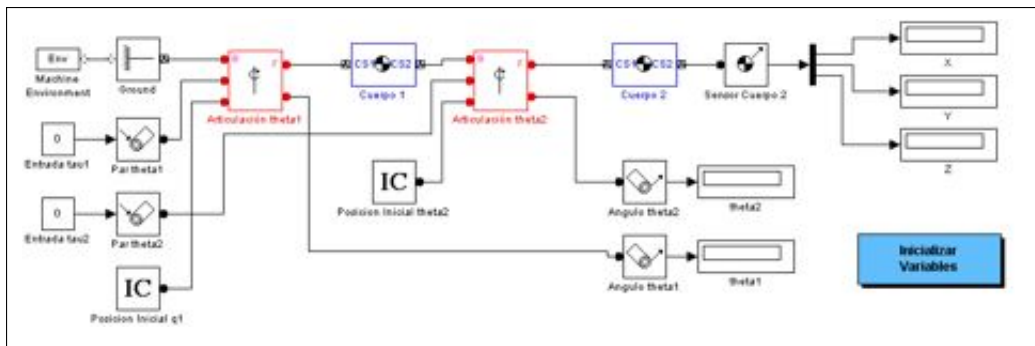


Figura 1.29: Bloques SimMechanics del Brazo Robótico de 2 GDL

- Obtención de un modelo dinámico y de los parámetros del mismo. El modelo dinámico utilizado es el de Lagrange-Euler, y que para un robot planar de dos grados de libertad es:

$$M(\theta) \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + B(\theta, \dot{\theta}) + G(\theta) = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

La obtención de los valores de los parámetros de esta ecuación se detalla en el ejemplo y en el fichero *dinamica2GDL.m*. Ejecutando este fichero se obtienen los valores de los parámetros gracias al uso de `symbolic` de Matlab®. Con este modelo se puede utilizar la técnica de desacoplo por inversión del modelo para desacoplar el sistema del robot de 2 gdl.

```

>> dinamica2GDL
>> G(1)

ans =
5
m2*((981*a2*cos(theta1 + theta2))/200 + (981*a1*cos(theta1))/100)
+ (981*a1*m1*cos(theta1))/200

>> G(2)
10
ans =

(981*a2*m2*cos(theta1 + theta2))/200

15
>> B(1)

ans =

-a1*a2*m2*theta2p*sin(theta2)*(2*theta1p + theta2p)
20

>> B(2)

ans =

25
a1*a2*m2*theta1p^2*sin(theta2)

>>

```

3. Obtener el modelo del entorno según se explica en el documento adjunto. Para ello se utiliza un sencillo modelo con la distancia de penetración del extremo del robot en el objeto rígido.
4. Obtener el controlador dinámico según se explica en el documento adjunto. Se trata de, utilizando el modelo anterior, desacoplar el sistema e implementar un regulador PD para cada grado de libertad del robot. De esta forma se controla el robot y se le comanda colocarse en posición horizontal. La figura 1.30 muestra el esquema de bloques correspondiente al fichero *controlDinamicoFuerzas.mdl*.
5. Obtener un controlador de impedancia según se explica en el documento adjunto. Para ello se utiliza el esquema del fichero *ControlImpedancia.mdl* donde al esquema del control anterior se le ha añadido un nuevo esquema con un control de impedancia. Al ejecutar la simulación se puede observar (figura 1.31) el efecto del segundo lazo de control en fuerzas, ya que se observa una comparación del comportamiento del robot con y sin control de impedancias. Del mismo modo se puede graficar las fuerzas de interacción (modeladas como un muelle con la distancia de penetración) y observar la disminución de los valores de dichas fuerzas (figura 1.32).

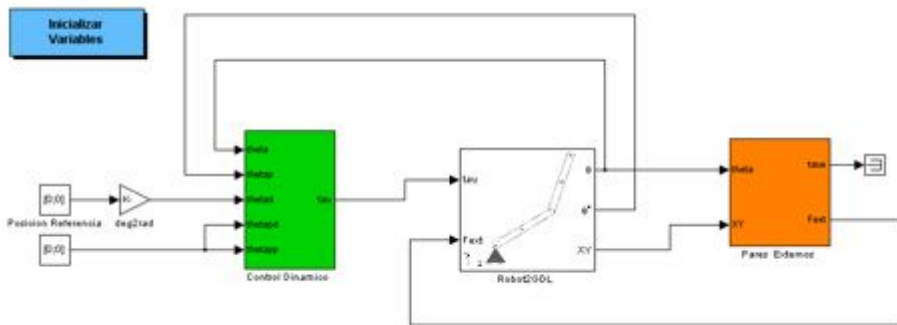


Figura 1.30: Modelo del entorno

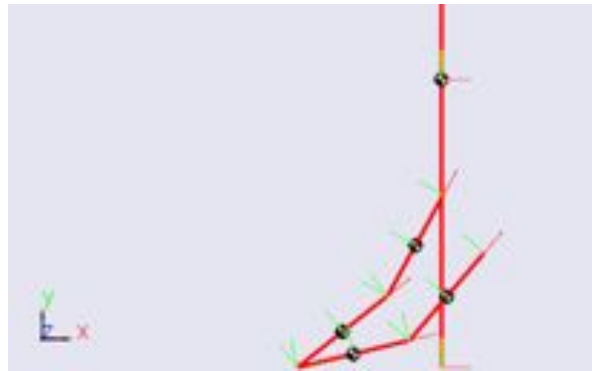


Figura 1.31: Comparativa Control de Impedancia vs Sin Control de Impedancia

1.6. Resumen del capítulo

En este capítulo se ha presentado una introducción a los robots médicos. Para ello se ha utilizado una clasificación por áreas de aplicación, de forma que se han comentado aplicaciones para el ámbito quirúrgico, ámbito de rehabilitación y ámbito asistencial.

Para el alumno es importante remarcar los conceptos presentados en la sección 1.4, de forma que quede claro la diferencia en los paradigmas de un robot industrial y un robot médico que interactúa con un hombre. Los temas de *soft robotics* y de diseño de actuadores y esquemas de control seguros cobran especial importancia para lograr una correcta interacción hombre-máquina.

En el apartado ROBOTICA MEDICA APLICADA se muestran dos ejemplos completos, uno para el análisis cinemático del movimiento de un robot de rehabilitación que tiene que evolucionar por un espacio de trabajo compartido con un fisioterapeuta, y un segundo ejemplo del efecto de un control de impedancia cuando se produce una interacción del robot con su entorno.

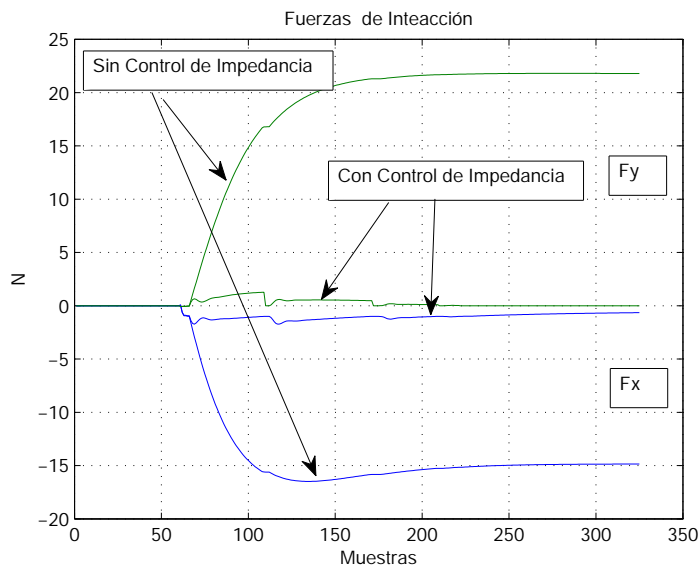


Figura 1.32: Comparativa en Magnitud. Control de Impedancia vs Sin Control de Impedancia

Bibliografía para ampliar

Handbook of robotics B. Siciliano, O. Khatib. Springer. El capítulo 52 presenta una visión general de “Medical Robotics and Computer-Integrated Surgery”, que aunque más centrado en robótica quirúrgica, constituye un excelente documento de revisión del estado de arte. Los capítulos 53 “Rehabilitation and Health care robotics” y 54 “Domestic robotics” complementan esta visión.

Robotica Aplicada R. Saltaren, L. Puglisi, J.M. Sabater-Navarro, F. Thomas. Este libro está concebido para la docencia de robots seriales y paralelos utilizando Matlab. Los capítulos 1 y 2 presentan problemas, retos y soluciones tecnológicas generales de la robótica que son aplicables a los conceptos presentados en este capítulo.

Soft Robotics Soft Robotics: form torque feedback-controlled lightweight robots to intrinsically compliant systems. A. Albu-Schäffer, O. Eiberger, M. Grebenstein, S. Haddain, C. Ott, T. Winbock, S. Wolf, G. Hirzinger. IEEE Robotics & Automation Magazine, September 2008. Se trata de un artículo sobre control de robots.

Compliant Actuator Designs Van Ham et al. IEEE Robotics & Automation Magazine, pp.81-94. Sept.2009. Este magnifico artículo describe los trabajos realizados en Universidad Vrije de Bruselas sobre nuevos diseños de actuadores.

Capítulo 2

Fundamentos de robótica aplicada

“Los robots del futuro no pueden funcionar sin las clavijas del pasado.”

Popular

Indice

2.1. Introducción	44
2.2. Herramientas para el análisis de robots	44
2.2.1. Localización de un cuerpo rígido en \mathbb{R}^3	44
2.2.2. Representación de la posición en \mathbb{R}^3	44
2.2.3. Representación de la orientación en \mathbb{R}^3	45
2.2.4. Representación conjunta de la localización de un cuerpo en \mathbb{R}^3	54
2.2.5. Convenio de Denavit-Hartenberg	57
2.3. Cinemática de robots	60
2.3.1. Cinemática directa	61
2.3.2. Cinemática inversa	62
2.3.3. Cinemática diferencial	62
2.4. Dinámica de robots	64
2.5. ROBOTICA MEDICA APLICADA	66
2.5.1. Ejemplo de análisis de la cinemática un robot serial de 6 GDL	66
2.5.2. Ejemplo de análisis de la dinámica de un robot 6UPS con SimMechanics	69
2.6. Resumen del capítulo	74
2.7. Ejercicios	76
Bibliografía para ampliar	77

2.1. Introducción

Desde un punto de vista docente, este capítulo debe utilizarse para aquellos alumnos de master que no hayan cursado asignaturas de robótica básica durante su formación de grado. En él pueden encontrar justificación a los desarrollos posteriores que se utilizan en el análisis de algunos robots ejemplo.

En este segundo capítulo se resumen las herramientas necesarias para poder seguir los desarrollos de los siguientes apartados de ROBOTICA MEDICA APLICADA. Este capítulo no pretende cubrir el temario de un curso básico de robótica, y se remite al lector interesado a consultar la bibliografía recomendada, [70], donde podrá encontrar ampliación de los contenidos aquí mostrados.

2.2. Herramientas para el análisis de robots

2.2.1. Localización de un cuerpo rígido en \mathbb{R}^3

En un instante determinado, un cuerpo rígido se encuentra localizado en el espacio tridimensional cuando se conocen la posición de un punto perteneciente al mismo y la orientación de un sistema de referencia con origen en dicho punto. Además se puede considerar la localización actual de un cuerpo como el resultado de la evolución de las coordenadas de posición y orientación debidas al movimiento del cuerpo. Por tanto, la definición de la localización de un cuerpo en el espacio viene dada por la transformación sufrida para llevar dicho cuerpo desde una posición inicial (definida por el sistema considerado inercial) hasta una posición final (definida por la posición actual del cuerpo).

Se define una transformación de un cuerpo rígido como una aplicación $g : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ que cumpla las condiciones:

1. $\|p(t) - q(t)\| = cte \quad \forall t \quad \forall p, q \in \mathbb{R}^3$
2. Se conserva el producto vectorial $g_*(v \times w) = g_*(v) \times g_*(w) \quad \forall v, w \in \mathbb{R}^3$

donde $g_*(\cdot)$ es la acción producida por g sobre vectores definidos por dos puntos ($g_*(v) = g(q) - g(p)$).

Esta definición implica que los sistemas de referencia ortogonales sobre los que se aplica una transformación continúan siendo ortogonales.

2.2.2. Representación de la posición en \mathbb{R}^3

Para la localización espacial de puntos necesitaremos tres coordenadas independientes. La representación más natural es la representación cartesiana, aunque existen otras representaciones que son igualmente válidas, como las coordenadas cilíndricas o esféricas.

Normalmente los sistemas de referencia se definen mediante tres ejes perpendiculares entre sí con un origen común. En este libro, y mientras no se indique lo contrario, se utilizan *triedros ortonormales a derechas* $(x, y, z) \in \mathbb{R}^3$ de forma que se cumplan las

relaciones siguientes:

$$\mathbf{z} = \mathbf{x} \times \mathbf{y}$$

$$\mathbf{y} = \mathbf{z} \times \mathbf{x}$$

$$\mathbf{x} = \mathbf{y} \times \mathbf{z}$$

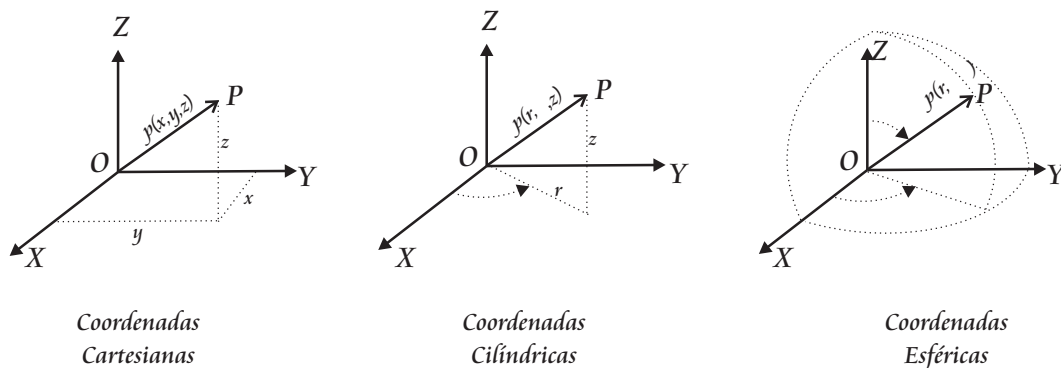


Figura 2.1: Representación de la posición en \mathbb{R}^3

En la figura 2.1 se muestra la representación de la posición en \mathbb{R}^3 utilizando coordenadas cartesianas, cilíndricas y esféricas:

Coordenadas cartesianas. Un punto P queda representado en el sistema $OXYZ$ por las proyecciones sobre los ejes coordenados del vector \mathbf{p} que une el origen O con el punto P .

Coordenadas cilíndricas. Un punto P queda representado en el sistema $OXYZ$ por las coordenadas polares (r, θ) de la proyección del vector \mathbf{p} en el plano OXY y la coordenada z que marca la cota sobre el plano OXY .

Coordenadas esféricas. Un punto P queda representado en el sistema $OXYZ$ por el módulo del vector \mathbf{p} y dos ángulos: el ángulo que forma la proyección del vector \mathbf{p} en el plano OXY con el vector X (θ) y el ángulo que forma el vector \mathbf{p} con el eje Z (ϕ).

2.2.3. Representación de la orientación en \mathbb{R}^3

La representación de la orientación de un cuerpo rígido en el espacio admite diferentes expresiones. La orientación en el espacio tridimensional viene definida por tres grados de libertad, luego será necesario un mínimo de tres parámetros linealmente independientes para poder representar una orientación.

Sean dos sistemas de referencia, $\{A\}$ definido por $OXYZ$ y $\{B\}$ definido por $OUVW$, cuyos orígenes son coincidentes (figura 2.2). Se van a utilizar las siguientes formulaciones para representar la orientación del sistema $OUVW$ sobre el sistema $OXYZ$.

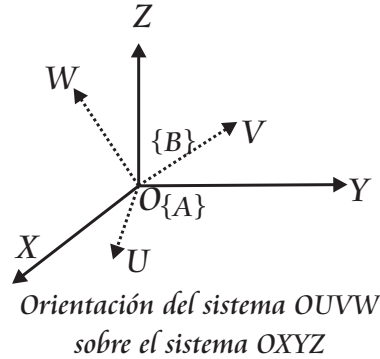


Figura 2.2: Orientación de un sistema de referencia

2.2.3.1. Matriz de rotación

En robótica es común el uso de matrices de rotación, debido a la comodidad que presenta el uso del algebra matricial para trabajar con orientaciones espaciales. La matriz de rotación es una matriz 3x3 ortogonal de cosenos directores que define la orientación del sistema móvil $\{B\}$, definido por los vectores $OUVW$, respecto al sistema considerado inercial $\{A\}$, definido por los vectores $OXYZ$, y que tiene la forma siguiente:

$${}^A\mathbf{R}_B = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = [\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3] \quad (2.1)$$

2.2.3.2. Obtención de la Matriz de rotación como matriz de cosenos directores

Una de las expresiones de la matriz de rotación más cómoda de desarrollar es su expresión en función de los cosenos directores que definen el giro del sistema. La figura 2.3 muestra la obtención de la matriz de rotación que define el giro de un ángulo α de un sistema plano OUV (definido por el origen O y los versores \mathbf{u} y \mathbf{v}) sobre el sistema OXY (definido por el origen O y los versores \mathbf{i} y \mathbf{j}).

Un punto P del plano se puede representar en ambos sistemas como:

$$\begin{aligned} \mathbf{p}_{xy} &= [p_x, p_y]^T = p_x \cdot \mathbf{i} + p_y \cdot \mathbf{j} \\ \mathbf{p}_{uv} &= [p_u, p_v]^T = p_u \cdot \mathbf{u} + p_v \cdot \mathbf{v} \end{aligned}$$

dado que el punto es el mismo expresado en un sistema u otro

$$p_{xy} = p_{uv}$$

$$p_x \cdot \mathbf{i} + p_y \cdot \mathbf{j} = p_u \cdot \mathbf{u} + p_v \cdot \mathbf{v}$$

Multiplicando escalarmente por el versor \mathbf{i} esta ecuación, y análogamente por \mathbf{j} ,

$$\begin{aligned} p_x + 0 &= p_u \cdot \mathbf{u} \cdot \mathbf{i} + p_v \cdot \mathbf{v} \cdot \mathbf{i} \\ 0 + p_y &= p_u \cdot \mathbf{u} \cdot \mathbf{j} + p_v \cdot \mathbf{v} \cdot \mathbf{j} \end{aligned}$$

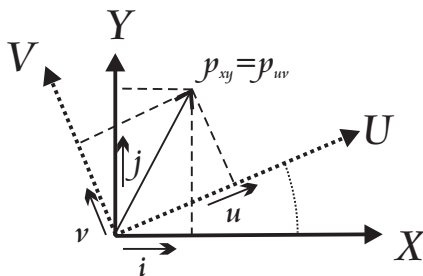
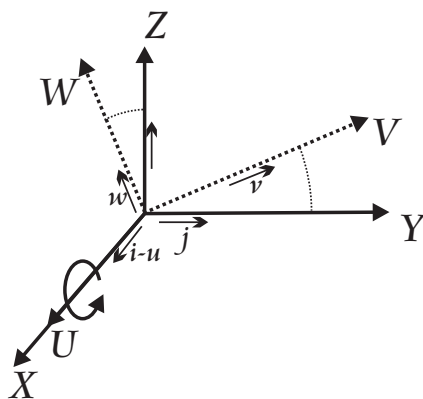


Figura 2.3: Rotación básica en el plano

Figura 2.4: Rotación básica alrededor del vector OX

es decir,

$$\begin{bmatrix} p_x \\ p_y \end{bmatrix} = \mathbf{R} \begin{bmatrix} p_u \\ p_v \end{bmatrix}$$

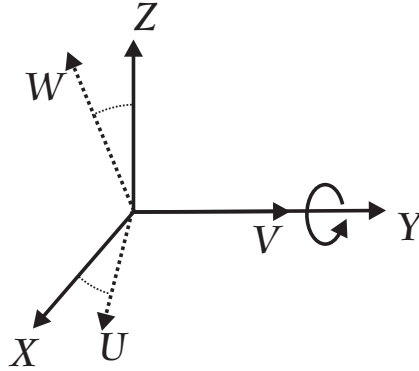
donde

$$\mathbf{R} = \begin{bmatrix} \mathbf{i} \cdot \mathbf{u} & \mathbf{i} \cdot \mathbf{v} \\ \mathbf{j} \cdot \mathbf{u} & \mathbf{j} \cdot \mathbf{v} \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix}$$

Es fácil comprobar que la matriz definida es una matriz ortonormal ($\mathbf{R}^{-1} = \mathbf{R}^T$), y que en el caso de $\alpha = 0$ corresponde a la matriz identidad.

En el caso de un espacio tridimensional, el razonamiento es análogo. La figura 2.4 muestra el giro de un ángulo α sobre el eje OX de un sistema $OUVW$ con respecto a un sistema $OXYZ$, de forma que un punto P se representaría en ambos sistemas como,

$$\begin{aligned} \mathbf{p}_{xyz} &= [p_x, p_y, p_z]^T = p_x \cdot \mathbf{i} + p_y \cdot \mathbf{j} + p_z \cdot \mathbf{k} \\ \mathbf{p}_{uvw} &= [p_u, p_v, p_w]^T = p_u \cdot \mathbf{u} + p_v \cdot \mathbf{v} + p_w \cdot \mathbf{w} \end{aligned}$$

Figura 2.5: Rotación básica alrededor del vector OY

es decir,

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \mathbf{R} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix}$$

donde

$$\mathbf{R} = \begin{bmatrix} \mathbf{i} \cdot \mathbf{u} & \mathbf{i} \cdot \mathbf{v} & \mathbf{i} \cdot \mathbf{w} \\ \mathbf{j} \cdot \mathbf{u} & \mathbf{j} \cdot \mathbf{v} & \mathbf{j} \cdot \mathbf{w} \\ \mathbf{k} \cdot \mathbf{u} & \mathbf{k} \cdot \mathbf{v} & \mathbf{k} \cdot \mathbf{w} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} = \mathbf{R}(\mathbf{x}, \alpha) \quad (2.2)$$

En el caso de que el giro fuese de un ángulo ϕ sobre el eje OY (figura 2.5) la expresión de la matriz \mathbf{R} quedaría como,

$$\mathbf{R}(\mathbf{y}, \phi) = \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix} \quad (2.3)$$

Por último, en el caso de que el giro fuese de un ángulo θ sobre el eje OZ (figura 2.6) la expresión de la matriz \mathbf{R} quedaría como,

$$\mathbf{R}(\mathbf{z}, \theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

Las ecuaciones (2.2), (2.3) y (2.4) constituyen las expresiones de las tres matrices básicas de rotación.

De esta definición de la matriz de rotación como matriz de cosenos directores se extrae una interesante interpretación geométrica. Por inspección directa de la matriz \mathbf{R} se observa que las columnas de dicha matriz representan los versores $[\mathbf{i}, \mathbf{j}, \mathbf{k}]$ expresados en el sistema $OUVW$, mientras que leída por filas, las filas de la matriz \mathbf{R} representan a los versores $[\mathbf{u}, \mathbf{v}, \mathbf{w}]$ expresados en el sistema $OXYZ$. Esta rápida interpretación geométrica permite trabajar fácilmente con las matrices de rotación.

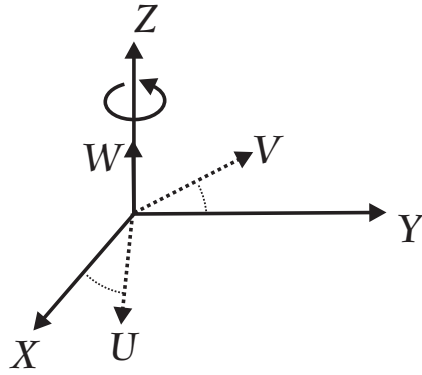


Figura 2.6: Rotación básica alrededor del vector OZ

2.2.3.3. Obtención de la Matriz de rotación como par de rotación. Fórmula de Rodrigues

Teorema 2.2.1 *Teorema de Euler* Cualquier orientación $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ es equivalente a una rotación de un ángulo $\theta \in [0, 2\pi)$ alrededor de un eje fijo $\mathbf{u} \in \mathbb{R}^3$.

De acuerdo con el teorema de Euler, otra forma de expresar una rotación es mediante el par de rotación (θ, \mathbf{u}) , que se define como el ángulo θ girado alrededor de un vector genérico $\mathbf{u} = [u_x, u_y, u_z]^T$, de forma que el sistema $OXYZ$ se transforma en el sistema $OUVW$ (figura 2.7).

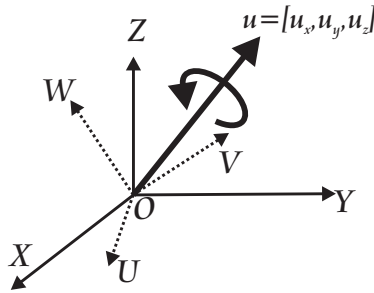


Figura 2.7: Representación de orientación mediante el par de rotación

Para obtener una expresión que defina esta transformación en función de los parámetros del par de rotación (θ, \mathbf{u}) , utilizamos la representación de la figura 2.8. Sea \mathbf{s}' el vector de posición del punto P' cuyas coordenadas se asumen fijas en $OUVW$. La figura 2.8 muestra la rotación de un ángulo θ alrededor del eje \overline{OC} . Como resultado de esta rotación, el punto P' se traslada al punto P . El vector de posición del punto P en el sistema $OXYZ$ se denomina \mathbf{s} . El vector $\Delta \mathbf{s}$ define el cambio del vector de posición del punto P' , es decir:

$$\mathbf{s} = \mathbf{s}' + \Delta\mathbf{s} \quad (2.5)$$

donde

$$\Delta\mathbf{s} = \mathbf{b}_1 + \mathbf{b}_2 \quad (2.6)$$

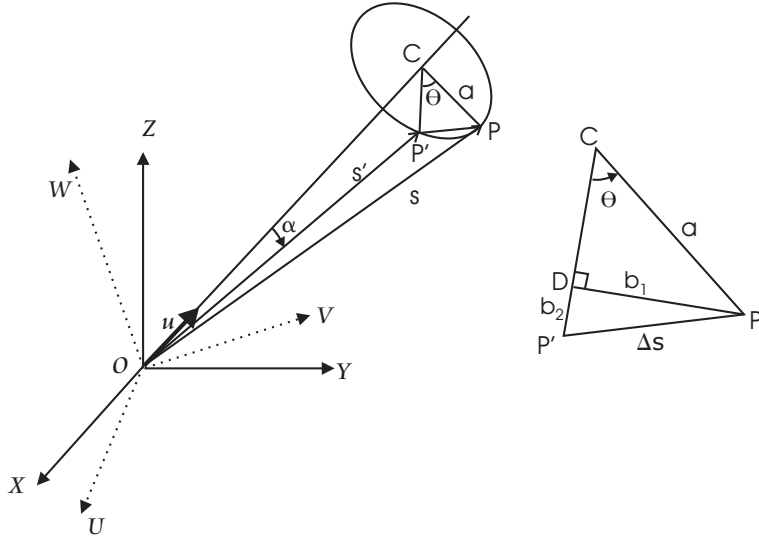


Figura 2.8: Rotación finita de un punto P'

Donde el vector \mathbf{b}_1 es perpendicular al plano OCP' y por tanto tiene la dirección de $(\mathbf{u} \times \mathbf{s}')$ donde \mathbf{u} es el vector unitario a lo largo del eje de rotación \overline{OC} . La magnitud del vector \mathbf{b}_1 viene dada por:

$$|\mathbf{b}_1| = a \sin \theta \quad (2.7)$$

donde $a = |\mathbf{s}'| \sin \alpha = |\mathbf{u} \times \mathbf{s}'|$, luego ($\mathbf{b}_1 = \text{módulo} \times \text{vector unitario}$):

$$\mathbf{b}_1 = (a \sin \theta) \frac{\mathbf{u} \times \mathbf{s}'}{|\mathbf{u} \times \mathbf{s}'|} = (\mathbf{u} \times \mathbf{s}') \sin \theta \quad (2.8)$$

El vector $\mathbf{b}_2 = \overline{DP'}$ = $\overline{CP'} - \overline{CD}$ tiene una magnitud dada por:

$$|\mathbf{b}_2| = a - a \cos \theta = (1 - \cos \theta)a = 2a \sin^2 \left(\frac{\theta}{2} \right) \quad (2.9)$$

Donde se ha utilizado la igualdad trigonométrica $(-2 \sin^2(\theta/2) = \cos \theta)$. Por otra parte, \mathbf{b}_2 es perpendicular tanto a \mathbf{u} como a $\overline{DP'}$, cuya dirección viene dada por el vector unitario $\frac{\mathbf{u} \times \mathbf{s}'}{a}$, por tanto:

$$\mathbf{b}_2 = 2a \sin^2 \left(\frac{\theta}{2} \right) \frac{\mathbf{u} \times (\mathbf{u} \times \mathbf{s}')}{a} = 2 [\mathbf{u} \times (\mathbf{u} \times \mathbf{s}')] \sin^2 \left(\frac{\theta}{2} \right) \quad (2.10)$$

luego

$$\mathbf{s} = \mathbf{s}' + (\mathbf{u} \times \mathbf{s}') \sin \theta + 2 [\mathbf{u} \times (\mathbf{u} \times \mathbf{s}')] \sin^2 \left(\frac{\theta}{2} \right) \quad (2.11)$$

Por otra parte, se define $so(n)$ como el espacio de las matrices simétricas skew, cuya definición es:

$$so(n) = \{S \in \mathbb{R}^{n \times n} : S^T = -S\} \quad (2.12)$$

En este libro utilizaremos el operador ($\tilde{\cdot}$) para definir la matriz skew obtenida a partir del vector \mathbf{u} , de forma que si $\mathbf{u} = (u_1, u_2, u_3)$, se define:

$$\tilde{\mathbf{u}} \equiv \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \quad (2.13)$$

lo que da lugar a la identidad $\mathbf{u} \times \mathbf{s}' = \tilde{\mathbf{u}}\mathbf{s}' = -\tilde{\mathbf{s}}'\mathbf{u}$, siendo u_1, u_2 y u_3 las componentes del vector unitario \mathbf{u} y s'_1, s'_2, s'_3 las componentes del vector \mathbf{s}' . La ecuación (2.11) puede describirse como:

$$\mathbf{s} = \mathbf{s}' + \tilde{\mathbf{u}}\mathbf{s}' \sin \theta + 2(\tilde{\mathbf{u}})^2\mathbf{s}' \sin^2 \left(\frac{\theta}{2} \right) = \left[\mathbf{I} + \tilde{\mathbf{u}} \sin \theta + 2(\tilde{\mathbf{u}})^2 \sin^2 \left(\frac{\theta}{2} \right) \right] \mathbf{s}' \quad (2.14)$$

donde \mathbf{I} es una matriz identidad 3x3.

En general, se definirá una transformación de un vector algebraico \mathbf{s} como:

$$\mathbf{s} = \mathbf{R}\mathbf{s}' \quad (2.15)$$

donde \mathbf{s}' es un vector descrito respecto del sistema de referencia considerado móvil, \mathbf{s} es el vector expresado en el sistema considerado fijo y \mathbf{R} viene expresada por la conocida *fórmula de Rodrigues*:

$$\mathbf{R} = \left[\mathbf{I} + \tilde{\mathbf{u}} \sin \theta + 2(\tilde{\mathbf{u}})^2 \sin^2 \left(\frac{\theta}{2} \right) \right] \quad (2.16)$$

La matriz de rotación \mathbf{R} es una matriz que describe el vector \mathbf{s}' respecto del sistema de referencia absoluto inercial.

2.2.3.4. Composición de matrices de rotación

Las matrices de rotación pueden combinarse utilizando el producto matricial para formar nuevas matrices de rotación. En general, si un sistema de referencia $\{C\}$ tiene una rotación ${}^B\mathbf{R}_C$ respecto a un sistema $\{B\}$, y $\{B\}$ tiene una orientación ${}^A\mathbf{R}_B$ respecto a otro sistema $\{A\}$, la orientación de $\{C\}$ respecto a $\{A\}$ viene dada por,

$${}^A\mathbf{R}_C = {}^A\mathbf{R}_B {}^B\mathbf{R}_C \quad (2.17)$$

De esta forma, utilizando la regla de composición para matrices de orientación con las matrices $\mathbf{R}(\mathbf{x}, \alpha)$, $\mathbf{R}(\mathbf{y}, \phi)$, $\mathbf{R}(\mathbf{z}, \theta)$, que vienen definidas por las ecuaciones 2.2, 2.3 y 2.4, una rotación general puede expresarse como composición de tres giros básicos sobre los ejes cartesianos de un sistema fijo $OXYZ$ mediante la siguiente expresión:

$$\mathbf{R} = \mathbf{R}(\mathbf{z}, \theta)\mathbf{R}(\mathbf{y}, \phi)\mathbf{R}(\mathbf{x}, \alpha) =$$

$$\begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} = \quad (2.18)$$

$$\begin{bmatrix} \cos \theta \cos \phi & -\sin \theta \cos \alpha + \cos \theta \sin \phi \sin \alpha & \sin \theta \sin \alpha + \cos \theta \sin \phi \cos \alpha \\ \sin \theta \cos \phi & \cos \theta \cos \alpha + \sin \theta \sin \phi \sin \alpha & -\cos \theta \sin \alpha + \sin \theta \sin \phi \cos \alpha \\ -\sin \phi & \cos \phi \sin \alpha & \cos \phi \cos \alpha \end{bmatrix}$$

2.2.3.5. Propiedades de la matriz de rotación

Sea $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ una matriz de rotación y $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3 \in \mathbb{R}^3$ las columnas de dicha matriz. Como las columnas de \mathbf{R} son ortonormales, se sigue que:

$$\mathbf{r}_i^T \mathbf{r}_j = \begin{cases} 0, & \text{si } i \neq j \\ 1, & \text{si } i = j \end{cases}$$

y $\mathbf{R}\mathbf{R}^T = \mathbf{R}^T\mathbf{R} = \mathbf{I}$.

El determinante de la matriz de rotación es $\det \mathbf{R} = \pm 1$, que en el caso de triedros a derechas toma el valor positivo:

$$\det \mathbf{R} = \mathbf{r}_1^T(\mathbf{r}_2 \times \mathbf{r}_3) = \mathbf{r}_1^T(\mathbf{r}_1) = 1$$

En general se define el espacio de las matrices de rotación SO (*special orthogonal*) como:

$$SO(n) = \{\mathbf{R} \in \mathbb{R}^{n \times n} : \mathbf{R}\mathbf{R}^T = \mathbf{I}, \det \mathbf{R} = +1\} \quad (2.19)$$

siendo de especial interés en robótica los casos de $SO(2)$ para robots planares y $SO(3)$ para robots espaciales. Se puede demostrar que $SO(3) \subset \mathbb{R}^{3 \times 3}$ junto con la operación de multiplicación de matrices y la matriz identidad como elemento neutro forman un grupo, que llamaremos *grupo de rotaciones* en \mathbb{R}^3 .

2.2.3.6. Otras representaciones: ángulos de Euler, roll-pitch-yaw

Ángulos de Euler.

La orientación de todo sistema $\{B\}$ $OUVW$ respecto al sistema $\{A\}$ $OXYZ$ puede obtenerse a partir de tres ángulos ϕ, θ, ψ , denominados como ángulos de Euler. Consideremos el sistema $\{B\}$ inicialmente coincidente con $\{A\}$ (figura 2.9). Primero se rota el sistema $\{B\}$ alrededor del eje OZ del sistema $\{B\}$ un ángulo ϕ , luego se rota alrededor del (nuevo) eje OX del sistema $\{B\}$ un ángulo θ para finalmente rotar el sistema $\{B\}$

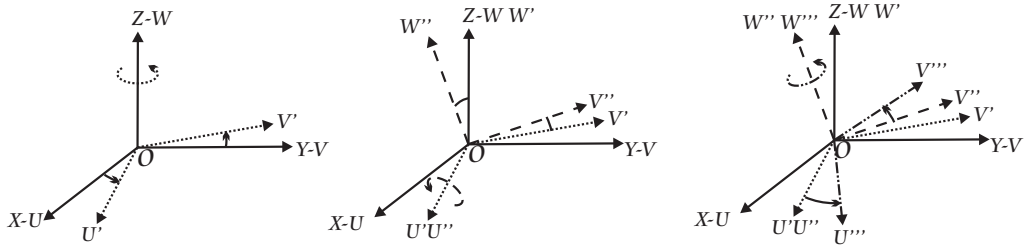


Figura 2.9: Representación de orientación mediante los ángulos 313 de Euler

alrededor del (otra vez nuevo) eje OZ del sistema $\{B\}$. Dichos ángulos representan los giros sufridos por el sistema $OXYZ$ sobre unos ejes determinados de un triedro ortonormal, por tanto, además de los valores de los ángulos ϕ, θ, ψ , es necesario especificar los ejes sobre los cuales se han realizado dichas rotaciones.

Existen varias combinaciones de 3 giros sobre ejes ortogonales que definen un giro general. Normalmente se expresan con los nombres de los ejes coordenados, de forma que los ángulos ZXZ de Euler representan un giro en OZ , un giro posterior en el eje OX desplazado y otro giro posterior respecto al eje OZ móvil. También es usual nombrar dichos ángulos con la secuencia numérica que representa dichos ejes, es decir los ángulos ZXZ se representan como ángulos 313. Otras combinaciones usuales en robótica son los ángulos 323.

La figura 2.9 muestra un esquema de los ángulos de Euler 313 (ZXZ) que representan un giro de un ángulo ϕ sobre el eje OZ , seguido de un giro de un ángulo θ sobre el eje OU y finalmente un giro de un ángulo ψ sobre el eje OW . La expresión de la matriz de rotación que define una rotación 313 de Euler se obtiene por composición de las matrices de rotación básicas:

$$\mathbf{R}_{313} = \mathbf{R}(\mathbf{z}, \phi)\mathbf{R}(\mathbf{u}, \theta)\mathbf{R}(\mathbf{w}, \psi) =$$

$$\begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} = \quad (2.20)$$

$$\begin{bmatrix} \cos \phi \cos \psi - \sin \phi \cos \theta \sin \psi & -\cos \phi \sin \psi - \sin \phi \cos \theta \cos \psi & \sin \phi \sin \theta \\ \sin \phi \cos \psi + \cos \phi \cos \theta \sin \psi & -\sin \phi \sin \psi + \cos \phi \cos \theta \cos \psi & -\cos \phi \sin \theta \\ \sin \phi \sin \psi & \sin \theta \cos \psi & \cos \theta \end{bmatrix}$$

Ángulos Roll-Pitch-Yaw (alabeo, guiñada, cabeceo).

Se trata de la representación utilizada generalmente en aeronáutica. En este caso los giros se definen respecto a los ejes inerciales del sistema $OXYZ$. Si se parte de los sistemas $OXYZ$ y $OUVW$ coincidentes, se puede colocar al sistema $OUVW$ en cualquier orientación siguiendo los siguientes pasos, (ver figura 2.10):

1. Girar el sistema $OUVW$ un ángulo α con respecto al eje OX . Es el denominado Yaw o guiñada.
2. Girar el sistema $OUVW$ un ángulo ϕ con respecto al eje OY . Es el denominado Pitch o cabeceo.
3. Girar el sistema $OUVW$ un ángulo θ con respecto al eje OZ . Es el denominado Roll o alabeo.

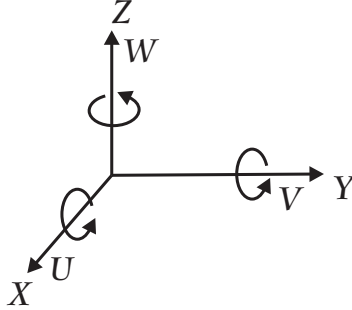


Figura 2.10: Representación de orientación mediante Roll-Pitch-Yaw

2.2.4. Representación conjunta de la localización de un cuerpo en \mathbb{R}^3

Como hemos dicho anteriormente, en general, el movimiento de un cuerpo rígido implica transformaciones de posición y de rotación. En esta sección introduciremos una representación conjunta de estas transformaciones.

Sea $\mathbf{p} \in \mathbb{R}^3$ el vector de posición del origen del sistema $\{B\}$ respecto al sistema $\{A\}$, y $\mathbf{R} \in SO(3)$ la matriz que define la rotación del sistema $\{B\}$ respecto al sistema $\{A\}$, definimos como *Grupo Especial Euclideo* $SE(3)$ a una configuración del sistema $\{B\}$, como un par (\mathbf{p}, \mathbf{R}) cuyo espacio es el producto de \mathbb{R}^3 con $SO(3)$:

$$SE(3) = \{(\mathbf{p}, \mathbf{R}) : \mathbf{p} \in \mathbb{R}^3, \mathbf{R} \in SO(3)\} = \mathbb{R}^3 \times SO(3) \quad (2.21)$$

2.2.4.1. Coordenadas homogéneas

Generalmente, para describir objetos geométricos se utilizan ecuaciones que no son homogéneas en todas sus coordenadas. Así por ejemplo, la ecuación de un plano cuya normal es $t\mathbf{i} + u\mathbf{j} + v\mathbf{k}$ y que dista del origen O una cantidad $\frac{-s}{\sqrt{t^2+u^2+v^2}}$ viene dada por

$$tx + uy + vz + s = 0 \quad (2.22)$$

en la que el grado de los tres primeros términos es 1 mientras que para el cuarto término es 0. Esta es una ecuación no homogénea.

Esta representación de objetos en coordenadas no homogéneas (x, y, z) no es una buena representación para objetos que se encuentran muy alejados del origen, y además se puede demostrar que la derivación en coordenadas no homogéneas puede ocasionar inconsistencias matemáticas. Para resolver este problema se va a utilizar una representación en coordenadas homogéneas, que además permitirá una representación conjunta de las transformaciones de posición y de orientación estudiadas previamente.

Se describe el 3-espacio proyectivo, denominado \mathbb{P}^3 , como el espacio $(3+1)$ dimensional de coordenadas (x, y, z, ω) en el que:

- No todas las coordenadas pueden ser nulas. Por tanto se excluye el punto $(0, 0, 0, 0)$.
- Existe una relación con el espacio euclideo \mathbb{R}^3 tal que $(x, y, z, \omega) = (\alpha x, \alpha y, \alpha z, \alpha \omega)$ $\forall \alpha \in \mathbb{R}$.

Por tanto, el punto (x, y, z) puede ser representado en coordenadas homogéneas como un vector $(x, y, z, \omega)^T \in \mathbb{R}^4$ o como $(\frac{x}{\omega}, \frac{y}{\omega}, \frac{z}{\omega}, 1)^T \in \mathbb{R}^4$, con $\frac{1}{\omega} \neq 0$. El espacio euclideo es, por tanto, un plano con $\omega = 1$ en el 3-espacio proyectivo.

Consideremos la ecuación (2.22) del plano. Si denotamos las coordenadas del plano por $\pi = [t, u, v, s]^T$ la ecuación del plano viene dada por:

$$\pi^T r = [t, u, v, s] \begin{bmatrix} x \\ y \\ z \\ \omega \end{bmatrix} = 0 \quad (2.23)$$

Multiplicando el vector π por cualquier escalar real $\alpha \neq 0$, obtenemos el mismo plano $(\alpha\pi)$.

La ecuación (2.23) puede leerse como que representa todos los puntos del plano π , y es homogénea en (x, y, z, ω) , pero también puede interpretarse como que representa todos los planos que pasan por el punto (x, y, z, ω) , y que es homogénea en (t, u, v, s) . De esta forma, si $\omega = 0$, la ecuación representa todos los planos en el infinito. Si $s = 0$, la ecuación representa todos los planos que contienen el origen.

En resumen, utilizaremos un espacio proyectivo y sus correspondientes coordenadas homogéneas para la localización de los cuerpos rígidos en el espacio.

2.2.4.2. Matrices de transformación homogénea

Consideremos ahora el sistema $\{A\}$, definido por $(OXYZ)$ solidario al cuerpo A , y el sistema $\{B\}$, definido por $(O'UVW)$ solidario al cuerpo B , tal y como muestra la figura 2.11. De la definición de matriz de rotación, obtenemos la relación:

$${}^A \mathbf{r}_P = {}^A \mathbf{R}_B^B \mathbf{r}_P + {}^A \mathbf{r}_{O'} \quad (2.24)$$

siendo ${}^A \mathbf{r}_{O'}$ la transformación de posición del origen O' .

Esta ecuación se puede expresar de una forma más compacta, utilizando coordenadas homogéneas con $\omega = 1$ como:

$$\begin{bmatrix} {}^A \mathbf{r}_P \\ 1 \end{bmatrix} = \begin{bmatrix} {}^A \mathbf{R}_B & | & {}^A \mathbf{r}_{O'} \\ \hline \mathbf{0}_{1 \times 3} & | & 1 \end{bmatrix} \begin{bmatrix} \mathbf{B} \mathbf{r}_P \\ 1 \end{bmatrix} \quad (2.25)$$

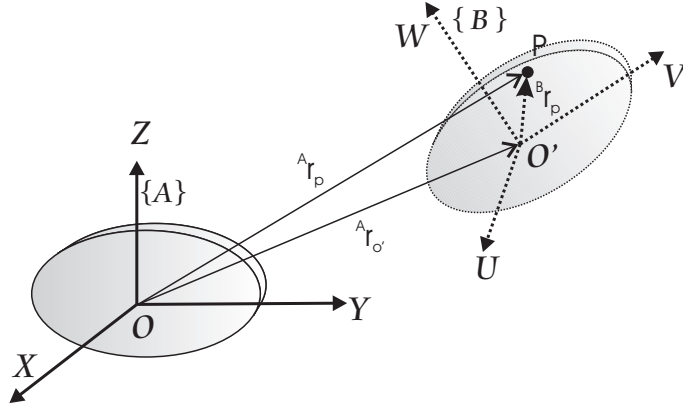


Figura 2.11: Movimiento de un cuerpo rígido

donde la transformación de rotación (definida por ${}^A\mathbf{R}_B \in SO(3)$) y la transformación de posición (definida por ${}^A\mathbf{r}_{O'}$ en \mathbb{R}^3) vienen representadas por la matriz de transformación homogénea:

$${}^A\mathbf{A}_B = \begin{bmatrix} {}^A\mathbf{R}_B & | & {}^A\mathbf{r}_{O'} \\ \hline \mathbf{0}_{1 \times 3} & | & 1 \end{bmatrix} \quad (2.26)$$

En el caso particular en que se defina una rotación pura, la matriz toma la forma:

$${}^A\mathbf{A}_B = \begin{bmatrix} {}^A\mathbf{R}_B & | & \mathbf{0}_{3 \times 1} \\ \hline \mathbf{0}_{1 \times 3} & | & 1 \end{bmatrix} \quad (2.27)$$

y en el caso de una traslación pura:

$${}^A\mathbf{A}_B = \begin{bmatrix} \mathbf{I}_{3 \times 3} & | & {}^A\mathbf{r}_{O'} \\ \hline \mathbf{0}_{1 \times 3} & | & 1 \end{bmatrix} \quad (2.28)$$

La composición de matrices de transformación homogénea se realiza de forma análoga a la composición de matrices de rotación. Consideremos tres cuerpos rígidos A, B y C , con sistemas de referencia $\{A\}$, $\{B\}$ y $\{C\}$ solidarios a dichos cuerpos (figura 2.12).

Si la transformación de coordenadas de $\{B\}$ a $\{A\}$ viene expresada por la matriz de transformación ${}^A\mathbf{A}_B$ y la transformación de $\{C\}$ a $\{B\}$ por ${}^B\mathbf{A}_C$, entonces la transformación de $\{C\}$ a $\{A\}$ viene dada por:

$${}^A\mathbf{A}_C = {}^A\mathbf{A}_B {}^B\mathbf{A}_C$$

$${}^A\mathbf{A}_C = \begin{bmatrix} {}^A\mathbf{R}_C & | & {}^A\mathbf{r}_{O''} \\ \hline \mathbf{0}_{1 \times 3} & | & 1 \end{bmatrix} = \begin{bmatrix} {}^A\mathbf{R}_B \times {}^B\mathbf{R}_C & | & {}^A\mathbf{R}_B \times {}^B\mathbf{r}_{O''} + {}^A\mathbf{r}_{O'} \\ \hline \mathbf{0}_{1 \times 3} & | & 1 \end{bmatrix} \quad (2.29)$$

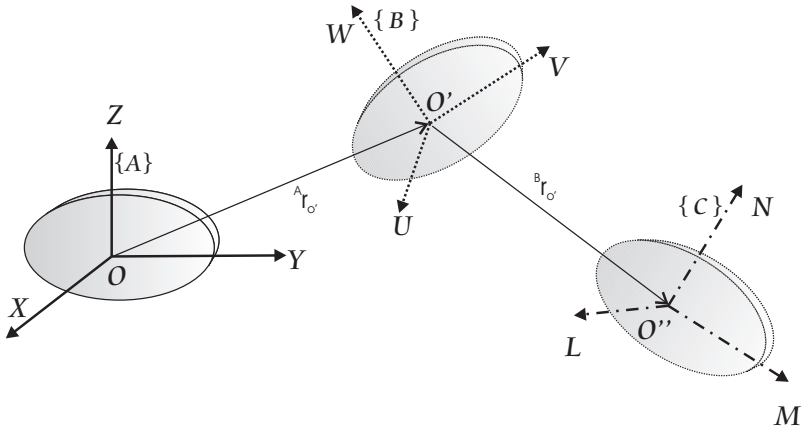


Figura 2.12: Tres cuerpos rígidos con tres sistemas de referencia $\{A\}$, $\{B\}$, $\{C\}$

Utilizando la composición de matrices de transformación homogénea, y haciendo $\{C\}$ igual a $\{A\}$ en la expresión (2.29), es decir, superponiendo los sistemas de forma que la transformación ${}^B\mathbf{A}_C$ es la inversa de la transformación ${}^A\mathbf{A}_B$, se obtiene la útil expresión de la matriz inversa de una matriz de transformación homogénea:

$$[{}^A\mathbf{A}_B]^{-1} = \begin{bmatrix} [{}^A\mathbf{R}_B]^T & -[{}^A\mathbf{R}_B]^T \mathbf{A}r_{O'} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (2.30)$$

2.2.5. Convenio de Denavit-Hartenberg

La representación presentada en 1955 por Jacques Denavit, en aquel momento un estudiante de doctorado que trabajaba con Dick Hartenberg, y que se conoce como representación de Denavit-Hartenberg (D-H), se ha convertido en una metodología clásica para el análisis de dispositivos robóticos. Básicamente se trata de asociar un sistema de referencia para cada cuerpo que forma la cadena cinemática, de manera que la elección de este sistema de referencia no es libre, y debe de estar restringida por una serie de reglas que permiten que la localización de un sistema de referencia respecto al inmediatamente anterior se defina utilizando tan sólo 4 parámetros, en lugar de los 6 necesarios para definir una localización genérica.

Denavit y Hartenberg establecieron 4 transformaciones básicas que relacionaban un sistema de referencia con otro. La secuencia de estas transformaciones es la siguiente (figura 2.13):

1. En primer lugar un giro respecto a \mathbf{z}_{i-1} de magnitud θ_i .
2. Posteriormente, se realiza una traslación respecto al mismo eje \mathbf{z}_{i-1} con una magnitud de d_i .

3. A continuación se realiza otra traslación, ahora respecto al eje \mathbf{x}_{i-1} de magnitud a_i .
4. Y, finalmente se realiza un giro respecto a \mathbf{x}_{i-1} de magnitud α_i .

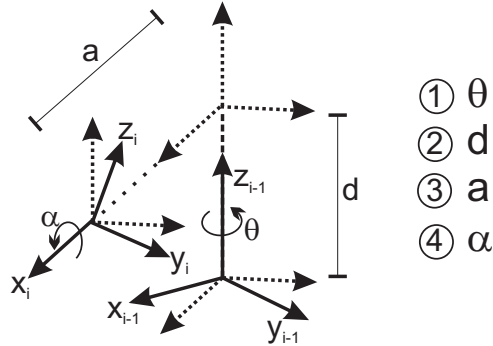


Figura 2.13: Transformaciones básicas de Denavit-Hartenberg.

Si se cumplen estas transformaciones, el sistema al que habremos llegado es un sistema que llamamos de Denavit-Hartenberg. En este caso, y utilizando las matrices de transformaciones básicas, encontramos una expresión general de la matriz de transformación homogénea que relaciona dos sistemas de referencia de Denavit-Hartenberg:

$${}^{i-1}\mathbf{A}_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.31)$$

$${}^{i-1}\mathbf{A}_i = \begin{bmatrix} c\theta_i & -c\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\alpha_i c\theta_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.32)$$

En el caso de trabajar con eslabones encadenados, se asocia un sistema de Denavit-Hartenberg a cada uno de los eslabones (siguiendo una serie de reglas para asegurar que los sistemas son efectivamente sistemas de Denavit-Hartenberg), tal como se observa en la figura 2.14. En este caso, para obtener los parámetros de Denavit-Hartenberg de la transformación, procederemos de la siguiente forma:

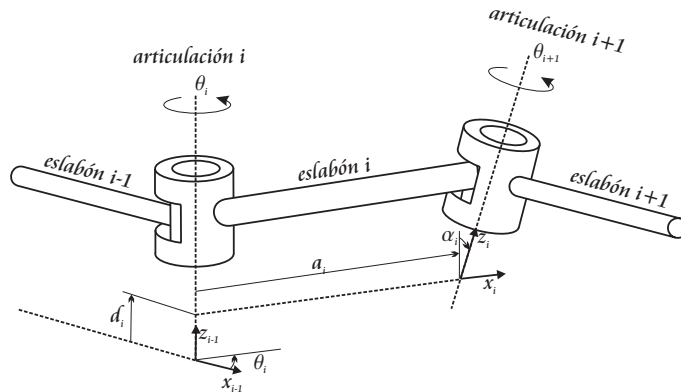


Figura 2.14: Definición de los parámetros de Denavit-Hartenberg de un eslabón

- θ_i : Es el ángulo de la articulación desde el eje \mathbf{x}_{i-1} hasta el eje \mathbf{x}_i , medido respecto del eje \mathbf{z}_{i-1} , usando la regla de la mano derecha. Corresponde a un parámetro variable en las articulaciones rotacionales.
- d_i : Es la distancia medida desde el origen del sistema i-1, a lo largo del eje \mathbf{z}_{i-1} hasta la intersección del eje \mathbf{z}_{i-1} con el eje \mathbf{x}_i . Corresponde a un parámetro variable en las articulaciones prismáticas.
- \mathbf{a}_i : Es la distancia de separación entre los orígenes de los sistemas de referencia i-1 e i, medida a lo largo del eje \mathbf{x}_i hasta la intersección con el eje \mathbf{z}_{i-1} (o la distancia más corta entre los ejes \mathbf{z}_{i-1} y \mathbf{z}_i , cuando éstos no se interceptan).
- α_i : Es el ángulo que separa los ejes \mathbf{z}_i y \mathbf{z}_{i-1} , medido respecto del eje \mathbf{x}_i .

Reglas para la elección de los sistemas de Denavit-Hartenberg

Para poder aplicar el anterior método de D-H a un robot serial es importante que los sistemas elegidos cumplan con las transformaciones de Denavit-Hartenberg. Para ello se presentan a continuación unos consejos para obtener dichos sistemas:

- Se parte de una configuración cualesquiera del robot, si bien es aconsejable colocarlo en una posición sencilla de analizar.
- Se numeran los eslabones, asignando el 0 para la base y n-1 para el último eslabón, siendo n el número de grados de libertad del robot.
- El sistema de coordenadas ortonormal dextrógiro de la base ($\mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0$) se establece con el eje \mathbf{z}_0 localizado a lo largo del eje de movimiento de la articulación 1 y apuntando hacia fuera del hombro del brazo del robot.

- El sistema de referencia de cada eslabón se coloca al final del mismo, en el extremo de la articulación a la cual está conectado el eslabón siguiente.
- El ángulo o desplazamiento de cada eslabón siempre se mide tomando como base el sistema de referencia del eslabón anterior.
- Al colocar el sistema de referencia del *eslabón- i* , se deben seguir las siguientes reglas:
 - El eje \mathbf{z}_i del sistema de referencia debe quedar alineado a lo largo de la articulación. El sentido del eje \mathbf{z}_i es indiferente, pero una vez elegido marcará el sentido positivo del valor θ_i .
 - El eje \mathbf{x}_i debe colocarse en la línea normal común a \mathbf{z}_{i-1} y \mathbf{z}_i . La casuística de esta regla se detalla para tres casos:
 - Los ejes \mathbf{z}_{i-1} y \mathbf{z}_i se cortan en el espacio. La normal común es aquella que sale (o entra) del plano definido por los vectores \mathbf{z}_{i-1} y \mathbf{z}_i (figura 2.15 a) y b)).
 - Los ejes \mathbf{z}_{i-1} y \mathbf{z}_i son paralelos en el espacio. En este caso la normal común es aquella que, siendo coplanar a los vectores \mathbf{z}_{i-1} y \mathbf{z}_i , es normal a ambos (figura 2.15 c)).
 - Los ejes \mathbf{z}_{i-1} y \mathbf{z}_i son colineales en el espacio. Se trata de un caso degenerado del anterior, en el cual todo vector \mathbf{x}_i será normal común. Habitualmente se elige de forma que sea paralelo al anterior \mathbf{x}_{i-1} (figura 2.15 d)).

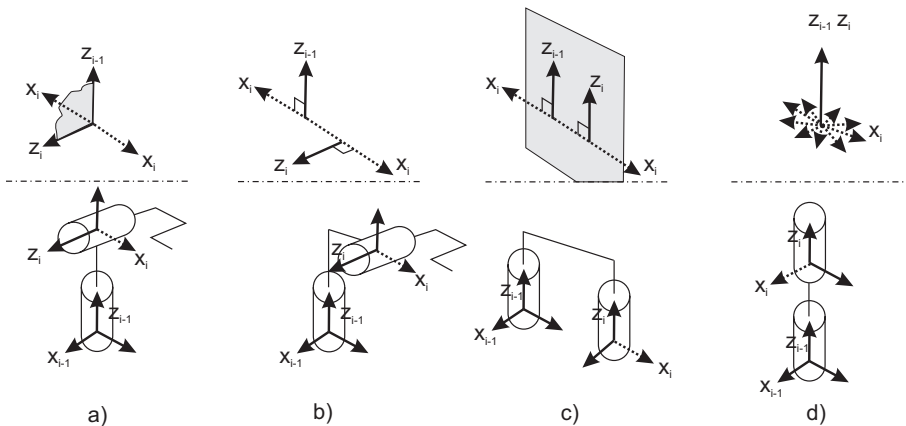


Figura 2.15: Ubicación de los ejes x_i en los sistemas de Denavit-Hartenberg

2.3. Cinemática de robots

Como se ha comentado anteriormente, este capítulo no pretende revisar las técnicas de análisis cinemático y dinámico de robots, y se remite al lector interesado a revisar

otros textos recomendados. En este apartado se cita la teoría necesaria para poder seguir los desarrollos del resto del libro. Para ello, y siguiendo el enfoque práctico, se presentan unos ejemplos (no exhaustivos) para ilustrar la resolución de los problemas planteados.

El lector debe considerar que algunos de estos ejemplos son particulares para esa morfología de robot, ya sea un **robot serial**, un **robot paralelo** o uno **móvil**. Se recomienda revisar textos más completos con ejemplos de otros robots.

Clásicamente se estudian dos problemas fundamentales en el análisis de la cinemática del robot. El primero de ellos, consiste en determinar la posición y orientación del extremo final de la cadena cinemática, a partir de los valores de las coordenadas articulares y las características geométricas del robot, y es conocido como **problema cinemático directo**. Análogamente, la solución del **problema cinemático inverso** permite hallar las variables articulares conocida la posición y orientación del extremo de la cadena cinemática (figura 2.16).

Para la resolución de estos problemas se pueden utilizar varias formulaciones, la más conocida es la formulación clásica basada en la representación de Denavit-Hartenberg y el álgebra matricial de las *matrices de transformación homogénea*, aunque también hay otras formulaciones más generales para el estudio de sistemas multicuerpos, como son la *teoría de screws* [81] o la propia teoría de multicuerpos [27].

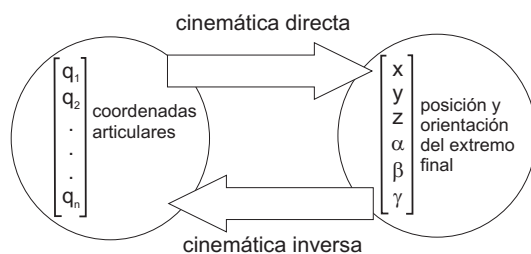


Figura 2.16: Relación entre cinemática directa e inversa.

En primer lugar, el apartado 2.3.1 recuerda la resolución de la cinemática directa de un robot serial utilizando la formulación de Denavit-Hartenberg.

2.3.1. Cinemática directa

En esta sección se plantea primeramente un algoritmo para resolver la cinemática directa de **cadena seriales**. La sección 2.5.1 muestra un ejemplo de uso de este algoritmo para un robot serial de 6 grados de libertad.

1. Colocamos al robot en una posición cómoda para colocar los sistemas D-H.
2. Se obtiene los valores de los parámetros D-H para cada eslabón.
3. A partir de los valores de los parámetros D-H, se calculan las matrices ${}^{i-1}\mathbf{A}_i$ _{DH} para cada transformación utilizando la expresión (2.32).
4. Obtenemos la matriz ${}^0\mathbf{A}_n$ _{DH} como producto de las matrices anteriores.

En el caso de los **robots de configuración paralela**, la solución de la **cinemática directa** es más tediosa. A diferencia del problema cinemático inverso, hallar una solución analítica al problema cinemático directo es muy compleja. Hallar una solución basada en el análisis puramente geométrico resultan en polinomios de elevado grado con múltiples soluciones [81]. Algunos autores [59], [42], [6], han dado soluciones analíticas, haciendo uso de la formulación de Denavit-Hartenberg, pero solo para algunas configuraciones estructurales sencillas. Para el caso particular del mecanismo 6UPS, Merlet [59] ha encontrado hasta 40 soluciones diferentes para un mismo conjunto de estado articular. Otra alternativa propuesta para hallar solución al problema de la cinemática directa consiste en la utilización de sensores redundantes [13]. Sin embargo, aparecen otros problemas asociados con determinar el número adecuado de sensores y la influencia de los errores de medición en la determinación de la cinemática. La sección 2.5.2 presenta un ejemplo de la plataforma clásica 6UPS.

2.3.2. Cinemática inversa

La cinemática inversa consiste en hallar los valores de las coordenadas articulares del robot $q = (q_1, q_2, \dots, q_n)$ a partir de la posición y orientación del extremo del robot.

En el caso de los **robots seriales**, la metodología más comúnmente utilizada es la resolución por métodos geométricos, aunque también existen otros enfoques como la inversión de las matrices de transformación homogénea. El enfoque geométrico hace uso de la morfología de los robots seriales (a menudo los tres primeros ejes se utilizan para el posicionamiento del robot y el resto para la orientación del efector final) y el desacople cinemático para encontrar relaciones cerradas particulares para cada robot. Hay que destacar que, al igual que comentábamos para el problema directo de los robots paralelos, la cinemática inversa de robots seriales puede tener varias soluciones, incluso infinitas en el caso de robots redundantes.

En el caso de los **robots paralelos**, el problema cinemático inverso se resuelve por lo general con una simple observación de la geometría del mecanismo paralelo.

2.3.3. Cinemática diferencial

Sea \mathbf{q}^1 el vector de variables articulares, y sea \mathbf{x} el vector que describe la posición cartesiana de la plataforma móvil. Las restricciones cinemáticas impuestas por las patas se expresan como $\mathbf{f}(\mathbf{x}, \mathbf{q}) = \mathbf{0}$. Diferenciando con respecto al tiempo se obtiene la relación entre la velocidad cartesiana y las velocidades articulares:

$$\mathbf{J}_x \dot{\mathbf{x}} = \mathbf{J}_q \dot{\mathbf{q}} \quad (2.33)$$

donde $\mathbf{J}_x = \frac{\delta \mathbf{f}}{\delta \mathbf{x}}$ y $\mathbf{J}_q = -\frac{\delta \mathbf{f}}{\delta \mathbf{q}}$.

Por tanto, la matriz Jacobiana \mathbf{J} , puede escribirse como:

$$\dot{\mathbf{q}} = \mathbf{J} \dot{\mathbf{x}} \quad (2.34)$$

¹Tenga cuidado con este término, en este libro se utiliza el símbolo \mathbf{q} para denotar el espacio articular como así también al vector de coordenadas generalizadas que describe el estado de un cuerpo rígido. El significado del mismo quedará definido de acuerdo al contexto en el cual se mencione.

donde $\mathbf{J} = \mathbf{J}_q^{-1} \mathbf{J}_x$.

2.3.3.1. Análisis de singularidades

Una importante característica de la matriz jacobiana es la posibilidad de utilizarla para analizar las configuraciones singulares de un robot. Como hemos mencionado con anterioridad, la matriz jacobiana de un mecanismo vincula el espacio de las velocidades articulares con el espacio de las velocidades del efector final. Por lo tanto definiremos a una configuración singular del mecanismo en términos de la matriz Jacobiana:

- Si la matriz Jacobiana es singular, entonces el mecanismo se encuentra en una configuración singular.

Recuerde, que una matriz es singular si pierde su rango, esto sucede, si algunas de sus filas o columnas son completamente nulas, o si son linealmente dependiente entre sí (filas y columnas), y además su determinante es cero. Note usted, que si la matriz Jacobiana del mecanismo es singular, su determinante será cero, por lo tanto no existe su matriz inversa, y la relación de velocidades será indeterminada. Recordemos que, en el caso de robots paralelos, la matriz jacobiana tiene dos componentes: una matriz jacobiana directa J_x y una matriz jacobiana inversa J_q . Por lo tanto podremos distinguir tres tipos de singularidades:

Singularidad de Cinemática Directa: en este tipo la matriz jacobiana J_x es singular, es decir $\det(J_x) = 0$. En este tipo de singularidad, la plataforma móvil puede realizar movimientos infinitesimales en ciertas direcciones con los actuadores completamente bloqueados. Es decir que el mecanismo ha ganado uno o mas grados de libertad.

Singularidad de Cinemática Inversa: este tipo de singularidad se presenta cuando el $\det(J_q) = 0$. En este caso, existen direcciones en las que la plataforma móvil no puede realizar movimientos diferenciales. Por lo general este tipo de singularidades se encuentran en los límites del espacio de trabajo.

Singularidad Combinada: ocurre cuando las matrices jacobianas directa e inversa son singulares, es decir $\det(J_x) = 0$ y $\det(J_q) = 0$.

2.3.3.2. Mapeado de fuerzas

Adicionalmente, la matriz jacobiana proporciona una herramienta muy útil en el análisis de robots que interactúan con el entorno. Una vez obtenida la matriz jacobiana, se utiliza la ecuación 2.35 para obtener las fuerzas en el extremo de la herramienta (F_{hta}) a partir de las fuerzas en los actuadores (F_{act}).

$$F_{hta} = \{J^T(q)\}^{-1} * F_{act} \quad (2.35)$$

Para la obtención de la F_{act} se utiliza la propiedad de linealidad entre la corriente consumida (i) y la fuerza (F_{act}) (ecuación 2.36)

$$F_{act(i)} = K * I_{motor(i)} \quad (2.36)$$

2.4. Dinámica de robots

El *modelo dinámico* del robot relaciona el movimiento del robot y las fuerzas implicadas en el mismo. Por lo tanto, el modelo dinámico establece relaciones matemáticas entre las coordenadas articulares (o las coordenadas del extremo del robot), sus derivadas (velocidad y aceleración), las fuerzas y pares aplicados en las articulaciones (o en el extremo) y los parámetros del robot (masas de los eslabones, inercias, etc.)

El modelo dinámico de un brazo robot puede obtenerse considerando diferentes formulaciones clásicas que provienen de las leyes de la mecánica newtoniana y lagrangiana: formulaciones de Lagrange-Euler, de Newton-Euler, ecuaciones generalizadas de D'Alembert, etc. Hay que tener en cuenta que las ecuaciones de movimiento obtenidas con estas **formulaciones son equivalentes** en el sentido que todas ellas describen la misma conducta dinámica del robot. Sin embargo, cada una de ellas presenta características diferentes que la hacen más apropiada para ciertas tareas. Por ejemplo, la formulación de Lagrange-Euler presenta un modelo simple y elegante, dando como resultado una serie de ecuaciones diferenciales no lineales de 2^o orden acopladas útiles para el estudio de estrategias de control en el espacio de estados de las variables articulares del robot, pero que se presentan ineficaces para aplicaciones en tiempo real dado el elevado tiempo de cálculo que requieren las operaciones con matrices de transformación homogénea. Por otra parte, las ecuaciones de Newton-Euler obtenidas por Luh poseen una formulación que las hace computacionalmente eficientes, aunque su estructura recursiva destruye la expresión analítica del modelo dinámico dificultando el análisis para el control. La dificultad es enorme debido a que se obtienen expresiones fuertemente no-lineales que constan de cargas inerciales, fuerzas de acoplo entre las articulaciones y efectos de las cargas de gravedad, con la dificultad añadida de que los pares/fuerzas dinámicos dependen de los parámetros físicos del manipulador, de la configuración instantánea de las articulaciones, de la velocidad, de la aceleración y de la carga que soporta el robot. Aunque las ecuaciones del movimiento son equivalentes ya sean analíticas o numérico-recursivas, los diferentes planteamientos dependen de los objetivos que se quieran conseguir con ellos. En algunos casos es necesario solucionar el problema dinámico de un robot para lograr tiempos de cálculo rápidos en la evaluación de los pares y fuerzas articulares para controlar el manipulador, y en otros casos son necesarios planteamientos para facilitar el análisis y la síntesis del control.

De forma muy resumida, (se remite al lector a otros textos para ampliación), se puede decir que:

- El algoritmo de Lagrange-Euler utiliza consideraciones energéticas para obtener una ecuación algebraica que permite disponer de un modelo matemático formal.

$$\tau = \mathbf{D}(q)\ddot{q} + \mathbf{H}(q, \dot{q}) + \mathbf{C}(q)$$

donde

τ es el vector de fuerzas generalizadas (fuerzas lineales y torques) en las articulaciones.

$\mathbf{D}(q)$ es una matriz simétrica de dimensión $n \times n$ relacionada con la aceleración, dependiente únicamente de la posición de los eslabones del robot (q), y que

contabiliza el efecto de las masas en movimiento (inercias). Se conoce como matriz inercial.

$\mathbf{H}(q, \dot{q})$ es un vector que contabiliza las fuerzas de Coriolis, centrípeta y centrífuga, y es de dimensión $n \times 1$.

$\mathbf{C}(q)$ es un vector de fuerzas gravitatorias, de dimensión $n \times 1$ y dependiente únicamente de la posición.

- El algoritmo de Newton-Euler utiliza las matrices de rotación ${}^0\mathbf{R}_i$ y una formulación recursiva para generalizar las ecuaciones de balance de fuerzas y pares de Newton a sistemas de cadena cinemática abierta. Este algoritmo numérico presenta una complejidad $O(n)$ y es fácilmente programable. Una implementación en Matlab de este algoritmo se encuentra en [70].
- El algoritmo numérico de Walker y Orin utiliza la ecuación algebraica proporcionada por L-E y el cálculo numérico de N-E para resolver el problema dinámico directo de cadenas abiertas. Para su resolución se utiliza la técnica del cuerpo compuesto.

Los objetivos de los modelos dinámicos están enfocados a resolver dos tipos de problemas:

- **Problema dinámico inverso** de un robot manipulador. La dinámica inversa permite calcular las fuerzas y pares de reacción en las articulaciones y los pares y fuerzas requeridos por los accionamientos de potencia del manipulador, a partir de las condiciones cinemáticas de posición, velocidad y aceleración del robot, definidas a través de una planificación de trayectorias o de una serie de configuraciones geométricas que se quiere estudiar, y de los efectos de las fuerzas externas (carga útil, fuerzas de contacto, fuerzas de colisión, etc.). La aplicación del cálculo computacional de la dinámica inversa está dirigida hacia el diseño mecánico del robot y al cálculo de los accionamientos de potencia requeridos, de acuerdo a la respuesta dinámica que se desea que tenga el robot.
- **Problema dinámico directo** de un robot manipulador. La solución del problema dinámico directo comprende el cálculo de la aceleración, velocidad y posición de un robot a partir de la dinámica de pares o fuerzas impuestas por los accionamientos de potencia del robot, los efectos inerciales, las fuerzas externas (carga útil, fuerzas de contacto o colisiones, etc.) y el sistema de control.

simMechanics

Se remite al lector a los libros recomendados [70] para un detallado estudio del modelo dinámico. En este libro se va a hacer uso de la herramienta de Matlab/Simulink SimMechanics, que permite modelar y simular la dinámica de robots, tanto seriales como paralelos. Se aconseja consultar la documentación de Matlab sobre SimMechanics (<http://www.mathworks.es/help/toolbox/physmod/mech/>). En particular, en el User's Guide se muestra un ejemplo de análisis de una plataforma de Stewart ([ejemplo](#)). Asimismo en el adjunto a este libro se puede consultar la documentación de la Escuela de

Biomecánica que se celebró en Bogotá en 2009, donde se presentaron ejercicios para la resolución de robots paralelos como órtesis y prótesis utilizando simMechanics.

2.5. ROBOTICA MEDICA APLICADA

2.5.1. Ejemplo de análisis de la cinemática un robot serial de 6 GDL

Se va a realizar la cinemática directa del robot rotacional de 6 GDL mostrado en la figura 2.17. En la figura se han representado los sistemas de referencia de cada eslabón según las normas de Denavit-Hartenberg. En este robot todas las variables articulares corresponden a los giros en las articulaciones. Obsérvese que existe una rotación θ_4 entre los eslabones 3 y 4. Por lo tanto las variables articulares son: $(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6)$. Hay que destacar que el sistema de referencia $(\mathbf{x}_3, \mathbf{y}_3, \mathbf{z}_3)$ se ha trasladado al principio del eslabón 3 cumpliendo con la representación de D-H. En la tabla 2.1 se muestran los parámetros de D-H de este robot. Como se puede apreciar el parámetro a_2 aparece con signo negativo debido a que esta distancia es medida a lo largo del eje \mathbf{x}_2 desde el origen del sistema de referencia $\{1\}$ hasta el origen del sistema de referencia $\{2\}$.

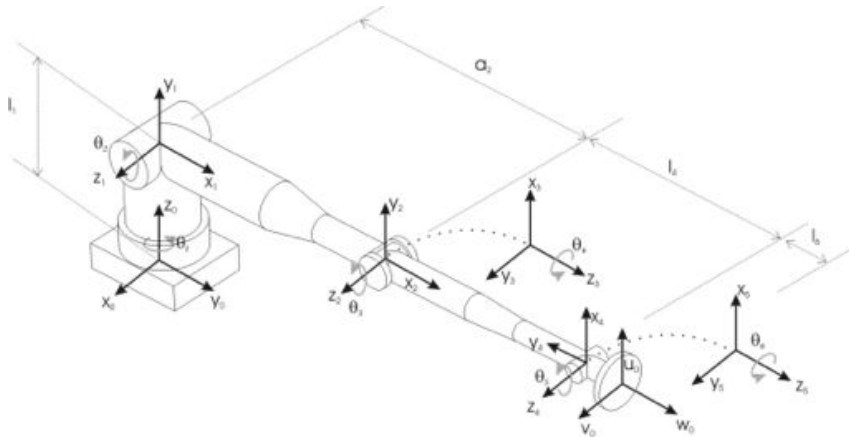


Figura 2.17: Parámetros de Denavit-Hartenberg de un robot rotacional 6 GDL.

La función `forkin6` calcula la cinemática directa del robot rotacional de 6 GDL anterior. Esta función devuelve la matriz de transformación del primer sistema de coordenadas al último en función del vector de variables articulares pasado como parámetro. Este vector representa los 6 grados de libertad del robot y se introduce en los parámetros de D-H como variable. Las dimensiones del robot también han sido introducidas en los parámetros de D-H considerando los valores mostrados en la tabla 2.1. El lector debe notar las modificaciones realizadas en el código respecto a la función utilizada en el cálculo de la cinemática directa del robot prismático de 4 GDL, debido al diferente número de eslabones de cada robot y a los parámetros de D-H de cada robot.

Tabla 2.1: Parámetros de Denavit-Hartenberg para el robot rotacional de la figura 2.17

Eslabón	θ_i	d_i	a_i	α_i
1	$\theta_1 + \pi/2$	$l_1 = 0,315m$	0	$+\pi/2$
2	θ_2	0	$a_2 = 0,45m$	0
3	θ_3	0	0	$\pi/2$
4	θ_4	$l_4 = 0,5m$	0	$-\pi/2$
5	θ_5	0	0	$\pi/2$
6	θ_6	$l_6 = 0,08m$	0	0

```

% FORKIN6 Direct Kinematic.
% A06 = FORKIN6(Q) devuelve la matriz de transformación del
% primer sistema de coordenadas al último en función del vector Q
% de variables articulares.
5 %
% See also HOMOGENEOUSDH.

function A06 = forkin6(q)

10 % Parámetros Denavit-Hartenberg del robot
teta = q;
d = [0.315 0 0 0.5 0 0.08];
a = [0 0.45 0 0 0 0];
alfa = [+pi/2 0 pi/2 -pi/2 pi/2 0];

15 % Matrices de transformación homogénea entre
% sistemas de coordenadas consecutivos
A01 = homogeneousdh(teta(1), d(1), a(1), alfa(1));
A12 = homogeneousdh(teta(2), d(2), a(2), alfa(2));
20 A23 = homogeneousdh(teta(3), d(3), a(3), alfa(3));
A34 = homogeneousdh(teta(4), d(4), a(4), alfa(4));
A45 = homogeneousdh(teta(5), d(5), a(5), alfa(5));
A56 = homogeneousdh(teta(6), d(6), a(6), alfa(6));

25 % Matriz de transformación del primer al último
% sistema de coordenadas
A06 = A01 * A12 * A23 * A34 * A45 * A56;

```

Para la resolución de la cinemática inversa, se ha utilizado el método geométrico, obteniendo unas ecuaciones algebraicas cerradas que permiten la resolución del problema. Hay que resaltar que el problema inverso de un robot antropomórfico como el de la figura 2.17 tiene diferentes soluciones, dependiendo de la configuración del robot. Es decir, el robot puede alcanzar la posición y orientación deseada adoptando para ello diferentes configuraciones. En este ejemplo

La función **invkin6** resuelve la cinemática inversa del robot rotacional de 6 GDL.

Para ello toma como argumento la matriz homogénea A , y los parámetros CODO y MUÑECA para definir las posibles configuraciones. Ha de tenerse en consideración que determinados puntos y orientaciones que pertenezcan al espacio de trabajo del robot, no podrán alcanzarse con algunas configuraciones. Por ello se recomienda al lector que trate de experimentar con estos puntos extremos con la herramienta que a continuación se presenta, y analice los resultados.

En el ejemplo mostrado a continuación se puede comprobar como después de asignar un vector de coordenadas articulares aleatorio, lo cual ya incluye una determinada configuración de CODO y MUÑECA y obtener la matriz homogénea del extremo de robot correspondiente a este vector, si sobre esta matriz se aplica la función INVKIN6 con los valores correctos de CODO y MUÑECA para el vector de coordenadas articulares, se obtiene el vector q original. Se recomienda utilizar la función INVKIN6 con vectores de coordenadas articulares sencillos de analizar.

```
>>q=rand(6,1)

q =

    0,6721
    0,8381
    0,0196
    0,6813
    0,3795
    0,8318

>>A=forkin6(q)

A =

   -0,7400   -0,3846    0,5518    0,5756
    0,6484   -0,1900    0,7372    0,4819
   -0,1787    0,9033    0,3900    0,3387
         0         0         0    1,0000

>>invkin6(A,-1,1)

ans=

    0,6721
    0,8381
    0,0196
    0,6813
    0,3795
    0,8318
```

2.5.2. Ejemplo de análisis de la dinámica de un robot 6UPS con SimMechanics

Para poder realizar la modelización dinámica del mecanismo, es necesario asignarles a cada elemento sus propiedades físicas, (masas y tensores de inercia) y su localización espacial. Para simplificar el modelado dinámico, se considera a los elementos que pertenecen al mecanismo como primitivas mas simples (cilindros sólidos y cilindros huecos), presentadas en la figura 2.18(a). Las mismas, nos permiten obtener de manera mas sencilla la masa y los tensores de inercias, a partir de las características geométricas y la densidad del material de los elementos.

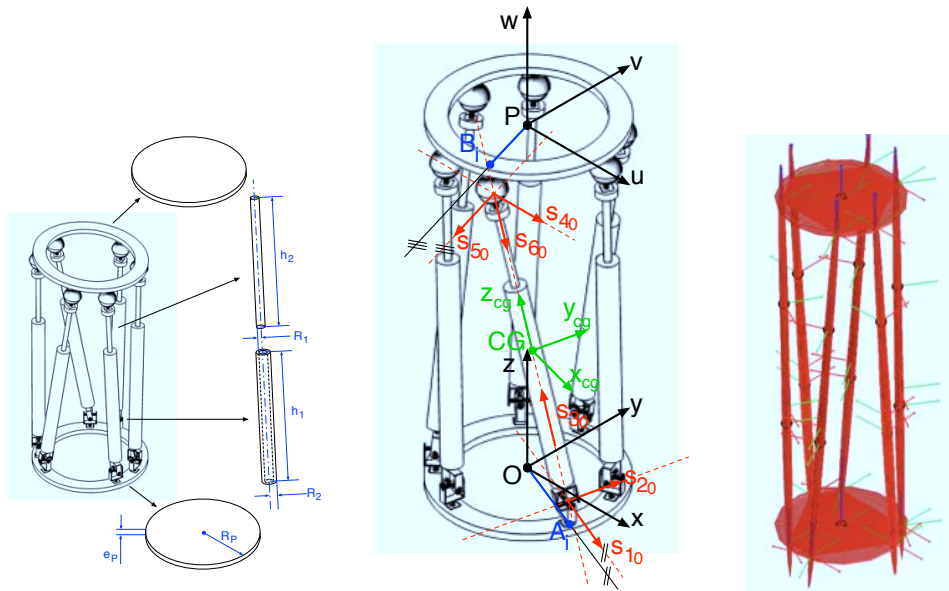


Figura 2.18: Esquema dinámico del mecanismo 6UPS. (a) Modelado de los componentes del mecanismo a partir de geometrías simples. (b) Sistemas de referencias asociados a los elementos del mecanismo. (c) Modelo físico del mecanismo

Además de las propiedades físicas de los elementos que componen al mecanismo, también es necesario definir la localización (posición y orientación) espacial inicial de cada uno de los cuerpos, como así también la dirección inicial de los ejes de las articulaciones. Como se puede observar en la figura 2.18(b), la dirección del eje de la articulación prismática se corresponde con la dirección de la pierna a la que pertenece y se obtiene de acuerdo a:

$$s_{30} = \frac{\overrightarrow{B_i A_i}}{\|\overrightarrow{B_i A_i}\|} \quad (2.37)$$

donde $\overrightarrow{B_i A_i} = \overrightarrow{OP} + \overrightarrow{PB_i} - \overrightarrow{OA_i}$. El primer eje de rotación que compone a cada

articulación universal es paralelo al plano que contiene a la plataforma base, y el mismo queda definido por:

$$s_{10} = \frac{\overrightarrow{A_i \hat{O}}}{\|\overrightarrow{A_i \hat{O}}\|} \quad (2.38)$$

La dirección del segundo eje de rotación de la articulación universal, se lo puede hallar de acuerdo a:

$$s_{20} = \frac{s_{20} \times \hat{z}}{\|s_{20} \times \hat{z}\|} \quad (2.39)$$

La dirección de s_{60} tiene la dirección opuesta a la dirección de la pierna, por lo tanto $s_{60} = -s_{30}$. La dirección de s_{50} viene dada por:

$$s_{50} = \frac{\overrightarrow{B_i \hat{P}}}{\|\overrightarrow{B_i \hat{P}}\|} \quad (2.40)$$

Y el último eje de la articulación esférica se puede obtener de acuerdo a:

$$s_{40} = \frac{s_{50} \times s_{60}}{\|s_{50} \times s_{60}\|} \quad (2.41)$$

La orientación de los eslabones que componen a cada pierna viene dado por la matriz de rotación conformada de la siguiente manera:

$${}^O R_{CG} = [\hat{x}_{CG}^T \quad \hat{y}_{CG}^T \quad \hat{z}_{CG}^T] \quad (2.42)$$

donde \hat{x}_{CG}^T , \hat{y}_{CG}^T , y \hat{z}_{CG}^T son los cosenos directores del sistema de coordenadas ubicado en el centro de gravedad del cuerpo referidos al sistema global O_{xyz} . Siendo $\hat{z}_{CG} = s_{30}$ e $\hat{y}_{CG} = s_{20}$ y \hat{x}_{CG} se puede obtener de acuerdo a:

$$\hat{x}_{CG} = \frac{s_{20} \times s_{30}}{\|s_{20} \times s_{30}\|} \quad (2.43)$$

A partir de las consideraciones mencionadas, vamos a desarrollar una nueva función `PropFisicas(plataforma,R,r,esp,den)` que genere a partir de la geometría del mecanismo sus propiedades físicas y sus marcos de referencias locales.

En la ventana de comandos de Matlab escribiremos las siguientes líneas y debemos obtener la siguiente estructura con los datos de las masas, inercia y sistemas locales, de cada cuerpo.

```
>> plataforma = Geometria(30,30,100);
>> propiedades = PropFisica(plataforma,3,2.5,3);
propiedades =
  Mov: [1x1 struct]
  Bas: [1x1 struct]
  CilHue: [1x1 struct]
  CilSol: [1x1 struct]
  Uni: [1x6 struct]
  Pris: [1x6 struct]
```

```

10  Esf: [1x6 struct]
     Pierna: [1x6 struct]
     L0: [100.8397 100.8397 100.8397 100.8397 100.8397 100.8397]
     >>

```

Note usted que `propiedades.Uni`, `propiedades.Pris`, `propiedades.Esf` y `propiedades.Pierna` son vectores de seis elementos, donde cada elemento es una estructura, y está vinculado a las propiedades del objeto en cuestión de la pierna *i*.

2.5.2.1. Modelo Dinámico en SimMechanics

Empleando las librerías de SimMechanics, armaremos el modelo del mecanismo 6UPS. Para ello, empezaremos por construir el modelo de una solo pierna UPS que vincule a la plataforma fija con la plataforma móvil, tal como se muestra en la figura 2.19

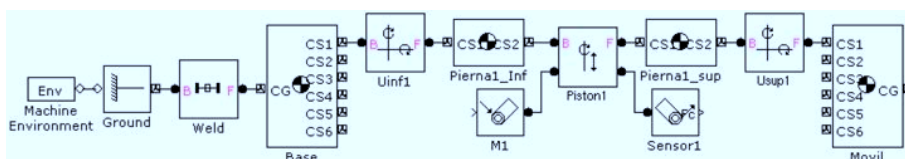


Figura 2.19: Modelo en SimMechanics de la cadena UPS.

Observe que las articulaciones prismática y esférica de la cadena cinemática, han sido sustituida por una articulación cilíndrica y una articulación universal. Esto se debe, a que por lo general en la practica así es como se consigue la cadena cinemática 6UPS.

Empleando las variables generadas con la función `PropFisica`, se configura las propiedades de los bloques tal como se muestran en las siguientes figuras. Los cuerpos `Pierna1_sup` y `Movil`, se configuran de la misma manera que los cuerpos `Pierna1_inf` y `Base`, empleando las estructuras correspondientes, generadas con la función `PropFisica`.

Las articulaciones de la cadena cinemática UPS, se configuran tal como se muestran en las siguientes figuras. Para las articulaciones esféricas no es necesario definir a los ejes, sin embargo, dado que el mecanismo tiene una naturaleza de lazo cerrado, definiremos a las articulaciones esféricas de las piernas 1-5 como las articulaciones “abiertas” para realizar las simulaciones. Observe, que en la articulación prismática se ha conectado un actuador y un sensor, por lo que se deben añadir dos puertos a la articulación. El actuador que se agrega a la articulación prismática, dotará de movimiento a las piernas, por lo que el mismo debe ser configurado como `motion`.

Para simplificar el esquema, seleccionamos los componentes de la cadena UPS y la encapsulamos en un subsistema (ver figura 2.20(a)). Observe, que al encapsular la cadena cinemática UPS, se han generado una entrada y una salida mas de señal: `qact-i` y `fact-i` respectivamente. La señal de entrada es un vector de tres elementos (posición, velocidad y aceleración) que corresponde al movimiento relativo entre los eslabones de la dicha pierna. La señal `fact-i`, es el valor de la fuerza calculada en la articulación correspondiente al movimiento que se le impone a la misma.

Empleando el subsistema de la cadena UPS creado, se completa el modelo del mecanismo 6UPS (figura 2.20(b)). Para finalizar, se conecta al conector del centro de gravedad del cuerpo de la plataforma móvil un *Body Sensor*, con el mediremos las variables de posición, velocidad y aceleración de la misma.

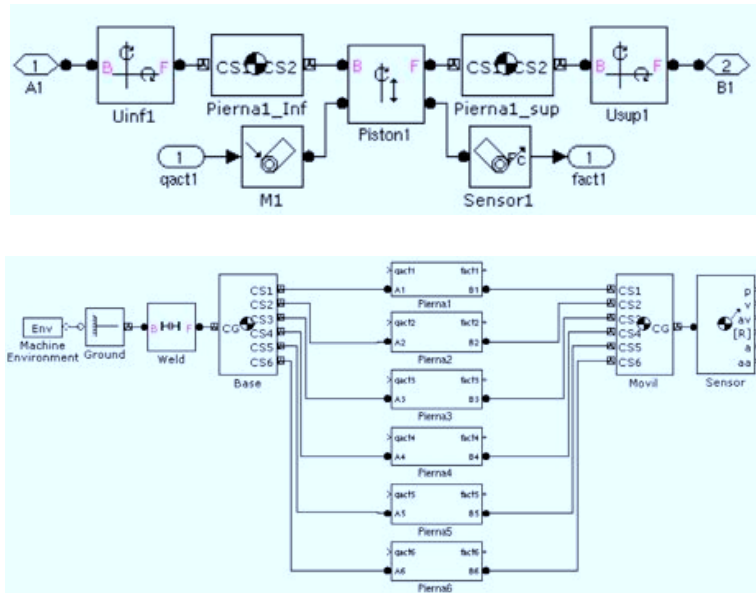


Figura 2.20: Modelo de SimMechanics del Mecanismo 6UPS. (a) subsistema de la cadena cinemática UPS. (b) modelo completo del mecanismo 6UPS.

2.5.2.2. Dinámica Inversa del Mecanismo 6UPS

En esta sección emplearemos el modelo dinámico del mecanismo 6UPS, para encontrar las fuerzas que deben ejercer los actuadores para que el mecanismo realice una trayectoria definida. Para ello definiremos una trayectoria circular de la plataforma móvil de acuerdo a $r = [30 * \sin 0,5 * t, 30 * \cos 0,5 * t, 150]$ manteniendo la orientación fija, es decir $p = [1, 0, 0, 0]$. Para cada uno de los puntos que pertenecen a dicha trayectoria, se calculará el estado de los actuadores empleando la cinemática inversa del mecanismo, empleando una nueva función, denominada *CinInv*, que será ejecutada desde un bloque *MATLAB Function* de Simulink.

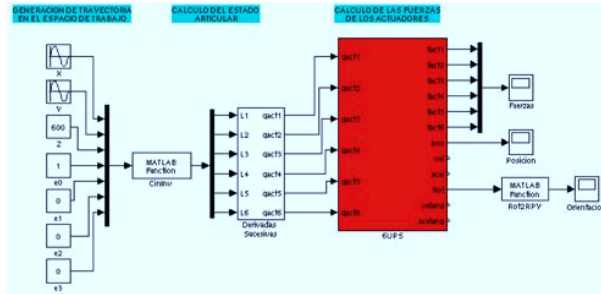


Figura 2.21: Modelo dinámico inverso completo del mecanismo 6UPS

```

function L = CinInv(q)
plataforma = Geometria (30 ,30 ,100); %
q = q?;
L = Cinematica_Inversa(q,plataforma); %
5 q0 = [ 0, 0, plataforma.dist, 1, 0, 0, 0 ]; %
L0 = Cinematica_Inversa(q0,plataforma); %
L = L-L0; %
return %
% ----- Fin CinInv -----

```

Esta función nos devuelve la posición relativa entre los elementos de cada pierna. Sin embargo, para dotar de movimiento al mecanismo, es necesario, obtener la velocidad y la aceleración relativa de los elementos de cada pierna. Por lo tanto, vamos a crear un nuevo subsistema de Simulink que calcule la velocidad y la aceleración de cada pierna, derivando sucesivamente la posición.

Para simplificar el esquema, seleccionaremos el modelo completo del mecanismo 6UPS (Figura (b)), y lo encapsularemos en un subsistema (6UPS). Este nuevo bloque, tiene seis señales de entradas que corresponden al movimiento relativo de cada pierna, y doce señales de salidas, que corresponden a las fuerzas calculadas en cada actuador y las variables medidas en la plataforma móvil. Es importante mencionar que la orientación obtenida de la plataforma móvil esta expresada como matriz de rotación, y para el análisis es mejor expresarla como “Roll-Pitch- Yaw”. Para ello se utiliza la función correspondiente.

Para finalizar, solo falta agregar los bloques `Scope` de Simulink para graficar las fuerzas desarrollada en cada actuador, como así también las posición y orientación de la plataforma móvil. El diagrama en Simulink para la simulación debe quedar como el que se muestra en la figura 2.21, y lo guardaremos como `DinInv6UPS`.

Ahora, solo falta configurar los parámetros del bloque *Machine Environment*. Definiremos el vector de gravedad como $[0, 0, -9, 81]m/s^2$, el modo de análisis será *Forward dynamics* y al resto de parámetros lo dejaremos con el valor por defecto. Para iniciar la simulación, en la ventana de comandos de Matlab, escribiremos las siguientes líneas:

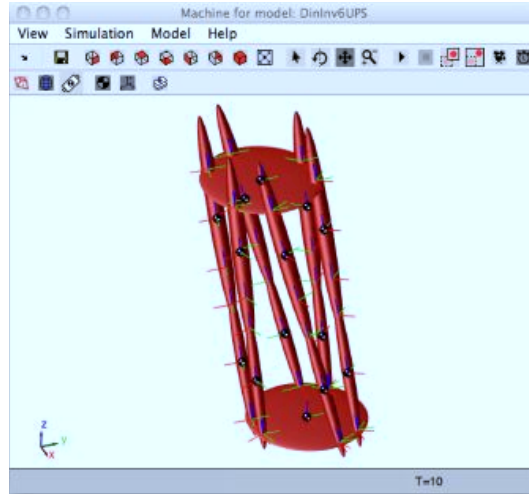


Figura 2.22: Visualización de la simulación del mecanismo 6UPS

```
>> plataforma = Geometria(30,30,100);
>> propiedades = PropFisica(plataforma,3,2.5,3);
>> DinInv6UPS;
>> tiempo = 10;
>> set_param('DinInv6UPS','StopTime',num2str(tiempo));
>> sim('DinInv6UPS')
```

Durante la simulación, se genera una ventana como la que se presenta en la figura 2.22 en la cual podremos observar el comportamiento del mecanismo. En la misma se representa a cada elemento físico como un elipsoide, en los que se remarcan los centros de gravedad y los sistemas de referencias locales de cada cuerpo. Haciendo doble click sobre los bloques Scope se observan las gráficas correspondientes a las fuerzas calculadas en cada pierna

2.6. Resumen del capítulo

Los aspectos claves que han sido tratados en este capítulo se resumen en los siguientes puntos:

1. *Representación de la posición en \mathbb{R}^3 .*
 - Coordenadas cartesianas
 - Coordenadas cilíndricas
 - Coordenadas esféricas

2. *Representación de la orientación en \mathbb{R}^3 .* La matriz de rotación o matriz de cosenos directores es la expresión de una transformación de rotación, y son matrices pertenecientes al espacio $SO(3)$,

$$SO(n) = \{\mathbf{R} \in \mathbb{R}^{n \times n} : \mathbf{R}\mathbf{R}^T = \mathbf{I}, \det \mathbf{R} = +1\}$$

3. *Fórmula de Rodrigues.*

$$\mathbf{R} = \left[\mathbf{I} + \tilde{\mathbf{u}} \sin \theta + 2(\tilde{\mathbf{u}})^2 \sin^2 \left(\frac{\theta}{2} \right) \right]$$

4. *Matrices de transformación homogénea.* La localización de un cuerpo rígido en el espacio tridimensional puede representarse como un elemento $g \in SE(3)$, que es una aplicación $g : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ que conserva la distancia y los ángulos entre puntos. Esta aplicación puede escribirse en coordenadas homogéneas como:

$$\mathbf{A} = \left[\begin{array}{c|c} \mathbf{A}_{\mathbf{R}_B} & \mathbf{A}_{\mathbf{r}_{O'}} \\ \hline \mathbf{0}_{1 \times 3} & 1 \end{array} \right]$$

siendo $\mathbf{A}_{\mathbf{r}_{O'}}$ la transformación de posición del origen O' .

5. *Composición de matrices de transformación homogénea.*
 6. *Modelo cinemático directo.* Ecuación en lazo cerrado de un robot serial.

$${}^0\mathbf{A}_n = {}^0\mathbf{A}_1 \times {}^1\mathbf{A}_2 \times {}^2\mathbf{A}_3 \times \dots \times {}^{n-1}\mathbf{A}_n$$

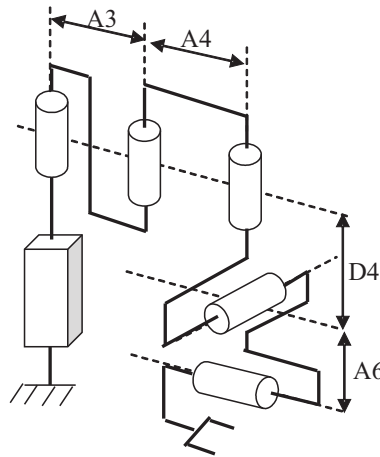
7. *Convenio de Denavit-Harbenberg.* Transformaciones de D-H. Parámetros de D-H. $\theta_i, d_i, a_i, \alpha_i$.

$${}^{i-1}\mathbf{A}_i = \begin{bmatrix} c\theta_i & -\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & \alpha_i c\theta_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

8. El modelo dinámico del robot relaciona el movimiento del robot y las fuerzas implicadas en el mismo. El problema dinámico inverso consiste en obtener los pares y fuerzas existentes a partir de la información cinemática del estado del robot. Se conoce como problema directo el obtener la aceleración, velocidad y posición a partir de la información de fuerzas impuestas por los accionamientos del potencia del robot.
9. Las diferentes formulaciones de la ecuación de movimiento de un robot son equivalentes en el sentido de que describen el mismo fenómeno físico, si bien difieren en su presentación de la información.
10. El algoritmo de Lagrange-Euler utiliza consideraciones energéticas para obtener una ecuación algebraica que permite disponer de un modelo matemático formal.

2.7. Ejercicios

1. Los orígenes de dos sistemas de referencia $\{A\}$ y $\{B\}$ coinciden. Se conocen las coordenadas de tres puntos respecto de los sistemas $\{A\}$ y $\{B\}$, es decir, ${}^A P_1$, ${}^A P_2$, ${}^A P_3$, ${}^B P_1$, ${}^B P_2$, ${}^B P_3$. Se pide determinar la matriz de rotación ${}^A \mathbf{R}_B$.
2. Los sistemas de coordenadas $\{A\}$ y $\{B\}$ son inicialmente coincidentes. El sistema de coordenadas $\{B\}$ se rota 60° alrededor de un vector paralelo a $[2, 4, 7]^T$ que pasa a través del punto $[3, 4, -2]^T$. Determine la matriz de transformación homogénea ${}^A \mathbf{T}_B$.
3. La matriz de rotación ${}^A \mathbf{R}_B$ es una matriz 3×3 con valores propios 1 , $e^{+\alpha i}$, y $e^{-\alpha i}$, donde $i = \sqrt{-1}$. ¿Cuál es el significado físico del vector propio asociado al valor propio 1 ? (Teorema de Euler).
4. El robot SCALPP de la figura es un robot de investigación para aplicaciones médicas basado en un hombro SCARA dotado de una muñeca particular de configuración no-esférica. La razón de ser de esta configuración especial es intentar salvar las singularidades cinemáticas de las muñecas esféricas, tan perjudiciales en aplicaciones médicas. Obtener los parámetros D-H del robot SCALPP.



5. Las siguientes ecuaciones constituyen el modelo dinámico de un robot polar de 2 grados de libertad de tipo RP (rotacional-prismático). Sin embargo, algunos términos están *obviamente* equivocados (no es necesario calcular el modelo dinámico para encontrar dichos términos). Encuentre los términos incorrectos indicando por qué son erróneos.

$$\begin{aligned}\tau_1 &= m_1(d_1^2 + d_2)\dot{\theta}_1 + 2m_2d_2\dot{d}_2\dot{\theta}_1 + g \cos(\theta_1) \left[m_1(d_1 + d_2\dot{\theta}_1) + m_2(d_2 + \dot{d}_2) \right] \\ \tau_2 &= m_1\dot{d}_2\ddot{\theta}_1 + m_2\ddot{d}_2 - m_1d_1\dot{\theta}_1 + m_2(d_2 + 1)g \sin(\theta_1)\end{aligned}$$

6. A partir de la definición de matriz jacobiana $\dot{x} = \mathbf{J}\dot{q}$, donde x representa el vector de velocidades cartesianas del extremo del robot referidas al sistema de la base y q el vector de coordenadas articulares, y de la relación $\tau = \mathbf{J}^T F$, donde τ son los pares articulares y F los pares cartesianos, se pide:

Obtener la ecuación matricial del modelo dinámico del robot en el espacio de trabajo, es decir, expresar el modelo dinámico del robot en coordenadas cartesianas.

Expresar en el espacio de la tarea el modelo dinámico dado por las siguientes ecuaciones en el espacio articular de un robot RR.

$$\tau_1 = m_2 l_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) + m_2 l_1 l_2 C_2 (2\ddot{\theta}_1 + \ddot{\theta}_2) + (m_1 + m_2) l_1^2 \ddot{\theta}_1 - m_2 l_1 l_2 S_2 \dot{\theta}_2^2 - 2m_2 l_1 l_2 S_2 \dot{\theta}_1 \dot{\theta}_2 + m_2 l_2 g C_{12} + (m_1 + m_2) l_1 g C_1$$

$$\tau_2 = m_2 l_1 l_2 C_2 \ddot{\theta}_1 + m_2 l_1 l_2 S_2 \dot{\theta}_2^2 + m_2 l_2 g C_{12} + m_2 l_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2)$$

$$J(q) = \begin{bmatrix} l_1 S_2 & 0 \\ l_1 C_2 + l_2 & l_2 \end{bmatrix}$$

Obtener las singularidades de dicho robot.

Bibliografía para ampliar

Robotica Aplicada R. Saltaren, L. Puglisi, J.M. Sabater-Navarro, F. Thomas. Este libro está concebido para la docencia de robots seriales y paralelos utilizando Matlab. Los capítulos 3, 5 y 8 muestran la teoría y ejemplos para la resolución práctica de la cinemática y el modelo dinámico de robots, tanto de cadena cinemática serial como paralela.

Flexible multibody system dynamics. Theory and applications. (1994) Mingjun Xie. En este texto se recoge la formulación matemática básica de los sistemas multicuerpo flexibles, para en la parte II del libro presentar diferentes aplicaciones entre las que se encuentra un modelo flexible del cuerpo humano.

A mathematical introduction to robotic manipulation. (1994) Richard M. Murray, Zexiang Li, S. Shankar Sastry. En el capítulo 2 de este libro se presenta un riguroso tratamiento matemático de las herramientas de localización de cuerpos rígidos basada en la formulación de algebra de Lie. Se presentan ejemplos y ejercicios utilizando coordenadas exponenciales en el tratamiento del análisis de la velocidad.

Robot analysis. The mechanics of serial and parallel manipulators. (2004) Lung-Wen Tsai. En el capítulo 1 se presenta las herramientas clásicas (coordenadas homogéneas) y las basadas en teoría de screws que son utilizadas en los capítulos posteriores del libro para el análisis de robots manipuladores. Se presenta una visión y un tratamiento muy claro de la naturaleza de los screws sin ahondar en su análisis matemático.

Robots and screw theory. Applications of kinematics and statics to robotics (2004) J.K. Davidson, D.H. Hunt. Este libro de la editorial Oxford presenta la teoría de Screws (no vista en este capítulo) para el análisis cinemático de robots.

Fundamentos de Robótica. A. Barrientos y otros. En este texto se pueden encontrar los fundamentos de lo expuesto en este capítulo.

Capítulo 3

Imagen médica. Fundamentos y aplicaciones

“Si consigo ver más lejos es porque he conseguido auparme a hombros de gigantes”
Isaac Newton

Indice

3.1. Introducción a la imagen médica	80
3.1.1. Principales hitos en la imagen médica	81
3.2. Uso de imágenes médicas	82
3.3. Tipos de imágenes médicas	83
3.3.1. Imágenes de radiografía	83
3.3.2. Imágenes por ultrasonidos	85
3.3.3. Imágenes de Tomografía Axial Computerizada	86
3.3.4. Imagen nuclear	88
3.3.5. Resonancia magnética	90
3.4. Análisis de imagen médica	94
3.5. ROBOTICA MEDICA APLICADA	94
3.5.1. Ejemplo de lectura de imágenes DICOM con Matlab®	94
3.5.2. Ejemplo de segmentación y reconstrucción 3D de la arteria aorta	100
3.6. Resumen del capítulo	105
3.7. Ejercicios	106
Bibliografía para ampliar	106

3.1. Introducción a la imagen médica

Desde la aparición de las radiografías hace un siglo las herramientas de diagnóstico médico basadas en imágenes han tenido una evolución espectacular gracias al auge que han experimentado las TICs (Tecnologías de la Información y las Comunicaciones) en los últimos años, catalizando el desarrollo de la Imagen Médica (IM). La IM representa un campo de investigación en el cual convergen muchas disciplinas como la Electrónica, la Instrumentación, el Procesamiento de Señales (en la fase de la adquisición, mejora y visualización de las imágenes), la ingeniería biomédica, la medicina o las ciencias computacionales en los aspectos de modelado, pero son las especialidades médicas como la Radiología y las especialidades particularmente concernidas en el problema médico bajo estudio (Cardiología, Neurología, etc.) las encargadas del análisis médico, del diagnóstico, o bien de su utilización en los procesos quirúrgicos.

La imagen médica se puede definir como el conjunto de técnicas y procesos utilizados para crear imágenes del cuerpo humano, de sus partes o aún de sus funciones para propósitos clínicos, entendidos como aquellos procedimientos médicos que permiten detectar, analizar y diagnosticar enfermedades o patologías. La IM incorpora la radiología, la medicina nuclear, la endoscopia, la ultrasonografía y la termografía, entre otras técnicas de generación de imágenes. Sin embargo, técnicas de medición y registro de señales como la electrocardiografía (ECG), la electroencefalografía (EEG) y la magnetoencefalografía (MEG) aunque no producen directamente imágenes, se pueden representar como gráficos que incluyen información espacial, convirtiéndose en sí mismos en una forma de IM o bien en datos auxiliares que permiten complementar las imágenes médicas tradicionales con información suplementaria.

En este contexto, las herramientas informáticas y los métodos de tratamiento digital utilizados en la IM son indispensables tanto para la adquisición de las imágenes médicas (interpretación de señales, reconstrucción 3D) como para su procesamiento y análisis. Asimismo, las aplicaciones de visualización, especialmente las de tipo 3D, son de gran utilidad pues facilitan al médico el manejo e interpretación de cantidades importantes de información visual, auxiliándolo no sólo en el análisis y diagnóstico sino aún en las fases de planeación quirúrgica y en su ejecución operatoria.

En efecto, las técnicas de reconstrucción 3D usadas en la visualización permiten la creación de imágenes volumétricas que pueden proyectarse sobre cualquier plano elegido por el médico para obtener una visión 2D. Además, esas imágenes volumétricas pueden convertirse, con la ayuda de un navegador (browser) en verdaderos mundos virtuales interactivos al interior de los cuales el médico puede navegar, eligiendo hacer transparentes algunos órganos o tejidos de la imagen y así poder ver detrás de dichos elementos (ver por ejemplo [visible body](#) o el laboratorio de google [google body](#)). Esto es de mucha utilidad en la creación de simuladores realistas del cuerpo humano que se emplean en los entrenadores quirúrgicos de técnicas mínimamente invasivas (laparoscopia).

Una imagen médica se obtiene mediante el registro de la interacción de un determinado tipo de radiación con los tejidos del cuerpo humano cuando éste es expuesto a aquella. Así, una imagen médica es la representación espacial de una o varias propiedades físicas del cuerpo humano puestas en evidencia por su interacción con la radiación. El conocimiento de dicha interacción y de sus propiedades permite generar modelos para el análisis e interpretación clínica de las imágenes médicas.

De este modo, la Imagen Médica puede definirse de manera alternativa como el conjunto de técnicas que producen imágenes del interior del cuerpo humano que pueden verse como la solución de un problema matemático inverso. Es decir, que las propiedades estructurales y funcionales de órganos y tejidos vivos (causas) se inferen a partir de sus efectos (señales provistas por los sistemas de IM): i) radiación (rayos X) absorbida en diferentes cantidades por los tejidos (hueso, músculo, grasa, piel). Se dice que las técnicas de IM son no invasivas en el sentido de que no es necesario penetrar físicamente la piel para obtener las imágenes deseadas; sin embargo, a nivel de radiación (electromagnética o radiactiva) la IM sí es invasiva. Algunas de las modalidades de IM alteran el entorno químico y físico del cuerpo del paciente durante el proceso de adquisición de datos y señales (fotones de alta energía, radiaciones gamma, rayos X o campos magnéticos de varios tesla).

3.1.1. Principales hitos en la imagen médica

Puede considerarse que el inicio de la IM lo marcan los estudios sobre los rayos X hechos por el físico alemán Wilhelm Konrad Von Röntgen en 1895 que le valieron el primer Premio Nobel de Física, el de 1901 y que son el origen de las ampliamente conocidas radiografías. Más tarde, en 1923, aparece el uso de sustancias radiopacas que permiten la adquisición de imágenes radiológicas de tejidos blandos como las angiografías. Basándose en los trabajos del austriaco J. Radón publicados en 1917. Alan M. Cormack (sudafricano) publicó en 1967 su trabajo sobre la TAC (Tomografía Axial Computarizada), que más tarde fue utilizado por Godfrey N. Hounsfield (británico) para desarrollar el primer prototipo TAC. En 1971 con este prototipo se obtuvo la primera imagen craneal en la que se podía apreciar la masa encefálica con claridad y, en 1979, ambos obtuvieron el Premio Nobel de Fisiología y Medicina, el único que ha sido ganado por ingenieros.

En esa misma década, 70s, aparece la técnica de obtención de imágenes 2D y 3D mediante la Resonancia Magnética generada por Sir Peter Mansfield (británico) y Paul Lauterbur (norteamericano) y que les valió el Premio Nobel de Fisiología y Medicina de 2003. La técnica de MRI (Imágenes por Resonancia Magnética) se basa en los principios descubiertos por Félix Bloch y Edward Purcell, distinguidos con el Premio Nobel de Física en 1952, hasta ese entonces utilizados solamente en el ámbito de la Química. En 1980 aparece la f-MRI (imágenes por Resonancia Magnética Funcional) basada en la resonancia magnética de alta resolución de contraste que permite hacer estudios funcionales.

Después de 40 años de investigaciones en medicina nuclear, en 1980 aparecen los sistemas de generación de imágenes tomográficas 3D basadas en los principios de emisión de fotones SPECT (Single Photon Emission Computed Tomography) y las basadas en los principios de emisión de positrones PET (Positron Emission Tomography). En el año 2000 el escáner PET/CT de los doctores David Townsend y Ron Nutt fue nombrada la invención médica del año por la revista TIME.

La imagen médica también se utiliza con propósitos de investigación en la ciencia médica para estudiar la anatomía y la fisiología normal del ser humano. Aunque la obtención y estudio de imágenes de órganos y tejidos extirpados del cuerpo mediante la fotografía o la microscopía también se utiliza con fines médicos o de investigación no son considerados como parte de la IM, sino de la Patología.

3.2. Uso de imágenes médicas

Al hablar de las aplicaciones de la imagen médica hemos de tener presente que continuamente están apareciendo nuevas técnicas y nuevas terapias que hacen uso de las crecientes capacidades de las IM. Tradicionalmente podemos pensar en las imágenes radiológicas utilizadas para diagnóstico, pero también tenemos la posibilidad de utilizar las nuevas imágenes de difusión para estudiar la distribución de los haces de axones de las neuronas en el cerebro. La siguiente lista muestra una clasificación del uso que actualmente se está dando a las diferentes imágenes médicas que se estudiarán en la sección 3.3.

Diagnóstico clínico. Ha sido y es la principal aplicación de la imagen médica. La capacidad de visualizar los tejidos y sus características ha permitido un avance espectacular en las técnicas de diagnosis. Baste como ejemplos las aplicaciones de mamografía (ejemplo en el bloque II del libro), análisis de roturas o malformaciones óseas, o detección de problemas cardiovasculares.

Etapas preparatorias en cirugía Junto con la diagnosis, la utilización de las imágenes médicas en las etapas preparatorias de una cirugía es, hoy por hoy, una técnica completamente habitual. Todas las especialidades quirúrgicas; ortopedia, neurocirugía, laparoscopia, etc. hacen uso extensivo de la información preparatorio disponible. Esta información se utiliza para planificar la operación y/o el tratamiento

Etapas intraoperatorias en cirugía. Constituye actualmente el mayor reto para los ingenieros y médicos. La utilización masiva de las imagen médica intraoperatoria es una próxima meta para un futuro cercano. Los actuales sistemas de navegación quirúrgica, o los sistemas de imagen intraoperatoria (C-arm, CT intra-operatorio, MRI intra-operatorio, US, etc.) están mostrando su potencial para mejorar el tratamiento quirúrgico, pero todavía presentan dificultades que hacen que su utilización no sea tan habitual como en las etapas pre-operatorias. El entorno no estructurado y dinámico que se presenta en una cirugía hace que sea difícil combinar una tecnología potente para la captación de imagen con un sistema ergonómico y ligero que no entorpezca el trabajo del cirujano. Una aplicación de la imagen médica en etapas intra-operatorias son los procedimientos actuales de cateterismo intervencionista. En el bloque II de este libro se muestran ejemplos de investigaciones encaminadas a mejorar este aspecto.

Radiología intervencionista. Al igual que en el caso anterior, están apareciendo gran cantidad de aplicaciones de radiología intervencionista, pero sin embargo, todavía quedan muchos retos por resolver. El aporte de energía directamente al tejido dañado es un reto apasionante, que en el futuro cercano dará muchas nuevas aplicaciones. Los problemas de visualización on-line de los resultados de esta operación de aporte de energía, como por ejemplo los problemas de visualización con ultrasonidos (US) cuando se produce la ablación de tejidos y por tanto existe evaporación, son retos apasionantes para el estudiante de ingeniería.

Diagnóstico funcional. Además, las nuevas tecnologías que permiten realizar un diagnóstico funcional, como por ejemplo el fMRI, el US doopler o las imágenes de difusión neuronales, van a permitir grandes avances en las actuales técnicas de diagnóstico.

3.3. Tipos de imágenes médicas

A la vista de los principios físicos de interacción que se utilizan para la obtención de las imágenes médicas, podemos decir que existen dos tipos de imágenes médicas: las que se obtienen por proyección y por reconstrucción. Las primeras surgen de la obtención directa de la interacción entre radiación y tejido. En este tipo está la radiología convencional y digital, la medicina nuclear y los ultrasonidos (US). Las segundas son una extensión de las primeras mediante diferentes proyecciones y su posterior reconstrucción. A este grupo pertenece la TAC, SPECT y PET, MRI e imágenes en 3D. Aunque SPECT y PET también son imágenes de medicina nuclear.

Adicionalmente se puede considerar un tercer tipo de imagen médica como aquella generada mediante procesamiento a partir de alguna de las imágenes anteriores. En esta categoría podríamos incluir la angiografía, las imágenes fMRI, las imágenes de difusión, etc.

3.3.1. Imágenes de radiografía

Los rayos X fueron descubiertos por Röntgen en 1895 cuando experimentaba con gases. Para producirlos se necesita un cátodo y un ánodo con una diferencia de potencial muy alta, en torno a las decenas de KV. La naturaleza de los Rayos X es electromagnética y son radiaciones ionizantes, capaces de separar electrones de sus núcleos. Los rayos X tienen una capacidad de penetración muy alta, con lo que la capacidad de resolución de las radiografías no es muy buena, en comparación con otros métodos. Para obtener la imagen de un objeto, se interpone éste entre emisor y receptor. Al detector llegará distinta intensidad en función de por dónde haya pasado la radiación, de tal manera que la imagen formada es una imagen con diferentes tonos de grises, negro para las zonas más radiadas y blanco para las menos. Los estudios en torno a este tipo de imagen están enfocados a reducir la radiación para el paciente y mejorar la calidad, en general el contraste, de la imagen. Los problemas típicos en la formación de la imagen de radiografía es la dispersión y errores en la proyección. La dispersión ocurre cuando los rayos atraviesan un objeto perdiendo concentración de energía. Las maneras de evitarlo son utilizando colimadores, parrillas antidifusoras o bien separando el objeto del detector ya que los rayos que sufren difusión han sido más atenuados y afectarán menos. Los problemas de proyección pueden ser de tres tipos: (ver figura 3.1)

- (a) si la fuente no es puntual los contornos del objeto aparecen difuminados.
- (b) si el haz no es perpendicular al detector habrá una deformación en la imagen.
- (c) si hay dos objetos separados que no están en el centro del haz entonces puede que sus dos proyecciones se solapen.

La figura 3.2 muestra unas imágenes de rayos X. Este tipo de imagen constituye el 80 % del volumen de imágenes médicas actual, aunque cada vez más se tiende a su digitalización. Para obtener radiografías digitales se pueden digitalizar con un escáner, sería una obtención indirecta. Una forma directa es irradiar sobre una placa luminiscente (fluorobromo de bario) capaz de almacenar la información y después leer la información

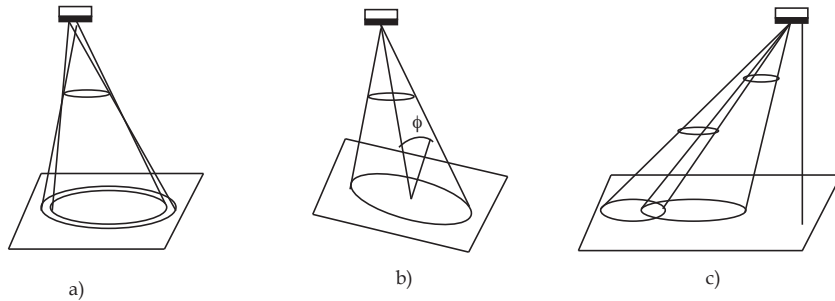


Figura 3.1: Tipos de errores de proyección



Figura 3.2: Imágenes de rayos X

mediante un láser He-Ne rojo de 633nm. A este último método se le llama radiografía digital por luminiscencia. Una ventaja, aparte de ser un método más directo, es que con un 10 % de la radiación anterior se obtiene la misma calidad, y esto es una ventaja muy a tener en cuenta. La desventaja es que la adquisición es más lenta, entorno a 2 ó 3 veces menor. Una tercera forma es obtener la imagen mediante vídeo con un CCD sensible a los rayos X puesto en el lugar de las placas.

Dado que la capacidad de penetración de los rayos X en el tejido es muy grande en ocasiones se utilizan sustancias radiopacas inyectadas en la zona que se quiere resaltar, esto es la angiografía. La imagen se forma mediante sustracción de dos imágenes, la imagen con la sustancia radiopaca y la imagen sin ella. Esta técnica se utiliza para poder visualizar tejidos blandos como estómago, intestinos, etc. En el ejemplo de la figura 3.3 se puede ver como los vasos sanguíneos, de color blanco, se pueden ver mucho más claros que con una radiografía convencional.

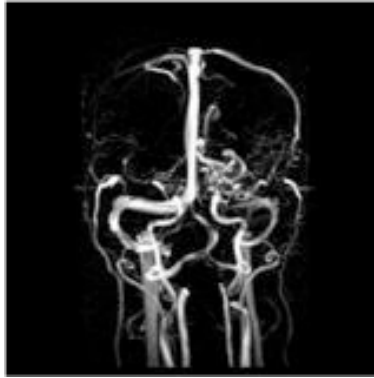


Figura 3.3: Imagen de angiografía

3.3.2. Imágenes por ultrasonidos

Los avances tecnológicos con US comenzaron a desarrollarse seriamente en los años 50 a raíz de las investigaciones militares de la II Guerra Mundial con el sonar. Su aplicación para fines más pacíficos, la medicina, supuso la aparición de sistemas de diagnóstico muy versátiles, baratos y portátiles. Actualmente las máquinas con esta tecnología tienen tamaños muy reducidos y se aplican en numerosos campos como el diagnóstico cardiológico, en ginecología, en cura de lesiones y cada vez más en cirugía. Los US son ondas acústicas que están por encima de espectro audible, por encima de 20 kHz. En medicina van desde 1 a 20 MHz. Las ondas acústicas son ondas de presión que deforman el medio en función de su elasticidad, o más bien son esta deformación. Evidentemente en el vacío no se pueden propagar. Se propagan muy bien a través de los medios acuosos y se refleja en las estructuras en función de su densidad. El sistema de imágenes por US consiste en un aparato llamado transductor que emite ondas de ultrasonidos. Estas ondas sonoras se transmiten hacia el área del cuerpo que se desea estudiar, y se recibe su eco. El transductor recoge el eco de las ondas sonoras y una computadora procesa la información, convirtiendo el eco en una imagen que se muestra por pantalla. La formación se basa en tres principios:

- La amplitud de las ondas recibidas depende de la densidad del objeto contra el que ha chocado la onda.
- Sabiendo la velocidad de propagación de las ondas en el medio se puede conocer la distancia a la que se encuentra el obstáculo.
- La variación en frecuencia de la onda recibida depende de la velocidad a la que se esté moviendo el objeto (efecto Doppler).

Las imágenes resultantes tienen unas resoluciones y definiciones muy bajas que no pueden competir con otras técnicas como las TAC. Pero tienen las ventajas de aportar

información funcional, cosa que no se puede obtener con TAC, además se pueden implementar en equipos portátiles y baratos, muy útiles para zonas rurales. Ejemplos de ecografías se pueden ver en la figura 3.4.



Figura 3.4: imagen US ecográfica

La técnica de ultrasonidos también es usada en el campo de la radiología intervencionista para la destrucción de cálculos en riñón, vesícula o vejiga y para disolver calcificaciones en caso de tendinitis muscular, mediante técnicas como el HIFU (high intensity focused ultrasound) que se explica en el capítulo 5.

3.3.3. Imágenes de Tomografía Axial Computerizada

La TAC fue descrita y puesta en práctica por el Dr. Godfrey Hounsfield en 1972, quien advirtió que los rayos X que pasaban a través del cuerpo humano contenían información de todos los constituyentes del cuerpo en el camino del haz de rayos, que, a pesar de estar presente, no se recogía en el estudio convencional con placas radiográficas, debido a su facilidad de penetración. El problema que tienen las radiografías es que los rayos X tienen un gran capacidad de penetración, con lo que se pierde capacidad para diferenciar objetos, sobre todo cuando éstos son tejidos blandos.

El TAC se basa en la reconstrucción de la información radiológica en varios planos de radiación, alrededor del paciente, en forma de secciones o rodajas, con lo que se puede obtener una reconstrucción volumétrica con mucho mayor contraste. Un sistema TAC consta de una fuente de rayos X colimada con un haz muy fino que permite espesores de solo unos pocos milímetros, un detector de radiación sensible a la energía recibida, el sistema mecánico que permita manejar y sincronizar todo esto y el sistema reconstructor que permita reconstruir la imagen a partir de las proyecciones.

3.3.3.1. Principio de funcionamiento

El aparato de TAC emite un haz muy fino de rayos X. Este haz incide sobre el objeto que se estudia y parte de la radiación del haz lo atraviesa. La radiación que no ha sido absorbida por el objeto, en forma de espectro, es recogida por los detectores. Luego el

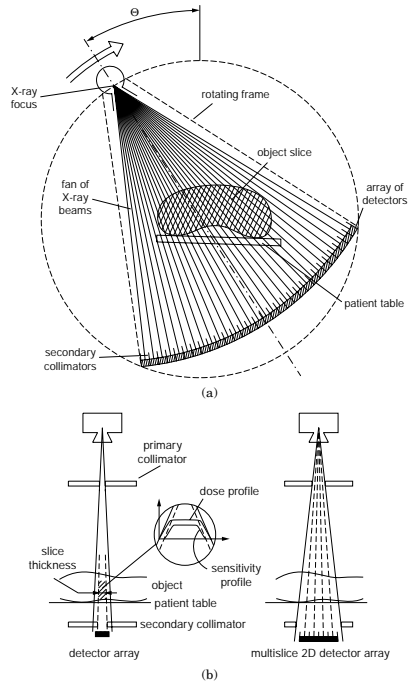


Figura 3.5: Principio de funcionamiento de un Scanner CT de 3ª generación. Extraído de [47]

emisor del haz, que tenía una orientación determinada (por ejemplo, estrictamente vertical a 90°) cambia su orientación (por ejemplo, haz oblicuo a 95°). Este espectro también es recogido por los detectores. El ordenador 'suma' las imágenes, promediándolas. Nuevamente, el emisor cambia su orientación (según el ejemplo, unos 100° de inclinación). Los detectores recogen este nuevo espectro, lo 'suman' a los anteriores y 'promedian' los datos. Esto se repite hasta que el tubo de rayos y los detectores han dado una vuelta completa, momento en el que se dispone de una imagen tomográfica definitiva y fiable.

Para comprender qué hace el ordenador con los datos que recibe lo mejor es examinar la figura 3.6:

- La imagen '1' representa el resultado en imagen de una sola incidencia o proyección (vertical, a 90°). Se trata de una representación esquemática de un miembro, por ejemplo un muslo. El color negro representa una densidad elevada, la del hueso. El color gris representa una densidad media, los tejidos blandos (músculos). El hueso, aquí, deja una zona de 'sombra'. Los músculos, una zona de 'penumbra'.
- La imagen '2' también representa el resultado en imagen de una sola incidencia o proyección, pero con un ángulo diferente (horizontal, a 180°)
- En la imagen '3' muestra qué hace el ordenador con las dos imágenes. Aquí la zona

de sombra ya está limitada al centro de la figura, pero la imagen presenta unos perfiles muy diferentes al objeto que se estudia (un cuadrado en vez de un círculo).

- En la imagen '4' el ordenador dispone de datos de cuatro incidencias: 45° , 90° , 135° y 180° . Los perfiles de la imagen son octogonales, lo que la aproximan mucho más a los contornos circulares del objeto real.

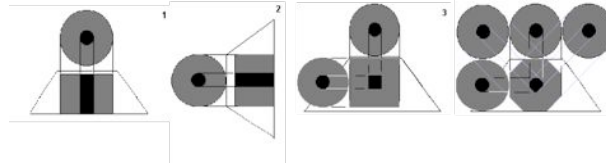


Figura 3.6: Principio de funcionamiento del TAC

Una vez que ha sido reconstruido el primer corte, la mesa donde el objeto reposa avanza (o retrocede) una unidad de medida (hasta menos de un milímetro) y el ciclo vuelve a empezar. Así se obtiene un segundo corte (es decir, una segunda imagen tomográfica) que corresponde a un plano situado a una unidad de medida del corte anterior. A partir de todas esas imágenes transversales (axiales) un computador reconstruye una imagen bidimensional que permite ver las distintas secciones de la pierna (o el objeto de estudio) desde cualquier ángulo. Los equipos modernos permiten incluso hacer reconstrucciones tridimensionales. Estas reconstrucciones son muy útiles en determinadas circunstancias, pero no se emplean en todos los estudios, como podría parecer. Esto es así debido a que el manejo de imágenes tridimensionales no deja de tener sus inconvenientes. Un ejemplo de imagen tridimensional es la imagen 'real'. Como casi todos los cuerpos son opacos, la interposición de casi cualquier cuerpo entre el observador y el objeto que se desea examinar hace que la visión de éste se vea obstaculizada. La representación de las imágenes tridimensionales sería inútil si no fuera posible lograr que cualquier tipo de densidad que se elija no se vea representada, con lo que determinados tejidos se comportan como transparentes. Aún así, para ver completamente un órgano determinado es necesario mirarlo desde diversos ángulos o hacer girar la imagen. Pero incluso entonces veríamos su superficie, no su interior. Para ver su interior debemos hacerlo a través de una imagen de corte asociada al volumen y aún así parte del interior no siempre sería visible.

3.3.4. Imagen nuclear

La medicina nuclear se basa en la utilización de isótopos radioactivos para formación de imágenes. Estos radioisótopos son inyectados en el tejido celular difundiendo por las cavidades mediante flujos, sobre todo por el torrente sanguíneo, y cuya actividad se puede detectar desde el exterior. La peculiaridad de estas imágenes es que aportan una información funcional del organismo, cosa que no pueden hacer otros tipos como CT o radiografía. Además si se combinan diferentes tomas con distintos ángulos se puede tener una reconstrucción tridimensional. Por contra la capacidad de contraste y resolución de estas imágenes es escasa. Los avances con aceleradores de partículas ha permitido que

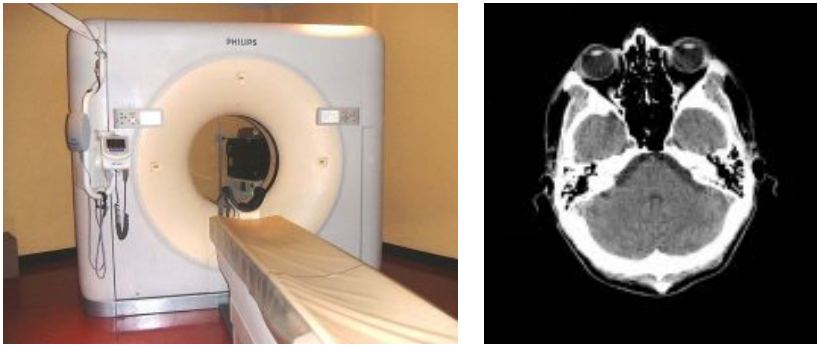


Figura 3.7: (a) Escáner TAC de 64 capas “Brilliance”, de Philips. (b) TAC de una cabeza humana

esta técnica haya tenido un gran desarrollo. Los radioisótopos son compuestos químicos, pero normalmente se administran como radiofármacos, moléculas que actúan en una determinada función u órgano. Deben tener un tiempo de vida corto a fin de irradiar lo menos posible al paciente. Normalmente se introducen mediante una inyección pero también se puede hacer mediante inhalación, vía oral, etc. En la actualidad, la detección de la radiación se realiza en la llamada cámara de Anger o gammacámara. A partir de varias orientaciones se puede hacer una reconstrucción tomográfica. Estas son la SPECT y PET. Los cristales de centelleo son de yoduro sódico activados con talio, sensibles a la radiación gamma, y muy caros. En ocasiones se suele anteponer al cristal un colimador o rejilla para eliminar las radiaciones en las direcciones no deseadas.

3.3.4.1. SPECT

La tomografía axial computerizada (TAC) introdujo grandes avances en las exploraciones médicas y también en medicina nuclear. Supuso el paso de 2 a 3 dimensiones. Al igual que las TAC, las SPECT (Single Photon Emission Computer Tomography) analiza la radiación entorno a una circunferencia exterior al cuerpo a analizar mediante detectores colimados. La diferencia es que con TAC se irradia y con SPECT la radiación viene del interior, de los radioisótopos. Por contra con SPECT no se consigue la misma nitidez en las imágenes como con TAC. Para obtener la imagen tomográfica se utiliza una gammacámara rotatoria que proporciona datos cualitativos e incluso cuantitativos. Si se toman distintos planos, se puede construir una representación 3D lo que permite realizar los cortes dentro del cuerpo con la orientación que se quiera. La reconstrucción de las proyecciones se consigue mediante cálculos matriciales al igual que las TAC.

3.3.4.2. PET

Constituye la parte más reciente de la medicina nuclear, se forma a partir de emisores de positrones, es decir, radiaciones. Las radiaciones beta tienen un tiempo de vida corto, se propagan hasta encontrarse con un electrón, entonces se aniquilan y se producen

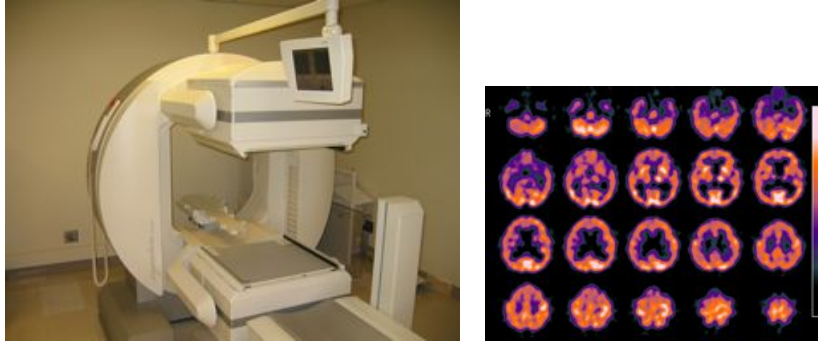


Figura 3.8: (a) Escáner “Precedence”, de Philips. (b) SPECT de una cabeza humana

dos fotones que se alejan en sentidos contrarios; si el radioisótopo emite otro par de fotones entonces la posición del positrón está en esa misma línea. Al igual que en TAC y SPECT la detección se produce a lo largo de una circunferencia que rodea el cuerpo de análisis. Los sistemas PET son extremadamente caros pero son de interés para el estudio metabólico de tejidos y órganos, como por ejemplo, el metabolismo de la glucosa cerebral, el metabolismo del oxígeno o procesos miocárdicos.

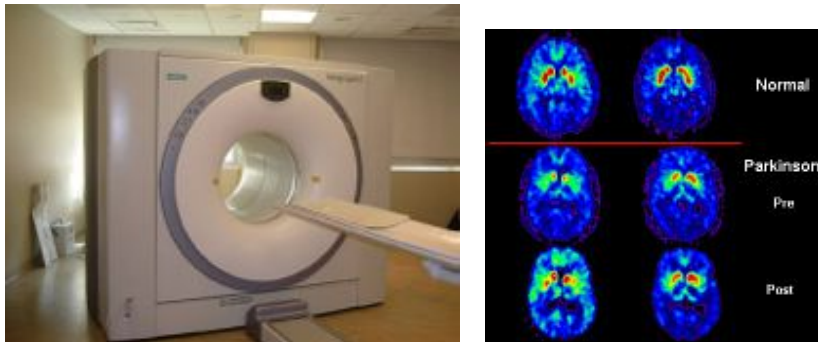


Figura 3.9: (a) Escáner PET (b) PET de una cabeza humana

3.3.5. Resonancia magnética

La RM es un método de diagnóstico que ha resultado ser de gran utilidad en la medicina desde su introducción en 1976. La RM se basa en la capacidad de algunos núcleos para absorber ondas de radiofrecuencia cuando son sometidos al efecto de un campo magnético. Dicha capacidad genera una señal que es detectada por un receptor y tratada en un ordenador de manera similar a como lo hace la TAC para producir imágenes. Respecto a otras técnicas como los rayos X y la TAC, se usa cada vez más por sus mejores prestaciones, como permitir cortes más finos, y en varios planos, ser más sensible para demostrar accidentes vasculares cerebrales, tumores y otras patologías, y

no utilizar radiaciones ionizantes. Como desventajas cabe mencionar su mayor coste, el prolongado tiempo para obtener las imágenes y el tener que excluir a portadores de marcapasos y otros objetos metálicos intracorpóreos. La RM se basa en el hecho de que por debajo de los 60 Mhz hay una ventana de radiofrecuencia que produce fenómenos de resonancia en los tejidos, en concreto con los núcleos de hidrógeno de las moléculas de agua, dando información de la densidad de átomos. Además, el spin de éstos también podrá proporcionar información fisiológica de los tejidos. Las bases de este fenómeno fueron sentadas en 1946 por Felix Bloch y Edward M. Purcell. En 1972 R. Damadian probó que los tejidos cancerosos tenían diferentes comportamientos de resonancia. La aplicación era inmediata. Se aplicaron los mismos métodos de reconstrucción propuestos por P. Laurtebur en las TAC. A partir de aquí el desarrollo fue espectacular hasta ser uno de los pilares fundamentales de la imagen médica. Los fundamentos físicos son:

- Se orientan los átomos que se van a analizar, normalmente de hidrógeno, en dirección de un campo magnético.
- Se les aplica un campo electromagnético a su frecuencia de resonancia para cambiar su momento magnético.
- Cuando el impulso de radiofrecuencia cesa el momento magnético vuelve a su posición inicial liberando energía electromagnética que se puede captar desde el exterior con un detector o antena.
- Los datos obtenidos se procesan para formar la imagen de resonancia en niveles de grises en función de la energía captada de cada región.

También son de interés los núcleos C_{13} , F_{19} , P_{31} y Na_{23} , que son también susceptibles de magnetización. Pero es el H el más utilizado por ser el más abundante el cuerpo humano y por el más fácil de tratar magnéticamente. El tipo de tejido viene marcado por los tiempos de respuesta de los protones. Teniendo en cuenta todos estos parámetros se puede llegar a obtener una imagen capaz de distinguir tejidos blandos con un gran contraste, como puede ser sangre y grasa, además con una resolución espacial muy buena. Evidentemente esta técnica también tiene sus inconvenientes, y es que las máquinas necesarias son muy aparatosas, grandes y costosas. Además, dado el gran número de cálculos que requiere la formación de las imágenes supone que el proceso sea muy lento e incómodo para el paciente. En la formación de las imágenes se pueden utilizar tres métodos: excitación local (por puntos, líneas o planos), por retroproyección filtrada o mediante transformadas de Fourier (igual que en TAC).

Un excelente material adicional sobre MRI se encuentra recopilado en www.mri-tutorial.com. De especial interés son los video-tutoriales sobre los principios del MRI y del fMRI.

3.3.5.1. Resonancia magnética funcional

La resonancia magnética funcional (fMRI) utiliza los principios generales que relacionan estrechamente la actividad neuronal con el metabolismo y el flujo sanguíneo, puede registrar cambios hemodinámicos cerebrales que acompañan la activación neuronal; además permite la evaluación funcional de regiones responsables de la sensorialidad,

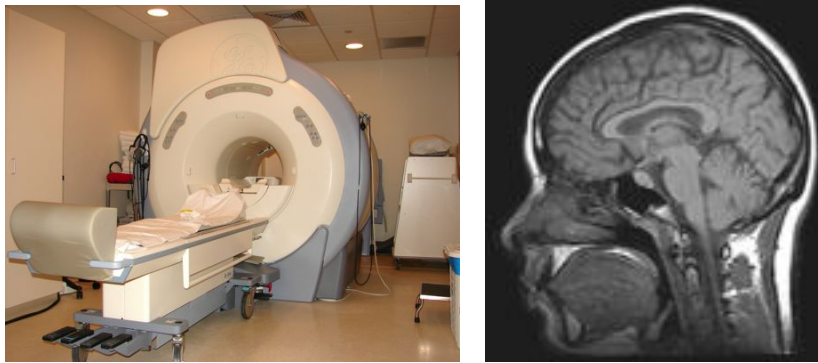


Figura 3.10: (a) Escáner MRI (b) MRI de una cabeza humana

motricidad, cognición y procesos afectivos en cerebros normales y patológicos. La fMRI aporta información importante en el estudio de actividad cerebral cortical y puede obtener al mismo tiempo imágenes anatómicas adecuadas para una ubicación precisa del área de interés incluyendo imágenes angiográficas o estudio espectroscópico.

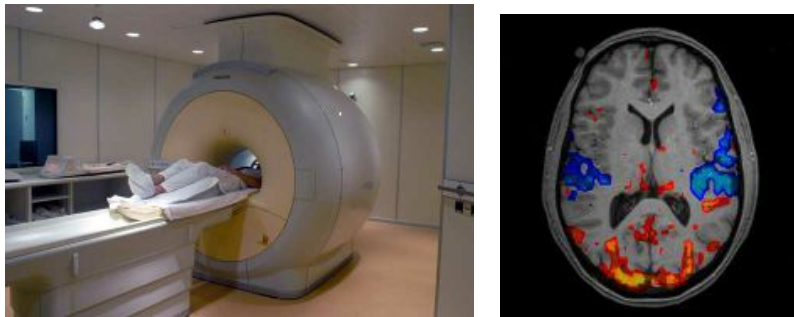


Figura 3.11: (a) Escáner fMRI (b) fMRI de una cabeza humana

Bases y funcionamiento de la Resonancia Magnética Funcional

La fMRI se basa en el acoplamiento neurovascular, que consiste en que toda neurona cuya actividad aumenta, requiere mayores cantidades de energía (glucosa) y oxígeno para funcionar adecuadamente. Por lo tanto, la activación de un área cerebral se acompaña de un incremento de su perfusión vascular. La fMRI mide este incremento en el flujo de sangre. En ese sentido la fMRI se parece a la tomografía por emisión de positrones (PET), pero a diferencia de ésta es una técnica incruenta que no requiere de la inyección de material radioactivo potencialmente dañino.

Existen varios métodos, que permiten en fMRI determinar el incremento del riego sanguíneo acoplado a una mayor activación neuronal. El más usado de todos es el

BOLD; el cual consiste en identificar el área cerebral activada por medio del contraste que se obtiene de la hemoglobina oxigenada. El hierro de la hemoglobina desoxigenada (sin oxígeno) es paramagnético; por el contrario, es diamagnético en la hemoglobina oxigenada (oxihemoglobina). La desoxihemoglobina produce un desfase de los espines y en consecuencia una disminución de la señal en T2. Cuando se activa un área cerebral (cortical) se produce un incremento de su perfusión sanguínea que excede largamente el consumo de oxígeno. En el área activada hay una gran cantidad de sangre oxigenada que remueve las no-homogeneidades del campo magnético (que dependen de la sangre no oxigenada) y que provoca que el desfase de los espines no se produzca y haya una mayor señal que emana del área activa con relación a la señal que producía en condiciones de reposo, en el que proporcionalmente hay menos sangre oxigenada y más sangre desoxigenada. Dentro de sus aplicaciones están la investigación neuropsicológica y neurológica, así como su utilidad en clínica y en neurocirugía.

3.3.5.2. Imagen de difusión

La técnica de difusión por resonancia magnética (dMRI) está basada en la detección in vivo del movimiento de las moléculas de agua. Las imágenes de resonancia magnética de difusión (dMRI) son imágenes de MRI en las que se visualiza el proceso de difusión de las moléculas, principalmente las moléculas de agua, en los tejidos biológicos. La principal aplicación de las imágenes dMRI está siendo el estudio y análisis de los desórdenes neurológicos, aunque recientemente se están aplicando también al abdomen y la pelvis. Las moléculas libres (ej. un recipiente de agua) tienen un movimiento browniano (es decir, al azar), sin embargo, las moléculas de agua en el organismo tienen una difusión restringida debido a la interacción con diferentes obstáculos, como macromoléculas, membranas, etc. El patrón de difusión puede revelar los detalles microscópicos del tejido estudiado.

A partir de la información de dMRI se pueden obtener diferentes imágenes, como la diffusion tensor imaging (DTI) en la que se puede observar la estructura de las fibras (axones de las neuronas) de la materia blanca y se generan modelos de conectividad del cerebro. Esto es muy útil para el análisis y planificación en neurología.

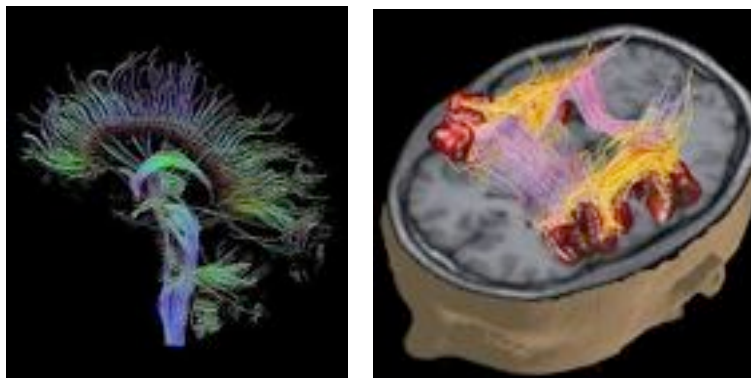


Figura 3.12: (a) imagen DTI (b) Superposición de la información DTI sobre una cabeza humana

3.4. Análisis de imagen médica

Las técnicas de análisis y procesamiento de imagen médica son esencialmente las mismas utilizadas en cualquier tipo de imagen digital. Se recomienda al lector novel consultar la documentación adicional sobre visión artificial, como por ejemplo el documento [vision.pdf](#) relativo a la toolbox de [Machine Vision](#) de Peter Corke (en inglés) o los apuntes del profesor C. Platero [link](#).

Además, es muy recomendable la lectura de textos clásicos de visión artificial como los recomendados al final de este capítulo.

En cuanto a la teoría detrás del análisis y procesamiento particular de imágenes médicas, se recomienda la lectura del libro *Medical Image Processing, Reconstruction and Restoration*, de Jirí Jan [47]. En la parte III de este libro, en los capítulos 9 a 14 se tratan de forma profusa todos los fundamentos del procesamiento de imágenes médicas.

Siguiendo el espíritu práctico de este libro, las etapas clásicas de procesamiento de imagen, como son los *filtros, segmentación, registro y reconstrucción 3D de imágenes* se van a estudiar con ayuda de los excelentes tutoriales del programa *3DSlicer* que se pueden encontrar en este [enlace](#). Se recomienda realizar el segundo ejemplo de este capítulo para adquirir destreza en el manejo de este programa.

Adicionalmente la segunda parte de este libro contiene varios ejemplos resueltos de aplicaciones software realizadas sobre VTK e ITK, como por ejemplo la reconstrucción 3D de imágenes CT utilizando el algoritmo Marching Cubes, o la segmentación de imagen mediante técnicas de umbralización en el ejemplo de la mamografía.

3.5. ROBOTICA MEDICA APLICADA

En esta sección de ROBOTICA MEDICA APLICADA se presentan dos ejercicios sencillos, el primero de ellos realizado con las herramientas de Matlab[®] para imagen médica DICOM, y el segundo, más completo, adaptado a partir de los tutoriales de *3DSlicer* (www.slicer.org). Esta potente aplicación es una aplicación front-end, modular y de código abierto para visualizar y analizar imágenes médicas. El programa está diseñado para ser multiplataforma. Está mantenido por una gran comunidad que ha aportado más de 1.5 millones de líneas de código. Los ejecutables y el código del proyecto se distribuyen bajo licencia BSD.

3DSlicer utiliza las librerías [ITK](#) y [VTK](#) entre otras. Estas librerías serán utilizadas en la segunda parte del libro y en los tutoriales de *Opensurg*, si bien aquí se ha preferido utilizar un programa ejecutable como *3DSlicer* que permite a todos los usuarios trabajar con las imágenes médicas.

3.5.1. Ejemplo de lectura de imágenes DICOM con Matlab[®]

El conjunto de imágenes correspondiente a la cavidad abdominal está almacenado en formato DICOM (*Digital Imaging and Communication in Medicine*). Es un estándar utilizado para el intercambio, manejo, almacenamiento y transmisión de imágenes médicas. Especifica la forma en que se debe guardar la información relativa de las exploraciones

en archivos digitales. Tienen una extensión `.dcm` y consisten en dos partes diferenciadas (figura 3.13).

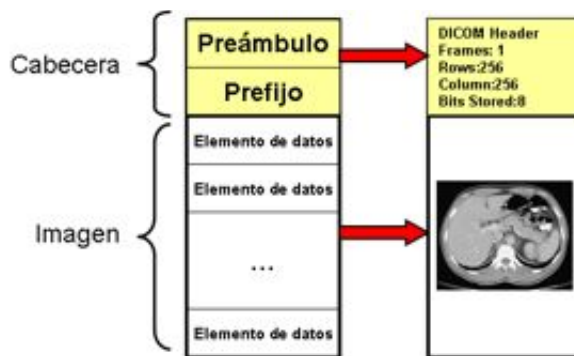


Figura 3.13: Ejemplo de estándar DICOM

- Cabecera: En la cabecera se especifican datos administrativos (datos del paciente, tipo de exploración, etc.) y los datos sobre la imagen (tamaño, etc.). Está compuesta por:
 - Preámbulo: Puede contener información sobre el nombre de la aplicación usada para crear el fichero o información que permita a aplicaciones acceder a los datos.
 - Prefijo: Diferencia si un fichero es DICOM o no.
- Imagen: Una imagen se define como una función bidimensional compuesta por su coordenadas espaciales (x,y) y su intensidad de gris. Consisten en un mapa de densidades.

Un *dataset* de imágenes es una serie ordenada de imágenes DICOM.

Para realizar el presente ejercicio necesitamos el software Matlab[®]. Los pasos a realizar son:

1. Establecemos como directorio de trabajo aquel donde hemos almacenado las imágenes que acompañan a este ejemplo en el cd adjunto. Las imágenes estarán nombradas como *“brain_xxx.dcm”*.
2. Abrimos la herramienta `imtool`. En esta interfaz se abre la imagen *brain_001.dcm* y utilizando el control de contraste de la barra de herramientas de la imagen variamos el contraste (`imcontrast`) para visualizarla correctamente (figura 3.15). También podemos leer las cabeceras de la imagen DICOM con el control de información de la barra superior. En el ejemplo utilizado se observa, entre otros datos, que es una imagen de 256x256, obtenida con un MR de GE Medical Series (figura 3.16).

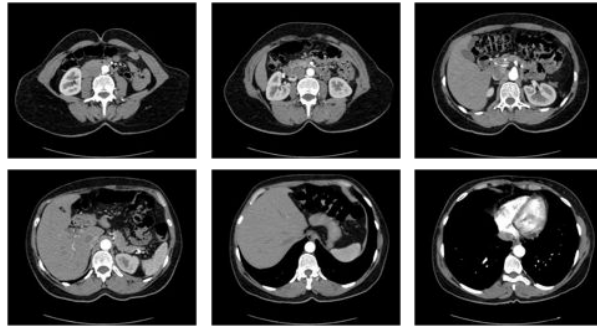


Figura 3.14: Ejemplo de Dataset de la cavidad abdominal

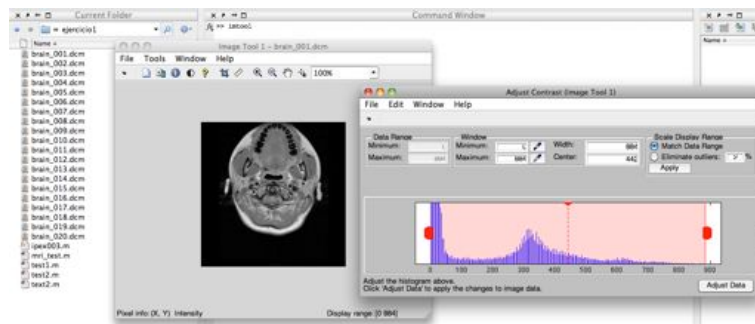


Figura 3.15: Herramientas imtool e imcontrast de Matlab®

3. Vamos a realizar lo mismo pero con comandos en línea de Matlab®. En primer lugar se recomienda leer la información del comando `dicomread`, para asegurarnos que tenemos la toolbox en nuestra instalación de Matlab®. Observar como se debe introducir el comando `imcontrast` tras abrir la imagen, ya que si no lo hacemos la visualización no es correcta. También se observa el comportamiento del comando `contour` (figura 3.17).

```

>help dicomread
>info=dicominfo ('brain_001.dcm')
>img1=dicomread(info);
>figure, imshow(img1), imcontrast
5 >figure, contour(img1)

```

4. Ahora vamos a ejecutar el fichero `test1.m` para la lectura de un dataset de 20 imágenes (figura 3.18).

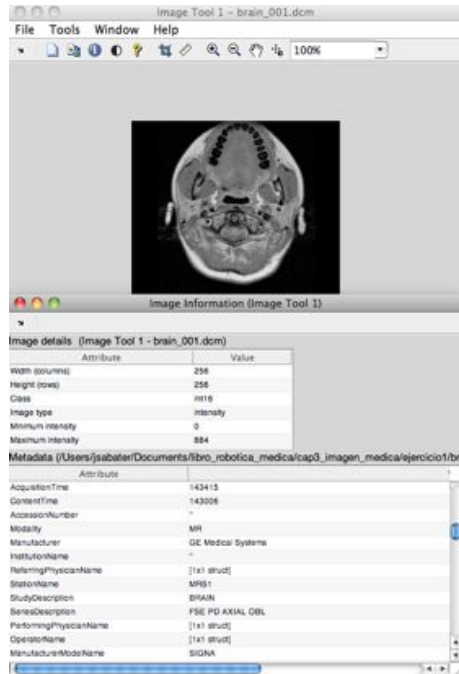


Figura 3.16: Lectura de la cabecera DICOM con imtool

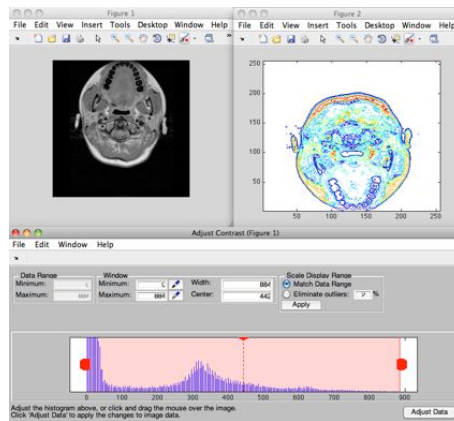


Figura 3.17: Comandos en linea para tratamiento de imágenes DICOM

```

% Ejemplo 1 para acceso a imagenes DICOM
% Primero reservamos memoria para 20 imagenes de 1 canal
% y dimensiones 256x256.
X = repmat(int16(0), [256 256 1 20]);
5 % Leemos las imagenes.
for p=1:20
    filename = sprintf('brain_%03d.dcm', p);
    X(:,:,1,p) = dicomread(filename);
end
10 % Mostramos un display multiple.
montage(X);
% ajustamos el contraste para visualizar correctamente
imcontrast;

```

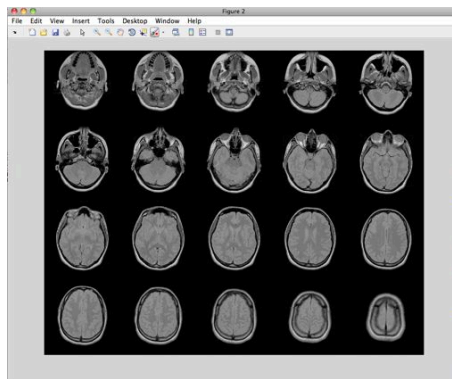


Figura 3.18: Ejecución del script *test1.m* de Matlab®

5. Ahora ejecutamos el fichero *test2.m* que contiene una función *test2* cuyo parámetro es el número de imágenes que se quiere visualizar. Lo ejecutamos como *test2(4)* o como *test2(6)*.

```

% Ejemplo 2. Accediendo a datos cabecera DICOM
% la funcion test2 tiene como parametro el numero de imagenes
% que queremos visualizar.
function test2(nFrames);
5 info=dicominfo('brain_001.dcm');
nRows = info.Rows;
nCols = info.Columns;
nPlanes = info.SamplesPerPixel;
X = repmat(int16(0), [nRows, nCols, nPlanes, nFrames]);
10 for p = 1:nFrames
    fname = sprintf('brain_%03d.dcm', p);
    X(:,:,,p) = dicomread(fname);
end

```

```

15 % El siguiente codigo evita tener que usar imcontrast.
minPixels = repmat(0, [1, nFrames]);
maxPixels = repmat(0, [1, nFrames]);
for p = 1:nFrames
    fname = sprintf('brain_%03d.dcm', p);
    info = dicominfo(fname);
20 minPixels(p) = info.SmallestImagePixelValue;
    maxPixels(p) = info.LargestImagePixelValue;
end
b = min(minPixels);
m = 2^16/(max(maxPixels) - b);
25 Y = imlincomb(m, X, -(m * b), 'uint16');
montage(Y)

```

6. Ahora vamos a ejecutar el ejemplo `test3.m` que muestra cómo se utilizan los comandos en línea para generar de forma fácil los planos sagitales y coronales a partir de los planos horizontales de un dataset de imágenes MRI. Los datos que se utilizan en este ejemplo se encuentran almacenados en el *dataset* `mri` que viene instalado por defecto con la instalación de la toolbox. El código del ejemplo se muestra a continuación, y el resultado en la figura 3.19.

```

% Ejemplo 3 para creacion de planos sagital y coronal
% ver comandos immovie, maketform y tformarray
% mri son datos ejemplo que vienen con la toolbox de Matlab
5 load mri;
figure;
immovie(D,map);
montage(D,map);
title('Horizontal Slices');
%-----STEP 1. Vamos a generar los planos sagitales
10 T3 = maketform('affine',[-2.5 0 0; 0 1 0; 0 0 0.5; 68.5 0 -14]);
S = tformarray(D,T3,R3,[4 1 2],[1 2 4],[66 128 35],[],0);
figure;
immovie(S,map);
S2 = padarray(S,[6 0 0 0],0,'both');
15 montage(S2,map);
title('Sagittal Slices');
%-----STEP 2. Vamos a crear los planos coronales
T4 = maketform('affine',[-2.5 0 0; 0 1 0; 0 0 -0.5; 68.5 0 61]);
C = tformarray(D,T4,R3,[4 2 1],[1 2 4],[66 128 45],[],0);
20 figure;
immovie(C,map);
C2 = padarray(C,[6 0 0 0],0,'both');
montage(C2,map);
title('Coronal Slices');

```

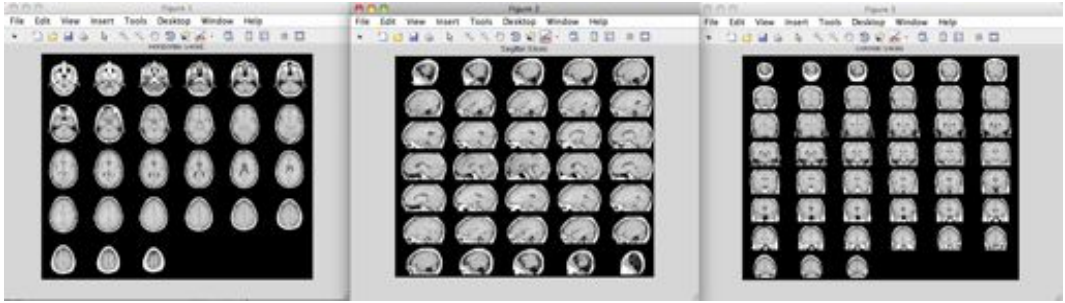


Figura 3.19: Ejecución del script *test3.m* de Matlab®

3.5.2. Ejemplo de segmentación y reconstrucción 3D de la arteria aorta

En este ejemplo se van a utilizar las imágenes del *dataset* CALIX obtenido del repositorio de OSIRIX. Antes de comenzar la segmentación de la cavidad abdominal se inicia el software *3DSlicer* desde el icono directo. Entonces aparece la interfaz gráfica de usuario (figura 3.20) mostrando el módulo por defecto *SlicerWelcome*. Los módulos proporcionan las diferentes funcionalidades del programa.

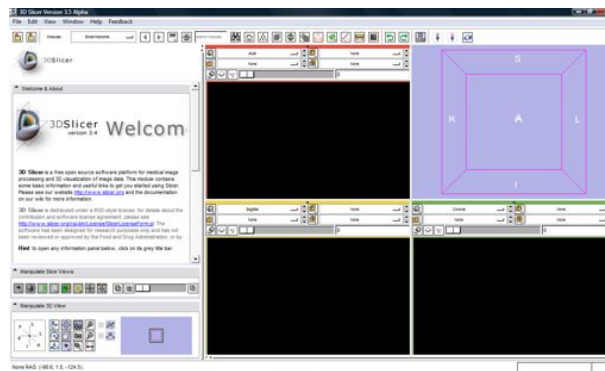


Figura 3.20: Módulo *SlicerWelcome*

A continuación se carga el conjunto de imágenes CT de la cavidad abdominal con formato DICOM siguiendo **File**->**AddVolume**, buscando el directorio donde estén almacenadas las imágenes del *dataset* CALIX y seleccionando el primer archivo *IM-0001-0001.dcm*. Las imágenes DICOM aparecen en la pantalla de visualización *2DViewer*. En la figura 3.21 se observa el resultado de la carga de las imágenes. En la parte Axial (izquierda superior) se muestra la imagen que contiene el archivo central del dataset, es decir, la imagen del archivo *IM-0001-0172.dcm*, mientras que la parte inferior izquierda (Sagittal) y la inferior derecha (Coronal) contienen las imágenes laterales y frontales. En la

parte derecha superior se puede observar el cruce de los tres cortes anteriores. Las barras de desplazamiento situadas en la parte superior de las imágenes permiten visualizar las rebanadas de imágenes que van seguidas de las mostradas.

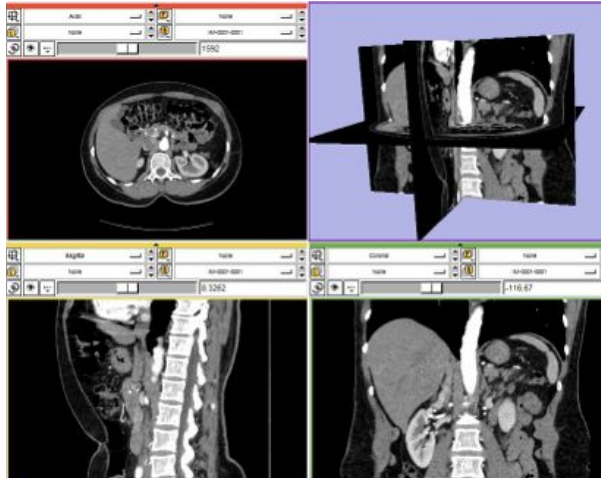


Figura 3.21: Visualización de las imágenes DICOM

En primer lugar, hay que extraer la arteria aorta junto con sus ramificaciones principales utilizando un método de la técnica *Level Set* para la segmentación de vasos. En este caso se utilizará el método llamado *Fast Marching*. El módulo de 3DSlicer que utiliza este método es el *Fast Marching Segmentation*. Antes de iniciar la técnica de segmentación, se realiza una extracción del *dataset* que contiene toda la arteria aorta y sus ramificaciones mediante un ROI (Region Of Interest). Gracias a esta extracción, se utilizarán menos recursos computacionales en el proceso de segmentación permitiendo la obtención de mejores resultados añadiendo menos semillas en el algoritmo *Fast Marching*. Esta tarea se realiza cumpliendo dos pasos:

1. En primer lugar se crea un objeto ROI llamado *Aorta* a través del módulo ROI. Después se sitúa en la parte de las imágenes que contiene toda la aorta. Para colocarlo en la posición deseada se puede realizar desplazando los vértices del objeto en la pantalla 3D de visualización o moviendo las barras de desplazamiento de éste módulo (figura 3.22).
2. En segundo lugar se debe extraer la parte de la imagen que está ocupando el objeto ROI creado en el paso anterior. Esta tarea se realizará con el módulo *Extract SubvolumeROI*. Primero se configura dicho módulo asignando como entrada el *dataset* de la cavidad y el objeto ROI. Para completar la extracción se pulsa el botón *Do ROI resample* (figura 3.23).

Una vez finalizado el proceso de extracción de la zona que contiene la aorta en el dataset de la cavidad abdominal, se selecciona el módulo *Fast Marching Segmentation* para

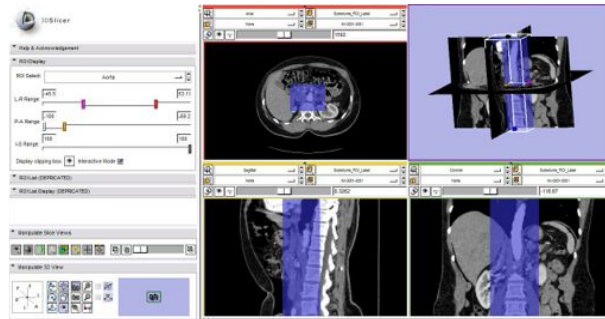


Figura 3.22: Creación del ROI

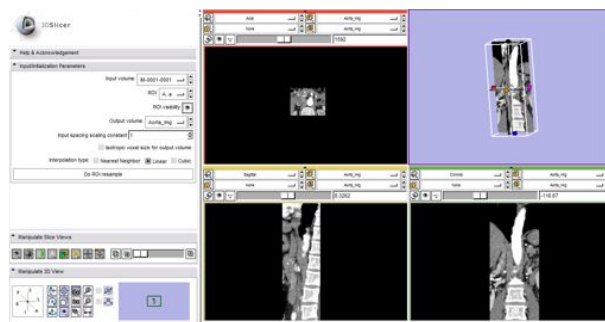


Figura 3.23: Extracción de la aorta con objeto ROI

realizar la segmentación (figura 3.24). Este módulo requiere de una configuración de varios pasos:

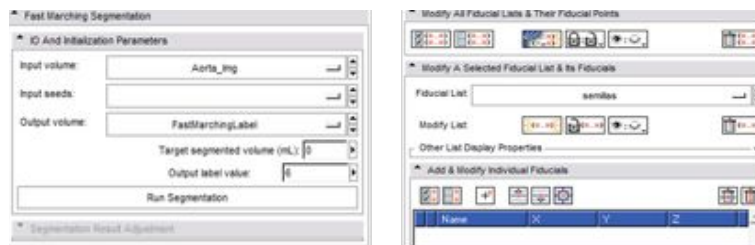


Figura 3.24: Modulo de Fast Marching Segmentation

1. Primero se incluye en Input volume la estructura que tiene que ser segmentada.
2. En segundo lugar se crea una lista de semillas para extraer y aislar el núcleo de

la aorta. Estas semillas se expanden en función de la intensidad media observada en las regiones luminosas de los vasos. El núcleo se obtiene mediante el uso de un umbral inicial, basado en una alta intensidad, conectando las semillas a la aorta. En `Input seeds` se crea un `FiducialList` de nombre `Semillas` que contiene los puntos situados en la región que tiene que ser segmentada. A continuación se añaden las semillas mediante el módulo `Fiducials` (figura 3.24). Una vez seleccionado el módulo `Fiducials` se visualizan las imágenes de la cavidad abdominal donde se quieren añadir las semillas. En la figura 3.25 se muestra un ejemplo de colocación de una semilla justo encima del núcleo de la aorta. Estas semillas se pueden incluir con la herramienta `Place a new object` del menú principal. La posición de las semillas influye directamente en el proceso de segmentación, por lo que se realizan diferentes pruebas con múltiples colocaciones de semillas para comprobar cuales generan una mejor segmentación.

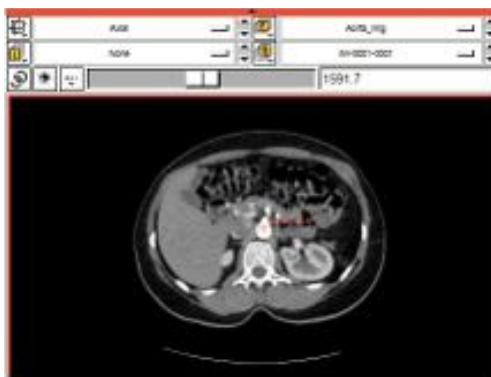


Figura 3.25: Colocación de la primera semilla

3. En `Output volume` se crea un `labelmap` que contendrá los resultados de la segmentación.
4. En `Target segmented volume (mL)` se incluye un número positivo que estima el volumen esperado de la estructura de segmentación en milímetros y en `Output label value` se asigna la capa de píxeles del resultado. Se añadirán los valores 120 mL y 2 respectivamente.
5. Una vez configurado, se pulsa `Run Segmentation` para construir el `labelmap` de nombre `Aorta_Label` que corresponde al núcleo de la aorta y a sus ramificaciones.
6. Al terminar el procesamiento aparece una estimación de segmentación con el color correspondiente a la capa de píxeles seleccionada, que en este caso es de color rosa claro. En la figura 3.26 se muestra un ejemplo de imagen con segmentación del núcleo de la aorta situada en el corte central de la cavidad abdominal.
7. Esta estimación puede ser ajustada adaptando la evolución del contorno. Esta tarea se realiza con el panel `Segmentation Result Adjustment` del mismo módulo.

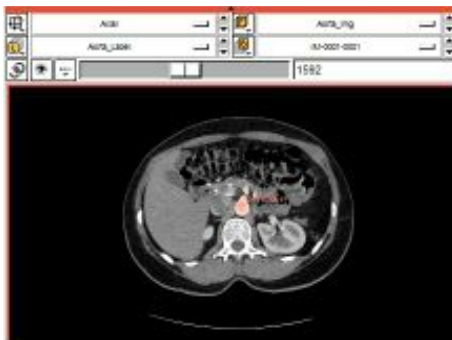


Figura 3.26: segmentación de la aorta

8. En este paso se modifica la barra de desplazamiento **Output segmentation volumen (mL)** para cambiar el contorno y obtener mejores resultados. En la figura 3.27 se muestra que la estimación de la segmentación se introduce en una parte de la columna vertebral (imagen izquierda). Al disminuir el volumen de segmentación se modifica el contorno, permitiendo conseguir que la parte no deseada desaparezca (imagen derecha).

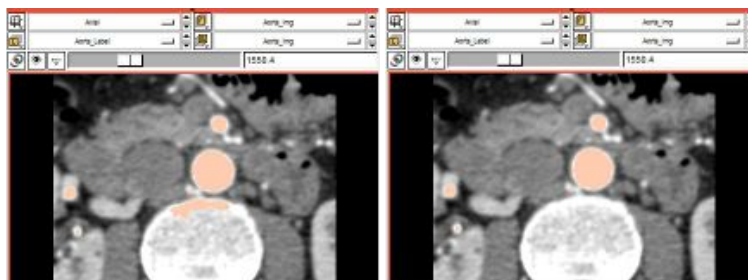


Figura 3.27: Adaptación del contorno de segmentación

9. Una vez comprobado que ninguna parte de la estimación está situada en una zona no deseada, se pulsa el botón **Accept Segmentation Result** para generar el labelmap de la aorta y sus ramificaciones.
10. A continuación se utiliza el módulo **Model Maker** para crear estructuras 3D de imágenes segmentadas (labelmap) de manera automática. Se pulsa **Apply** para generar el volumen de la aorta. Pasados unos segundos se obtiene la reproducción 3D de la aorta y sus ramificaciones. En la figura 3.28 se muestran diferentes perspectivas del volumen 3D.

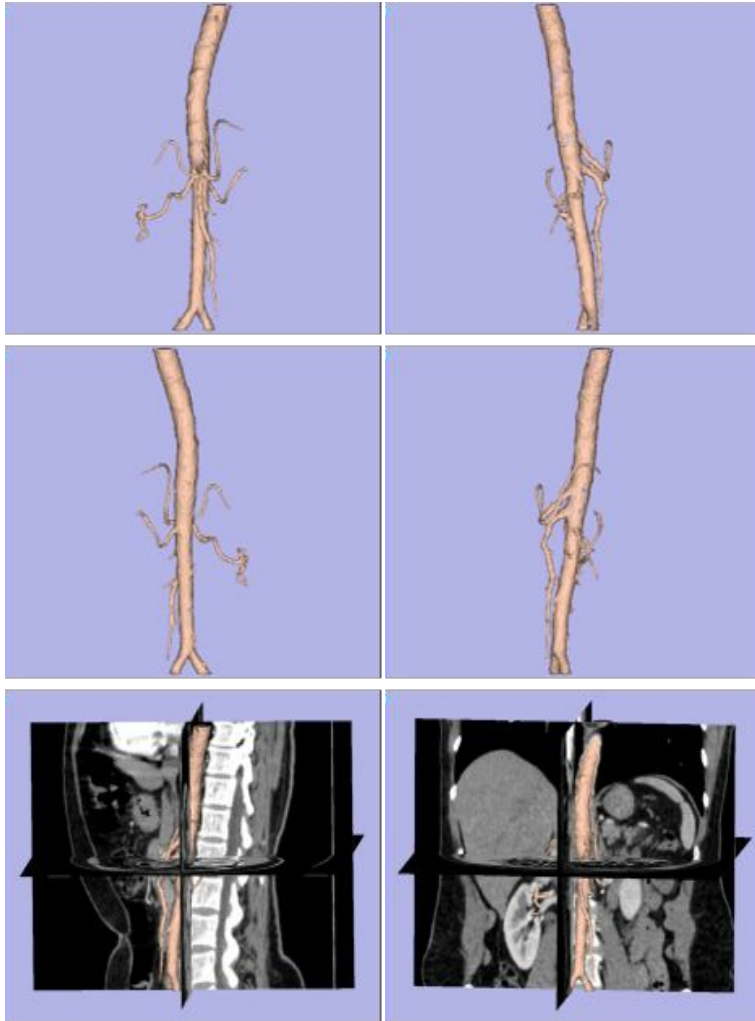


Figura 3.28: Representación 3D de la arteria aorta y sus ramificaciones

3.6. Resumen del capítulo

Los aspectos clave que han sido tratados en este capítulo se resumen en los siguientes puntos:

1. La imagen médica se puede definir como el conjunto de técnicas y procesos utilizados para crear imágenes del cuerpo humano, de sus partes o aún de sus funciones para propósitos clínicos, entendidos como aquellos procedimientos médicos que permiten detectar, analizar y diagnosticar enfermedades o patologías.

2. Los tipos de imágenes médicas estudiadas son:
 - Imágenes de radiografía
 - Imágenes por ultrasonidos
 - Imágenes TAC
 - Imágenes nucleares. SPECT y PET
 - Imágenes de resonancia magnética: MRI, fMRI, DWI
3. Aprender a utilizar el software 3DSlicer para análisis de imagen médica.

3.7. Ejercicios

1. En el cd anexo al libro se han añadido los tutoriales de la versión 4.2 del *3DSlicer*, así como toda la documentación y vídeos adicionales. Se recomienda realizar estos ejercicios en el orden establecido, empezando por el *SlicerWelcomeTutorial*.
2. De igual forma que en el ejemplo del apartado anterior, se pide utilizar el *dataset* MECANIX (<http://pubimage.hcuge.ch:8080>) para obtener una reconstrucción 3D de la estructura ósea de la cadera y la columna. Utilizar para ello la módulo ROI (región de interés) y el módulo *Model Maker*. El resultado debe ser algo como lo observado en la figura 3.29



Figura 3.29: Representación 3D de la pelvis y estructura ósea

Bibliografía para ampliar

Looking Within Anthoy Brinton Wolbarst. University of California Press. Este libro de 1999 presenta una entretenida lectura de la historia de la imagen médica, con numerosos ejemplos e ilustraciones sobre los principios físicos del CT y MRI.

Medical Image processing, reconstruction and restoration Jiri Jan. Taylor & Francis. Excelente libro para la comprensión de la imagen médica. Se divide en tres partes, una primera con los conceptos básicos del tratamiento de las imágenes como señales multidimensionales, una magnífica segunda parte donde se explican todos los principios físicos de cada tecnología (X-ray, CT, MRI, Nuclear, US, etc.) y una tercera parte de procesamiento y análisis de estas imágenes. Muy completo.

Slicer Wiki [Wiki del programa Slicer](#). En el apartado tutorials se pueden encontrar gran cantidad de tutoriales de calidad en los que se explican con ayuda del programa los procesamientos básicos de las imágenes médicas.

2-D and 3-D image registration A. Ardeschir Goshtasby. Wiley. Este libro se centra en el registro de imágenes, analizando las etapas de preprocesamiento, extracción de características, transformación, re-muestreo y fusión de imágenes. Con varios ejemplos de aplicaciones a imagen médica.

Capítulo 4

Tratamiento de señales biomédicas

“Haga los planes que haga, encontrará dificultades ocultas en alguna parte”
Ley de Van Gogh :-)

Índice

4.1. Introducción	110
4.2. Clasificación	111
4.2.1. Electromiograma. (EMG)	111
4.2.2. Electroencefalograma. (EEG)	112
4.2.3. Electrocardiograma (ECG)	115
4.2.4. Electrooculograma (EOG)	118
4.2.5. Electrorretinograma (ERG)	119
4.3. Análisis de señales fisiológicas	121
4.3.1. Conceptos generales de filtrado de señales	121
4.4. Hardware y Software para la adquisición y procesado de señales fisiológicas	122
4.5. ROBOTICA MEDICA APLICADA	125
4.5.1. Procesamiento de una señal EOG usando Matlab	125
4.5.2. Filtrado de señal EMG usando Matlab	128
4.6. Resumen del capítulo	130
Bibliografía para ampliar	130

4.1. Introducción

En este capítulo se revisará el origen, principio y la caracterización de los Biopotenciales y sus medidas, en particular las señales de Electrocardiografía (ECG), electroencefalografía (EEG), electromiografía (EMG) y electrooculograma (EOG). Estos biopotenciales representan la actividad eléctrica de los respectivos órganos: corazón, cerebro, músculos, y ojos. Los biopotenciales son adquiridos con la ayuda de electrodos que se convierten en interface entre el órgano o el cuerpo humano y el transductor, que posee características especiales para disminuir el ruido y eliminar los artefactos eléctricos y mecánicos de las señales originales.

El origen de los biopotenciales puede ser interpretado como la actividad eléctrica que se genera a nivel celular por las funciones de cada célula [37]. El potencial eléctrico a través de la membrana celular es el resultado de la diferencia de una concentración iónica que existe fuera y dentro de la célula. El gradiente de concentración electroquímica atraviesa una membrana semipermeable y como resultado se obtiene el potencial de Nernst. En el proceso la membrana celular separa altas concentraciones de iones de potasio y bajas concentraciones de iones de sodio (también en este proceso existen otros materiales como el calcio en menos proporciones). La diferencia de concentración iónica y el transporte de la misma concentración producen un potencial en reposo [28]. Algunas de estas células en nuestro cuerpo son excitables y producen el denominado potencial de acción, como resultado de un movimiento veloz a través de la membrana en respuesta a un estímulo eléctrico o un cambio en el gradiente eléctrico de la célula [45]. La excitación eléctrica de la célula genera corrientes alrededor del volumen conductor manifestándose como un potencial sobre el cuerpo específicamente en la piel.

Un elemento importante en la adquisición de los biopotenciales es el amplificador biomédico que consiste de un amplificador de instrumentación con características especiales, que incluyen una elevada amplificación, una impedancia de entrada, y la habilidad para rechazar las interferencias eléctricas, todas estas variables son ajustadas en un diseño electrónico para que la adquisición y medida sean fiables y verdaderas. Las etapas posteriores a la adquisición son el filtrado y la extracción de características de la señal, así como el aislamiento para su registro y transferencia a sistemas computarizados. La medida de las señales y el estudio de las características de los biopotenciales de nuestro cuerpo pueden proveer información vital para un proceso de diagnóstico y su resultado puede ser normal sin importar factores como amplitud o variaciones en la frecuencia o patológico y ser detectado por la forma y comportamiento de la señal. Como por ejemplo los latidos del corazón anormales o arritmias cardiacas pueden ser diagnosticados por la interpretación de un ECG.

Las señales biomédicas son registros espaciales, temporales o espacio-frecuencial de eventos tales como el latido del corazón o la contracción de un músculo. La actividad eléctrica, química o mecánica que ocurre durante estos eventos biológicos frecuentemente produce señales que pueden ser medidas y analizadas. En consecuencia las señales biomédicas o bioseñales contienen información que puede ser utilizada para explicar los mecanismos fisiológicos subyacentes en un evento o un sistema biológico específico.

A continuación nos introduciremos en la clasificación de los biopotenciales y sus características más relevantes.

4.2. Clasificación

4.2.1. Electromiograma. (EMG)

Las fibras musculares generan biopotenciales cuando son activadas. Esto quiere decir que en un proceso de contracción muscular, los potenciales de acción en las fibras musculares son transmitidos a la superficie [25]. Las señales EMG son registradas por medio de electrodos en la superficie de la piel que está por encima del músculo. Por ejemplo, un par de electrodos localizados sobre el bíceps y otros localizados sobre el tríceps podrían capturar la señal EMG generada cuando estos músculos están en contracción. La principal característica de la señal mioeléctrica hace referencia a que la amplitud de la señal es estocástica y aleatoria por naturaleza y puede ser razonablemente como una distribución gaussiana. El rango de la señal está en el rango entre 0 a 10 milivoltios (pico a pico). La mayor componente energética está limitada en el rango entre 0 y 500 Hertzios de frecuencia, y se puede considerar que la energía dominante está concentrada entre 50 y 150 Hertzios.

Por ser de origen estocástico y de amplitud baja, se considera que es una señal afectable a niveles de ruidos eléctricos, especialmente a sus bajas frecuencia. En la figura 4.1, se puede apreciar la forma de la señal mioeléctrica y su espectro de potencia donde se confirma la distribución frecuencial.

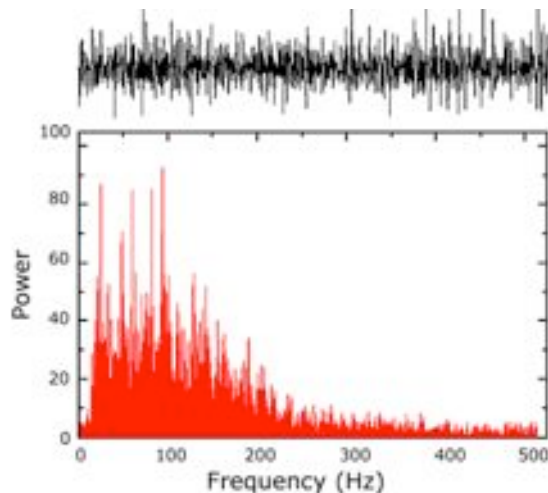


Figura 4.1: Señal EMG y su caracterización energética www.delsys.com

La manera de eliminar estos ruidos sobre la señal es por medio de filtros analógicos o digitales. Por ejemplo en aplicaciones de control y diagnóstico, la componente de ruido electromagnética en frecuencias como 60 o 50 Hertzios se eliminan por medio de filtros rechaza-bandas que eliminará este ruido pero también eliminará componentes frecuenciales importantes en el análisis. Recomendaciones para la eliminación de estas componentes

sin la utilización de filtros analógicos es utilizar cables apantallados que protegen y aíslan la penetración de estos ruidos en la etapa de amplificación diferencial. En la figura 4.2, se muestra la etapa de entrada de un amplificador biomédico y su conexión al cuerpo humano.

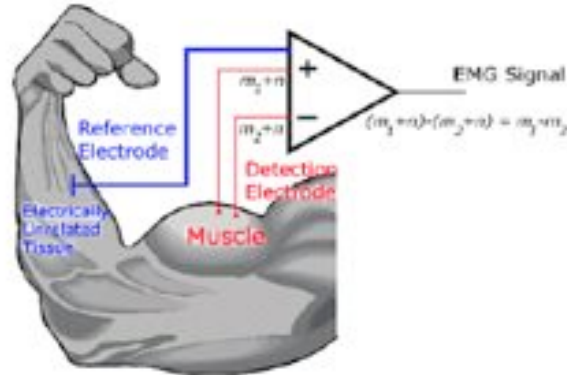


Figura 4.2: Adquisición de señal EMG, músculo bíceps www.delsys.com

El registro de la actividad eléctrica directamente sobre las fibras musculares puede ser clínicamente valorada y permite identificar desórdenes neuromusculares [3].

Otra posibilidad es el uso de electrodos invasivos, que pueden acceder a fibras y nervios internos que pueden proporcionar señales menos contaminadas de ruidos y más selectivas. En la actualidad finas agujas de acero inoxidable han sido insertadas o implantadas para obtener de forma local registros de fibras o uniones del sistema nervioso y neuro-muscular. La importancia de adquirir y registrar señales mioeléctricas es especialmente importante en el ambiente clínico, por ejemplo en el diagnóstico, análisis de la actividad muscular, obtención del tiempo de respuesta en la contracción, y finalmente en el ambiente de rehabilitación, donde se trabaja en el uso de estas señales para generar comandos y órdenes para el control de prótesis, órtesis y sillas de ruedas robotizadas.

4.2.2. Electroencefalograma. (EEG)

La Electroencefalograma es el registro y evaluación de los potenciales eléctricos generados por el cerebro y obtenidos por medio de electrodos situados sobre la superficie del cuero cabelludo. Dicho registro posee formas muy complejas que varían mucho con la localización de los electrodos y entre individuos. Esto es debido al gran número de interconexiones que presentan las neuronas y por la estructura no uniforme del encéfalo [25].

Los estudios sobre las señales producidas en la EEG las correlaciona con el resultado de la superposición de potenciales eléctricos producidos por la actividad sináptica sincronizada de grupos de neuronas.

En su formación, no todas las actividades cerebrales tienen el mismo impacto. Factores que influyen en la forma de una onda particular son la profundidad, la orientación y la



Figura 4.3: Colocación de electrodos para investigación, mapeo EEG. Gentileza de Sabbatini, R (2006). Universidad de Sao Paolo-Brasil. Tomado de la URL: <http://www.cerebromente.org.br>

simetría de las conexiones de la zona cerebral activa. Para el análisis clínico de señales EEG, se utilizan las propiedades, tales como: forma de la onda, frecuencia, amplitud, distribución espacial, relación de fase, sincronización, persistencia y reactividad. Para la interacción cerebro computadora o máquina (BCI, BMI) se pueden usar mediciones con estas propiedades para su operación [85]. En las señales de la EEG pueden apreciarse la forma de onda regulares, como sinusoides, pero también otras aperiódicas y con formas irregulares. Las frecuencias de interés se hallan en la banda entre 0.1 Hz y 40 Hz.

Existen determinadas ondas que presentan características de alguna región y aparentemente están asociadas a distintas funciones cerebrales, por lo que su frecuencia corresponde a una de las correlaciones más importantes en la investigación de señales de la EEG y estructuras subyacentes. Teniendo en cuenta otras características de la señal, es necesario tener presente que, como generalmente hay varios electrodos que cubren al mismo tiempo diferentes áreas de la cabeza, entonces la señal puede manifestarse como una distribución espacial en diferentes canales o en caso contrario solo será una señal focalizada.

Existen patrones básicos de bandas de frecuencia de ondas, las cuales han sido clasificadas. Estas ondas pueden estar en forma aislada o bien pueden estar mezcladas, pero las distintas actividades cerebrales y estados mentales pueden producir algunas de estas ondas. En particular, existen seis grupos de ondas importantes, que corresponden a las ondas **beta**, **alfa**, **mu**, **teta**, **delta** y **gama**.

Ondas Beta

Las ondas Beta se encuentran en el rango de frecuencias entre 12 Hz y 30 Hz. Su voltaje es usualmente bajo, cercano a los 5-30 microVoltios. Las ondas Beta generalmente se asocia a pensamiento activo, estado de alerta y resolución de problemas. Durante actividades mentales intensas puede alcanzar hasta los 50 Hz. Esta onda ocurre principalmente en el lóbulo frontal y parietal (ver Figura 4.4).

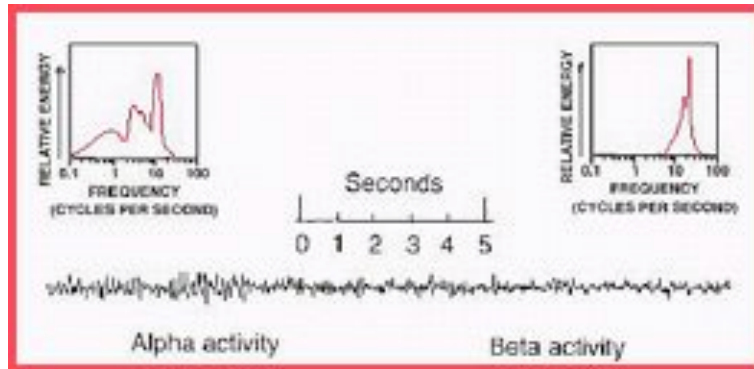


Figura 4.4: Ondas ritmo beta. Fuente Baztarrica (2002)

Ondas Alfa

Las ondas Alfa se encuentran en el rango de frecuencias entre los 8 Hz y 12 Hz, con amplitudes generalmente superiores a los 20 microVoltios. Las ondas Alfa se asocian a relajación y actividad mental leve. Ondas Alfa se manifiestan principalmente en el lóbulo Occipital (posiciones O) y frontal (posiciones F) (la nomenclatura de las posiciones está tomada del sistema internacional de colocación de electrodos), aumentando su manifestación con los ojos cerrados y pueden reducirse abriendo los ojos, presentando estímulos no familiares al sujeto como flashes de luz, también la ansiedad y la actividad mental intensa disminuye su manifestación en el EEG, donde se observa las relaciones anatómicas de los ojos y los lóbulos occipitales, en los cuales se produce las ondas alfa con el cierre de los ojos.

Las ondas tienen un origen dado por un fenómeno denominado de reclutamiento. En la corteza cerebral no se producen ondas alfa si no hay conexiones con el sistema reticular activador, de este modo se ve que las ondas alfa son el producto de la sincronización de muchísimas neuronas.

Ondas Mu

Las ondas Mu son ondas espontáneas que se encuentran en la banda de frecuencia entre 8 Hz y 13 Hz, igual que las ondas Alfa. Ellas se registran principalmente en la zona

Sensorio-motora, área 3,2,1 de Brodmann y se registran en la posición C del sistema de colocación de electrodos, manifestando una atenuación durante el movimiento o intento de movimiento de extremidades.

Ondas Teta

Las ondas Teta se encuentran en el rango de frecuencias entre 4 Hz y 8 Hz, y poseen amplitudes que generalmente superan los 20 microVoltios. Teta se asocia a estados emocionales de estrés, como decepción o frustración, y sueño MOR (Movimiento Rápido de Ojos o en siglas del inglés REM). También se ha mostrado que está relacionada a funciones mentales que involucran el acceso a material subconsciente, meditación profunda e inspiración creativa.

Ondas Delta

Las ondas Delta se ubican en el rango de frecuencias entre 0.5 Hz y 4 Hz con amplitud variable. Éstas se manifiestan principalmente durante sueño profundo, o en ciertos estados de meditación. Su presencia en estado de vigilia indican defectos o daños cerebrales. Es fácil confundir actividad Delta con artefactos musculares, como por ejemplo aquellos producidos por los músculos de la nuca. Sin embargo, un sistema BCI en línea puede discriminar fácilmente entre una onda Delta y una respuesta muscular, ya que la primera se origina profundamente en el cerebro y es severamente atenuada, mientras que la segunda presenta amplitudes mucho mayores.

Ondas Gama

Las ondas Gama se encuentran desde los 30 Hz hacia arriba. Se piensa que éstas reflejan el mecanismo subyacente de la conciencia. Se ha observado que las ondas Gama están asociadas a procesos de comunicación, feedback y sincronización entre las distintas zonas neuronales - módulos cerebrales - durante la realización de actividades mentales complejas.

4.2.3. Electrocardiograma (ECG)

El electrocardiograma es el registro directo de los biopotenciales del músculo cardíaco (miocardio) durante su excitación y que conlleva a las funciones de contracción y relajación para el bombeo de sangre de órganos y venas hacia el corazón y posteriormente el traslado a los pulmones para el intercambio de gaseoso (liberación de dióxido de carbono y captura de oxígeno). El sistema circulatorio sanguíneo está compuesto por el corazón como bomba impulsora, las arterias que llevan la sangre a los pulmones y al resto del cuerpo, la red de capilares, y las venas, que regresan la sangre al corazón. El paso de la sangre por el corazón se puede resumir como llenado y contracción de las aurículas seguido por el llenado y contracción de los ventrículos. El corazón está dotado de un sistema especial para generar rítmicamente impulsos que produzcan la contracción periódica del músculo cardíaco, y luego para conducir estos impulsos a todo el músculo cardíaco. La

mayor parte de las enfermedades del corazón dependen de anomalías de este sistema especial de excitación y conducción (Figura 4.5).

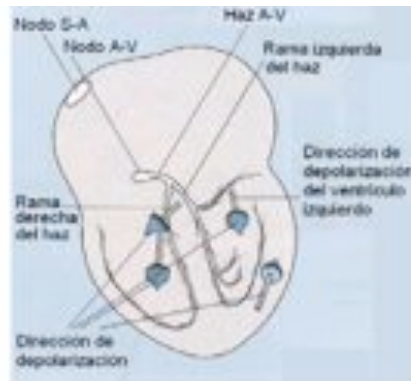


Figura 4.5: Sistema de estimulación eléctrica del corazón donde se muestran los nodos A-V. Extraído de web

En la figura 4.6 se ilustra la generación del potencial de acción en las fibras excitables del corazón y la superposición de estos potenciales para generar la señal Electrocardiograma.

Las células del miocardio, desde el punto de vista eléctrico, son neutras. Tienen en su superficie exterior iones positivos y en la interior iones negativos. La célula utiliza energía para mantener esta configuración, asociada al llamado potencial de membrana en reposo ($PMR=85mV$). El estímulo eléctrico se propaga mediante procesos de despolarización y re-polarización que modifican temporalmente el potencial en reposo.

La despolarización y re-polarización se pueden considerar ondas de transporte de energía que nacen en los puntos y zonas de estímulo eléctrico, (ver Fig. 4.6) atraviesan el corazón y se difunden a través de los tejidos que lo rodean. La despolarización avanza y utiliza las células que forman fibras musculares para su transporte, permutando los iones de la membrana. Esta permutación cambia los centros de cargas, alterando el potencial de membrana en reposo y haciendo necesaria la re-polarización (proceso de recuperación del estado inicial). En cada latido se verifica un par de estas ondas, las cuales provocan que los centros de las cargas (positiva y negativa) se separen, creando un polo positivo y un polo negativo (téngase en cuenta que la suma total de las cargas sigue siendo cero). Esta distribución de cargas cambia en el tiempo de latido, como cambia el volumen del corazón, por lo que los polos (centros de carga) se alejan, se acercan y rotan, cambiando en general su posición en el espacio.

La suma de estos biopotenciales genera una señales eléctricas que definen funciones y procesos propios en cada cámara y el control de las válvulas, ver figura 4.7. Las características de la señal ECG permiten el diagnostico de enfermedades. Por ejemplo, si analizamos la figura 4.7, podremos identificar que el latido se refleja en el electrocardiograma como una serie de “ondas principales” que se ilustran en la figura, denominadas P, Q, R, S y T. La onda P se verifica durante la despolarización de las aurículas, el complejo

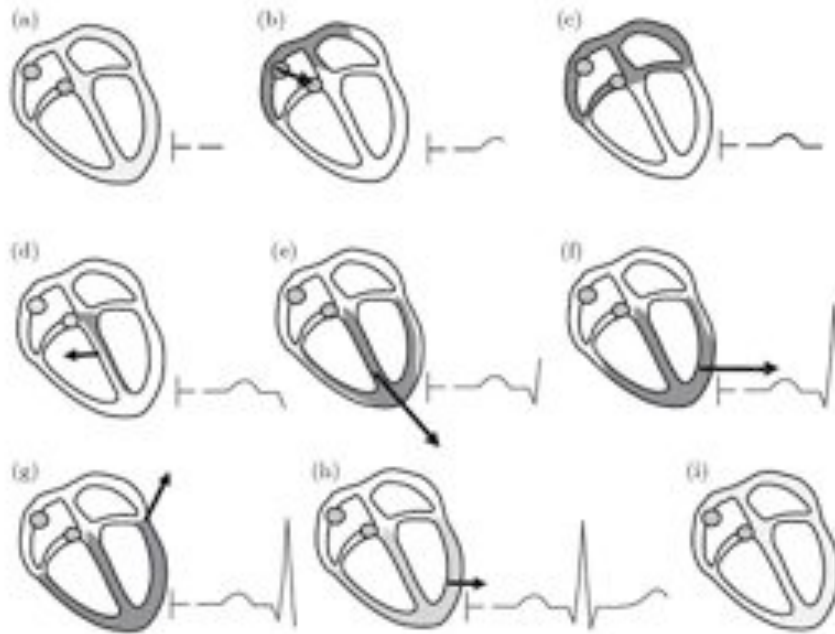


Figura 4.6: La generación de biopotenciales en los diferentes nodos del corazón

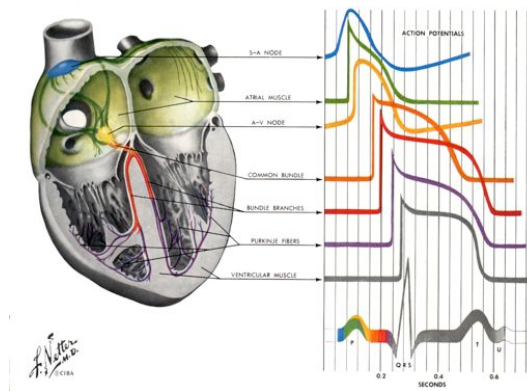


Figura 4.7: Bipotenciales y su relación con los biopotenciales en regiones de actividad eléctrica del corazón

QRS durante la despolarización de los ventrículos (mucho más intenso eléctricamente) y la onda T durante la repolarización de los ventrículos. El especialista examina la forma y duración de estas ondas para el diagnóstico, ver figura 4.8.

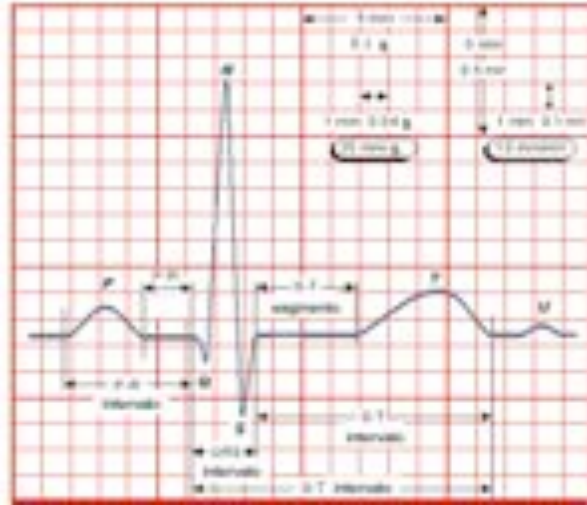


Figura 4.8: Electrocardiograma de un latido, derivación I. Eje vertical: biopotencial; eje horizontal: tiempo

4.2.4. Electrooculograma (EOG)

El electrooculograma (EOG) es un examen que consiste en colocar pequeños electrodos cerca de los músculos de los ojos para medir el movimiento de éstos. Este examen es utilizado en la polisomnografía. En condiciones habituales existe una diferencia de potencial de aproximadamente de 0,4 a 5 mV entre la córnea y la membrana de Bruch situada en la parte posterior del ojo. El origen de esta diferencia se encuentra en el epitelio pigmentario de la retina y permite considerar la presencia de un dipolo, el cual puede ser representado por un vector cuyo brazo coincide con el eje antero-posterior del globo ocular, donde la córnea corresponde al extremo positivo y la retina al extremo negativo de dicho dipolo. Ahora bien, el potencial producido por este dipolo es susceptible de ser registrado a través de sistemas de registro tanto unipolar como bipolar, mediante la colocación de electrodos en la piel cercana al ojo. Al medir el potencial producido por un dipolo, la magnitud (voltaje) y polaridad del potencial registrado dependerán, en gran medida, de la angulación del dipolo con respecto a los electrodos pertenecientes a dichos sistemas de registro, en la figura 4.9 se muestra un registro de un examen clínico.

Existen cuatro tipos de movimientos oculares, cada uno controlado por un sistema neural distinto pero que comparten la misma vía final común, las neuronas motoras que llegan a los músculos extraoculares.

1. Los movimientos sacádicos: movimientos súbitos y enérgicos de tipo espasmódico, ocurren cuando la mirada cambia de un objeto a otro. Colocan nuevos objetos de interés en la fovea y disminuyen la adaptación en la vía visual, que podría ocurrir si la mirada se fijara en un solo objeto por períodos prolongados.
2. Los movimientos suaves de persecución (de búsqueda): movimientos oculares de

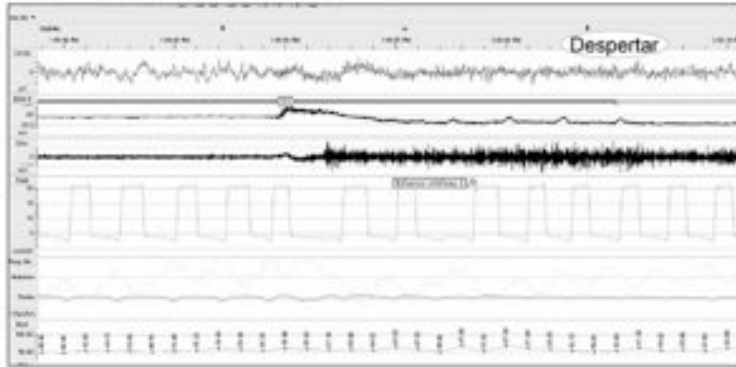


Figura 4.9: Registro de biopotenciales EOG en un estudio del sueño

seguimiento que se producen cuando se observa un objeto en movimiento.

3. Los movimientos vestibulares (movimientos de ajuste): ocurren como respuesta a estímulos iniciados en los conductos semicirculares, para mantener la fijación visual mientras se mueve la cabeza.
4. Los movimientos de convergencia: aproximan los ejes visuales entre sí cuando se enfoca la atención en objetos cercanos al observador.

Aún cuando una persona se fije en un objeto estacionario, sus ojos no están inmóviles, sino que exhiben muy pequeños movimientos involuntarios. Hay tres tipos de movimientos involuntarios: vibración, saltos lentos y golpeteos.

1. Vibración: una serie de pequeñas vibraciones de los ojos entre 30-80 Hz.
2. Saltos lentos: movimientos involuntarios que resultan en movimiento de saltos de los ojos; estos saltos significan que aunque los objetos estén estacionarios, la imagen salta a través de la fóvea.
3. Movimientos de golpeteo (microsacádicos): como la imagen salta en el extremo de la fóvea, el tercer mecanismo involuntario causa un reflejo de salto del globo ocular de tal manera que la imagen es proyectada nuevamente hacia la fóvea.

4.2.5. Electrorretinograma (ERG)

La electrorretinografía, se utiliza para medir la respuesta eléctrica de células en la retina, incluyendo los fotorreceptores (cónos y bastones). Los electrodos se colocan sobre la córnea y la piel cercana al ojo. Durante una grabación, el paciente mira un estímulo estándar y la señal resultante se interpreta en términos de su amplitud de voltaje y su duración. Los estímulos incluyen destellos (flash ERG) y modelos de rejillas (pattern ERG). Se aplica principalmente en oftalmología, donde el electrorretinograma (ERG) se

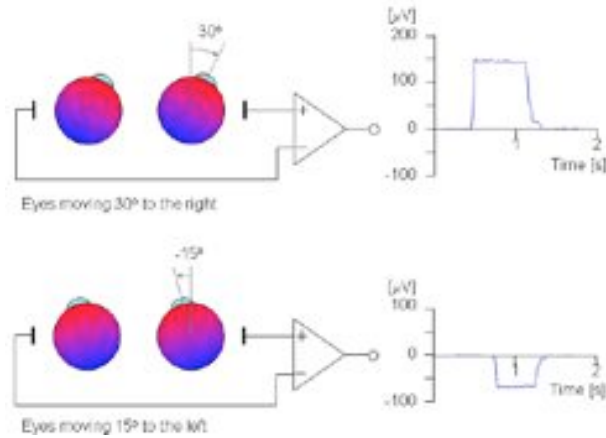


Figura 4.10: Registro de biopotenciales EOG en un estudio del movimiento ocular sacádico

utiliza para el diagnóstico de varias enfermedades de la retina [60]. La técnica de EGR multifocal se utiliza para registrar respuestas separadas a diferentes posiciones de la retina. Los electroretinogramas puede dividirse en tres componentes: una onda a inicial, causada por corrientes iónicas extracelulares generados por fotorreceptores durante la fototransducción; una onda “b” que corresponde a la actividad de una célula bipolar, una última onda “c”, que se genera por el epitelio pigmentado de la retina (retinal pigment epithelium en inglés) y células Müller. Según del tipo de ERG que se utilice, la onda “c” puede ser positiva, negativa o nula [56]. En la figura 4.11, se presenta un estudio clínico de la respuesta eléctrica de las células de la retina.

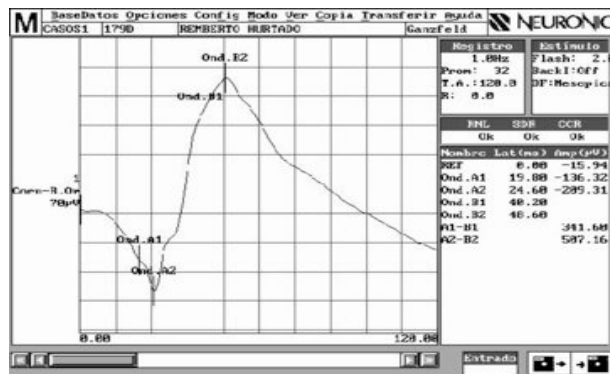


Figura 4.11: Registro de biopotenciales Electroretinograma escotópico

4.3. Análisis de señales fisiológicas

Las señales fisiológicas pueden provenir de fenómenos mecánicos o eléctricos dentro de nuestro cuerpo y son transmitidas a la superficie o piel como biopotenciales. En ambos casos se necesita un traductor para poder adquirirlas, por ejemplos electrodos, micrófonos piezoeléctricos. En la mayoría de los casos las señales adquiridas son analógicas y lo interesante es extraer de ellas las características o parámetros más relevantes de la señal. En general una señal analógica es $F(t)$, una función matemática continua en la que es variable su amplitud (representando una cantidad de señal) en función del tiempo; mientras que una señal digital $F'(t)$ se caracteriza porque las magnitudes de la misma se representan mediante valores discretos en lugar de variables continuas, ver figura 4.12.

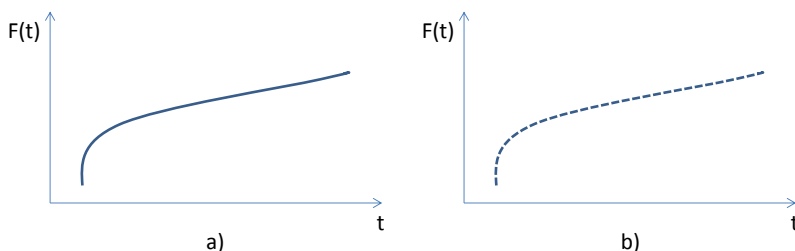


Figura 4.12: Representación gráfica de señales continua a), y discreta en el caso b)

Ahora bien, las señales de cualquier circuito o comunicación electrónica pueden ser afectadas de forma no deseada mediante el ruido electrónico, lo que ocurre siempre en mayor o menor medida. La gran desventaja respecto a otros tipos de señales es que en las señales analógicas cualquier variación o pérdida de información provocada por el ruido es de difícil recuperación. Este inconveniente afecta en gran medida al correcto funcionamiento y rendimiento de los dispositivos o sistemas analógicos. Las técnicas para la extracción de características son bien conocidas en el caso análogo. Lo habitual es usar filtros y rectificadores, mientras que en el caso digital se usan filtros digitales y funciones de promediados, como por ejemplo mediana, valor medio y valor media cuadrática o RMS. A continuación introduciremos factores y generalidades de las técnicas clásicas del filtrado de señales eléctricas.

4.3.1. Conceptos generales de filtrado de señales

Como se comentó en la introducción, las señales eléctricas analógicas son propensas al ruido, especialmente si estas señales son biopotenciales y adquiridas en la superficie de la piel. En donde el valor de la señal puede ser mayor que el ruido electromagnético, las técnicas clásicas para eliminar este ruido es por medio de filtros; en muchos casos estos ruidos están o en las bajas frecuencias entre 10 y 60 Hertzios, o ruido de alta frecuencia por encima de los 1000 Hertzios.

Para extraer la máxima información o energía de la señal se utilizan diferentes tipos de filtros analógicos, pasa-bajos, pasa-altos, pasa-banda o rechaza-banda, filtros que se

construyen con amplificadores operacionales y existen diferentes tipos tales como Butterworth, Chebyshev, y su orden proporciona la selectividad y la década por decibeles proporciona la pendiente con que el filtro llega a su frecuencia de corte.

En el caso de extraer características frecuenciales y temporales de la señal, otra técnica importante es digitalizar la señal con un conversor analógico a digital y adquirirlas en un computador para utilizar algoritmos y filtros digitales. Ejemplos de procesamiento de señales biológicas extraídas en personas amputadas y utilizadas en el control de prótesis se muestran a continuación.

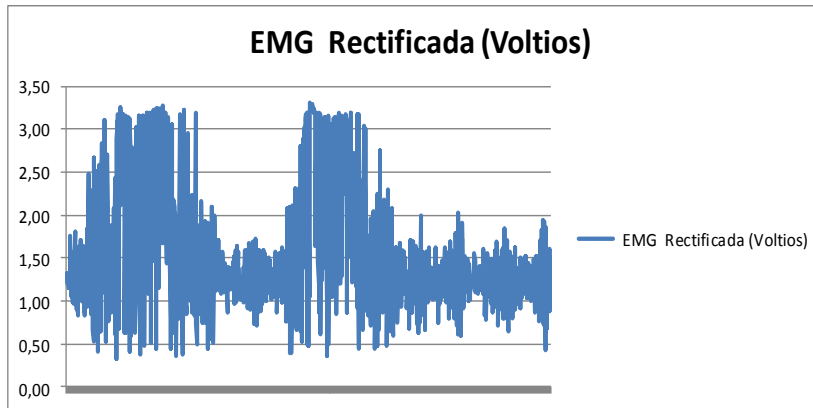


Figura 4.13: Señal EMG rectificada y extraída del musculo bíceps de un amputado

4.4. Hardware y Software para la adquisición y procesamiento de señales fisiológicas

Las bioseñales suelen ser muy pequeñas, contienen ruido innecesario y pueden estar enmascaradas por otras bioseñales de diferentes fenómenos biológicos. Con el objetivo de extraer la información de una bioseñal, lo que puede ser crucial para entender un sistema o evento biológico particular, comúnmente son utilizados sofisticados equipos y técnicas de adquisición de datos.

Es importante preservar la información de la bioseñal original a lo largo de todo el procedimiento de adquisición de los datos. Las señales, primeramente son detectadas en un medio biológico, como puede ser una célula o la superficie de la piel, utilizando un sensor. El sensor convierte la medición física en una salida eléctrica y ofrece una interfaz entre el sistema biológico y un instrumento de registro eléctrico. Es muy importante que el sensor utilizado para detectar las señales biológicas no afecte las propiedades y características de la señal que está midiendo.

Después que la bioseñal ha sido detectada utilizando un sensor, usualmente es amplificada y filtrada. Los amplificadores operacionales son circuitos electrónicos que usualmente son utilizados para incrementar la amplitud de las bioseñales. Un filtro analógico puede

ser, entonces, utilizado para eliminar el ruido o para compensar las distorsiones causadas por el sensor. La amplificación y el filtrado de la bioseñal pueden ser también necesarios para ajustarla a las especificaciones del hardware del sistema de adquisición de los datos. Las señales continuas deben ser limitadas a cierta banda de frecuencias antes de que la señal pueda ser digitalizada utilizando la conversión A/D, ver figura 4.14

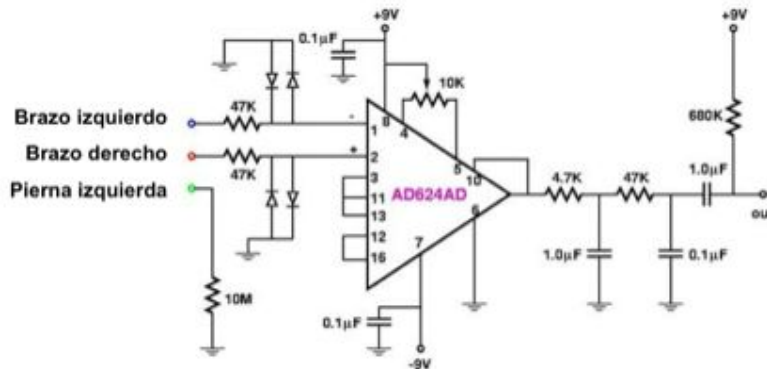


Figura 4.14: Circuito electrónico para la adquisición de biopotenciales y filtros analógicos

Para cambiar la forma de la bioseñal de onda analógica continua a señal digital se utiliza un convertidor A/D. Un convertidor A/D es un voltímetro controlado por un ordenador, que mide una señal de entrada analógica y brinda una representación numérica de la señal en su salida. La onda analógica originalmente detectada por el sensor y posteriormente amplificada y filtrada es una señal continua. El convertidor A/D transforma la señal analógica continua en una señal digital discreta. La señal discreta consiste de una secuencia de números que puede ser fácilmente almacenada y procesada en un ordenador. La conversión A/D es particularmente importante porque, debido a los avances de la tecnología de los ordenadores, se incrementa el almacenamiento y análisis de las bioseñales basados en los ordenadores. En esta definición están patentes los cuatro procesos que intervienen en la conversión analógico-digital:

1. Muestreo y Retención: El Muestreo (en inglés, Sampling) consiste en tomar muestras periódicas de la amplitud de onda. La velocidad con que se toman estas muestras, es decir, el número de muestras por segundo, es lo que se conoce como frecuencia de muestreo. La retención (En inglés, Hold) se refiere a que las muestras tomadas han de ser retenidas por un circuito de retención, el tiempo suficiente para permitir evaluar su nivel en el proceso de cuantificación.
2. Cuantificación: Se mide el nivel de voltaje de cada una de las muestras. Consiste en asignar un margen de valor de una señal analizada a un único nivel de salida.
3. Codificación: La codificación consiste en traducir los valores obtenidos durante la cuantificación al código binario. Hay que tener presente que el código binario es

el más utilizado, pero también existen otros tipos de códigos que también son utilizados.

4. Compresión: Si bien no es un proceso que se pueda incluir en la conversión analógico-digital, sí suele ser el destino de la señal digital que se obtiene de ella. Para reducir el tamaño de la cadena de valores binarios se comprime la señal según un determinado algoritmo matemático.

Durante el muestreo y la retención, podemos decir que la señal es analógica puesto que aún puede tomar cualquier valor en cualquier instante. A partir de la cuantificación es cuando la señal ya toma valores determinados según una escala y en momentos determinados también del tiempo, la señal ya es digital. Los cuatro procesos tienen lugar en el convertidor analógico-digital. Existen ya software en el mercado que permiten procesar y extraer características de señal, entre ellos se pueden mencionar el Matlab, Labview y en código abierto el Scilab. En la mayoría de software las funciones ya existen y sus parámetros pueden ser ajustados hasta obtener gráficamente la eliminación del ruido que afecta a la señal, un ejemplo de estos procesos se muestra en la siguiente figura 4.15.

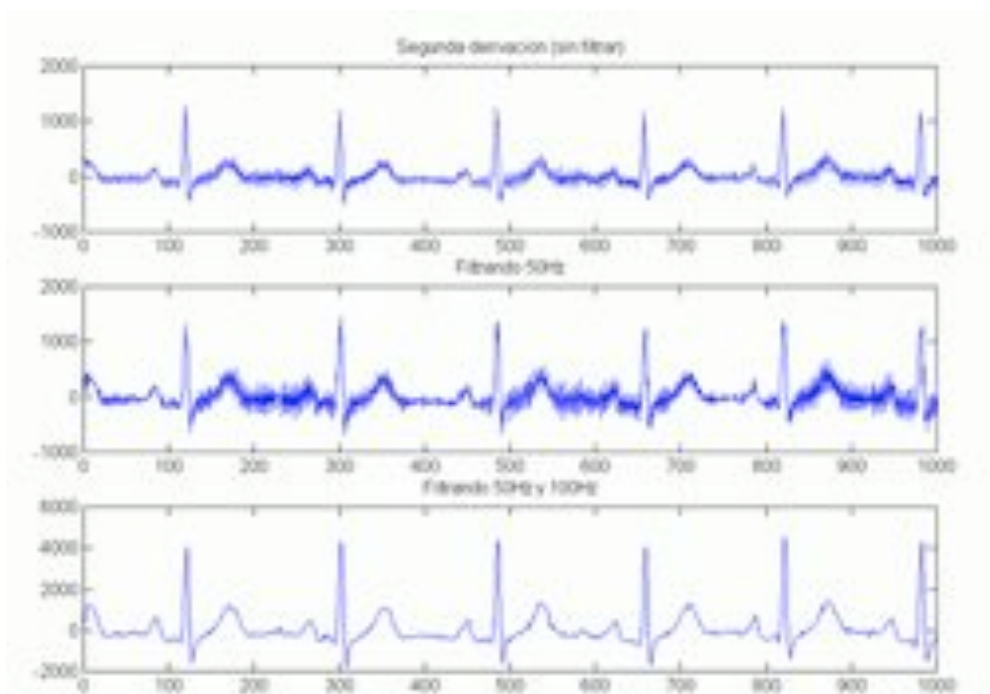


Figura 4.15: Ejemplos de filtrado digital de una señal ECG y eliminación de ruido entre 50 y 100 Hz.

4.5. ROBOTICA MEDICA APLICADA

Las aplicaciones en Robótica Médica de la adquisición y procesado de características de biopotenciales son múltiples y crecen cada día. Ejemplos que podemos citar son: intención de apertura, agarre en prótesis de manos, intención de ejecutar un paso, y con ello activar rodillas, cadera y tobillos robóticos. Otros ejemplos se dirigen hacia la actividad neuronal para la generación de órdenes de navegación en silla de ruedas y ambuladores robóticos. Pero no sólo en esta área se ve la importancia de los biopotenciales, para el diagnóstico en el caso de enfermedades cardiacas, neuronales y cerebrales los estudios de ECG, EMG, y EEG facilitan el análisis y la detección de enfermedades.

En la última sección se presentan dos ejemplos. En el primero de ellos se realiza el procesamiento de una señal EOG, mientras que en el segundo se presentan unos ejemplos de filtrado de una señal EMG en Matlab y se proponen ejercicios adicionales para los lectores.

4.5.1. Procesamiento de una señal EOG usando Matlab

Las interfaces basadas en electrooculografía (EOG) permiten la comunicación hombre-máquina a través del movimiento ocular. Para obtener los comandos de control, las señales electrooculográficas son registradas y suelen ser procesadas mediante algoritmos basados en la derivada. Este tipo de algoritmos son capaces de detectar los cambios bruscos en la señal producidos por los movimientos oculares. El tipo de movimiento que debe efectuar el usuario consiste en mirar rápidamente en la dirección deseada volviendo inmediatamente a la posición central (de reposo).

Un ejemplo sencillo de algoritmo de procesamiento permite obtener el movimiento ocular efectuado en cuatro direcciones (Figura 4.16). Para ello, las señales se procesan en ventanas de 1 segundo, es decir, se obtiene un comando ocular cada segundo. Se procesan las señales correspondientes al canal horizontal (para detectar movimientos horizontales) y el canal vertical (para detectar movimientos verticales).

El algoritmo sigue los siguientes pasos:

1. Se adquieren las señales EOG.
2. Se calcula la media móvil para eliminar el ruido y obtener una señal clara.
3. Se calcula la derivada para detectar el cambio en la dirección de los ojos. Si la persona mira en una dirección determinada, la señal cambia abruptamente. A este cambio rápido le sigue una caída lenta que es detectable por la derivada, generando un valor alto (positivo o negativo) en el momento en el que ocurre.
4. Se usa un umbral para distinguir la detección del movimiento frente al ruido o la señal obtenida cuando no hay movimiento de los ojos. Este umbral puede ser diferente para los canales horizontal y vertical y depende del usuario.
5. Se buscan los máximos y los mínimos con el fin de saber la dirección del movimiento ocular. En este caso se buscan secuencias max/min y min/max.

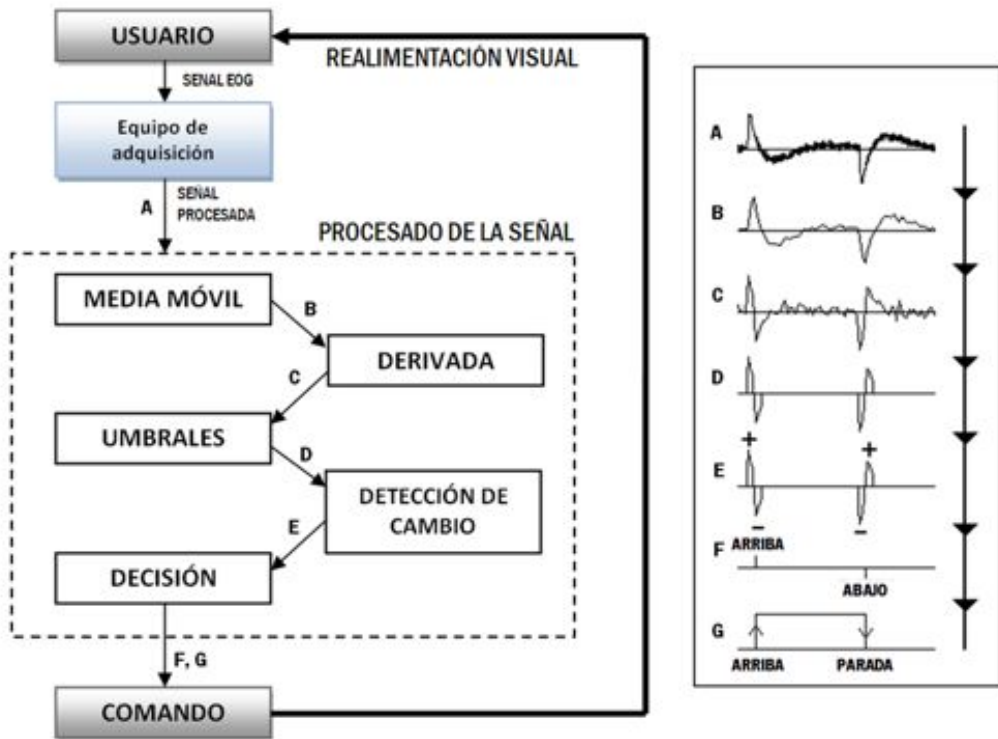


Figura 4.16: Algoritmo de procesamiento electrooculográfico

- Finalmente, se calcula la dirección en la que se ha mirado. El comando proporcionado por el algoritmo corresponderá a uno de los 4 posibles movimientos de los ojos: arriba, abajo, izquierda y derecha, o bien, a reposo.

En este ejemplo, se muestra la ejecución de este algoritmo en un conjunto de señales donde se han efectuado una serie de movimientos predefinidos almacenados en `eog_signal.mat`. Para ello, se ejecutará la función `eog_main.m` que incluye la función `eog_processing.m`, que ejecutará el algoritmo sobre la ventana de muestras seleccionada. El archivo `eog_signal.mat` contiene dos vectores, uno para el canal horizontal y otro para el vertical, con las señales registradas a una frecuencia de muestreo de 120 Hz.

En la Figura 4.17 se muestra paso a paso el procesamiento de la señal al ejecutar la función `eog_processing.m`. Esta función se ejecutará para cada ventana de procesamiento de un segundo. La función `eog_main.m` se encarga de gestionar las iteraciones de procesamiento en función de la frecuencia de muestreo y define los umbrales para cada uno de los canales.

Los umbrales seleccionados se basan en el estudio cualitativo de las señales y varían en función del usuario. En este caso se han seleccionado los siguientes umbrales:

```

% Definición de umbrales para cada canal:
% [umbral_superior umbral_inferior]
umbrales_horizontal=[30 -30];
umbrales_vertical=[18 -18];

```

Una vez seleccionados los umbrales se ejecuta el algoritmo `eog_processing.m` en ventanas de 1 segundo a lo largo de toda la señal registrada:

```

% Se calcula el número de ventanas de 1 segundo
iteraciones=floor(length(horizontal)/fm);

% Bucle de procesamiento (ventanas de 1 segundo)
5 for i=1:iteraciones
    % Calculo de los indices del vector para obtener cada ventana en
    % funcion de la frecuencia de muestreo
    indices=(1+(i-1)*fm):(i*fm);

10    % Obtención de la salida del canal horizontal a partir de la
    % ventana y los umbrales
    [salida_horizontal]=eog_processing(horizontal(indices), ...
        umbrales_horizontal);

15    % Obtención de la salida del canal vertical a partir de la
    % ventana y los umbrales
    [salida_vertical]=eog_processing(vertical(indices),...
        umbrales_vertical);

20    % Almacenamiento en un vector de cada salida
    direccion_horizontal=[direccion_horizontal salida_horizontal];
    direccion_vertical=[direccion_vertical salida_vertical];
end

```

Siguiendo la Figura 4.17, los pasos seguidos por la función `eog_processing.m` son los siguientes:

1. Suavizado inicial: se suaviza la señal de entrada registrada
2. Cálculo de la derivada de la señal: permite obtener los cambios de dirección ocular.
3. Umbralizado: se utilizan dos umbrales (mínimo y máximo) para eliminar el ruido y seleccionar los picos generados por el movimiento ocular.
4. Detección de la dirección ocular: se estudian los picos obtenidos y se seleccionan los valores máximos y mínimos. A continuación, se buscan las secuencias max/min o min/max que corresponden al movimiento en una dirección y la contraria, para cada canal.

La ejecución del algoritmo genera la siguiente salida:

```
>> [direccion_horizontal, direccion_vertical]=eog_main(hor,ver,120)
direccion_horizontal =
    1    0   -1    0    0    0    0    0    0    0    1
  0   -1    0
direccion_vertical =
    0    0    0    0    0    1    0   -1    0    1    0   -1
  0    0    0
```

Como se puede observar, el usuario de este ejemplo, ha mirado dos veces a la derecha y dos veces a la izquierda (canal horizontal) y dos veces arriba y dos veces abajo (canal vertical).

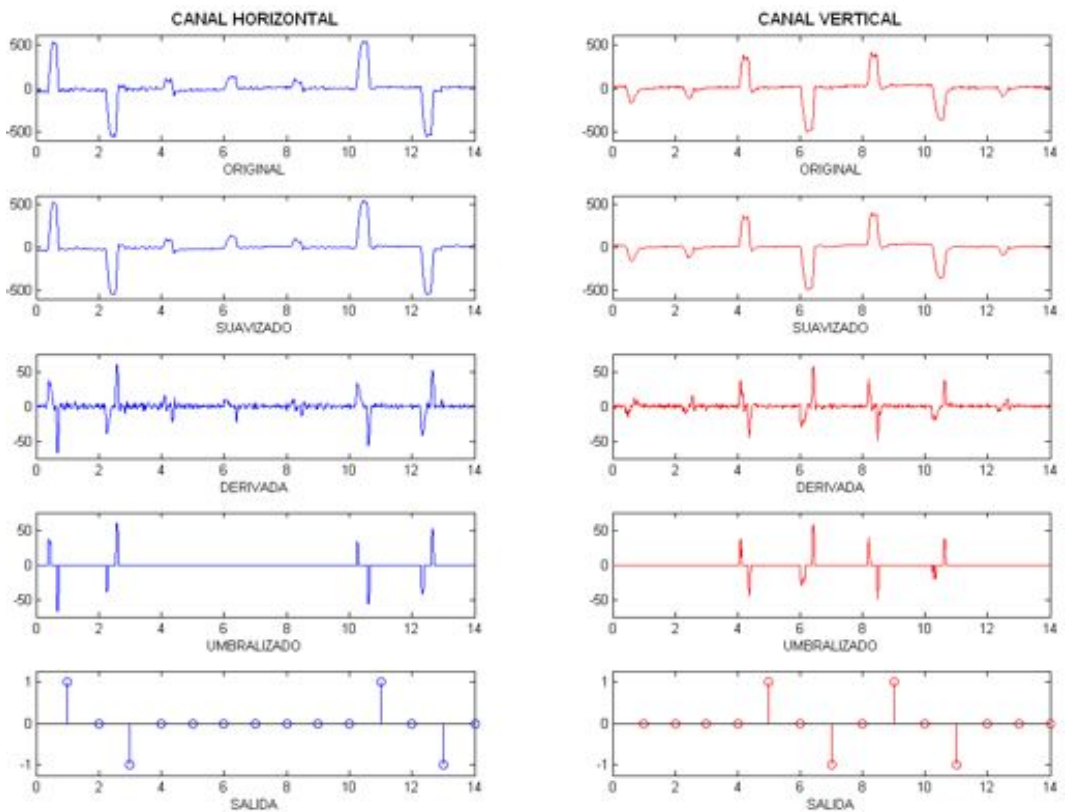


Figura 4.17: Procesamiento paso a paso de la señal para cada canal

4.5.2. Filtrado de señal EMG usando Matlab

Utilice la herramienta de Matlab para diseñar y aplicar un filtro digital pasa-bajo FIR, con frecuencia de corte de 200Hz y una frecuencia de muestreo de 1000 Hertzios. Apro-

veche el ejemplo que se anexa para cambiar el orden de filtro y compare sus resultados (8, 16 y 32). Pasos:

- Abra Matlab y cargue el archivo de EMG1.xls.
- Renombre la variable en matlab como data.m
- Abra el ejemplo filtro2.m

```

% Primero se lee la señal a partir de la ruta en que se encuentra el
% fichero o siga los pasos si la variable se carga en Workspace
% *****
%fid = fopen('c:\señales\señal01');
5 %signal = fread(fid);
%signal = signal.'; % Esta sentencia es para manejar los datos como
% un vector fila.
%status = fclose(fid);
% *****
10 % Ahora, se diseña el filtro pasabajas. En este ejemplo,
% utilizaremos un filtro FIR2 para crear
% un filtro digital de orden 8, 16 y 32, frecuencia de corte 200 Hz
% y suponiendo que la señal ha
% sido muestreada a 1000 Hz (Frecuencia de Nyquist = 1000
15 %Hz/2 = 500 Hz).
%
order = 8; % Orden del filtro FIR.
wn=[0:1/500:1]; % Vector de frecuencias normalizadas.
mag=zeros(1,501);
20 % Respuesta en magnitud al vector de frecuencias wn.
mag(1:2)=10;
coef = fir2(8, wn, mag, kaiser((order+1),4));
freqz(coef,1); % Esta sentencia sirve para ver gráficamente la
% respuesta en magnitud y fase del filtro diseñado.
25 %
% Se procede a filtrar la señal por el filtro diseñado. Se utiliza
% la función FILTER.
%
out = filter(coef, 200, data);
30 Hpsd=dspdata.psd(out)

% Ahora se ver·n las dos señales: original y filtrada.
%
35 %subplot(2,1,1), plot(data), title('Señal Original'), grid on
subplot(2,1,1), plot(Hpsd), title('Señal Original'), grid on
subplot(2,1,2), plot(out), title('Señal Filtrada'), grid on

```

- Cambie los valores del orden del filtro a 8, 16 y 32
- Compare la respuesta de los filtros.

- Utilice el ejercicio anterior para caracterizar la misma señal, con funciones como `rms1`, `means1` y `psd` (espectro de potencia), compare los resultados y explique cómo se puede utilizar estas caracterizaciones para utilizarla en controles de prótesis.
- Para el siguiente ejercicio se ha rectificado una señal EMG del bíceps y la señal de salida se ha pasado por un filtro pasa-bajo con frecuencia de corte de 5 Hz (`FILT_EMG.xls` y `REC_EMG.xls`), grafique la señales, utilice el filtro del primer ejercicio para mejorar la salida y cambie el orden del filtros a 8, 16 y 32, compare sus respuestas.

4.6. Resumen del capítulo

1. En este capítulo se ha presentado el origen de los biopotenciales, sus características, además de los pasos que hay que seguir para su adquisición en la piel y el acondicionamiento electrónico que debemos adicionar para su posterior procesado.
2. Se han aprendido los principios físicos de las siguientes señales:
 - a) EMG
 - b) EEG
 - c) ECG
 - d) EOG
 - e) ERG
3. Se ha aprendido y practicado con el uso de Matlab para el análisis y procesamiento de señales biológicas.

Bibliografía para ampliar

EEG Signal Processing Sanel, Saeld, Chambers, Jonathan. Chichester, England Hoboken, NJ, John Wiley & Sons, 2007. En el siguiente libro se centra en los fundamentos del procesamiento de señales EEG, incluyendo no sólo algoritmos de procesamiento de señales sino además algoritmos de clasificación y aplicaciones del procesamiento de señales EEG.

Toward brain-computer interfacing Dornhege, Guido; Millán, José del R.; Hinterberger Thilo; McFarland, Dennis J.; Klaus-Robert, Müller. Edit. Cambridge (Massachusetts) MIT Press 2007. Para profundizar en la aplicación de señales EEG con aplicación a las interfaces cerebro-computador, se puede consultar este siguiente libro.

Biosignal processing: principles and practices Hualou Liang, Joseph D. Bronzino, Donald R. Peterson, CRC Press, 2012. Libro completo que cubre el proceso de análisis de señales biomédicas, incluyendo imagen médica MRI y fMRI.

Capítulo 5

Robótica quirúrgica. Fundamentos y aplicaciones

“Con ayuda de la robótica y la navegación quirúrgica podemos ubicarnos con mucha precisión, pero en el lugar precisamente equivocado”
Leo Joskowicz

Indice

5.1. Introducción	132
5.2. El proceso global de cirugía asistida por computador (CAS)	133
5.2.1. El problema de la incertidumbre	135
5.2.2. Limitaciones de la robótica quirúrgica	135
5.3. Clasificación de dispositivos robóticos para cirugía	136
5.4. Ejemplos de dispositivos robóticos para cirugía	138
5.4.1. Sistema de radiocirugía con fuente externa. Cyberknife.	138
5.4.2. Sistema HIFU con fuente de energía interna	140
5.4.3. Sistema de telecirugía. DaVinci	141
5.4.4. Sistema de cirugía ortopédica. Robodoc	143
5.4.5. Sistema de robot montado en paciente. SpineAssist	145
5.4.6. Sistema de robots colaborativos	147
5.5. Simuladores quirúrgicos	148
5.5.1. Requerimientos de los simuladores quirúrgicos	148
5.5.2. Simuladores comerciales y de investigación	150
5.6. Navegación quirúrgica	151
5.6.1. Tecnologías de los sistemas de navegación	152
5.6.2. Tipos de sistemas de navegación	154

5.7. ROBOTICA MEDICA APLICADA	157
5.7.1. Análisis del espacio de trabajo de una muñeca paralela de uso quirúrgico.	157
5.8. Resumen del capítulo	164
5.9. Ejercicios	165
Bibliografía para ampliar	165

5.1. Introducción

“La robótica quirúrgica representa un cambio radical en el trabajo del quirófano, convirtiendo al robot en un integrante más del equipo quirúrgico.”

La frase anterior encierra una paradoja clásica en robótica, y que es recurrente cuando se empieza a hablar de robótica quirúrgica. Al considerar al robot como un *integrante más*, se tiene la percepción del robot como un **“cirujano robótico inteligente”**, con autonomía y capacidad de decisión. Sin embargo, es más acertado, y más cercano a la realidad, ver al robot quirúrgico como una **“herramienta quirúrgica avanzada”**, es decir, como un dispositivo puesto en manos del cirujano que le permite incrementar sus capacidades, hacerlo más preciso, mitigar los problemas de fatiga, mejorar la destreza, etc. De esta forma **la autonomía de los robots quirúrgicos está muy delimitada a determinadas tareas muy específicas**, mientras que la mayoría de ocasiones el robot es una herramienta más que funciona bajo el esquema de control teleoperado.

En palabras del doctor Julio Mayol, cirujano y coordinador de la Unidad de Cirugía Guiada por la Imagen del hospital Clínico San Carlos, *“este nuevo enfoque representa una visión de futuro y requiere un esfuerzo de cambio conceptual del abordaje quirúrgico. Todo ello la convierte en una cirugía íntegra, intuitiva e inteligente”*. Según Russell Taylor, uno de los pioneros en el campo de la robótica quirúrgica, ésta pretende crear una sociedad entre el cirujano y el robot de forma que se aumenten las capacidades de éste y se puedan realizar los procedimientos de forma más eficiente que podrían hacerlos por separado cirujano o robot. Robots como el **DaVinci** o **Robodoc** son pioneros en la integración de robots en un quirófano, y están convirtiéndose en compañeros del cirujano. **Sin embargo, y como pioneros que son, están en continua evolución** para llegar a disponer de robots más eficientes, más ergonómicos y sobre todo que permitan realizar nuevos procedimientos quirúrgicos que aumenten las posibilidades y destrezas que actualmente dispone el cirujano.

En la actualidad existen muchos debates abiertos alrededor de la aplicación de robots en el quirófano. Como hemos comentado, algunos de estos debates hacen referencia al uso de esquemas de control de robots autónomos o de robots teleoperados. Otros discuten acerca de la morfología de los robots, que pueden ser de grandes dimensiones y estar fijos a la propia estructura del quirófano (table-mounted robots), como el Cyberknife (www accuray.com) o el **DaVinci**, o pequeños y estar fijados a la propia estructura ósea del paciente (patient-mounted robots), como el SpineAssist (www mazorst.com), evitando desplazamientos relativos entre la base del robot y el paciente. También aparecen

los robots móviles, capaces de navegar por el interior del cuerpo humano, y por extensión, los robots colaborativos, pequeños robots de capacidades limitadas que trabajan conjuntamente para realizar una determinada tarea quirúrgica.

Por otra parte, toda esta complejidad inherente a la revolución tecnológica implica un aprendizaje de las nuevas capacidades disponibles. Históricamente, el aprendizaje en cirugía ha sido un aprendizaje por oportunidad, es decir, si un paciente tiene apendicitis, los estudiantes pueden aprender aspectos técnicos de la apendicetomía. Afortunadamente esta situación ha cambiado, en parte debido a las nuevas técnicas docentes y, cada vez más, gracias al uso de simuladores quirúrgicos se pueden generar diferentes escenarios, y además incrementando la complejidad de forma gradual. De esta forma, las intervenciones se pueden practicar de forma segura en un simulador antes de operar a un paciente. Sin embargo, la sensación de realidad que ofrecen estos simuladores es todavía claramente mejorable [54].

Esta revolución global está teniendo una amplia respuesta en los diferentes países industrializados. Por ejemplo, en el plano docente, en la Universidad de California, San Diego, ingenieros, médicos y científicos, están uniendo fuerzas para desarrollar el nuevo Instituto de Ingeniería en Medicina, que trata de dar respuesta a la necesidad de colaboración entre ingenieros y médicos. En España, universidades como la Universidad Miguel Hernández han puesto en marcha programas de máster en bioingeniería donde aparecen asignaturas de robótica, imagen o señal aplicadas a medicina.

En lo referente a aplicaciones, tanto comerciales como proyectos de investigación, se puede encontrar un excelente resumen en los tutoriales recientemente publicados en “IEEE Robotics & Automation Magazine” [50], [51], [52].

5.2. El proceso global de cirugía asistida por computador (CAS)

El proceso global de una operación quirúrgica puede esquematizarse según se indica en la figura 5.1.

Empezando por la esquina superior izquierda, podemos observar que en la etapa de diagnóstico, aparecen diferentes tecnologías, como imágenes médicas de diferente naturaleza física, algoritmos de fusión de la información de dichas imágenes, conocimiento previo que se tiene de la anatomía, etc. Como hemos visto en el capítulo 3, existen diferentes principios físicos que se utilizan para obtener las imágenes, y por tanto, dependiendo del sistema utilizado, existirá una determinada incertidumbre en la información obtenida. En las etapas preoperatorias, las referentes a la planificación quirúrgica, existe también un gran número de tecnologías involucradas, algoritmos de reconstrucción 3D y visualización de dicha información, procedimientos para la generación de la planificación, etc. En cada uno de estos procedimientos, también existe una incertidumbre asociada a cada uno de ellos, es decir, dependiendo del algoritmo de reconstrucción 3D utilizado, se obtendrá un resultado con un nivel de precisión u otro.

Tenemos que pensar que los resultados de estas etapas previas son utilizados en la etapa intraoperatoria, y por tanto, es necesario conocer las incertidumbres que conllevan. En la etapa intraoperatoria, la navegación quirúrgica y la robótica hacen uso de la infor-

mación 3D y 2D previa, lo cual implica que nos basamos en la información proporcionada por las etapas previas para guiar con precisión nuestras herramientas ruborizadas. En general, las operaciones de calibración de estos sistemas, conocidas como registro quirúrgico, hacen uso de los modelos 3D estáticos obtenidos con técnicas de imagen de gran precisión, pero la naturaleza estática de estos modelos hace difícil su adaptación a la realidad quirúrgica, en la cual los tejidos y órganos son muchas veces deformables. Por otra parte, al igual que en los casos anteriores, cada tecnología utilizada intraoperatoriamente tiene una determinada incertidumbre, es decir, un navegador tiene unas determinadas características de funcionamiento, un robot tiene una determinada precisión, etc.

También hemos de pensar que en las etapas postoperatorias, cuando se realiza una revisión de los resultados quirúrgicos, también se vuelven a utilizar imágenes médicas. Idealmente, estas etapas de revisión también sirven para incrementar el conocimiento de los equipos quirúrgicos y mejorar pasos realizados en etapas previas (este es el significado de la flecha que cierra el lazo).

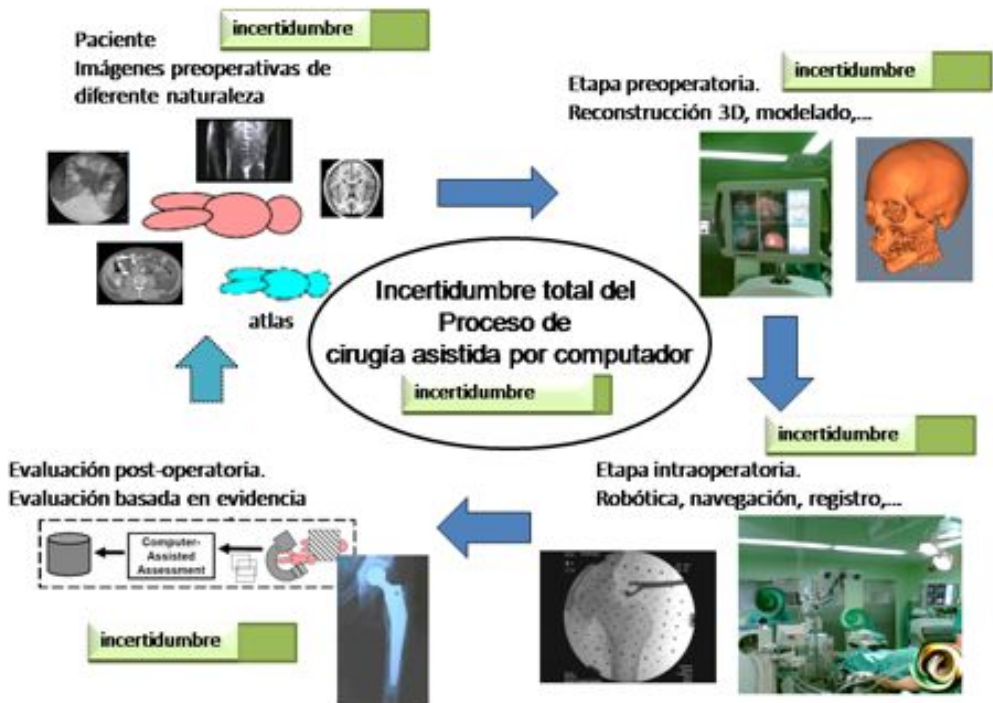


Figura 5.1: Proceso CAS global

5.2.1. El problema de la incertidumbre

Siguiendo con el razonamiento anterior, es muy importante seguir trabajando para mejorar todas y cada una de las tecnologías que se utilizan en el ciclo CAS, y por ejemplo diseñar robots que puedan tener una mayor precisión de posicionamiento. Sin embargo, a la vista de la figura 5.1, el objetivo final es disminuir la incertidumbre global, por lo que carece de sentido utilizar un robot de extremada precisión guiado por un sistema de navegación de muy mala precisión.

Esta situación justifica la frase introductoria de este capítulo “*Con ayuda de la robótica y la navegación quirúrgica podemos ubicarnos con mucha precisión, pero en el lugar precisamente equivocado*” (Prof. Leo Joskowicz).

Por tanto, en un capítulo dedicado a la robótica quirúrgica deben de tratarse todas las áreas que aparecen en el ciclo global, si bien en este caso nos vamos a centrar únicamente en la etapa intraoperatoria. El estudiante interesado puede consultar el capítulo 3 y los textos recomendados para ampliar sus conocimientos.

5.2.2. Limitaciones de la robótica quirúrgica

Como a toda tecnología incipiente, la robótica quirúrgica está afectada por una serie de factores que condicionan su aceptación final como un “miembro más” del equipo quirúrgico. Algunos de estos factores están relacionados con el estado actual de la tecnología, pero también intervienen otros factores. Revisemos brevemente algunos de ellos:

Tecnología pionera. Como hemos comentado anteriormente, la aparición de la robótica quirúrgica es reciente, y como tal, los sistemas que se utilizan son los precursores de los que se utilizarán próximamente. Se tratan de sistemas con tecnologías innovadoras, y en muchas ocasiones su aceptación requiere de una voluntad por parte del equipo quirúrgico para aceptar/corregir/mejorar diversos aspectos. Esta voluntad está permitiendo que en la interacción *personal clínico - ingenieros* se esté avanzando hacia sistemas más evolucionados y cómodos para trabajar.

Tecnología cara. De forma muy simplificada, podemos decir que el valor del uso de la robótica en operaciones quirúrgicas se establece según la siguiente ecuación:

$$\text{valor} = \text{beneficios} - \text{costes}$$

Según la cual el primer término nos indica que el uso de la robótica debe proporcionar unos beneficios respecto a la no utilización de la misma. Este aspecto evidente es importante que se tenga en consideración. Añadir la robótica a un procedimiento quirúrgico debe hacerse cuando se pueda evidenciar que se producen beneficios, como por ejemplo, nuevas capacidades de destreza que permiten nuevos abordajes, incremento de las habilidades del equipo quirúrgico, menor tiempo de hospitalización o menores complicaciones. En caso contrario, los costes de la robótica (por otra parte altos por ser una tecnología pionera) no justifican su uso.

Curva de aprendizaje pronunciada La pendiente de la curva de aprendizaje también es un factor que se debe considerar a la hora de hablar de los factores que

afectan a la aceptación de la robótica. Una pendiente pronunciada produce un rechazo en la aceptación por parte del equipo clínico. Es importante que se tenga tendencia a simplificar el uso de esa tecnología, que no la tecnología en sí misma, para tratar de allanar la curva de aprendizaje.

Nuevas barreras tecnológicas Este aspecto hace referencia a los nuevos retos tecnológicos que aparecen por el uso de la robótica en ambiente quirúrgico. El más conocido es el uso de la tecnología haptics. La aparición de robots hace que se pierda la sensación de tacto y de fuerza. Para proporcionar de nuevo esa sensación al cirujano se desarrollan sistemas hapticos, como sistemas maestros, joysticks, o guantes actuados. Sin embargo es importante que este sistema haptico que aparece no entorpezca la labor principal del cirujano, por lo que se debe adaptar la tecnología haptics existente a los entornos quirúrgicos. Otras barreras tecnológicas que hay que superar es la mejora en la instrumentación, debido a los nuevos abordajes que se pueden hacer con robots. Un ejemplo serían la necesidad de ópticas más precisas y pequeñas, o el diseño de nuevos efectores finales para los sistemas robóticos.

5.3. Clasificación de dispositivos robóticos para cirugía

Se podrían realizar diferentes clasificaciones de robots para cirugía, en función de su esquema de control (autónomo o teleoperado), o de su posición con respecto al paciente (table-mounted-robot o patient-mounted-robot), o bien de su arquitectura mecánica (robot serial, paralelo o robot móvil).

En este apartado se ha optado por mostrar una clasificación basas en la tarea que realiza el robot, y en su área de aplicación. No se pretende aquí hacer un estudio de cada uno de los robots citados sino simplemente realizar una pequeña clasificación no exhaustiva. Algunos de ellos se presentan en más detalle en la siguiente sección a modo de ejemplo. Aquí, y siguiendo la norma del libro de añadir referencias e-link, se muestran unas tablas conteniendo ejemplos de robots clasificados según su área de trabajo, y se anima al estudiante a utilizar dicho enlace para empezar una búsqueda de la información disponible sobre ese dispositivo. Algunos de ellos son proyectos finalizados y abandonados, pero se ha creído adecuado añadirlos desde un punto de vista docente.

Tarea	Sistema	Institución/- compañía	e-link
cirugía maxilofacial	OTTO	Medical Faculty Charité	enlace a IEEE Xplore
cirugía maxilofacial	Robopoint	Surgical Robotics Lab	enlace a ScienceDirect

Tabla 5.1: Sistemas robóticos de cirugía maxilofacial

Tarea	Sistema	Institución/- compañía	e-link
Guiado de cámara endoscópica	AESOP	Intuitive Surgical Inc. (computer motion)	enlace a ZEUS
Guiado de cámara endoscópica	LER	TIMC-IMAG, France.	enlace a IMAG
Guiado de cámara endoscópica	FreeHand	FREEHAND (pro-surgics)	webpage
Tele-ecografía	Hippocrate	LIRMM	enlace a PubMed
Endoscopio	Hyper Endoscope	Biomedical Micro-mechanics Lab	enlace a artículo
Tele-ecografía	Otelo	Sinters SA	enlace a webpage
Cirugía endoscópica	MC ² E	LRP-LIRMM-CEA	enlace a artículo

Tabla 5.2: Sistemas robóticos de “Imaging” en cirugía

Tarea	Sistema	Institución/- compañía	e-link
Cirugía remota	ARTEMIS	FZK	enlace a PubMed
Cirugía remota	Blue Dragon	Bionics Lab UCSC	webpage
Cirugía remota	Imhotep	nBio lab - UMH	webpage
Cirugía remota	DaVinci	Intuitive Surgical	webpage
Compensación del temblor	PADyC	TIMC-IMAG	webpage
Cirugía cardíaca	HeartLander	Carnegie Mellon Univ	webpage

Tabla 5.3: Sistemas robóticos de cirugía abdominal y torácica

Tarea	Sistema	Institución/- compañía	e-link
Ablación	CIBERKNIFE	Accuray Inc.	webpage
Ablación próstata	Ablatherm	Edap	webpage

Tabla 5.4: Sistemas robóticos de radiocirugía

Se remite al estudiante a los enlaces, libros recomendados y a las referencias [50], [51], [52] citadas anteriormente para ampliar los conocimientos sobre estos dispositivos.

Tarea	Sistema	Institución/- compañía	e-link
Guiado bajo CT	MINERVA	EPFL	enlace a PubMed
Guiado bajo CT	Neuroarm	Clinical Neuro- science, Calgary	webpage
Neurocirugía	Neuromate	Renishaw (ISS)	webpage
Neurocirugía	Neurobot	Imperial College	enlace a IEEE Xplo- re
Porta-Microscopio	Surgiscope	ISIS (Elekta)	webpage

Tabla 5.5: Sistemas robóticos de cirugía neurocirugía

Tarea	Sistema	Institución/- compañía	e-link
Rodilla	ACROBOT	Stanmore Implants Worldwire	webpage enlace del Imperial College
Cadera	ARTHROBOT	Telerobotics lab, Seoul	enlace a IEEE Xplo- re
Rodilla y cadera	CASPAR	URS Gmbh	
Cadera	CRIGOS		enlace a IEEE Xplo- re
Rodilla	PRAXITELES	OmniLifeScience (Praxim)	webpage
Cadera	ROBODOC	Curexo	webpage
Columna vertebral	SPINEASSIST	Mazor Robotiics	webpage

Tabla 5.6: Sistemas robóticos de cirugía ortopédica

5.4. Ejemplos de dispositivos robóticos para cirugía

En este apartado se van a ampliar la información de algunos de los dispositivos robóticos mencionados en las tablas anteriores. El objetivo es mostrar el amplio abanico de diferentes dispositivos existentes. Se recomienda seguir los enlaces proporcionados y ampliar los detalles con la bibliografía recomendada.

5.4.1. Sistema de radiocirugía con fuente externa. Cyberknife.

[Cyberknife](#) es un sistema robótico de radiocirugía diseñado para el tratamiento de tumores, tanto cancerígenos como no. Es posible utilizarlo tanto en próstata, pulmón, columna, hígado, páncreas y riñón.

La figura 5.2 muestra una imagen del sistema completo. Se puede observar que consta de un robot de grandes dimensiones que porta un sistema LINAC acelerador de partícula.

5.4. EJEMPLOS DE DISPOSITIVOS ROBÓTICOS PARA CIRUGÍA 139

las que dispara partículas de alta energía y de una camilla robotizada que permite el movimiento del paciente. El robot que porta el LINAC es un robot serial clásico (un Kuka KR240) capaz de posicionar con mucha precisión el sistema de Rayos X LINAC de su extremo. La camilla está robotizada y es capaz de posicionar al paciente.

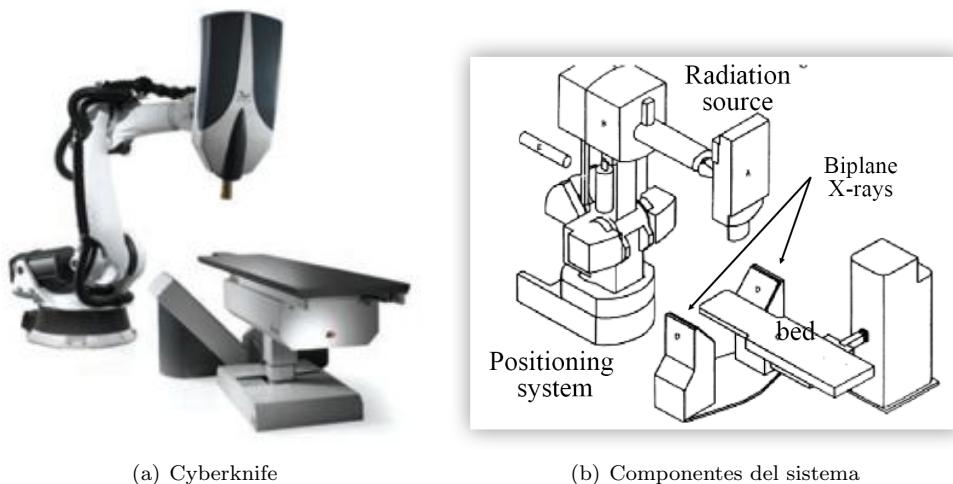


Figura 5.2: Sistema Cyberknife. Extraído de www.cyberknife.com

La figura 5.3 muestra el esquema de funcionamiento XSight (existen versiones particularizadas para pulmón (xsight-lung) o para columna (xsight-spine)). Se trata de un sistema adicional de guiado por imagen que permite al sistema Cyberknife prever el movimiento que se produce en el tejido tumoral como consecuencia de los movimientos respiratorios. También existen opciones de realizar este tracking con marcadores fiduciales.

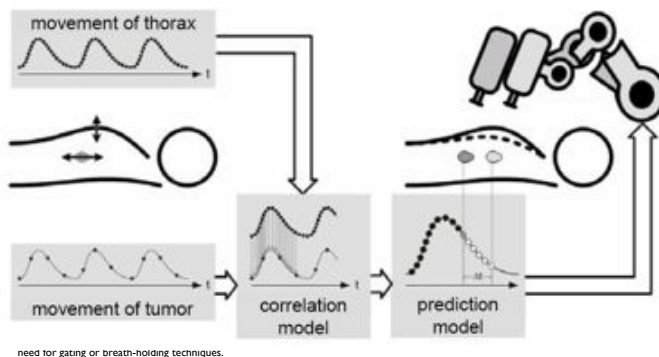


Figura 5.3: Sistema XSight de Accuray. Extraído de www.cyberknife.com

5.4.2. Sistema HIFU con fuente de energía interna

Un sistema que está adquiriendo un gran interés son los sistemas HIFU (High Intensity Focused Ultrasound), o Ultrasonidos de Alta Intensidad. Su principio de funcionamiento es sencillo, y se basa en la utilización de ondas de ultrasonidos de alta intensidad convergentes en un punto. La interferencia de las ondas produce un aumento de la presión acústica en ese punto, lo cual se traduce en una vibración de los tejidos. Como resultado de los procesos de dilatación y contracción de ese tejido, y debido a que los tejidos no son perfectamente elásticos, se produce una pérdida de energía en términos de calor, es decir se produce una cantidad de calor localizada y relativa a la amplitud de la onda sonora. La figura 5.4 esquematiza este principio.

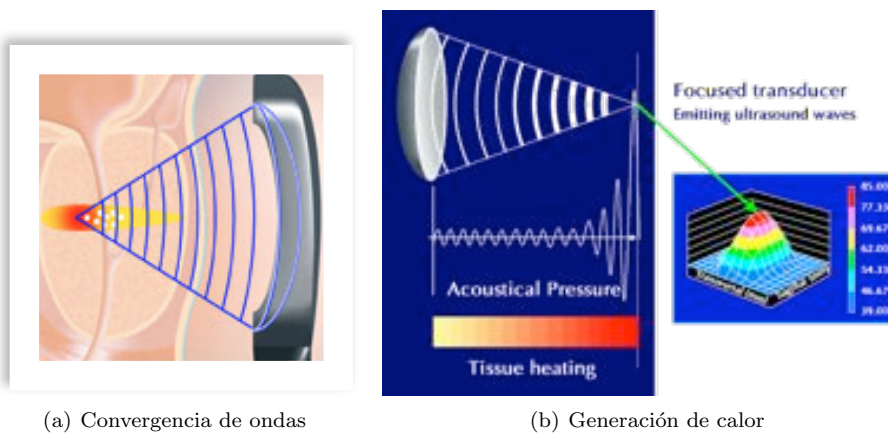


Figura 5.4: Principio de funcionamiento de los sistemas HIFU

Este principio está siendo aplicado con éxito en sistemas de ablación de tejidos, como por ejemplo el sistema HIFU para tratamiento de próstata comercializado por [Ablatherm](#). En este enlace se pueden encontrar vídeos explicativos del tratamiento de próstata mediante ultrasonidos localizados de alta intensidad. La figura 5.5 muestra imágenes del sistema de Ablatherm®.



Figura 5.5: Sistema HIFU de Ablatherm®. Extraído de [hifu-planet](#)

5.4.3. Sistema de telecirugía. DaVinci

El sistema *DaVinci* es actualmente el robot quirúrgico con mayor éxito e implantación a nivel mundial, (sólo en EE.UU. hay más de 1600 unidades, y en España, en 2012 hay 22 unidades de robot clínicos y 2 más con tares de formación). La [lista de procedimientos quirúrgicos](#) que se pueden realizar con el sistema *DaVinci* es amplia, y están apareciendo continuamente nuevos.

El sistema *DaVinci* (figura 5.6) está formado por un una columna que porta 3 ó 4 brazos robóticos, una columna de visualización endoscópica y una consola maestro donde se ubica el cirujano principal y los mandos del sistema y que proporciona imagen este-reoscópica. Los brazos del robot tienen una primera parte con articulaciones pasivas que se utilizan para posicionar al robot sobre el punto de trocar. La parte activa de cada brazo (figura 5.7) es un sistema de doble paralelogramo (se implementa con correas) que permite la creación de un centro de rotación remoto (RRC Remote Rotation Center) que se hace coincidir con el trocar o punto de acceso al paciente. De esta forma se garantiza que la rotación de la herramienta se produce sobre dicho punto, lo que es muy importante desde el punto de vista de la seguridad del paciente.



Figura 5.6: Primera generación del sistema *DaVinci* (1999). Extraído de [Intuitive Surgical](#)



Figura 5.7: Esquema del RRC del *DaVinci*

Gran parte del éxito del sistema DaVinci se debe al amplio [catálogo](#) y a la gran destreza que se logra con los instrumentos que se colocan sobre los brazos robóticos (figura 5.8). Comercialmente se conocen como EndoWrist y son instrumentos miniaturizados y actuados. La transmisión se realiza con cables y poleas desde el efector final de los brazos robóticos, donde se encuentran los motores. Esta transmisión, junto con los movimientos de orientación del RRC, permite dotar de hasta 7 grados de libertad al efector final de la herramienta, lo cual permite una gran destreza y amplitud de movimientos.



Figura 5.8: Ejemplos de instrumentos EndoWrist®. Extraído de [Intuitive Surgical](#)

El sistema actual del DaVinci, el [DaVinci Si HD](#) (figura 5.9), lanzado en 2009, incorpora una serie de mejoras como son la posibilidad de disponer de 2 consolas en las que pueden trabajar 2 cirujanos, incorporar un sistema de visión 3D para el resto del equipo quirúrgico, pacos para [SILS](#) (Single Input Laparoscopic Surgery - Cirugía Laparoscópica de puerto único) o sistemas de fluorescencia para identificar los tejidos.

En los últimos tiempos existe mucha discusión acerca de la aplicación de la ecuación ($\text{valor} = \text{beneficios} - \text{costes}$) al sistema DaVinci, pero es indudable el papel de pionero que este sistema representa en cuanto a la integración de un robot en el quirófano. Estas discusiones tratan acerca del termino de coste principalmente, pero también sobre mejoras que deben ir apareciendo en este tipo de robots teleoperados, como son:

- Mejora en el tiempo de “set-up” del robot. El tiempo que actualmente se requiere para posicionar el robot al inicio de la operación es relativamente alto. Esto se hace con el paciente bajo anestesia, por lo que es prioritario minimizar dicho tiempo.
- Mejoras en el intercambio de herramientas. Durante una operación se producen múltiples cambios de herramientas, por lo que mimimizar el tiempo necesario para cada intercambio es también muy importante.
- Mejoras en cuanto a la ergonomía del robot. Durante el tiempo que dura la operación, puede suceder que los 4 brazos del robot, que están actuando sobre un campo quirúrgico pequeño, tiendan a colisionar entre si por el exterior del paciente. Esto requiere parar la operación, volver a realizar un re-ajuste (“set-up”) del robot y continuar con la operación. Este tiempo es tiempo extra que debe ser eliminado.
- Mejoras en cuanto a la ergonomía del cirujano. En esta área se reclama principalmente la adición de capacidades hapticas reales al cirujano. Hemos de pensar que, debido a la cinemática de la laparoscopia, se ha perdido capacidad haptica, y el cirujano sólo dispone de la realimentación visual para realizar operaciones como la

5.4. EJEMPLOS DE DISPOSITIVOS ROBÓTICOS PARA CIRUGÍA 143

sutura. Adicionalmente se trabaja en nuevas aplicaciones que mejoren la ergonomía del cirujano.



Figura 5.9: Sistema DaVinci Si HD (2009). Extraído de [Intuitive Surgical](#)

5.4.4. Sistema de cirugía ortopédica. Robodoc

Pero si existe un dispositivo que debemos considerar como pionero en cuanto a la aplicación clínica real de robots en el quirófano, este podría ser el [Robodoc](#). Robodoc es un sistema para la cirugía ortopédica que puede utilizarse para operaciones de reemplazo de cadera o de rodilla (THR - Total Hip Replacemet, TKR - Total Knee Replacement). La idea original del Robodoc es debida al Dr. Hap Paul, un veterinario que en 1986 ideó el sistema que fue creado gracias al apoyo de IBM. Este sistema fue un adelantado a su tiempo, puesto que no se disponía de las tecnologías de imagen y de navegación actuales, y sin embargo se conocen más de 5000 cirugías entre 1994 y 2004 realizadas con Robodoc.

La figura 5.10 muestra como ejemplo cómo era el embalaje de las primeras versiones de Robodoc, y que contaban con el propio robot (un robot de configuración Scara) y toda la electrónica de control del mismo. Nos podemos imaginar que estas primeras versiones del sistema eran un sistema caro, con tecnología todavía no madura, no rentable y complicado.

Sin embargo, Robodoc ha continuado su evolución, con diferentes modelos y versiones (figura 5.11), lo que ha permitido que en la actualidad exista una empresa ([Curexo](#)) que comercialice el sistema actualizado.



Figura 5.10: Primeras versiones de Robodoc. Cortesía de André Bauer, MD.



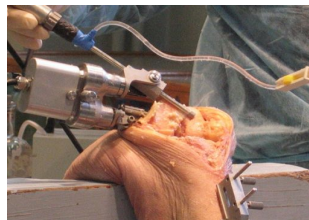
1991



1992



2002



2006



(b) Actual

(a) Evolución de Robodoc

Figura 5.11: Historia de Robodoc®. Cortesía de André Bauer, MD.

5.4.5. Sistema de robot montado en paciente. SpineAssist

Uno de los problemas que se necesitan solucionar para utilizar los ejemplos de robots anteriores es el relativo a conocer la ubicación del robot con respecto al paciente. Los ejemplos anteriores, el robot está sobre un carro, columna o plataforma móvil, de forma que cuando se coloca sobre el paciente, lo primero es relacionar la localización de ambos actores. Esto es lo que se conoce como **registro quirúrgico**, y se realiza utilizando las herramientas de localización vistas en el capítulo 2. Este paso es crucial, ya que permite correlacionar las imágenes preoperativas de las que se disponga con la realidad clínica del quirófano.

Otra de las posibilidades de diseño de robot quirúrgico son los llamados robots-montados-en-paciente (patient-mounted-robots). Como su nombre indica, son robots de pequeñas dimensiones que se fijan sobre el paciente. De esta forma no hay movimiento relativo entre el paciente y el robot. Una de las ventajas de este concepto es que si la base del robot (pieza sobre la que se fija/ancla el robot) es compatible con la tecnología de imagen que se vaya a utilizar (CT, X-ray, Fluoroscopia, MRI, etc) se pueden tomar imágenes del paciente con la base ya fijada al cuerpo, de forma que el registro quirúrgico ya esté realizado, y de forma automática se puedan realizar los registros de las imágenes preoperativas con la realidad clínica.

Obviamente esta opción depende de la aplicación quirúrgica a realizar, pues la restricción del tamaño del robot también afecta a su comportamiento dinámico y a la capacidad de fuerzas que puede ejercer. En áreas como la cirugía maxilofacial, la neurocirugía o la ortopédica (ver ejemplo del [praxiteles](#)) se están encontrando operaciones donde la aplicación de robots montados en paciente es beneficiosa.

Como ejemplo de un robot montado en paciente se muestran imágenes del robot [MARS](#), (comercializado bajo el nombre de [SpineAssist](#) por la compañía [Mazor Robotics](#)).

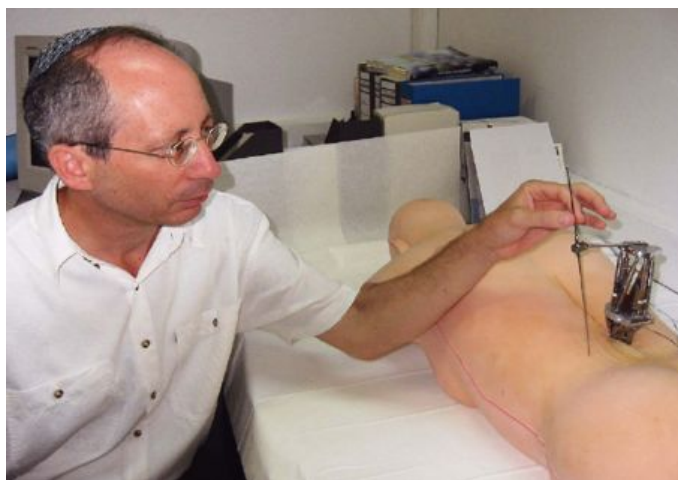


Figura 5.12: Concepto de robot montado en paciente. Mars y Dr. Moshe Shoham

Como se puede apreciar en las figuras [5.12](#) y [5.13](#), la arquitectura del robot MARS

es una arquitectura paralela basada en una plataforma de Stewart-Gough de pequeño tamaño (esta plataforma fue estudiada y analizada con SimMechanics® en la sección 2.5.2). Una vez fijada la plataforma inferior, la plataforma superior tiene 6 grados de libertad, y por tanto puede posicionarse y orientarse en cualquier localización dentro de su espacio de trabajo. Esto permite que pueda utilizarse como guía y ayuda al cirujano. La aplicación mostrada aquí es la de la inserción de tornillos para la fijación de vértebras, aunque en la literatura se puede encontrar artículos con aplicaciones a neurocirugía.



Figura 5.13: Prototipo del Mars. Plataforma de Stewart-Gough

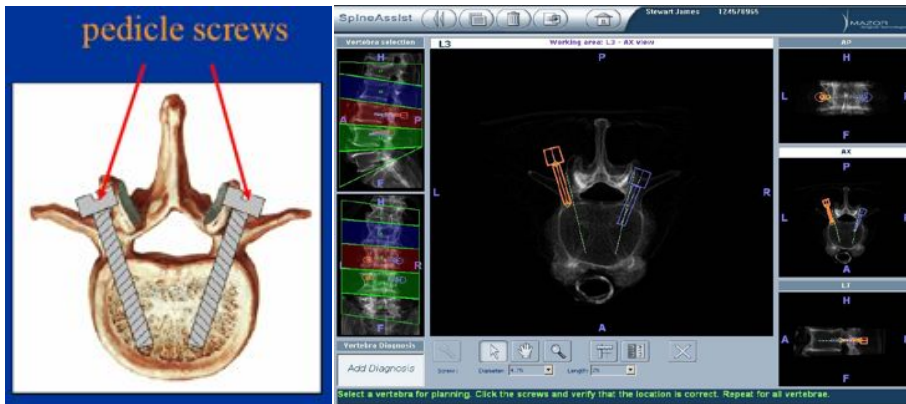


Figura 5.14: Aplicación del SpineAssist®

La figura 5.14 muestra una imagen del software CAD diseñado para la planificación del sistema comercial SpineAssist. En ella se puede observar como el sistema, tras haber realizado la fusión de las imágenes preoperatorias con las imágenes de fluoroscopia, puede utilizar herramientas CAD para seleccionar el tornillo a utilizar, seleccionar la localización de dicho tornillo y el ángulo de entrada. Para ello se puede hacer la planificación sobre el modelo 3D, teniendo información de los tres planos (sagital, coronal y axial) y evitando así el contacto con la médula. Esta información es luego traspasada al dispositi-

tivo mecánico (SpineAssist) que se encuentra en el campo quirúrgico y que se posiciona para indicar la trayectoria que debe realizarse.

5.4.6. Sistema de robots colaborativos

Otra tendencia de la robótica en este campo se dirige al desarrollo de micro-robots que se internan de forma completa dentro del cuerpo del enfermo. La primera generación de estos ingenios han sido las cápsulas robóticas que se introducen por la boca para la diagnosis del aparato digestivo [83]. No se trata de un dispositivo pasivo que sigue el camino natural, sino que posee un sistema de guiado mixto basado en pequeñas patas y un campo magnético externo que le confieren libertad de movimientos. Otras soluciones orientadas a la cirugía utilizan los denominados robots modulares y reconfigurables que se pasan a través del esófago como una serie de cápsulas y se ensamblan y configuran en el estómago del enfermo, en función de la tarea que tiene que realizar [35]. Estas investigaciones se han recogido dentro del proyecto europeo del Sexto Programa Marco **ARES** (Assembling Reconfigurable Endoluminal Surgical System). La evolución del mencionado proyecto con un fin similar pero orientado al uso de micro-robots colaborativos es el proyecto del Séptimo Programa Marco **ARAKNES** (Array of Robots Augmenting the Kinematics of Endoluminal Surgery). En este caso los distintos micro-robots se encuentran dotados de diferentes herramientas quirúrgicas de intervención y diagnóstico, colaborando entre ellos para realizar una determinada tarea quirúrgica.

La figura 5.15 muestra diferentes imágenes de los robots colaborativos diseñados en la Universidad de Nebraska.

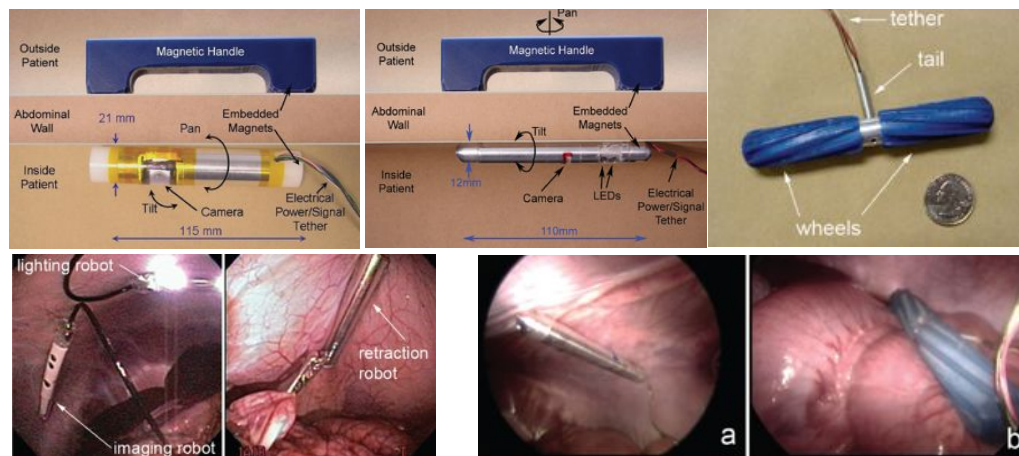


Figura 5.15: Robots cooperativos quirúrgicos. Extraído de [53]

Asimismo, se han ideado micro-robots para intervenir de forma directa en la cavidad abdominal, de entre los que se destaca el micro-robot de la Universidad de Nebraska dotado de dos brazos con instrumentos quirúrgicos y un sistema de visión estéreo [53]. El robot se maneja desde una consola exterior al enfermo que fija el robot en el interior de la pared abdominal mediante un campo magnético. Otras versiones utilizan ventosas como

medio de locomoción sobre la pared abdominal o bien toman la forma de pequeños robots móviles que navegan sobre las vísceras. Estos autores apuntan que el futuro de estos micro-robots se basan en trabajar de forma colaborativa dentro de la cavidad abdominal y del desarrollo de interfases adecuados para gobernarlos.

5.5. Simuladores quirúrgicos

El entrenamiento habitual de los cirujanos en procedimientos quirúrgicos se ha llevado a cabo hasta ahora mediante la utilización de cadáveres, animales vivos o intervenciones quirúrgicas reales guiadas por expertos. El empleo de cadáveres presenta problemas por su elevado coste de mantenimiento y el cada vez más reducido número de cuerpos disponibles. Con respecto a la utilización de animales, se presentan problemas éticos respecto al empleo de animales vivos en experimentación, y por supuesto, la existencia, también en este caso, de un elevado coste de mantenimiento. En último caso, el aprendizaje a través de operaciones reales bajo la vigilancia de un experto presenta inconvenientes en cuanto al número reducido de ocasiones que el cirujano puede realizar una técnica quirúrgica determinada, y el número limitado de situaciones patológicas cuya intervención, el médico, podría realizar, respecto a la variedad de patologías con las que se podría enfrentar en su día a día.

A raíz de estos inconvenientes, en el momento actual, hay un creciente interés por el desarrollo de Sistemas de Simulación Quirúrgicos, frente al empleo de métodos tradicionales. Los simuladores quirúrgicos permiten:

- Reducir los costes asociados al empleo de cadáveres y de animales vivos en el entrenamiento de técnicas quirúrgicas.
- Aportar experiencia al cirujano con una mayor diversidad de casos patológicos y complicaciones.
- Permitir la repetición de las tareas quirúrgicas tantas veces como sea necesario para que el aprendizaje de las mismas sea completo.
- Ofrecer la posibilidad de revisar los procedimientos realizados para estudiar sus ventajas e incluso mejorarlo mediante la modificación de las técnicas empleadas.
- Permitir la búsqueda de soluciones para mejorar los procedimientos que contengan gran dificultad.
- Permitir la planificación y la práctica sobre la anatomía de un paciente específico en la fase preoperatoria.
- Mejorar la comunicación entre los miembros del equipo médico o entre doctores y pacientes.

5.5.1. Requerimientos de los simuladores quirúrgicos

Los simuladores quirúrgicos imponen dos restricciones a los entornos: **realismo virtual** y **realismo físico**. La primera requiere que los órganos se muestren lo más fielmente

posible a cómo aparecen en la cirugía real. La segunda requiere que el comportamiento de los órganos sea lo más similar posible al de los órganos reales. La complejidad de este comportamiento excede la capacidad de cálculo de los ordenadores convencionales, requiriendo alcanzar una solución de compromiso entre realismo y prestaciones.

Según este esquema, podemos dividir las partes de un simulador en:

Display visual Las aproximaciones que podemos encontrar para implementar un display visual son múltiples, y van desde la generación de software propio, hasta las más comunes, que hacen uso de alguna de las múltiples librerías que se pueden encontrar. Muchas de ellas están basadas en [OpenGL](#), como por ejemplo [VTK](#), [IgsTK](#) o [Ogre](#). Para lograr un realismo visual, se precisa una frecuencia de refresco de entre 20 Hz y 60 Hz.

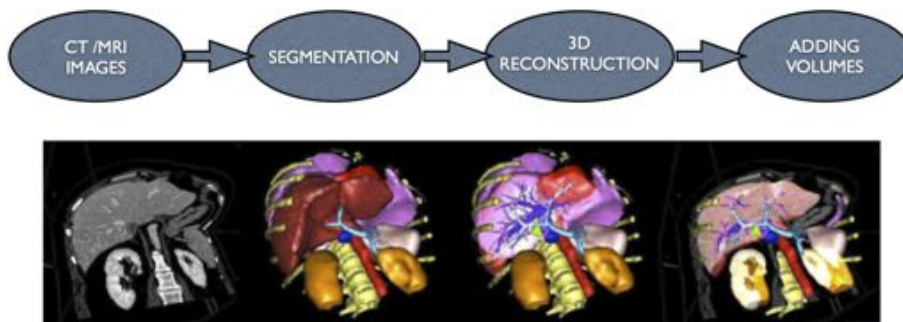


Figura 5.16: Proceso de generación de un display visual

Display háptico En cuanto a los display hápticos que se utilizan en los simuladores, podemos encontrar diferentes opciones, como los dispositivos Phantom de [Sensable-Geomagic](#), los dispositivos de [Force Dimension](#), los Falcon de [Novint](#) o dispositivos propios como el FBB de Karl Storz. Dada el mayor ancho de banda relativo a la sensación de tacto que poseen los seres humanos, éstos sistemas precisan de una frecuencia de refresco de entre 300 Hz y 1000 Hz para tener una sensación realista en este tipo de realimentaciones. Actualmente se pueden encontrar investigaciones encaminadas a añadir sensación táctil a la sensación de reflexión de fuerzas que producen los dispositivos hápticos.

Módulo de colisiones El módulo de colisiones es especialmente importante en los simuladores quirúrgicos. Tradicionalmente se han venido evolucionando algoritmos clásicos de la informática gráfica, como el Axis Aligned Bounding Boxes (AABB), el Oriented Bounding Boxes (OBB), etc. La tendencia actual es a utilizar las capacidades de multiprocesamiento de las GPU (Graphic Processing Unit) para lograr algoritmos más eficientes y realistas.

Motor físico El motor físico es el soler dinámico que resuelve el modelo implementado. En el caso de la cirugía es especialmente importante lograr un equilibrio entre el realismo y la velocidad de cómputo, a la vez que se implementan modelos más

complejos que permitan trabajar con modelos tanto de cuerpo rígido (para modelar huesos y partes rígidas) como modelos reformables (para modelar órganos sólidos o tejidos).

5.5.2. Simuladores comerciales y de investigación

Un magnífico trabajo de recopilación de información sobre los distintos simuladores puede encontrarse en el website del ETZH, bajo el nombre de “**A Web-Based Repository of Surgical Simulator Projects**” (www.virtualsurgery.vision.ee.ethz.ch).

De forma resumida, y siguiendo un esquema de dificultad de implementación en cuanto al realismo físico, podemos clasificar los simuladores en tres categorías, ordenadas de menor a mayor dificultad de implementación, como:

Needle-based Los simuladores de inserción de aguja, necesitan de un modelo físico más sencillo, puesto que lo más importante es simular el comportamiento de la aguja o del catéter, y por tanto se pueden utilizar modelos de una dimensión. Algunos ejemplos son el CathSim, el sistema Cathi-System o el [Angio-Mentor](#).

Minimal Invasive Surgery En los simuladores de cirugía mínima invasiva, el campo quirúrgico que se debe simular contiene toda la dificultad en lo referente al modelo físico, ya que puede contener tejidos, huesos (simuladores de artroscopia) y órganos sólidos (simuladores de laparoscopia). Sin embargo, el pequeño tamaño de este campo quirúrgico, unido al modelo rígido de las herramientas de MIS, permite algunas simplificaciones. Algunos de estos simuladores son el Xitact o el MIST de [Mentice](#), el [Vest](#) (Virtual Endoscopic Surgery Trainer) de Select-It o el [Lapsim](#) de Surgical Science.

Open Surgery La complejidad de este problema hace que los ejemplos que se pueden encontrar suelen combinar soluciones informáticas con el uso de maniquis o modelos reales, como el HATS de HT medical.

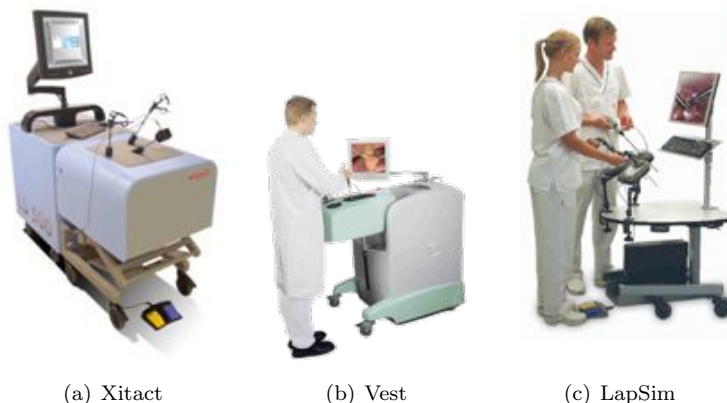


Figura 5.17: Simuladores comerciales de MIS

5.6. Navegación quirúrgica

Una de las tecnologías que se está convirtiendo en imprescindible en el quirófano, sobre todo en determinadas áreas que trabajan con sólidos rígidos, como son la ortopedia o la neurocirugía son los sistemas de navegación quirúrgica.

*Por navegación quirúrgica se entiende los sistemas que permiten la **localización 3D** de los diferentes actores que intervienen en el procedimiento quirúrgico; a saber, **herramientas quirúrgicas, campo quirúrgico e información anatómica** del paciente (imágenes pre-operativas o intra-operativas).*

Los sistemas de navegación permiten conocer la localización relativa de estos actores y por tanto las relaciones geométricas que los unen.

Las limitaciones de la cirugía sin navegación que tratan de solucionar los sistemas de navegación quirúrgica son:

- Capacidad limitada para integrar la información de la imagen médica o planificación pre-operatoria en la realidad quirúrgica del quirófano. No es sencillo correlacionar la información pre-operatoria disponible con la realidad quirúrgica. Este hecho disminuye la utilidad de esta información, al no poder ser utilizada intra-operativamente. Esta dificultad es todavía mayor cuando la planificación se ha realizado sobre información en 2D (fiura 5.18.)
- Dificultad en la interpretación geométrica de las posiciones y orientaciones de los distintos actores (herramientas, tejidos, información) en 3D. Una vez realizado el estudio pre-operativo, se calculan los parámetros geométricos de la operación, como por ejemplo los ángulos de abordaje; sin embargo, no es inmediato trasladar esta información a la operación real.
- Uso intensivo de sistemas de imagen intra-operatoria de radiación ionizante. La necesidad de información en tiempo real del desarrollo de la operación implica que en ocasiones se debe hacer un uso elevado de equipos como el C-arm de fluoroscopia, con el fin de comprobar el desarrollo de la operación.

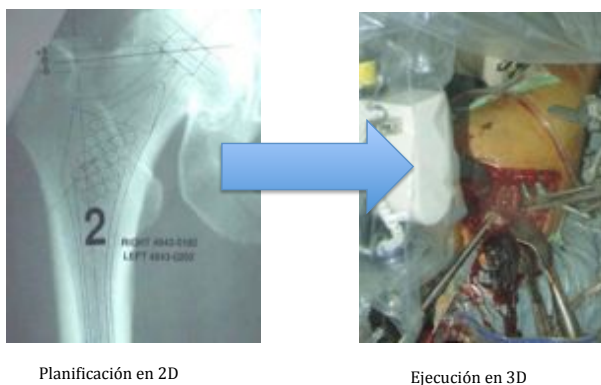
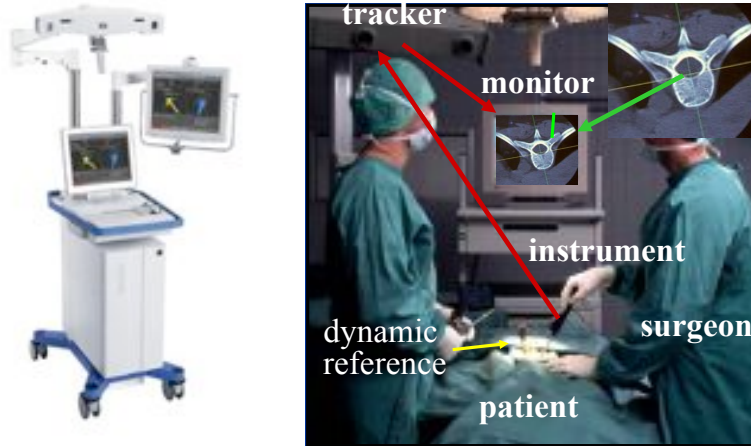


Figura 5.18: Limitaciones de la cirugía sin navegación

5.6.1. Tecnologías de los sistemas de navegación

Básicamente los sistemas de navegación están formados por un sistema de localización 3D (conocido como tracker) y un software de visualización de imágenes medicas que incluye una serie de opciones como son los algoritmos de registro 2D-3D y 3D-3D. La figura 5.19(a) muestra un sistema de navegación de la casa Stryker, en el que se puede observar el tracker y el software asociado.



(a) Sistema de navegación quirúrgica. Cortesía de Stryker®
 (b) Concepto de navegación quirúrgica. Cortesía de Leo Joskowicz.

Figura 5.19: Sistema de navegación quirúrgica

La idea general de estos sistemas es poder conocer en tiempo real la localización de las herramientas quirúrgicas y visualizarlas con relación al campo quirúrgico (figura 5.19(b)). Para lograr esto, el tracker debe ser capaz de:

- localizar la herramienta quirúrgica y posicionarla dentro de la imagen médica (flechas rojas de la figura 5.19(b)),
- pero además, el tracker también debe ser capaz de localizar el campo quirúrgico (localizado a través del sistema de referencia (dynamic reference) señalado de amarillo en la figura 5.19(b)),
- y además, el sistema debe ser capaz de superponer la información disponible (imagen médica) sobre el campo quirúrgico. Esta operación se conoce como registro.

5.6.1.1. Ópticos

La tecnología más extendida en los tracker son los trackers ópticos, basados en construir un par estereoscópico de cámaras. Ejemplos de trackers ópticos son la familiar **Polaris** de NDI-Digital o los sistemas de captura de movimiento de **Optitrack**.

El principio de funcionamiento de estos sistemas es el mismo que el de los pares estereoscópicos de visión. Utilizan la triangulación para obtener la localización de unos marcadores que se colocan sobre el objeto a seguir. Estos marcadores deben ser fácilmente identificables por el sistema de visión. Dentro de los sistemas ópticos existen diferentes tipos, como aquellos que utilizan marcadores pasivos reflectantes o aquellos que utilizan marcadores activos (leds activos).

Para ilustrar el proceso de triangulación, la figura 5.20 muestra el modelo simplificado de par estereoscópico con ejes alineados. En la realidad, los ejes ópticos no estarán alineados, sino convergentes, y por tanto aparecerán los parámetros de calibración del par. En este caso utilizaremos las matrices de transformación homogénea estudiadas en el capítulo 2 para obtener el modelo completo.

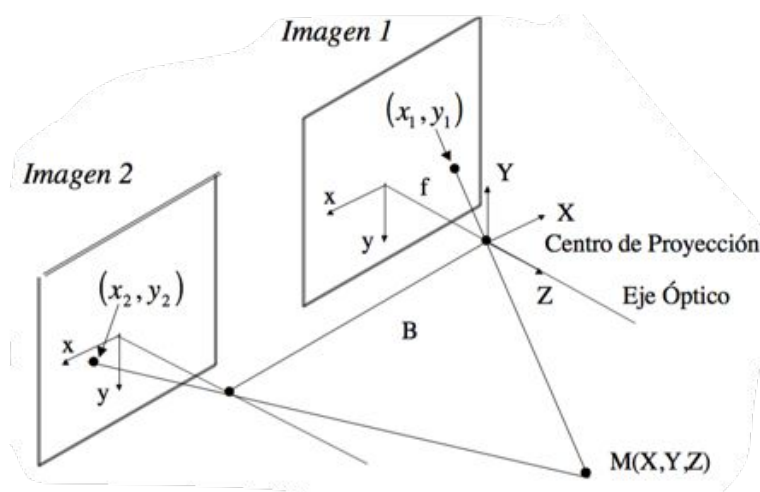


Figura 5.20: Modelo estereoscópico de ejes convergentes

En este modelo simplificado:

- En la primera imagen: $X = \frac{x_1}{f} Z$, e $Y = \frac{y_1}{f} Z$.
- En la segunda imagen: $X = \frac{x_2}{f} Z + B$ e $Y = \frac{y_2}{f} Z$.
- De donde obtenemos las ecuaciones: $Z = \frac{fB}{x_1 - x_2}$; $X = \frac{Bx_1}{x_1 - x_2}$; $Y = \frac{By_1}{x_1 - x_2}$

El principal problema operativo que sucede con estos navegadores son los problemas de oclusión óptica, debido a que un objeto (como por ejemplo cualquier miembro del equipo quirúrgico) se coloque momentáneamente en la línea de visión del par estereoscópico. En ese momento se pierde la localización. Existen diferentes soluciones a este problema, como la redundancia de trackers. Sin embargo, la experiencia con los navegadores y la correcta ubicación del tracker con respecto a la operación quirúrgica que se esté realizando, minimiza el riesgo de este problema.

5.6.1.2. Electromagnéticos

También existen trackers electromagnéticos, que no necesitan tener visión directa de los sensores para localizarlos. El tracker es capaz de detectar la localización espacial de los sensores (pequeñas espiras) debido a las variaciones del campo electromagnético que estos sensores producen. El campo magnético creado por el tracker es de baja intensidad, y por tanto puede atravesar los tejidos y localizar los sensores incluso en el interior del cuerpo.

El principal problema de estos sistemas es que actualmente su campo de acción, es decir el volumen del espacio que son capaces de analizar para localizar los sensores, es menor que en los tracker ópticos. Por esta razón se están utilizando principalmente en operaciones con campo quirúrgico pequeño, como en otorrinolaringología.

Un ejemplo de estos trackers puede encontrarse en el modelo [Aurora](#) de NDI-Digital (figura 5.21).



Figura 5.21: Sistema electromagnético de tracking

5.6.2. Tipos de sistemas de navegación

5.6.2.1. Basados en información CT

Los sistemas de navegación basados en información CT se fundamentan en la utilización de un modelo 3D reconstruido a partir de la información pre-operativa CT. El objetivo es minimizar la necesidad de imágenes de fluoroscopia intra-operativa. De forma resumida, los pasos a seguir en una operación con navegación son:

- El primer paso es obtener la reconstrucción 3D del órgano a partir de la información del CT. En este paso se utilizan las técnicas y algoritmos vistos en el capítulo 3, como el algoritmo *Marching Cubes*. En la figura 5.22 se ejemplifica con la reconstrucción de un fémur.
- El siguiente paso es realizar el registro del órgano. El objetivo es obtener la transformación rígida entre las coordenadas del CT y las coordenadas del sistema de navegación. Este es un paso que puede realizarse con diferentes apropiaciones:

Pin-based registration Se puede realizar utilizando fiduciales (o puntos clave).

Los fiduciales son puntos fácilmente reconocibles que se utilizan para calcular la transformación rígida correspondiente. Los fiduciales pueden ser “pins” artificiales (p.ej. tornillos en el hueso) que han sido añadidos quirúrgicamente antes de tomar las imágenes CT y que son fácilmente identificables en las

imágenes CT. Este sistema es fácil de implementar y muy preciso, pero tiene el inconveniente de necesitar una cirugía extra para la colocación de los fiduciales.

Pinless registration Se pueden realizar adquiriendo una nube de puntos del órgano. Para ellos se utiliza una herramienta palpado (puntero) que permita tocar un determinado punto anatómico e identificar su localización. Una vez creada esa nube de puntos, se utilizan algoritmos de registro 3D-3D como el ICP (Iterative Closest Point) para obtener la transformación de coordenadas. Este método es más complejo, debido al algoritmo de registro entre un punto y una superficie. Además, el uso de algoritmos de naturaleza iterativa precisa que el resultado obtenido por el algoritmo de registro sea verificado por el cirujano.

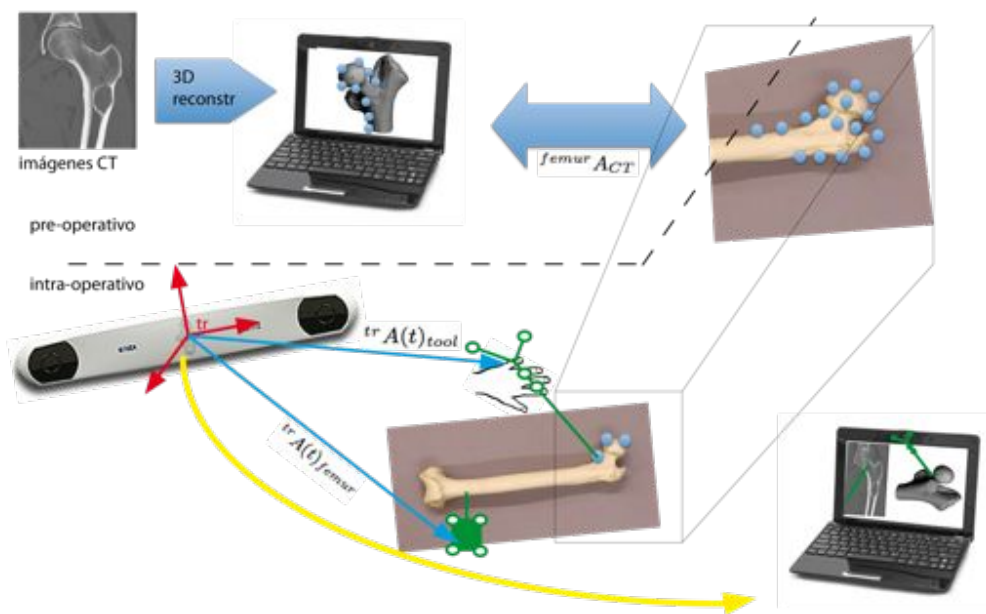


Figura 5.22: Navegación basada en información CT

En la figura 5.22 se ha marcado como ${}^{femur}A_{CT}$ a la matriz de transformación solución del proceso de registro. Esta matriz nos indica como relacionar el modelo 3D obtenido (y las imágenes CT de las cuales se obtiene) con el fémur real en su posición actual en el quirófano. Esta matriz será constante una vez obtenida.

- Monitorizar en tiempo real tanto las herramientas quirúrgicas, (obtener ${}^{tr}A(t)_{tool}$) como la localización del órgano (en la figura ${}^{tr}A(t)_{femur}$).
- Actualizar en tiempo real (flecha amarilla en la figura) la información de la localización del modelo 3D (y de las imágenes 2D de CT) junto con las localizaciones de la herramienta.

Las principales ventajas de esta técnica de navegación son que se precisa de un “planning pre-operativo”, que se evita el uso de la fluoroscopia y que se dispone de capacidad de navegación 3D. Por contra, sólo es aplicable en aquellos casos en que se disponga de un modelo 3D de calidad, a la vez que requiere que la anatomía del órgano en el que se trabaja esté accesible para poder adquirir los puntos necesarios (o los fiduciales) para poder hacer un registro. Además, el proceso de registro necesita de un tiempo de quirófano.

5.6.2.2. Basados en información de Fluoroscopia

Otro sistema de navegación intra-operatoria es el uso intensivo de la fluoroscopia como imagen intra-operatoria. La ventaja de este sistema es que no es necesaria la etapa de registro, tan solo se necesita una calibración del brazo C-arm para conocer la posición del cabezal óptico respecto al órgano visualizado. También se necesita corregir la distorsión de las imágenes de fluoroscopia. Se trata de una metodología robusta y que no es sensible a las oclusiones ópticas.

En contra tiene que se trata de una navegación 2D, ya que en ningún momento se presenta la información en 3D, y además se hace un uso intensivo de las radiaciones ionizantes, con el consiguiente problema para las manos del cirujano, que de forma repetitiva realice estas operaciones. Se ha utilizado de forma profusa en ortopedia, debido a la facilidad de trabajar con los huesos.

También han aparecido sistemas 3D de fluoroscopia, que utilizan técnicas similares de reconstrucción (ver el ejemplo `test3.m` del ejercicio 1 del tema 3) para presentar la información de fluoroscopia según los planos sagital, coronal y axial.

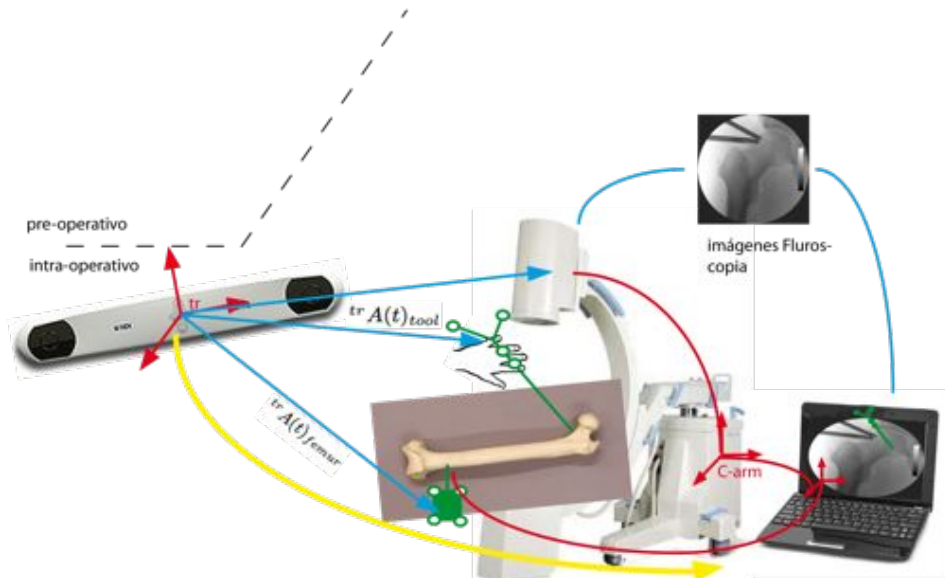


Figura 5.23: Navegación basada en fluoroscopia

5.6.2.3. Imageless - Sistemas sin imagen

Los sistemas de navegación “imageless” como el [OrthoPilot](#) se basan en la utilización de un modelo genérico del órgano (en nuestro caso el fémur) y adquirir la superficie 3D real del mismo mediante la nube de puntos de contacto. Se requiere también el movimiento de la articulación para adquirir el modelo cinemático de la misma, y conocer los ángulos y planos de rotación de la articulación. Este sistema se ha utilizado en ortopedia, principalmente en TKR y THR (ver por ejemplo este [artículo](#)).

Adicionalmente han aparecido métodos más evolucionados de navegación, que hace uso de combinaciones como la utilización de CT+Fluoroscopia o CT+US (ultrasonido) para tratar de tener la precisión de las imágenes CT y la capacidad intra-operatoria de imágenes como el US o la fluoroscopia.

5.7. ROBOTICA MEDICA APLICADA

5.7.1. Análisis del espacio de trabajo de una muñeca paralela de uso quirúrgico.

La cirugía mínima invasiva se caracteriza por la utilización de pequeñas incisiones menores de 10 mm de forma que los instrumentos penetran en el cuerpo atravesando dichas incisiones. En estos puntos de entrada se colocan los trocares, de forma que la cinemática de las herramientas es la vista en la figura 5.24 en la que el punto de entrada es una articulación esférica y además existe un grado de libertad de traslación a lo largo del eje de la herramienta quirúrgica. Los dispositivos robóticos que se diseñen han de obedecer dicha cinemática y, por supuesto se deben añadir criterios de seguridad. La solución más común es el uso del RRC (remote rotation center) (ver ejemplo en la figura 5.7), que coloca el centro de rotación de la muñeca del robot sobre el trocar. Además,

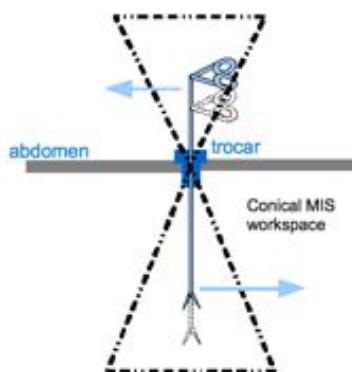


Figura 5.24: Restricciones cinemáticas en MIS

es muy importante realizar un trabajo previo de optimización del espacio de trabajo requerido. A menudo se diseñan dispositivos con espacios de trabajo mucho mayores que el necesario para realizar la tarea. Ello implica complicaciones mecánicas, y problemas

de ergonomía en el quirófano. En diferentes trabajos científicos se encuentran referencias sobre el espacio de trabajo objetivo de las operaciones de MIS laparoscópicas. Dicho espacio corresponde a un cono de 60° como el mostrado en la figura 5.25. Por otra parte, la presencia de singularidades cinemáticas dentro del espacio de trabajo útil del dispositivo robótico es algo indeseable [81].

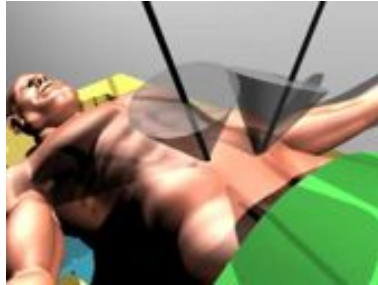
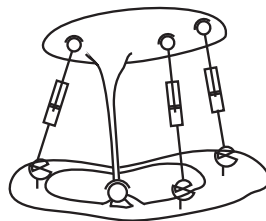


Figura 5.25: Espacio de trabajo en forma cónica en cirugía MIS abdominal

En este ejercicio se va a realizar análisis del espacio de trabajo de una muñeca esférica, de configuración paralela. El objetivo del mismo es comparar el espacio de trabajo de esta muñeca con el espacio de trabajo objetivo (el cono de laparoscopia).

La muñeca que se va a analizar es una plataforma de arquitectura paralela 3UPS1S de dimensiones reducidas. La plataforma 3UPS1S (figura 5.26(a)) está compuesta por tres cadenas cinemáticas o “patas”, donde cada una de las mismas tiene tres articulaciones, una articulación esférica (S) que une la plataforma inferior con cada “pata”, una articulación prismática (P) a lo largo de cada “pata” y una articulación universal (U) que une la cadena cinemática con la plataforma superior. Adicionalmente hay una articulación esférica (S) que une directamente las plataformas inferior y superior. Como se muestra en la figura 5.26(a), la 3UPS1S tiene la característica especial de que el punto de rotación del mecanismo **P** situado en la articulación 1S se encuentra por debajo de la plataforma inferior. Esta característica permite situar el punto **P** sobre el trocar, de forma que las rotaciones producidas en la herramienta MIS lo hacen sobre el punto de inserción como puede observarse en la figura 5.26(b).



(a) Esquema cinemático de la 3UPS1S



(b) Dibujo-boceto de la ubicación del dispositivo durante MIS

Figura 5.26: Muñeca laparoscópica

5.7.1.1. Metodología de análisis

Las plataformas esféricas paralelas consisten en una plataforma superior unida a otra inferior a través de varias cadenas cinemáticas o “patas”. Habitualmente, el número de patas es igual al número de grados de libertad (GDL) de la plataforma. Las plataformas esféricas (o muñecas esféricas paralelas) tienen 3 GDL de orientación.

El espacio de trabajo de una plataforma paralela de orientación se define como: “aquellas orientaciones posibles de la plataforma superior”. El término “posible” indica que las restricciones impuestas por las articulaciones y/o las dimensiones y geometría de la plataforma esférica no impiden que esa orientación sea realizable. Se pueden encontrar en libros de robótica avanzada diferentes formas de representar estas posibles orientaciones. En este ejercicio haremos uso de la representación utilizando los cuaternios de Hamilton.

El siguiente cuaternio representa una determinada orientación:

$$\mathbf{p} = \begin{bmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos(\theta/2) \\ \mathbf{u} \sin(\theta/2) \end{bmatrix} \tag{5.1}$$

$$\mathbf{p}\mathbf{p}^T = \|\mathbf{p}\|^2 = e_0^2 + e_1^2 + e_2^2 + e_3^2 = 1 \tag{5.2}$$

donde, de acuerdo al teorema de Euler, \mathbf{u} es el vector unitario que representa el eje de rotación, y θ es el valor escalar de la rotación alrededor de ese eje.

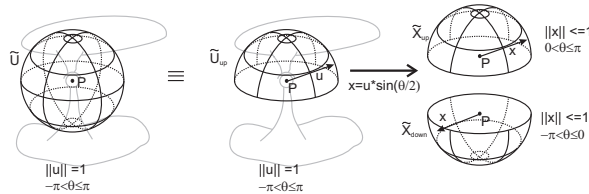


Figura 5.27: Explicación de la representación de las orientaciones en cuaternios

De la definición de \mathbf{u} , se puede observar que los posibles vectores unitarios correspondientes a los ejes de rotación forman una esfera ($\tilde{\mathbf{U}}$) de radio unidad en \mathbb{R}^3 (figura 5.27). Además, una rotación de un ángulo θ alrededor de un eje \mathbf{u} es lo mismo, (o tiene el mismo efecto) que una rotación de un ángulo $-\theta$ alrededor de un eje $-\mathbf{u}$. Por tanto, para analizar todas las posibles rotaciones (desde 0 a 2π , o desde $-\pi$ a π), sólo es necesario analizar una de las semiesferas. Nos centramos en la semiesfera superior ($\tilde{\mathbf{U}}_{up}$).

Tomando en cuenta esto, se analiza el comportamiento del cuaternio \mathbf{p} . Se puede observar que el primer término de la expresión $(\cos(\theta/2))$, será siempre positivo en el intervalo que va desde $-\pi$ a π , mientras que el término $\sin(\theta/2)$ tendrá valores positivos o negativos dependiendo el ángulo girado. Si analizamos la proyección en \mathbb{R}^3 de todos los posibles cuaternios (todas las posibles rotaciones), es decir, el producto de $\mathbf{u} \sin(\theta/2)$, se observa que si consideramos como vector unitario \mathbf{u} el correspondiente a la parte superior de la esfera, es decir $\mathbf{u} = [u_x, u_y, u_z]^T$ con $\|\mathbf{u}\| = 1$ y $u_z \geq 0$, y las rotaciones dentro del intervalo $-\pi < \theta \leq \pi$, entonces el producto $\mathbf{x} = \mathbf{u} \sin(\theta/2)$ representa todos los puntos

que están dentro y en la superficie de la esfera ($\tilde{\mathbf{X}}$) en \mathbb{R}^3 . En la esfera ($\tilde{\mathbf{X}}$), la parte superior implica que $x_z \leq 0$, luego u_z será siempre positivo, y por tanto $\sin(\theta/2) \leq 0$ y $-\pi < \theta \leq 0$. Análogamente, la semiesfera superior implica que $0 < \theta \leq \pi$. De esta forma, un punto de $\tilde{\mathbf{X}}_{\text{up}}$ representa una rotación de $\theta = 2 \arcsin(\|x\|)$ en la dirección de \mathbf{x} , y un punto de $\tilde{\mathbf{X}}_{\text{down}}$ representa una rotación de $\theta = -2 \arcsin(\|x\|)$ en la dirección de $-\mathbf{x}$.

Una de las ventajas de utilizar la representación de una esfera para las rotaciones es que el espacio de trabajo se visualiza como un objeto sólido, ya que en cada eje de rotación existirán dos valores críticos, el mínimo y el máximo valor permitido, y los valores que quedan fuera de ese intervalo serán valores no admisibles.

El procedimiento para calcular el espacio de trabajo de forma gráfica será el siguiente:

1. Para una rotación determinada, es decir un punto \mathbf{u} de la esfera $\tilde{\mathbf{U}}$ y un escalar θ , se corresponde un punto \mathbf{x} en la esfera unidad $\|x\| \leq 1$. El cuaternio que representa esta orientación se escribe como:

$$\mathbf{p} = [\cos(\theta/2), \mathbf{u} \sin(\theta/2)]^T = [\sqrt{1 - \|x\|^2}, \mathbf{x}]^T \quad (5.3)$$

Utilizando este cuaternio, se calcula la matriz de rotación de la platform superior.

2. Una vez obtenida la matriz de rotación, se utiliza la cinemática inversa de la plataforma paralela para obtener la localización del mecanismo.
3. Una vez conocida la localización del efector final, se evalúan las restricciones de los actuadores. En el caso de un actuador prismático, se calcula la longitud l y se verifica que su valor esté dentro del rango de valores admisibles para la carrera de dicho actuador $l_{\min} \leq l \leq l_{\max}$. Si no es así, dicha orientación no es admisible.
4. Se realiza la comprobación de los valores angulares admisibles en todas las articulaciones pasivas del mecanismo (en este caso esférica y universales).
 - En las articulaciones universales se comprueba que el ángulo entre la normal al plano que contiene la base de la articulación y el vector dirección alineado con la correspondiente pata, esté dentro de los márgenes permitidos por la geometría de dicha articulación mecánica.

5.7.1.2. Análisis del espacio de trabajo con teoría de screws

Espacio de trabajo objetivo

Antes de estudiar el espacio de trabajo del dispositivo propuesto, se va a realizar la representación del cono invertido de cirugía laparoscópica utilizando la formulación anterior. Para ello, como muestra la figura 5.28, se ha utilizado los ángulos roll-pitch-yaw $ropy$, dando como puntos válidos del espacio de trabajo aquellos que su ángulo $roll < \pm 30^\circ$, su ángulo $pitch < \pm 30^\circ$ y su ángulo $yaw < \pm 10^\circ$.

La figura 5.29(a) muestra la figura obtenida en la que se representa en la esfera $\tilde{\mathbf{X}}$ las orientaciones del cono laparoscópico. En ella se puede observar que las orientaciones posibles (el cono de laparoscopia) representan un cilindro sólido cuyo eje de revolución

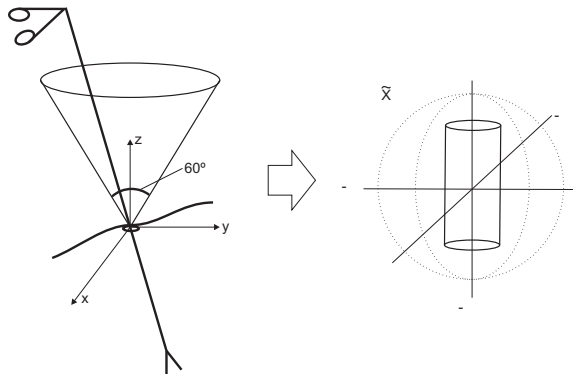


Figura 5.28: Representación del espacio de trabajo del cono laparoscópico

se extiende a lo largo de z . Como puede verse [71], la ventaja de esta representación es que las orientaciones posibles se representan como un sólido tridimensional.

Para obtener esta figura 5.29(a) se ha utilizado el siguiente código en Matlab®:

```

% programa para el calculo y grafico del espacio de trabajo objetivo
% se parte del cuaternio, se pasa a matriz R y a angulos rpy
% los limites admisibles se expresan en valores de los angulos rpy
%
5  indx = 0;
   cone_workspace(21,21,21)=0;
   for x = -1.0:0.1:1.0
       indx = indx + 1;
       indy = 0;
10  for y = -1.0:0.1:1.0
       indy = indy + 1;
       indz = 0;
       for z = -1.0:0.1:1.0
           indz = indz + 1;
15  usin_fi_2 = [ x; y; z ];
           % test if usin_fi_2 is inside a unit sphere
           a = norm(usin_fi_2);
           if ( a <= 1.0 )
               % first element of p
20  cos_fi_2 = sqrt( 1.0 - a*a );
               p = [cos_fi_2; usin_fi_2 ];
               % test for workspace and determinant
               [ok] = test_cone_lap_kinematics(p);
               % store results
25  cone_workspace(indx, indy, indz) = ok;
           else
               % the point is outside the sphere
               cone_workspace(indx, indy, indz) = 0;

```

```

        end
30     end
    end
end
[x, y, z] = meshgrid(-1:0.1:1.0, -1.0:0.1:1.0, -1.0:0.1:1.0);
p = patch(isosurface(x,y,z, cone_workspace, 0.9));
35 isonormals(x, y, z, cone_workspace, p);
set(p, 'FaceColor', 'green', 'EdgeColor', 'none', 'FaceAlpha', 0.5);
daspect([1 1 1])
view(135,30)
camlight;
40 lighting phong
grid on
title('Representacion del workspace objective'); %,'FontSize',16);
% save results
%save cone_workspace

```

Este código hace uso de las siguientes funciones auxiliares:

```

function ok = test_cone_lap_kinematics(p)
% matrix rotation from quaternion
R = p_to_R(p);
%de matriz de rotacion a angulos rpy
5 rpy = tr2rpy(R);
%comprobacion de los limites de angulso rpy
if (abs(rpy(1))<=(30*pi/180) && abs(rpy(2))<=(30*pi/180)
&& abs(rpy(3))<=(60*pi/180))
    ok=1;
10 else
    ok=0;
end
return

```

```

function A = p_to_R(p)
% This function computes the matrix rotation from a quaternion.
% use like this:
% A = p_to_R(p)
5 % where:
% A = is the matrix rotation
% p = is the quaternion vector
%
A = 2*[ p(1)^2 + p(2)^2-0.5      p(2)*p(3) - p(1)*p(4)      ...
10 p(2)*p(4) + p(1)*p(3); ...
      p(2)*p(3) + p(1)*p(4)      p(1)^2 + p(3)^2-0.5      ...
      p(3)*p(4) - p(1)*p(2); ...
      p(2)*p(4) - p(1)*p(3)      p(3)*p(4) + p(1)*p(2)      ...
      p(1)^2 + p(4)^2 - 0.5 ];
15 return

```

```

%TR2RPY Extrae de una transformacion homogenea sus correspondientes
% angulos RPY
%
% [R P Y] = TR2RPY(T) devuelve un vector que contiene los
5 %angulos RPY en radianes,
% Copyright (C) Peter Corke 1993

function rpy = tr2rpy(m)
10   rpy = zeros(1,3);

   if abs(m(1,1)) < eps & abs(m(2,1)) < eps,
       rpy(3) = 0;
       rpy(2) = atan2(-m(3,1), m(1,1));
       rpy(1) = atan2(-m(2,3), m(2,2));
15   else,
       rpy(3) = atan2(m(2,1), m(1,1));
       sp = sin(rpy(3));
       cp = cos(rpy(3));
       rpy(2) = atan2(-m(3,1), cp * m(1,1) + sp * m(2,1));
20       rpy(1) = atan2(sp * m(1,3) - cp * m(2,3), cp*m(2,2) - sp*m(1,2));
   end

```

Análisis del espacio de trabajo

Para el calculo de este espacio de trabajo, se consideran validas las rotaciones posibles que cumplen los siguientes restricciones de diseño:

- Las longitudes de los actuadores están dentro de los limites de carrera de los actuadores utilizados en el prototipo. $L_{min} < L_{act} < L_{max}$.
- El ángulo **S** comprendido entre la normal a la base y el vector \mathbf{u}_i que define la articulación prismática esta por debajo del umbral que establece la articulación S utilizada.
- El ángulo **U** comprendido entre la normal a la base y el vector \mathbf{u}_i que define la articulación prismática esta por debajo del umbral que establece la articulación U utilizada.

Además, la figura 5.29(c) muestra que el espacio de trabajo del dispositivo diseñado es mayor que el necesario para la realización de las tareas laparoscópicas, luego dicho dispositivo es “suficiente” para realizar dichas operaciones.

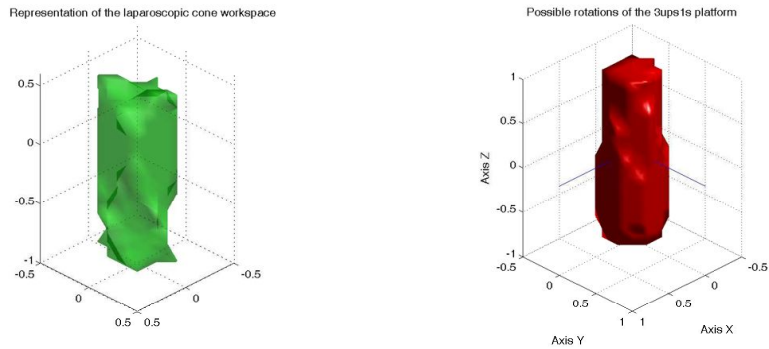
Para resolver este ejercicio se han desarrollado las siguientes funciones de Matlab®:

`ups1s_Geometric.m` - Función que define la geometría de la plataforma paralela, tanto las dimensiones como los límites de las articulaciones.

`ups1s_workspace_singularities.m` - Procedimiento iterativo que recorre todos los puntos de la esfera unidad haciendo, calculando el cuaternio correspondiente y haciendo una llamada a la función `test_3ups1s_kinematics.m` en cada punto para comprobar si dicho punto pertenece o no al espacio de trabajo.

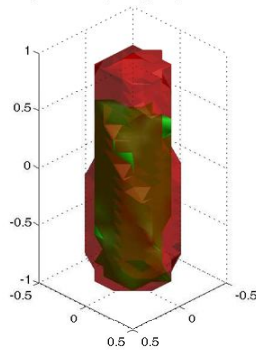
`test_3ups1s_kinematics.m` - Función que resuelve la cinemática inversa de la plataforma 3ups-1s.

Estas funciones que permiten resolver este problema se encuentran en el cd-adjunto.



(a) Representación en Matlab del espacio de trabajo objetivo (b) Representación del espacio de trabajo efectivo de la plataforma 3-UPS-1S

Comparative of objective workspace (green) and 3UPS1S workspace (red)



(c) Comparativa de los espacios de trabajo objetivo y efectivo de la solución propuesta

Figura 5.29: Análisis del espacio de trabajo

5.8. Resumen del capítulo

1. El robot quirúrgico es una **“herramienta quirúrgica avanzada”**, es decir, como un dispositivo puesto en manos del cirujano que le permite incrementar sus capacidades, hacerlo más preciso, mitigar los problemas de fatiga, mejorar la destreza, etc.
2. La cirugía asistida por computador (CAS) es un proceso global que engloba varias

etapas; diagnóstico, planificación, intervención, etapas post-operatorias. Es importante mejorar cada una de ellas para poder mejorar el proceso global.

3. Ejemplos de robots quirúrgicos:
 - Cyberknife
 - Ablatherm
 - Sistema Da Vinci
 - Robodoc
 - SpineAssist
 - Robots colaborativos
4. Los simuladores quirúrgicos imponen dos restricciones a los entornos: **realismo virtual** y **realismo físico**. La primera requiere que los órganos se muestren lo más fielmente posible a cómo aparecen en la cirugía real. La segunda requiere que el comportamiento de los órganos sea lo más similar posible al de los órganos reales.
5. Partes de un simulador quirúrgico: Display visual, display háptico, módulo geométrico de colisiones, motor físico.
6. Por navegación quirúrgica se entiende los sistemas que permiten la **localización 3D** de los diferentes actores que intervienen en el procedimiento quirúrgico; a saber, **herramientas quirúrgicas**, **campo quirúrgico** e **información anatómica** del paciente (imágenes pre-operativas o intra-operativas).

5.9. Ejercicios

1. Modificar las funciones de Matlab[®] anteriores para obtener una representación gráfica de las singularidades de la plataforma 3ups1s. Para ello se debe obtener en primer lugar la matriz jacobiana de dicha plataforma.

Bibliografía para ampliar

Handbook of robotics B. Siciliano, O. Khatib. Springer 2008. ISBN: 978-3-540-23957-4. El capítulo 52 presenta una visión general de “Medical Robotics and Computer-Integrated Surgery”, centrado en robótica quirúrgica, constituye un excelente documento de revisión del estado de arte.

Surgical Robotics. Sytems Applications and Visions J. Rosen, B. Hannaford, R.M. Satava. Springer, 2011. ISBN 978-1-4419-1125-4. Este libro de 2011 muestra un muy completo estado de arte de los robots quirúrgicos. Se trata de un libro de múltiples capítulos independientes en el que cada capítulo está escrito por los autores del sistema presentado. Muy interesante.

Medical Robotics Vanja Bozovic, I-Tech, 2008. ISBN 978-3-902613-18-9. Libro por capítulos complementario al anterior. Presenta sistemas diferentes a los del libro anterior y con puntos de vista y autores complementarios.

Laparoscopic and Robot-Assisted Surgery in Urology Jens-Uwe Stolzenburg, I.A. Turk and E.N. Liatsikos, Springer 2011. ISBN 978-3-642-00890-0. Libro clínico sobre los procedimientos de laparoscopia, incluyendo la laparoscopia con ayuda del robot DaVinci. Con muchas ilustraciones como los setups del DaVinci en determinados procedimientos.

Capítulo 6

Robótica de rehabilitación. Fundamentos y aplicaciones

“Un científico debe tomarse la libertad de plantear cualquier cuestión, de dudar de cualquier afirmación, de corregir errores.”
Julius Robert Oppenheimer

Indice

6.1. Introducción a la rehabilitación robótica	168
6.1.1. Revisión de las técnicas de rehabilitación en ACV	169
6.2. Sistemas de rehabilitación robótica	170
6.2.1. Sistemas de rehabilitación de miembro superior	170
6.2.2. Sistemas de rehabilitación de miembro inferior	178
6.2.3. Sistemas de rehabilitación de extremidades distales; mano y pie	185
6.3. La realidad virtual como ayuda a la rehabilitación física . .	191
6.4. ROBOTICA MEDICA APLICADA	193
6.4.1. Análisis de un dispositivo de rehabilitación de miembro superior (PuParm)	193
6.4.2. Análisis del gait en un dispositivo de rehabilitación de miembro inferior (GBO)	194
6.5. Resumen del capítulo	199
Bibliografía para ampliar	199

6.1. Introducción a la rehabilitación robótica

Hay un dato reciente que revela claramente la importancia y el auge que está teniendo la robótica de rehabilitación en los últimos tiempos. Según la Organización Mundial de la Salud (OMS) para el año 2050 se incrementará en un 73 % el número de personas mayores de 65 años en los países industrializados, edad en la cual el ser humano es más propenso a accidentes cerebrovasculares, de la misma forma, se presentará una tendencia ascendente de la demanda para la rehabilitación de esta población. Esto conducirá a un crecimiento constante en la atención médica y de rehabilitación.

El accidente cerebrovascular es una de las principales causas de discapacidad en adultos en el mundo. La mayoría de personas afectadas tienen algún tipo de parálisis importante y necesitan ayuda profesional para rehabilitación; consecuencias de los accidentes cerebrovasculares como debilidad muscular, pérdida de la amplitud de movimiento y el deterioro en la generación de fuerzas, generan deficiencias en el control motor, limitando así la capacidad para la vida independiente y la autosuficiencia económica de las personas afectadas.

Por esta razón, se han diseñado muchas intervenciones terapéuticas tradicionales que se utilizan en rehabilitación para promover la recuperación funcional. Las actividades terapéuticas típicas incluyen manipulaciones manuales de las extremidades del paciente, ya sea con el paciente en actitud pasiva, o con éste intentando realizar movimientos voluntarios. Este procedimiento ha demostrado ser capaz de recuperar la facilitación funcional del sujeto mediante la reorganización de la corteza motora.

Sin embargo, varios investigadores han demostrado que la cantidad, duración, contenido e intensidad de las sesiones de entrenamiento son variables importantes en el re-aprendizaje de las habilidades motoras y en el cambio de la arquitectura neuronal. De hecho, se ha observado que cuando el tratamiento presenta mayor intensidad que lo acostumbrado, se obtienen mejores resultados, que repercuten en la calidad de vida del paciente. Desafortunadamente, cuando la terapia se proporciona en un hospital o centro de rehabilitación, la duración de las sesiones suele ser de una hora, dos o tres veces a la semana.

Por esta razón, la posibilidad de incrementar la eficacia de la rehabilitación mediante el aprovechamiento de los avances tecnológicos en áreas como la robótica, mecatrónica y biomecánica es cada vez más popular en todo el mundo. En este caso, el fisioterapeuta debe programar y controlar un dispositivo mecatrónico capaz de replicar (y cuando sea posible, mejorar) las estrategias terapéuticas tradicionales, haciendo posible una dosificación de la terapia de forma cuantitativa, intensiva y repetitiva.

Los robots pueden ayudar a los pacientes en el desarrollo de una fuerza o movimiento, programados acorde con las necesidades de cada individuo, mientras que la retroalimentación continua proporciona al paciente una percepción subjetiva de mejoría.

Estas características hacen de la robótica un posible apoyo en el ámbito de la rehabilitación, tanto para los fisioterapeutas como para los pacientes. La robótica permite combinar una serie de capacidades, como son capacidades sensoriales, motoras, cognitivas, etc.

Endeffector-Based



Exoskeleton

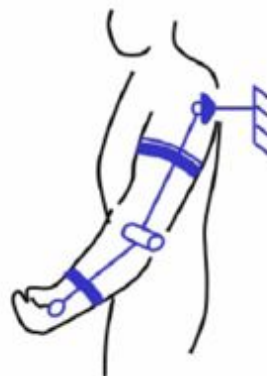


Figura 6.1: Idea de robot basado en efector final y exoesqueleto. Extraído de [15]

Por lo tanto, recientemente se han desarrollado varios sistemas robóticos para lograr este importante objetivo. Y dentro de estos sistemas se puede hacer una clasificación diferenciando dos arquitecturas distintas (Figura 6.1):

Robots basados en efector final: En estos sistemas (Figura 6.1 izquierda) sólo existe un punto de contacto entre el robot y el usuario. Tiene como ventaja la sencillez de ajuste del robot con el paciente; aunque entre las desventajas se puede destacar que la posición del miembro a rehabilitar no queda totalmente determinado con un sólo punto de sujeción. Esta desventaja queda solucionada en algunos dispositivos como el iPAM o el ROBOTHERAPIST3D con el uso de dos o más brazos robots independientes.

Exoesqueletos: En estos sistemas (Figura 6.1 derecha) el robot está fijado a lo largo del miembro, teniendo una estructura de articulaciones parecida a éste. La ventaja de este tipo de robots es que la posición del miembro queda totalmente determinada; mientras que entre las desventajas destaca que los ejes del robot tienen que estar alineados con los ejes anatómicos del miembro en cuestión, lo cual es particularmente difícil en articulaciones como el hombro o la cadera.

6.1.1. Revisión de las técnicas de rehabilitación en ACV

Para iniciar el proceso de rehabilitación de personas que hayan sufrido un accidente cerebrovascular se parte del diagnóstico al paciente, el cual puede evidenciar desde una pequeña lesión que se traduce en una pérdida de fuerza transitoria, hasta una afectación motora mucho mayor, pérdida de visión, problemas de habla etc. Las técnicas utilizadas en la rehabilitación son:

- **Técnicas pasivas:** el objetivo de esta técnica es la prevención de las complicaciones musculoesqueléticas y especialmente, las retracciones musculares secundarias a las

modificaciones histomorfológicas y postespásticas. A esta técnica están asociadas las órtesis de postura.

- Técnicas funcionales: estas técnicas buscan apoyar al paciente con sistemas sensoriales, sistemas de estimulación eléctrica, equipos que permitan ejercitar las extremidades del pacientes, como arnés, cintas para caminatas, etc, todo estos equipos acompañados de un programa de entrenamiento buscan mantener la masa muscular del paciente y devolverle su movilidad.
- Técnicas neuromotoras: a diferencia de las anteriores, que se basan en un enfoque muscular, esta técnica enfrenta de manera global las perturbaciones del paciente en la realización de movimientos, en otras palabras buscar establecer la sinergia de los músculos del cuerpo.

6.2. Sistemas de rehabilitación robótica

En las siguientes secciones de este capítulo, se pasará a comentar algunos de los sistemas robóticos para rehabilitación desarrollados por distintas universidades, tanto para miembros superiores como para miembros inferiores.

6.2.1. Sistemas de rehabilitación de miembro superior

En la tabla 6.1 se resume algunos de los sistemas de rehabilitación actuales para miembro superior, indicando para cada caso la tipología del robot y algunos detalles como puede ser el número de grados de libertad activos.

A continuación se describirán los sistemas expuestos en esta tabla, comentando ciertos detalles tales como las características del sistema, el tipo de actuadores que utiliza, el tipo de esquemas de control, los grados de libertad del sistema robótico y alguna otra característica importante de estos sistemas.

- **GENTLE/S (HapticMASTER):** Usa un maestro háptico para generar un espacio de trabajo que fomenta la terapia mediante robots para estimular y motivar a los pacientes. En la figura 6.2 se muestra una imagen del sistema GENTLE/S.

En esta imagen se observa que el sistema tiene tres puntos de fijación, aunque en realidad dos de ellos (los puntos del codo y antebrazo) no están fijados a la parte actuada del sistema, sino a una parte del sistema utilizada para compensar el peso del brazo (compensación de gravedad). El otro punto de fijación del sistema es un punto de contacto entre la mano del paciente y el maestro háptico.

El maestro háptico utilizado en este sistema es el *HapticMASTER*, desarrollado por la misma universidad *University of Reading, UK* [30]. Este maestro háptico está compuesto por tres grados de libertad actuados mediante servomotores eléctricos.

Al tratarse de un dispositivo háptico, aparte de necesitar un lazo de control, es necesario el uso un lazo de renderización háptica, esto es, un lazo en el que se vayan actualizando la interacción del efector final y el resto de objetos que exista en la

INSTITUCIÓN	SISTEMA	TIPO	DETALLES
University of Reading (UK)	GENTLE/S	Un punto de fijación	Proporciona un espacio de trabajo 3D en la muñeca. Brazo apoyado mediante el uso de un sistema de suspensión pasivo.
ETH (Suiza)	ARMIN	Exoesqueleto	4 g.d.l. activos, tres rotaciones sobre el hombro y una sobre el codo. Traslación pasiva del hombro permitida.
University of Leeds (UK)	iPAM	Dos puntos de fijación	Dispositivo robótico dual de 6 g.d.l. que permite manipulación completa del brazo y hombro.
MIT/Interactive Motion Technologies Inc. (USA)	MIT-MANUS	Un punto de fijación	2 g.d.l. para movimiento en un plano. Una muñeca adicional permite un tercer grado de libertad para controlar en la muñeca.
HOCOMA (Suiza)	ARMEO	Exoesqueleto	Robot con todos los grados de libertad pasivos. Sólo realiza tratamiento resistivo.
MOTORIKA (Israel)	REOGO	Un punto de fijación	3 g.d.l. para movimientos en el espacio. Realiza tratamiento pasivo y activo.
Universidad Miguel Hernández (España)	SINU	Dos puntos de fijación	6 g.d.l. para movimientos en el espacio. Realiza tratamiento pasivo y activo.
Universidad Miguel Hernández (España)	PuParm	un punto de fijación	2 g.d.l. para movimientos en el plano. Realiza tratamiento pasivo y activo.

Tabla 6.1: Algunos sistemas para rehabilitación de miembro superior

realidad virtual. Dicho lazo háptico trabaja a una frecuencia de 2500 Hz, siendo unos 100 Hz la frecuencia mínima para obtener una sensación táctil adecuada.



Figura 6.2: GENTLE/S

En cuanto al esquema de control utilizado en este sistema (figura 6.3), mide la fuerza ejercida por el usuario mediante un sensor de fuerzas. Un modelo interno calcula la Posición, Velocidad y Aceleración (*PVA* en la figura) que un objeto (virtual) tendría como resultado de ser tocado por dicha fuerza.

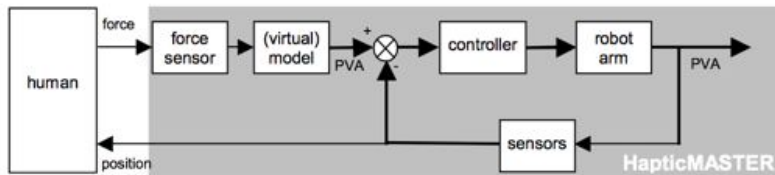


Figura 6.3: Esquema de control utilizado en GENTLE/S

- **MIT-MANUS:** (Figura 6.4) es un sistema robótico desarrollado por el Instituto de Tecnología de Massachusetts (MIT) [61].

Este sistema robótico se basa en una estructura SCARA de dos grados de libertad, en la que el movimiento se realiza en un plano.

Proporciona movimiento continuo pasivo y entrenamiento sensoriomotriz para tareas de agarre.

- **ARMIN:** es un sistema robótico de arquitectura exoesquelética, diseñado por el Instituto Federal Suizo de Tecnología de Zurich, Suiza [7], para ayudar, mejorar y



Figura 6.4: MIT-MANUS

evaluar la rehabilitación neurológica y ortopédica. En la figura 6.5 se muestra una imagen del sistema ARMIN.

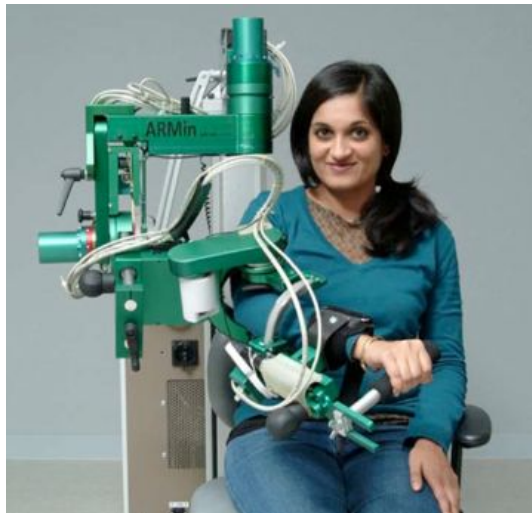


Figura 6.5: ARMIN

En la figura 6.6 se muestra un esquema de la estructura mecánica del sistema, donde se detallan los grados de libertad del robot y el tipo de actuadores utilizado en cada caso; además de los sensores dispuestos en cada articulación. En la figura se puede observar que este sistema hace uso de actuadores eléctricos y distintos sensores para poder registrar las posiciones y fuerzas realizadas por el paciente en cada momento.

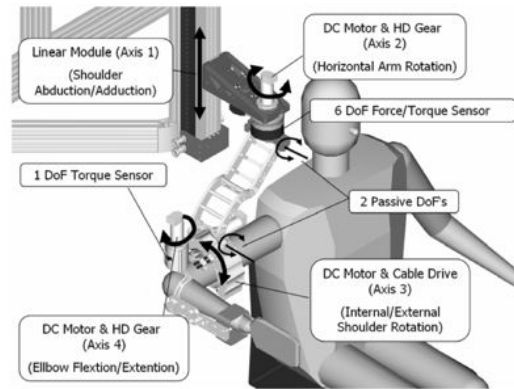


Figura 6.6: Estructura mecánica de ARMIN

El sistema ARMIN dispone de cuatro modos de control distintos:

- *Prerecorded trajectory mode.* En este modo de control, el fisioterapeuta realiza una trayectoria inicial guiando al paciente y robot mientras se va grabando la posición en cada momento. Una vez guardada la trayectoria, el robot la repite a la velocidad que el fisioterapeuta vea oportuna.
- *Predefined motion therapy mode.* El fisioterapeuta puede elegir entre un número variado de ejercicios de terapia estándar preprogramados en el sistema.
- *Point and reach mode.* Cuando se selecciona este modo, se visualiza en un entorno tridimensional las posiciones de la trayectoria que debe seguir la mano del paciente. En caso de que el paciente no sea capaz de realizar el movimiento deseado, el robot ayuda y guía tanto como sea necesario.
- *Patient guided force supporting mode.* En este modo, la trayectoria la define el paciente. El robot mide las posiciones y velocidades y predice las fuerzas y pares requeridas mediante el uso de un modelo mecánico del paciente y robot. De acuerdo con un factor de ayuda ajustable, el robot realiza una parte de las fuerzas y pares necesarias para el movimiento.

Los dos primeros modos son puramente controles de posición, y las fuerzas y pares se miden únicamente para monitorizarlas. En los últimos dos modos, se han desarrollado estrategias de control basadas en control de impedancia y admitancia.

- **iPAM:** (Figura 6.7) es un sistema robótico desarrollado por la Universidad de Leeds, UK [43], con una arquitectura basada en efector final.

Este dispositivo se compone de dos brazos robóticos de tres grados de libertad activos cada uno. El efector de uno de los brazos se fija al brazo del paciente, mientras que el efector final del otro se fija al antebrazo. Con ello se pretende controlar una manipulación completa de brazo y hombro.

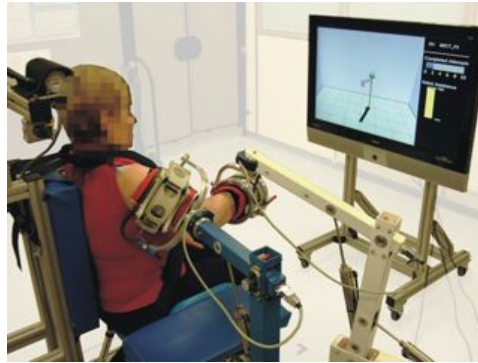


Figura 6.7: iPAM

A diferencia de otros sistemas actuados que hemos comentado anteriormente, iPAM no utiliza motores eléctricos para sus articulaciones actuadas, sino que hace uso de cilindros neumáticos. Esto se hace para añadir algo más de seguridad a la interacción hombre-máquina directamente en el Hardware del sistema, por la flexibilidad intrínseca que tiene la actuación neumática debida a la compresibilidad del aire.

- **ARMEO:** es un sistema robótico para rehabilitación desarrollado por HOCOMA, Suiza [38]. En la figura 6.8 se muestra una imagen de este sistema. ARMEO es



Figura 6.8: ARMEO

un sistema de rehabilitación para miembro superior pasivo, es decir, no hace uso

de ningún tipo de actuador. Este sistema se basa en una compensación mecánica de la gravedad, en la que el soporte del brazo contrarresta el peso del brazo del paciente. De este modo se refuerza la función residual del brazo y se fomenta el movimiento activo de éste en un área de trabajo tridimensional. A pesar de no disponer de sistemas de actuación, sí dispone de varios sensores para poder registrar el movimiento y así poder realizar una realimentación visual mediante aplicaciones tridimensionales en las que se simulan actividades de la vida diaria o simplemente juegos.

- **REOGO:** Es un sistema robótico para rehabilitación desarrollado por MOTORIKA, Israel [62]. En la figura 6.9 puede verse una imagen de este sistema. Se trata de un brazo robot que permite al paciente realizar movimientos en tres dimensiones sobre una pantalla virtual, mientras su brazo descansa en un soporte que compensa la gravedad. ReoGo es un sistema de rehabilitación para miembro superior, donde



Figura 6.9: REOGO

el paciente sigue una serie de movimientos predefinidos en una pantalla virtual, o se deja guiar mientras el brazo robot se mueve por estos puntos. La distancia a alcanzar en cada uno de estos puntos se define dependiendo de la capacidad de cada paciente. La interfaz del sistema permite de manera visual y auditiva estimular el esfuerzo del usuario en cada uno de los ejercicios.

- **SINU:** El robot SINU (los sinu eran los sanadores del antiguo egipto encargados de la rehabilitación de personas) es un sistema robótico desarrollado en la Universidad Miguel Hernández [2] para la realización de tareas de rehabilitación para personas con daño cerebral sobrevenido. Se trata de un sistema formado por dos brazos robots con 3 grados de libertad actuados cada uno de ellos. Uno de los brazos sujeta al paciente por su brazo, y el otro en la mano. La cadena cinemática cerrada que se forma junto con el brazo del paciente permite ejecutar tareas de rehabilitación del brazo en 6 grados de libertad. Adicionalmente, el sistema incorpora una interfaz de realidad virtual para la motivación del paciente. La figura 1.3 muestra una imagen del sistema completo. El sistema Sinu tiene la particularidad de que puede utilizarse con el paciente sentado en una silla o con el paciente en la camilla. Este sistema ha derivado en el producto comercial [RoboTherapist 3D](#).

El robot está actuado por actuadores neumáticos giratorios, lo que le confiere un nivel de seguridad intrínseco. El sistema de control utiliza un esquema de actuación antagonista, basado en lo visto en la sección 1.4.1 con una válvula servoproporcional de presión para cada cámara neumática del actuador rotacional.

El módulo de realidad virtual incorpora sencillos ejercicios para la motivación del paciente, tanto en la realización de las tareas de rehabilitación neuromuscular como en tareas de la vida diaria, como llevarse un vaso a la boca (figura 6.10).



Figura 6.10: Realidad virtual para el ejercicio de beber agua

- **PuParm:** El sistema PuParm está formado por un brazo robótico planar de actuación neumática y configuración paralela y por una interfaz de realidad virtual

para la motivación del paciente. La actuación del Puparm es mediante actuadores neumáticos rotatorios de giro. El sistema está diseñado para la investigación de la respuesta del paciente durante la realización de las tareas de neuro-rehabilitación [9].

La figura 6.11 muestra el brazo robot planar, de configuración SCARA, los sensores para la estimación del estado del paciente, (vistos en el apartado 1.4.2) y el módulo de interacción visual, que en este caso es una tarea 2D de guiado de un coche en una trayectoria recta. El análisis del robot PuParm se realiza en el ejercicio 6.4.1. Este sistema ha derivado en el producto comercial *RoboTherapist 2D*.

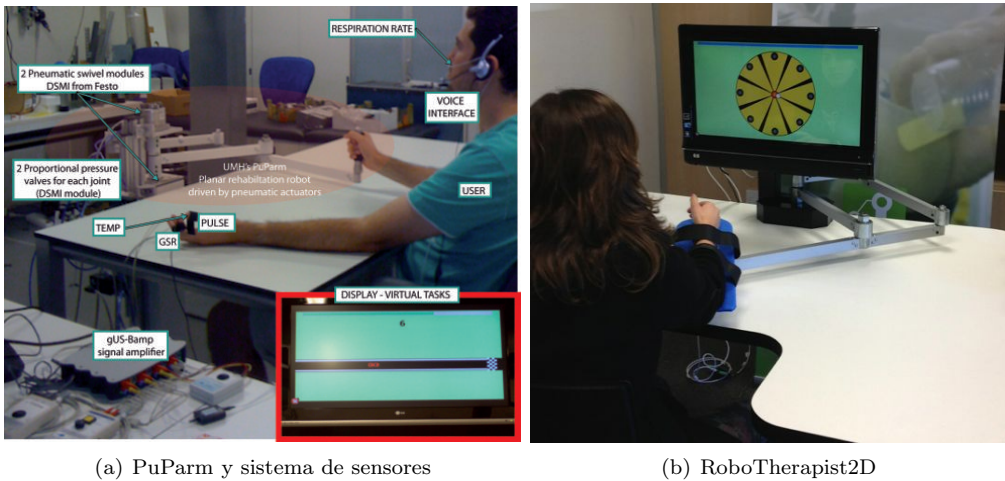


Figura 6.11: Sistema de investigación PuParm y comercial RoboTherapist2D

6.2.2. Sistemas de rehabilitación de miembro inferior

En la siguiente tabla (tabla 6.2) se resume algunos de los sistemas de rehabilitación actuales para miembro inferior, indicando para cada caso la tipología del robot y algunos detalles como puede ser el número de grados de libertad activos. A continuación se describirán los sistemas expuestos en la tabla 6.2, comentando ciertos detalles tales como las características del sistema, el tipo de actuadores que utiliza, el tipo de esquemas de control, los grados de libertad del sistema robótico y alguna otra característica importante de estos sistemas.

- **GBO:** (*Gravity Balancing Orthosis*) es un sistema robótico, de arquitectura exoesquelética, para rehabilitación de miembro inferior desarrollado por la Universidad de Delaware, USA [29]. En la figura 6.12 se muestra una imagen del sistema. El sistema GBO es un sistema pasivo, esto es, no hace uso de ningún tipo de actuación en sus articulaciones. El principio de este sistema es, como indica su nombre, hacer

INSTITUCIÓN	SISTEMA	TIPO	DETALLES
University of Delaware (USA)	GBO	Exoesqueleto	Sistema pasivo. El paciente entrena mediante la alteración de la gravedad. El propio paciente controla el movimiento.
University of Delaware (USA)	ALEX	Exoesqueleto	Sistema con actuación en las articulaciones de la rodilla y cadera.
IPK (Germany)	HapticWALKER	Basado en efector final	Dispositivo basado en efector final, que usa un dispositivo háptico como sistema rehabilitador.
HOCOMA (Suiza)	LOKOMAT	Exoesqueleto	Sistema actuado mediante motores eléctricos. Introduce realidad virtual tanto gráfica como auditiva.
MOTORIKA (Israel)	REOAMBULATOR	Exoesqueleto	Sistema actuado mediante motores eléctricos y realidad virtual para rehabilitación.
MOTEK MEDICAL (Netherlands)	CAREN	Entorno de rehabilitación	Entorno de rehabilitación basado en una plataforma robótica paralela.

Tabla 6.2: Algunos sistemas para rehabilitación de miembro inferior

una compensación mecánica de la gravedad. Esto es así ya que hacen uso de la idea de que a velocidades bajas, la gravedad es la fuerza dominante en las articulaciones de la pierna. En la figura figura 6.13 se muestra de forma esquemática el principio



Figura 6.12: GBO

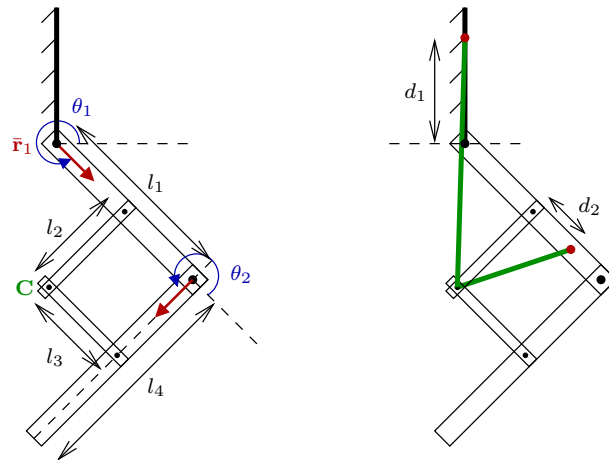


Figura 6.13: Principio de funcionamiento de GBO. Las líneas en color verde representan los resortes que realizan la compensación de fuerzas gravitacionales.

de funcionamiento de la compensación de gravedad que utiliza el sistema GBO. Como se puede observar hace uso de resortes para mantener el balance de gravedad adecuado en el movimiento de abducción de la pierna.

- **ALEX:** Es un sistema robótico, de arquitectura de exoesquelética, para rehabilitación de miembro inferior desarrollado por la Universidad de Delaware (al igual que el sistema GBO) [29]. En la figura 6.14 se muestra una imagen del sistema. A diferencia del GBO, este sistema sí dispone de sistemas de actuación para sus

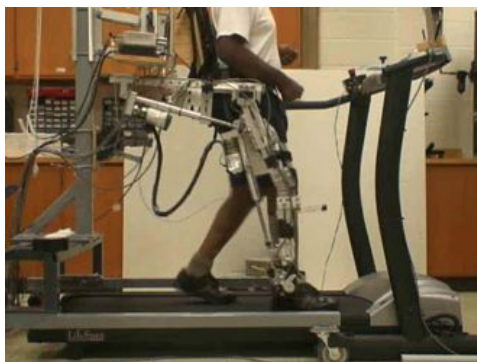


Figura 6.14: ALEX

articulaciones. Posee servomotores eléctricos en las articulaciones del exoesqueleto que coinciden con la cadera y la rodilla. El modo de funcionamiento de este sistema se basa en fuerzas que ayudan al pie a seguir una trayectoria concreta (definida mediante un túnel) durante la rehabilitación, de forma similar a la terapia realizada por el fisioterapeuta. Además, ofrece una realimentación visual de la trayectoria que sigue el pie del paciente. En cuanto al control, el sistema utiliza un control híbrido de fuerza y posición para cada servomotor. Además hace uso de una compensación de gravedad y de lazo cerrado para eliminar los efectos de la fricción. Un esquema de este tipo de control se muestra en la siguiente figura (figura 6.15).

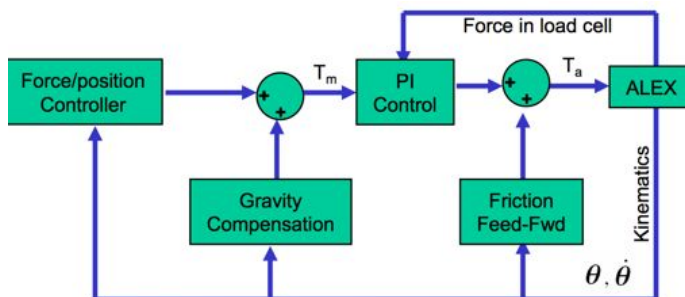


Figura 6.15: Esquema de control articulaciones ALEX

- **HapticWALKER:** Es un sistema robótico para rehabilitación de miembro inferior basado en efector final, desarrollado en el Instituto Fraunhofer (IPK), Alemania [44], el cual puede verse en la siguiente figura (figura 6.16).

Está diseñado con 6+1 grados de libertad para cada pie, y dispone de sensores de fuerza/par de 6 GDL en cada uno de los soportes de los pies. El peso del paciente se elimina mediante un sistema pasivo de compensación de peso. Este sistema dispone de varios modos de funcionamiento:

- *Position control:* En este modo de funcionamiento se utiliza un control de posición del dispositivo robótico, con lo cual es la máquina la que realiza el movimiento completo y el paciente se deja guiar por ella.
- *Compliant control:* En este modo de funcionamiento el paciente realiza movimientos voluntarios y la máquina asiste al movimiento. El nivel de asistencia es variante con el aprendizaje del paciente.
- *Haptic device:* El último modo de funcionamiento es el de un dispositivo háptico. Esto es, que se comporta como un dispositivo que realimenta fuerzas según un entorno virtual. Este modo de funcionamiento sólo puede utilizarse con pacientes muy avanzados en su rehabilitación o con sujetos sanos, ya que es el paciente el que tiene que hacer todo el movimiento e interactuar con las fuerzas que pueda ejercer el dispositivo.



Figura 6.16: HapticWALKER

El sistema de control del HapticWALKER puede verse en la siguiente figura (figura 6.17).

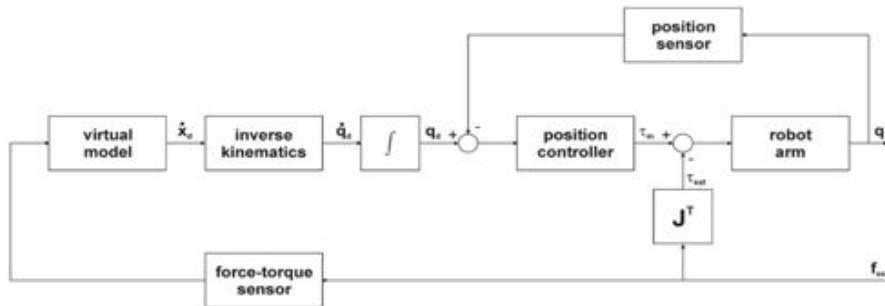


Figura 6.17: Esquema de control de HapticWALKER

- LOKOMAT:** Es un sistema robótico para rehabilitación de miembro inferior desarrollado por HOCOMA, Suiza [38]. En la figura 6.18 se muestra una imagen de éste sistema. Como puede observarse, este sistema está basado en una arquitectura exoesquelética. Al igual que el sistema ALEX, LOKOMAT es un sistema actuado.



Figura 6.18: LOKOMAT

Dispone de motores eléctricos para las articulaciones de la cadera y las rodillas, además de sensores de fuerza para cada articulación. Para mantener al paciente erguido dispone de un sistema de compensación de peso mediante resortes y poleas. En cuanto al control, utiliza un esquema de control de impedancias (figura 6.19). El sistema LOKOMAT hace uso de entornos virtuales para que el paciente disponga de realimentación sonora y visual que ayude a tener un nivel de motivación siempre

alto.

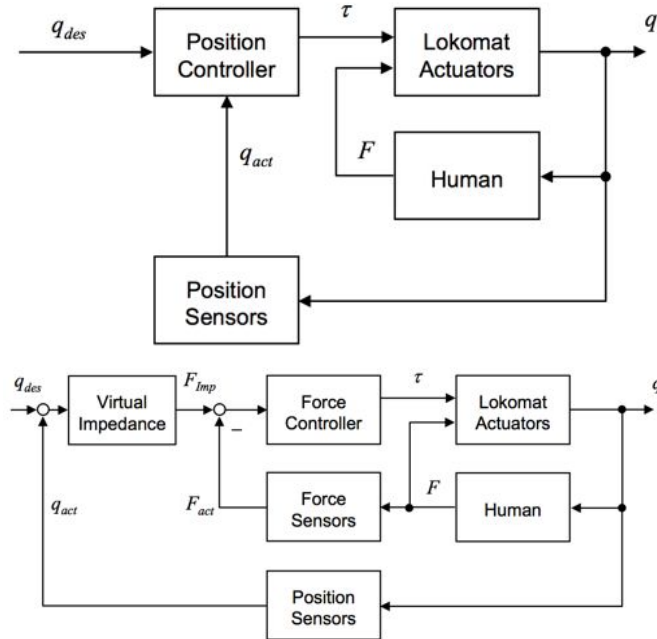


Figura 6.19: Esquemas de control en LOKOMAT

- **REOAMBULATOR:** es un sistema robótico para rehabilitación de miembro inferior desarrollado por la empresa MOTORIKA, Israel [62]. El sistema está basado en una arquitectura exoesquelética, como se presenta en la figura 6.20. El sistema está compuesto por un par de piernas robóticas sincronizadas que le permiten al paciente caminar de manera segura, utilizando patrones estándar de rehabilitación de marcha. Dispone de diversos sensores que monitorean constantemente los movimientos del paciente para ajustar la potencia y velocidad del sistema. Dependiendo del estado del paciente el robot contribuye con el movimiento de marcha, brindando la fuerza necesaria para caminar.
- **CAREN:** es un entorno para rehabilitación de miembro inferior desarrollado por MOTTEK MEDICAL, Países Bajos [58]. El sistema está basado en una plataforma robótica paralela, y se muestra en la figura 6.21. El sistema utiliza una plataforma paralela de 6 g.d.l. con motores hidráulicos. La plataforma posee dos metros de diámetro y un sistema de ocho cámaras en tiempo real que monitorean la marcha del paciente. Éste está inmerso en un sofisticado ambiente virtual con video y sonido, que le presenta diversos obstáculos y protocolos de marcha. La plataforma paralela se mueve de acuerdo al ambiente en pantalla, induciendo determinados



Figura 6.20: REOAMBULATOR

movimientos en el paciente necesarios en su proceso de rehabilitación. El sistema puede utilizarse en varias terapias tales como rehabilitación ortopédica, balance y mejora de la postura, rehabilitación neurológica, entrenamiento de marcha, tratamiento y diagnóstico de fobias, etc, tan variadas como programas virtuales puedan ser concebidos en el sistema.



Figura 6.21: CAREN

6.2.3. Sistemas de rehabilitación de extremidades distales; mano y pie

Durante el proceso de aprendizaje, desde que somos bebés, nuestro sistema nervioso va adquiriendo conciencia de nuestros miembros partiendo de nuestro nervioso central y llegando a nuestras extremidades distales. Si observamos cualquiera de las representaciones

del homúnculo de Penfield (figura 6.22) se aprecia la importancia de estas extremidades en nuestra forma de interactuar con el entorno. Por esta razón, la rehabilitación neuromuscular debe contemplar siempre el efecto que los dispositivos robóticos producen en nuestras extremidades.

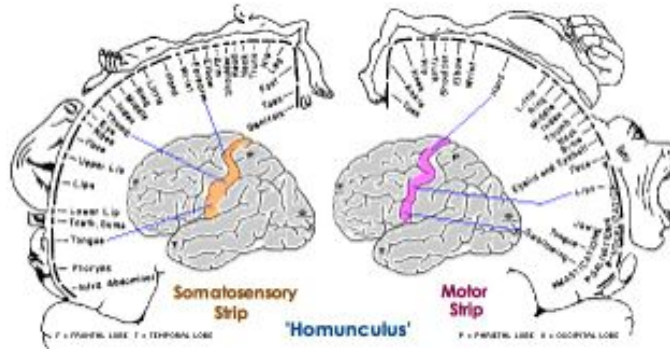


Figura 6.22: Representación del homúnculo de Penfield. Extraído de la web

Actualmente existen escasas referencias y muy pocos dispositivos diseñados específicamente para la rehabilitación de extremidades distales, siendo la mayor parte de ellas sistemas robóticos exosqueléticos que difícilmente pueden utilizarse con pacientes.

Realizar un diagnóstico correcto de la lesión inicial de la mano junto con una continua monitorización del progreso realizado son dos importantes parámetros a tener en cuenta para llevar a cabo una efectiva rehabilitación de la misma. Notar que una valoración subjetiva de todos los posibles parámetros a obtener de la mano es una tarea propensa a cometer errores. Por ello, usar guantes sensibles (*sensing glove*) puede llegar a ser altamente eficaz para diagnosticar el daño existente en la mano, obteniendo excelentes resultados. Aunque estos guantes sean aplicados para el diagnóstico y rehabilitación de la mano como se ha dicho, son incapaces por sí solos de aplicar una resistencia o fuerza auxiliar en la mano del paciente. La movilización forzada en las articulaciones es habitualmente necesaria como parte de la rehabilitación. Si nos referimos a *Hand Exoskeletons* en particular, es posible aplicar fuerzas a cada falange y en algunos casos a varias articulaciones por falange. Sus múltiples grados de libertad ofrecen buenos resultados en terapias de rehabilitación [73]. Exponemos seguidamente sistemas de rehabilitación de miembros superiores, en concreto para la mano, más relevantes que han contribuido a un avance tecnológico con resultados clínicos demostrables en este campo de la rehabilitación.

- VPL DataGlove:** El primer Sensing Glove (figura 6.23) aplicado para el estudio de la mano fue el DataGlove, desarrollado por VPL Research. Guante fino de licra que emplea fibra óptica como elemento sensor para la medida de la posición de las falanges. En su versión estándar usa dos fibras ópticas por falange para la medida de la flexión y extensión del metacarpiano y de la articulación interfalángica proximal [73]. En un extremo se encuentra un LED y en el otro un fotosensor. El cable de fibra óptica presenta pequeños cortes a lo largo de su longitud. Cuando el usuario

dobla un dedo, la luz escapa del cable de fibra óptica a través de estos cortes. La cantidad de luz alcanzado al fotosensor es medida y convertida a un valor de cuánto el dedo ha sido doblado.



Figura 6.23: Primera versión del VPL Data Glove

- Rutgers Master II:** Human-Machine Interface Laboratory, en Rutgers University, desarrolla el primer hand exos- keleton que elimina la necesidad de cables de transmisión ya que se sitúan los actuadores directamente sobre la palma de la mano del paciente. Por ello, se presenta como muy ligero de peso y sin problema alguno de portabilidad. Llamado Rutgers Master como nombre inicial y mejorado con nuevas versiones, vemos en la figura 6.24 el Rutgers Master II que consiste en cuatro microcilindros neumáticos situados en forma de L, alcanzando valores de 16N sobre la falange distal. La última versión llamada Rutgers Master II New Design basada en RMII, también aplica fuerza a la falange distal. Utiliza sensores de posición para medir la posición de la falange distal en relación a la palma de la mano. Del mismo modo, se colocan microcilindros neumáticos que actúan sobre los dedos índice, corazón, anular y pulgar [14].



Figura 6.24: Rutgers Master II

- MIT Hand Robot Alpha prototype II:** Otro diseño es el MIT Hand Robot Alpha Prototype II (figura 6.25) que permite una extensión de falanges para simular la acción de coger un objeto.

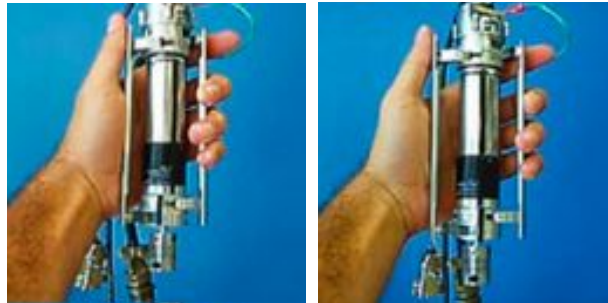


Figura 6.25: MIT Hand Robot Alpha Prototype II

- **HWARD:** El siguiente dispositivo basa su diseño en principios de reeducación motora neurobiológica tales como integración sensorial y la repetición de movimientos. HWARD ("Howard"), cuyas siglas responden a *Hand Wrist Assisting Robotic Device*, dispositivo de 3 GDL que ejercita movimientos de flexión y extensión de la mano así como movimientos de muñeca. El objetivo es reeducar a la mano mediante ejercicios de coger y soltar usando objetos reales durante la terapia para estimular la palma de la mano. Presenta actuadores neumáticos cilíndricos que pueden llegar a producir hasta 122N de fuerza, pero siempre controlado en un rango aproximado de 4-15N que son los niveles estimados para asistir los movimientos. Durante la terapia, el paciente coge y suelta variedad de objetos reales, diferentes en tamaño, textura o temperatura, mediante ayuda visual y auditiva para conseguir realizar tal movimiento siguiendo unas instrucciones, como por ejemplo poner pasta de dientes en su cepillo correspondiente. Durante este periodo, HWARD será el asistente personal del paciente, lo que provoca un ataque sensorial sobre éste con lo que se consigue una alta atención y motivación [79].
- **Hand Mentor:** Hablamos ahora del primer dispositivo comercial para la rehabilitación de manos con un sistema terapéutico llamado *Active Repetitive Motion*® producido por Columbia Scientific LLC. Se trata de un dispositivo de un solo grado de libertad que proporciona un control de fuerzas resistivo tanto a la mano como a la muñeca. La fuerza aplicada puede oponerse a la flexión o bien asistir la extensión de la mano. Incorpora sensores que monitorizan en todo momento la posición de la muñeca y de cada uno de los dedos durante los movimientos de extensión y flexión así como la medida de fuerzas aplicadas a la mano por el sistema actuador (www.handmentor.com).
- **Cybergrasp:** Originariamente fue desarrollado a través de un contrato con la marina de Estados Unidos, para aplicaciones de telerobótica. Es un dispositivo orientado a multitud de aplicaciones además de la teleoperación, como puede ser la recuperación médica o dispositivo háptico para realidad virtual.

Consiste en un exoesqueleto móvil de 15 GDL's y 350 g de peso por el que discurren unos cables que actúan como tendones y se encargan de transmitir la fuerza de

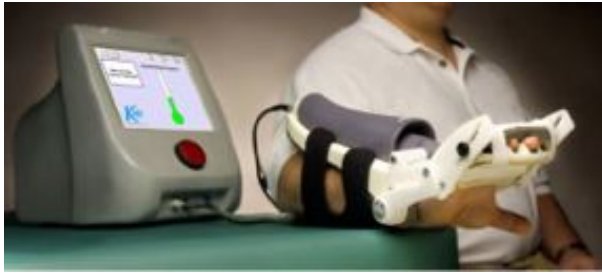


Figura 6.26: Hand Mentor

los actuadores (5 actuadores independientes) hasta unos anillos acoplados en los dedos. Los cables pueden llegar a realizar una fuerza de 12N por dedo. CyberGrasp (6.27) debe ser acoplado a otro producto para su funcionamiento, previamente a la colocación del exoesqueleto se debe acoplar el CyberGlobe, un guante para la adquisición de datos. También se pueden acoplar otros productos como GraspPack, una mochila con el paquete de actuadores y baterías que permite la movilidad del conjunto.



Figura 6.27: Cyberglove y Cybergrasp

- **Prototipo i-Phalanx:** Dentro de la acción OpenSurg se ha desarrollado un prototipo de bajo coste para la rehabilitación robótica de la mano. Los actuadores utilizados son muelles de Nitinol (un material con memoria de forma) y el control se ha realizado utilizando una tarjeta open-electronics de Arduino. La principal virtud de este prototipo es la capacidad de regular el movimiento de cada dedo de forma independiente y la seguridad que implica el uso de muelles como sistema de actuación. Otra de las características de este dispositivo es la facilidad para adaptarlo a la patología del paciente, de forma que se puede utilizar con casi todas las manos. El sistema incorpora un software de control y realidad virtual con ejercicios terapéuticos sencillos. Los detalles de la implementación de este sistema pueden ver en [67].

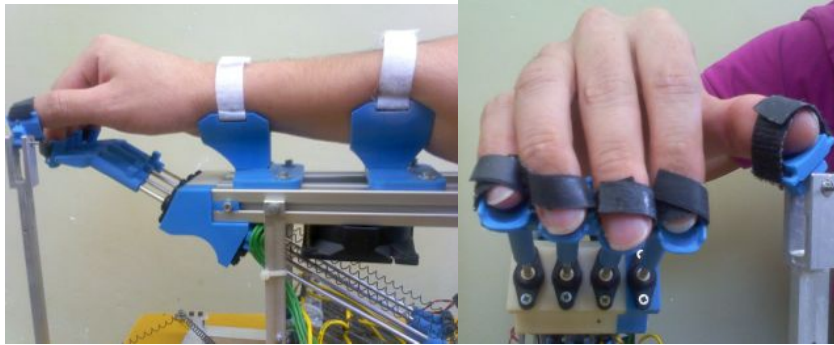


Figura 6.28: Primer prototipo de i-phalanx

- Prototipo guante sensorizado:** También dentro de OpenSurg se ha desarrollado una versión de guante sensorizado de bajo coste. El guante incorpora un sensor *flexsensor* para cada dedo que permite estimar la posición del mismo. Además, incorpora unos vibradores de pequeño tamaño en la yema de cada uno de los dedos, (2 en los casos del pulgar e índice) que permiten informar al usuario cuando se produce un contacto con un objeto virtual o cuando deslizamiento entre el objeto virtual y los dedos índice y pulgar. Se tiene la capacidad de variar la frecuencia de vibración de estos actuadores. El sistema completo se ha integrado dentro de ROS (Robot Operating System), que es un meta-sistema operativo muy extendido para la investigación con robots, de forma que su manejo es sencillo. Haciendo uso de las capacidades de ROS, se presenta la interacción del guante sensorizado con un modelo de mano robótica (con la Shadow Hand). En la figura 10.5 se observa el primer prototipo de este guante.

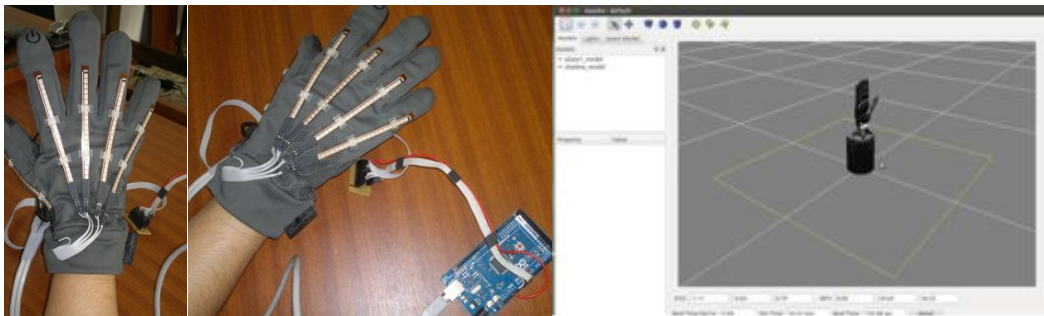


Figura 6.29: Primer prototipo de guante sensorizado háptico

6.3. La realidad virtual como ayuda a la rehabilitación física

La Realidad Virtual (RV) puede ser vista como “una avanzada forma de interfaz hombre-máquina que permite al usuario interactuar y sumergirse en un entorno generado por un computador de una manera natural [74]”. Además del computador, se pueden utilizar otros dispositivos como cascos y guantes equipados con sensores diseñados para simular una apariencia de realidad que permita al usuario tener la sensación de estar presente en ella, es precisamente la interacción en tres dimensiones lo que distingue la RV de la experiencia de simplemente observar un vídeo.

Inicialmente las aplicaciones de la RV se centraron en el terreno de los videojuegos, pero se han extendido a otros campos, como los simuladores de vuelo o la medicina. En el campo de la medicina, se han obtenido resultados alentadores, siendo dirigidas las aplicaciones a una variedad de problemas clínicos, incluyendo tratamientos psicológicos, trastornos post traumáticos, manejo del dolor, asistencia y rehabilitación de procesos cognitivos.

Realizar ejercicios de rehabilitación puede ayudar al cerebro a crear nuevas conexiones entre las células nerviosas. Sin embargo, las terapias estándar proporcionan sólo beneficios modestos en términos de mejora de la movilidad [72], así que la investigación continúa y la comprensión de esta tecnología en desarrollo será crucial para su integración efectiva.

La RV es una tecnología con una variedad de beneficios para ayudar al paciente en aspectos de rehabilitación física y cada vez se usa más esta tecnología para crear entornos de rehabilitación motrices [72]. Ofrece a la evaluación clínica opciones de rehabilitación que no están disponibles con los métodos tradicionales, así la RV hace posible la creación de tareas que pueden ser programadas, controladas y modificadas de acuerdo con las habilidades del usuario, permitiéndoles interactuar en un entorno potencialmente más motivador que el utilizado en la rehabilitación tradicional [57]. La rehabilitación virtual es capaz de proporcionar un entorno donde los individuos tienen la oportunidad de olvidarse de su situación y centrarse directamente en una tarea a desarrollar [74]. Adicionalmente, a los terapeutas les ofrece la posibilidad de medir objetivamente el comportamiento del paciente en un ambiente seguro en el cual pueden emprender tareas que pueden ser practicadas de forma reiterada, mientras se mantiene el control de las necesidades de tratamiento de forma individual y se puede aumentar gradualmente su complejidad [68].

Se han realizado diferentes trabajos donde se aplica la realidad virtual a la rehabilitación física, aunque muchos se caracterizan por utilizar un tamaño pequeño de muestras de pacientes [55]. Se han adoptado varias configuraciones de interfaces para aplicar la RV a la rehabilitación física con diferentes grados de inmersión, en general, son sistemas de entrenamiento virtual para mejorar el desempeño motor donde usualmente se tiene un profesor virtual que muestra movimientos que deben ser ejecutados correctamente en tiempo real junto con el paciente.

La RV fue aplicada por primera vez en la rehabilitación de las extremidades superiores de pacientes que sufrieron un accidente cerebrovascular [39], las pruebas se realizaron a dos pacientes que fueron evaluados en un entorno virtual diseñado para practicar diferentes movimientos (mano, codo y hombro) como si se desarrollaran en el mundo real. El experimento mostró que es posible la transferencia de habilidades desde el entrenamiento virtual al desempeño del mundo real, alcanzando incluso mejoras en la realización de tareas que no se entrenaron [39]. También, se han desarrollado varias aplicaciones de la RV para recuperar la marcha en pacientes con deterioro motor de sus extremidades después de sufrir un accidente cerebrovascular [23]. En [26], crea un entorno virtual integrado con una caminadora en una plataforma de 6 grados de libertad para recuperación de la función locomotor.

Otras soluciones incluyen RV semi-inmersiva con retroalimentación háptica o sensores electromagnéticos. En [46] se desarrolla un sistema virtual para rehabilitación de los movimientos de la mano, se utilizan dos dispositivos: un CyberGlove y un guante de fuerza RMII (Rutgers Master II), para permitirle al usuario interactuar con uno de los cuatro ejercicios de rehabilitación orientados a movimientos de alcance, velocidad o fuerza. En [57] se compara la cinemática de los movimientos realizados para alcanzar y sujetar tres objetos utilizando una inmersión completa en tres dimensiones mediante una pantalla que se acopla en la cabeza y además posee una respuesta háptica mediante un guante CyberGlove.

En los últimos años, un creciente número de terapeutas ocupacionales ha integrado tecnologías de videojuegos en programas de rehabilitación. 'Wiihabilitation', o el uso de la Wii en rehabilitación, ha tenido éxito para aumentar la motivación de los pacientes y alentar el movimiento de cuerpo completo [5].

Dentro de OpenSurg se ha desarrollado un sistema completo de rehabilitación utilizando dispositivos inerciales de bajo coste y el sistema Kinect de Microsoft. El sistema se conoce como Kineci, y combina los sistemas de medida EMG, sensores inerciales, Kinect y un software de realidad virtual para desarrollar terapias de rehabilitación. La figura 6.30 muestra una imagen de la terapia

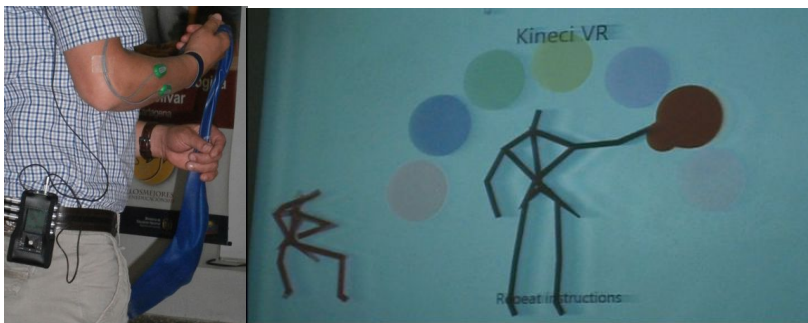


Figura 6.30: Sistema Kineci para rehabilitación con realidad virtual

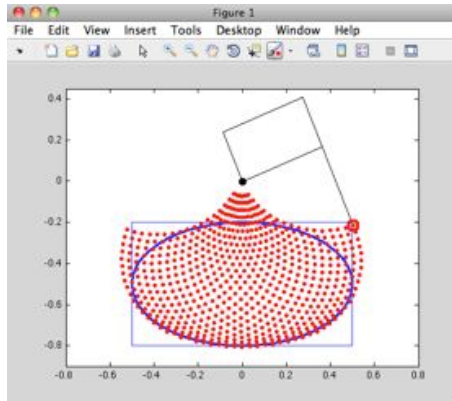


Figura 6.32: Espacio de trabajo del dispositivo PuParm

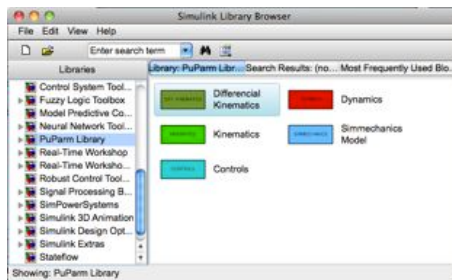


Figura 6.33: Librería de Simulink del dispositivo PuParm

6.4.2. Análisis del gait en un dispositivo de rehabilitación de miembro inferior (GBO)

En esta sección se analizará un dispositivo de rehabilitación para miembro inferior basado en el principio de compensación pasiva de fuerzas gravitacionales [82]. Este dispositivo [10], presentado en la figura 6.12, utiliza una combinación de resortes y de barras auxiliares para garantizar equilibrio estático en cualquier posición del espacio articular del dispositivo [4] (ver figura 6.13).

En este tipo de dispositivos los parámetros geométricos se calculan para hacer que el centro de gravedad coincida con un punto fijo de la estructura. Por ejemplo, en el caso del sistema presentado en la figura 6.12, las longitudes de las barras auxiliares l_2 y l_3 se calculan para que el centro de gravedad coincida siempre con el punto de intersección entre las barras 2 y 3, independientemente de las posiciones articulares θ_1 y θ_2 . Para calcular l_2 y l_3 se expresa primero la posición del centro de gravedad en función de las

masas y de las longitudes de las barras

$$\mathbf{r}_{cg} = \frac{\sum_{i=1}^4 m_i \cdot \mathbf{r}_i}{\sum_{i=1}^4 m_i} \quad (6.1)$$

m_i y \mathbf{r}_i representan respectivamente la masa y la posición del centro de gravedad de la barra i . Para calcular \mathbf{r}_i se definirán dos vectores unitarios $\bar{\mathbf{r}}_1$ y $\bar{\mathbf{r}}_2$ cuyas orientaciones están determinadas por las direcciones de las barras principales

$$\begin{aligned} \mathbf{r}_1 &= \alpha_1 l_1 \bar{\mathbf{r}}_1 \\ \mathbf{r}_2 &= (l_1 - l_3) \bar{\mathbf{r}}_1 + \alpha_2 l_2 \bar{\mathbf{r}}_2 \\ \mathbf{r}_3 &= (l_1 - l_3) \bar{\mathbf{r}}_1 + l_2 \bar{\mathbf{r}}_2 + \alpha_3 l_3 \bar{\mathbf{r}}_1 \\ \mathbf{r}_4 &= l_1 \bar{\mathbf{r}}_1 + \alpha_4 l_4 \bar{\mathbf{r}}_2 \end{aligned} \quad (6.2)$$

Las variables α_i son números entre 0 y 1. Por ejemplo, si α_i es igual 0,5, esto significa que el centro de gravedad de la barra i esta en el punto medio de esta. Al reemplazar (6.2) en (6.1) se obtiene

$$\mathbf{r}_{cg} = \frac{c_1}{m} \bar{\mathbf{r}}_1 + \frac{c_2}{m} \bar{\mathbf{r}}_2 \quad (6.3)$$

Siendo

$$\begin{aligned} m &\triangleq m_1 + m_2 + m_3 + m_4 \\ c_1 &\triangleq \frac{l_1 m_4 + m_2 (l_1 - l_3) + m_3 (l_1 - l_3 + \alpha_3 l_3) + \alpha_1 l_1 m_1}{m} \\ c_2 &\triangleq \frac{l_2 m_3 + \alpha_2 l_2 m_2 + \alpha_4 l_4 m_4}{m} \end{aligned}$$

Una vez calculado \mathbf{r}_{cg} , se calcula la posición deseada del centro de gravedad, denotada \mathbf{r}_{cg}^d . En el caso del presente dispositivo se busca que dicha posición coincida con el punto de intersección entre las barras 2 y 3 (punto C de la figura 6.13)

$$\mathbf{r}_{cg}^d = (l_1 - l_3) \bar{\mathbf{r}}_1 + l_2 \bar{\mathbf{r}}_2 \quad (6.4)$$

Al igualar las ecuaciones (6.3) y (6.4) y resolver para l_2 y l_3 se obtiene

$$\begin{aligned} l_2 &= \frac{\alpha_4 l_4 m_4}{m_1 + m_2 + m_4 - \alpha_2 m_2} \\ l_3 &= \frac{(1 - \alpha_1) l_1 m_1}{m_1 + m_4 + \alpha_3 m_3} \end{aligned} \quad (6.5)$$

Es importante recalcar que las expresiones para l_2 y l_3 se dedujeron suponiendo que las barras 1 y 3 son paralelas a las barras 2 y 4. Sin embargo, si la barra 1 se alinea con la barra 2, y la barra 3 se alinea con la 4, se obtiene una configuración singular en la cual las expresiones para l_2 y l_3 dejan de ser validas. En tal caso, el centro de gravedad no

se ubicará en la posición deseada. Una vez fijada la ubicación del centro de gravedad, se calculan las constantes de los resortes con el fin de garantizar que la energía potencial del mecanismo sea independiente de las variables articulares θ_1 , θ_2 . Dicha energía depende de las elongaciones de los resortes 1 y 2 y de la altura del centro de gravedad del mecanismo. Estas tres variables se denotaran respectivamente x_1 , x_2 y h

$$V = \frac{1}{2} k_1 x_1^2 + \frac{1}{2} k_2 x_2^2 + m g h \quad (6.6)$$

A partir de relaciones trigonométricas se obtiene

$$\begin{aligned} x_1^2 &= [\cos \theta_1 (l_1 - l_3 + l_2 \cos \theta_2) - l_2 \sin \theta_1 \sin \theta_2]^2 + \\ &\quad [\sin \theta_1 (l_1 - l_3 + l_2 \cos \theta_2) - d_1 + l_2 \cos \theta_1 \sin \theta_2]^2 \\ x_2^2 &= d_2^2 - 2 d_2 l_2 \cos \theta_2 + l_2^2 \\ h &= (l_1 - l_3) \sin \theta_1 + l_2 \sin (\theta_1 + \theta_2) \end{aligned}$$

Los parámetros d_1 y d_2 determinan la posición en la cual los resortes se conectan a la estructura mecánica (ver *Figura 6.13*). Al reemplazar los valores de x_1 , x_2 y h en la ecuación (6.6) se obtiene

$$V = \beta_0 + \beta_1 \cos \theta_1 + \beta_2 \cos \theta_2 + \beta_3 \cos (\theta_1 + \theta_2)$$

Siendo

$$\begin{aligned} \beta_0 &= \frac{1}{2} k_1 [d_1^2 + (l_1 - l_3)^2 + l_2^2] + \frac{1}{2} k_2 [d_2^2 + l_2^2] - m g d_1 \\ \beta_1 &= k_1 d_1 (l_1 - l_3) - m g (l_1 - l_3) \\ \beta_2 &= k_1 (l_1 - l_3) l_2 - k_2 d_2 l_2 \\ \beta_3 &= k_1 d_1 l_2 - m g l_2 \end{aligned}$$

Si los valores de k_1 y k_2 se seleccionan para que las constantes β_1 , β_2 y β_3 sean cero, la energía potencial sera constante e igual a β_0 . Al resolver las ecuaciones $\beta_1 = 0$ y $\beta_2 = 0$ ($\beta_1 = 0$ implica $\beta_3 = 0$) para k_1 y k_2 se obtiene

$$\begin{aligned} k_1 &= \frac{m g}{d_1} \\ k_2 &= \frac{m g (l_1 - l_3)}{d_1 d_2} \end{aligned} \quad (6.7)$$

Al variar d_1 y d_2 se puede variar el nivel de compensación gravitacional del dispositivo. El sistema de rehabilitación presentado en esta sección fue diseñado y construido en la universidad de *Delaware*.

6.4.2.1. Simulación de un sistema de rehabilitación para miembro inferior

Se va a realizar a continuación una simulación del sistema GBO modelado. Primero se realizará la simulación suponiendo que las longitudes de las barras auxiliares son diferentes de los valores dados por la ecuación (6.5). En tal caso la posición del centro de

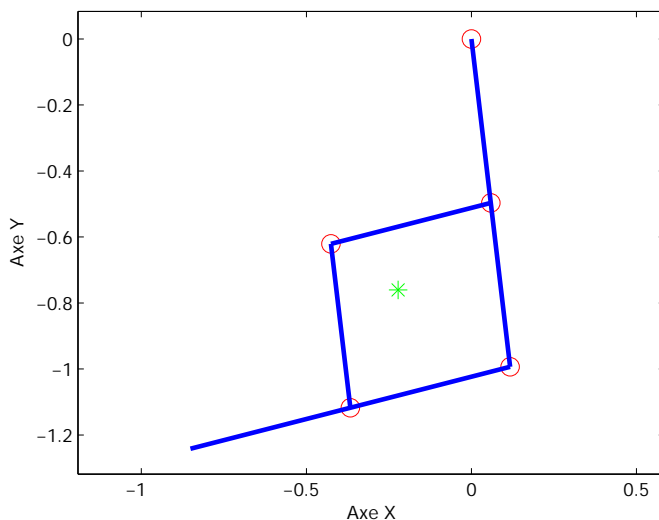


Figura 6.34: Evolución del centro de gravedad del mecanismo

gravedad dependerá de la posiciones articulares θ_1 y θ_2 . Para efectuar esta simulación: (i) cargue Matlab, (ii) abra el archivo `main_simulation.m`, (iii) ponga en comentario las líneas de código donde se calculan los valores de l_2 y l_3 , (iv) y asigne un valor de 0,5 a las variables l_2 y l_3 . Con estas modificaciones ejecute el programa `main_simulation.m` y observe la animación donde se muestra el comportamiento del mecanismo. La estrella de color verde indica la posición del centro de gravedad (ver figura 6.34)

Una vez realizada la etapa anterior, lleve el programa a su forma original y relance la simulación. En este caso debe observarse que para las configuraciones no singulares (las barras 1 y 2 tienen direcciones diferentes), el centro de gravedad coincide con el punto de intersección entre las barras 2 y 3. Cuando la configuración es singular, el centro de gravedad se ubica al exterior de la estructura. En la figura 6.35 se muestra la posición del centro de gravedad para una configuración articular singular.

Para evitar que el mecanismo llegue a una configuración singular, se impedirán las flexiones hacia adelante de la articulación de la rodilla (tal y como sucede en la realidad). Para ello se generará un *torque* de oposición en la articulación 2 cuando el valor de θ_2 sea mayor que cero. Dicho *torque* esta dado por la siguiente expresión :

$$\tau_2 = \begin{cases} 0, & \theta_2 \leq \epsilon \\ -k_p \theta_2 - k_v \dot{\theta}_2, & \theta_2 > \epsilon \end{cases} \quad (6.8)$$

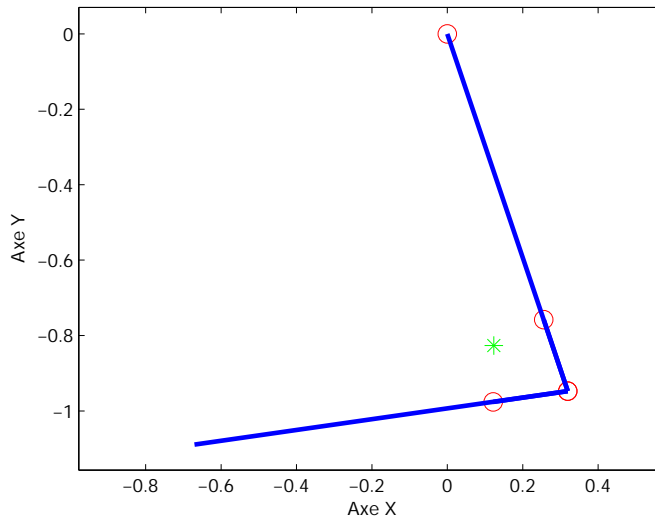


Figura 6.35: Posición del centro de gravedad para una configuración singular

Para implementar esta modificación (i) abra el archivo *Simulink* `dynamic_model` (figura 6.36) (ii) elimine el bloque `Torque Joint 2` (iii) implemente la ecuación (6.8) en un subsistema *Simulink* cuyas entradas deberán ser θ_2 y $\dot{\theta}_2$ y cuya salida será τ_2 (iv) conecte la salida del subsistema diseñado al punto donde estaba conectada la salida del bloque `Torque Joint 2`. Relance la simulación y observe la animación resultante.

La parte final de este ejercicio consiste en verificar que los valores de las constantes de los resortes, calculadas en la ecuación (6.7), permiten compensar las fuerzas gravitacionales. Para ello abra el archivo `main_simulation.m` y elimine los ceros que multiplican las expresiones para k_1 y k_2 (líneas 37 y 38). Lance la simulación para diferentes condi-

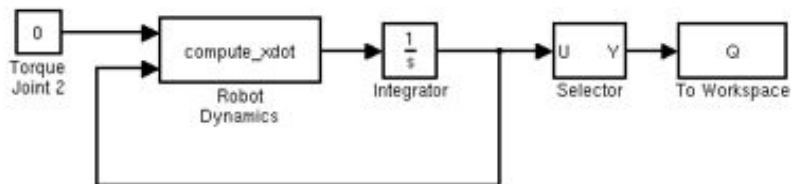


Figura 6.36: Implementación en *Simulink* de la ecuación (6.8)

ciones iniciales y constate que en todos los caso la configuración articular del mecanismo permanece invariante.

6.5. Resumen del capítulo

1. Una posible clasificación de los robots de rehabilitación los diferencia entre a los robots basados en efector final y los robots de tipo exosqueleto.
2. Se ha presentado una amplia lista de robots de rehabilitación, tanto a nivel comercial como prototipos clasificada en:
 - a) robots para rehabilitación de miembro superior
 - b) robots para rehabilitación de miembro inferior
 - c) robots para rehabilitación de extremidades distales
3. El uso de sistemas inmersivos de realidad virtual ayuda a la terapia de rehabilitación, gracias a la motivación que conlleva para el paciente.

Bibliografía para ampliar

Winter school on rehabilitation robots, 2008 . [enlace a la documentación de la escuela](#) Este curso, celebrado en Elche en 2008 contó con ponencias y cursillos de los más prestigiosos investigadores en el área de robótica de rehabilitación y asistiva.

F. Badesa, J. Sabater, L. López, N. Garcia, R. Morales, L. Rodriguez, *Optimización del diseño de un sistema robótico dual para rehabilitación de miembros superiores mediante método Kabat*, IV Jornadas Iberoamericanas de Tecnologías de Apoyo a la Discapacidad, AITADIS 2009.

Portal de fisioterapia y salud, *Fisaude*, Recuperado en Septiembre 5, 2011 de www.fisaude.com/fisioterapia/tecnicas/tecnica-de-kabat.html. Se describe el método Kabat en fisioterapia.

Luis Bernal, *Fisioterapia Propioceptiva*, Portal consultado en Septiembre 5, 2011 en <http://www.luisbernal.com/fisio/>. Documento en pdf que describe los fundamentos y técnicas de la fisioterapia propioceptiva.

Arcas Patricio, Miguel Ángel y otros. *Manual de Fisioterapia*. Módulo I. Generalidades. Mad, 2004.

E Viel. *El método Kabat*. Masson Barcelona, 1989.

Sandstron M; y otros Ahonen J, Lahtinen T. *Kinesiología y Anatomía aplicada a la actividad física*. Paidotribo, 2001.

Elena Agustino Blázquez. *Proprioceptive Neuromuscular Facilitation Approach*. L.W. Pedretti & M.B. Early, 2001.

Capítulo 7

Robótica asistencial. Fundamentos y aplicaciones

“No basta adquirir la sabiduría, es preciso usarla”
Marco Tulio Cicerón

Indice

7.1. Introducción a la robótica asistencial	202
7.2. Clasificación de robots asistenciales para personas con movilidad reducida	202
7.2.1. Sistemas fijos dedicados	202
7.2.2. Manipuladores montados sobre sillas de ruedas	204
7.2.3. Sistemas móviles autónomos	205
7.2.4. Sillas de rueda con guiado autónomo	205
7.2.5. Robots asistenciales para la navegación	206
7.3. Prótesis y órtesis mecatrónicas	207
7.3.1. Introducción	207
7.3.2. Justificación de la robotización	207
7.3.3. Dificultades y desafíos	208
7.3.4. Clasificación de las diferentes actividades	209
7.4. Dispositivos interfaz hombre-máquina para comunicación de pacientes discapacitados	215
7.4.1. Interfases Cerebro Computadora (Brain Computer Interface - BCI)	216
7.4.2. Interfases basadas en la interacción mioeléctrica	217
7.4.3. Interfases basadas en señales de electrooculografía	218

7.5. ROBOTICA MEDICA APLICADA	219
7.5.1. Evaluación de los modelos para una prótesis de mano	219
7.5.2. Diseño de una prótesis de mano	223
Bibliografía para ampliar	229

7.1. Introducción a la robótica asistencial

Un robot asistente es un dispositivo tecnológico adaptable que interactúa directamente con el usuario para mantener y/o aumentar las capacidades funcionales de las personas, a través de sensores, actuadores y procesamiento ([1], [36]). La concepción de los dispositivos de ayuda surge de la necesidad de cubrir tareas habituales, las cuales son impedidas por deficiencias neuropsicológicas y/o motoras. Actualmente, la robótica asistencial promete ser el medio más adecuado para que personas discapacitadas logren maximizar su autonomía funcional, particularmente en áreas asociadas con actividades rutinarias.

En general, los dispositivos robóticos de asistencia pueden ser divididos en dos categorías, robots autónomos y robots cuyas funcionalidades son compartidas con el usuario. Los robots autónomos reemplazan prácticamente la totalidad de las funciones rutinarias psicomotrices dependientes. Los procesos de diseño relacionados a la sensorica, funciones de procesamiento/control e implementación de los efectores no difieren sustancialmente de las otras aplicaciones robóticas. Sin embargo, los controladores de lazo cerrado implementados en la robótica asistencial conllevan múltiples consideraciones acerca de la interacción hombre-robot.

7.2. Clasificación de robots asistenciales para personas con movilidad reducida

7.2.1. Sistemas fijos dedicados

Se pueden distinguir dos tipos, sistemas de escritorio que conforman un puesto de actividad/trabajo para el paciente, y sistemas que aún siendo fácilmente transportables son ubicados frente al usuario por una tercera persona. Los sistemas de tipo escritorio asisten al usuario en un espacio de trabajo muy estructurado, permitiendo así la realización de tareas muy específicas. Las estaciones de trabajo robóticas más destacadas en los sistemas fijos son los RAID (Robot to Assist the Integration of the Disabled) y DeVAR (Desktop Vocational Assistant Robot). El sistema DeVAR (figura 7.1(a)) fue presentado dentro del programa de robótica de rehabilitación iniciado por el profesor Larry Leifer en 1978 [20]. Desde entonces fueron evaluados exitosamente en muchas personas con dificultades motoras ocasionadas por lesiones en la médula espinal [22] [80] y [21]. El sistema robotizado RAID [17], es una estación de trabajo, basada en un ordenador y que cuenta con el soporte de un robot para acercar objetos al usuario o realizar tareas de colaboración tales como archivar documentos o pasar hojas de un libro. El sistema RAID se desplaza sobre guías verticales y horizontales para alcanzar todos los puntos de interés.

Una visión general de la evaluación y desarrollo de las estaciones de trabajo RAID se presenta en el trabajo de Jones [48] (figura 7.1(b)), mientras que la pruebas realizadas por diferentes usuarios se describen por Danielsson y Holmberg ([18] y [19]). Otras estaciones

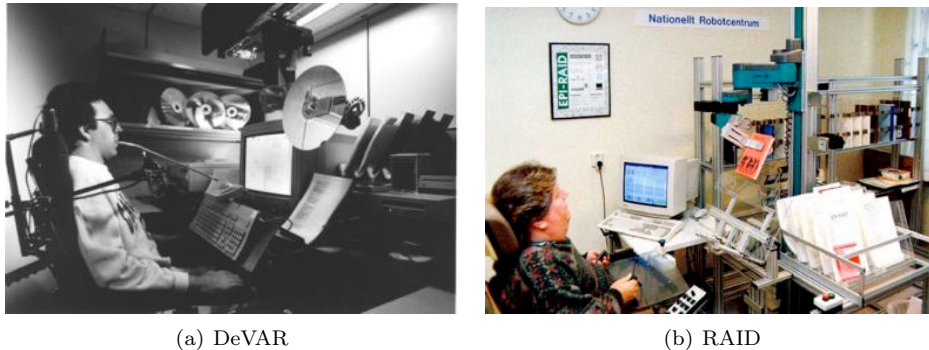


Figura 7.1: (a) Un estudiante con cuádruplejía de alto nivel comanda un sistema DeVAR para insertar un videojuego, (b) Sistema RAID desarrollado en Lund University Sweden

de trabajo más recientes son la cocina robotizada CAPDI, la cual permite la selección de tales utensilios ubicados en posiciones conocidas [16], y el manipulador flexible TOU. La implementación de sensores dentro del sistema de lazo cerrado de un sistema RAID y/o DeVAR proporciona al manipulador un gran nivel de independencia en la manipulación de los objetos [49]. Una de las principales desventajas de estas estaciones robóticas es la necesidad de la figura del asistente en la preparación de la puesta en funcionamiento del robot (por ejemplo, el abastecimiento de comida y/o bebida en los contenedores/bandejas del robot). La tendencia en estos sistemas, no es evitar la necesidad de disponer de una persona que asista al discapacitado, sino que una vez servida la comida y bebida, el usuario pueda comer y beber a su ritmo, autónomamente. Entre estos últimos se destacan el POWER FEEDER (US), el Neater Eater (UK), y MYSPOON (Japón) (figura 7.2.1). Esta solución permite, por una parte satisfacer al usuario al poder comer por sí solo, y por otra evitar la fatiga física/mental que supone la asistencia continuada entre el asistente y el discapacitado.

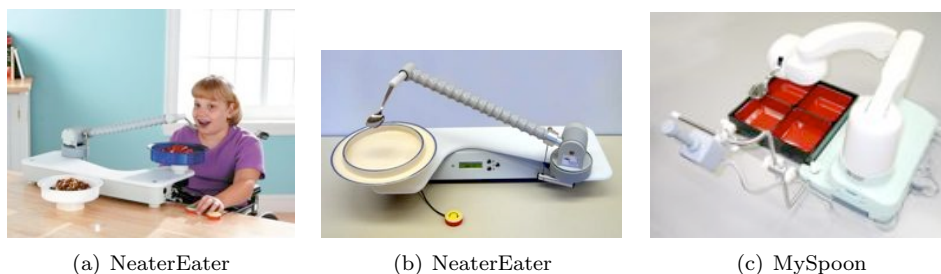


Figura 7.2: (a) y (b) Asistente Neater Eater, (c) Asistente Myspoon

7.2.2. Manipuladores montados sobre sillas de ruedas

Los sistemas robóticos montados en sillas de ruedas proporcionan al usuario autonomía total o parcial en la manipulación de objetos ubicados en su entorno, y la posibilidad del desplazamiento libre. El robot más conocido es el MANUS ([78]) el cual se comercializa desde 1999 por una empresa holandesa ([Exact Dynamics](#)). Este robot fue diseñado con el fin de manipular objetos dentro del entorno de trabajo del usuario, tal manipulación es controlada a través de un joystick. Normalmente, un pequeño teclado para la selección de opciones es también usado para el guiado de la silla (figura 7.3(a)). Este dispositivo de asistencia ha sido probado exitosamente en entornos laborales y domésticos. En el Robotics Research Laboratory del Korea Advanced Institute of Science and Technology (KAIST) se desarrollaron sistemas robóticos montados sobre estructuras móviles (similares a las sillas de ruedas) cuyo objetivo radica en la asistencia a personas discapacitadas en fábricas industriales [40]. Este grupo de investigación trabajó básicamente en tres prototipos robots TYPE I y TYPE II y el robot TYPE III. El RAPTOR, desarrollado por la Rehabilitation Technologies Division of Applied



(a) Manus



(b) Raptor

Figura 7.3: (a) silla de ruedas con el brazo robótico MANUS en una acción real, (b) Robot RAPTOR

Resources Corporation, fue el primer robot en ser aprobado por la agencia americana de salud (FDA). Este robot es menos complejo que el MANUS, con tan solo 4 grados de libertad (mas la garra) y con una interfase de usuario más simple (figura 7.3(b)). Esta fue la principal razón por al cual el RAPTOR ingresó al mercado americano con costo bastante bajo. Los brazos manipuladores montados sobre sillas han sido implementados exitosamente en sujetos con movilidad reducida de hasta aproximadamente el 80%. Los brazos robóticos han restituido efectivamente la capacidad de manipulación. Sin embargo tales sistemas presentan diseños constructivos poco flexibles (por ejemplo, se manejan exclusivamente con solo una mano, y el brazo robótico puede estorbar cuando no se está usando). Otros inconvenientes son la necesidad de una gran concentración en su manejo, lo cual puede llegar a fatigar y demorar bastante la realización de tareas complejas. El tamaño y peso del brazo robótico montado sobre la silla de ruedas pueden ser negativos para la movilidad y la conducción precisa por la casa. Los movimientos de agarre e inserción requieren precisión y pueden llegar a ser especialmente difíciles, puesto que es necesario un control tanto de la posición como de la orientación del brazo.

7.2.3. Sistemas móviles autónomos

Los sistemas móviles, o plataformas autónomas, transportan un manipulador robótico que permite alcanzar objetos fuera del limitado alcance de una persona discapacitada. En estos sistemas, la navegación de vehículos autónomos es el principal problema (ver proyectos WALKY (Neveryd y Bolmsj, [63]), CARE-O-BOT 2 (Graf et al, [31]), ARPH (Hoppenot y Colle, [41]) y HERMES (Bischoff y Graefe, [12])).

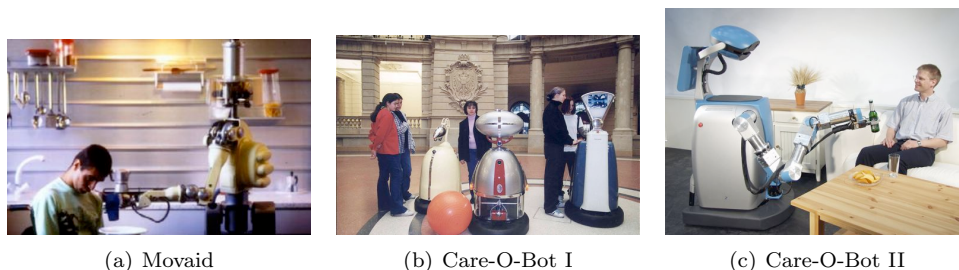


Figura 7.4: (a) Robot móvil MOVAID, (b) Tres diferentes estructuras del CARE-O-BOT I en el Museum für Kommunikation de Berlin, (c) Care-O-Bot II ofreciendo una bebida

MOVAID (Guglielmelli et al, [32]) fue un proyecto Europeo TIDE dirigido por el ARTS Lab Della Scuola Superiore Santa Anna (1994-1997). En este proyecto se desarrolló un sistema compuesto de estaciones fijas ubicados en diferentes puntos de una casa, y un robot móvil (figura 7.4(a)) capaz de navegar evitando obstáculos y manipular objetos mediante un brazo robótico. Este sistema emplea comandos de alto nivel para el control de la base móvil, mientras que su sistema de navegación autónoma está basado en odometría, ultrasonidos y láser. Fue diseñado para realizar tareas asistenciales del hogar, como calentar la comida en el microondas y servirla en la cama, limpiar la cocina y quitar sábanas entre otras.

El primer prototipo del CARE-O-BOT fue construido en 1998, por Fraunhofer IPA, Stuttgart, con capacidades funcionales de comunicación con el usuario, asistencia en la marcha humana y navegación autónoma y segura en ambientes interiores del hogar. Tres robots basados en la misma plataforma y con similares software de control, fueron instalados en el Museum für Kommunikation Berlin, para la constante operación, comunicación e interacción con los visitantes (marzo del año 2000) (figura 7.4(b)). La siguiente generación de CARE-O-BOT fue construida en el año 2001 y presentada por primera vez en HANNOVER MESSE 2002 y en Automation Fair, Stuttgart. CARE-O-BOT II esta equipado con un brazo robótico manipulador de 6 grados de libertad, apoyos ajustables para colaborar en la marcha humana al usuario, sensores de inclinación en la cabeza y un panel de control manual (figura 7.4(c)). En la actualidad el CARE-O-BOT III se encuentra en etapa de investigación (<http://www.care-o-bot-research.org/news>).

7.2.4. Sillas de rueda con guiado autónomo

La silla de ruedas es uno de los dispositivos de asistencia más comunes debido a la gran cantidad de personas con paraplejía y otros problemas de movilidad. La robótica

y las técnicas electromecánicas dotaron a la silla de ruedas tradicional de funcionalidades adicionales, tales como asistir en la navegación evitando obstáculos a través de sensores apropiados y facilitar su conducción con interfaces de usuario adaptados. Son muchos los desarrollos en el área de silla de ruedas inteligentes. Algunos ejemplos son, en Norteamérica, la NAVCHAIR de la Universidad de Michigan (Simpson et al, [77]), la WHEELSLEY del MIT (Yanco, [86]), y en Japón, la TAO de la empresa Applied Artificial Intelligence, Inc. En Europa, destacan los proyectos, SIAMO en España, el griego SENARIO, la silla ROBCHAIR en Portugal y los grupos de investigación alemanes OMNI en la Fernuniversität Hagen e INRO de la University of Applied Sciences Ravensburg-Weingarten.

7.2.5. Robots asistenciales para la navegación

Estos sistemas colaboran a la marcha humana proveyendo al usuario de un dispositivo de apoyo, como si se tratase de un andador. NURSEBOT (Roy et al, [69]) es un sistema robótico, desarrollado por la Universidad de Pittsburgh, Michigan y la Carnegie Mellon University, que permite asistir a personas de avanzada edad y/o con enfermedades crónicas, en sus vidas cotidianas. Fue diseñado para interactuar en el interior del hogar con el usuario realizando tareas tales como, recordatorio de dosis medicinales, video conferencia con los médicos, recolección de datos del usuario, asistencia en la manipulación de objetos en caso de usuarios con artritis y proporcionar algún tipo de interacción social. En general, estos sistemas se basan en plataformas altamente sensorizadas, capaces de navegar autónomamente en entornos interiores. Utilizan algoritmos probabilísticos y, tras un proceso de inicialización, son capaces de aprender y reconocer el entorno en forma inteligente. Su interfaz hombre-maquina esta basado en información visual y acústica, también añade un módulo básico de interacción social. Como alternativa, pueden estar dotados de sistemas de tele-presencia y permiten la conexión del usuario a Internet desde el propio robot. Mediante técnicas de visión artificial, es capaz de localizar y seguir la cara del usuario, lo cual le permite el monitoreo continuo.

Actualmente, el intercambio social entre el usuario y el robot (por ejemplo, mantener una conversación), es una de las principales tendencias en el desarrollo de estos sistemas (Pollack et al, [66]). Las investigaciones apuntan a crear un agente social artificial que transmita los mecanismos con los cuales las personas se comunican en su vida diaria (sensaciones, rasgos, expresiones, etc.). Para ello, se estudian los objetivos que deben cumplir este tipo de aplicaciones: percepción (interpretación de actividades y comportamientos), interacción natural (la comunicación debe ser natural), comportamiento social (el robot debe proporcionar realimentación sobre su estado, permitir a las personas interactuar con él de una manera sencilla y transmitir emociones a través de su expresividad facial, gestual o de vocalización) (Fong et al, [24]).

7.3. Prótesis y órtesis mecatrónicas

7.3.1. Introducción

La palabra prótesis proviene del griego: prós (πρός) 'por añadidura', 'hacia' thé-sis (θέσις) 'disposición'. El diccionario de la Real Academia Española adopta como definición: Procedimiento mediante el cual se repara artificialmente la falta de un órgano o parte de él; ó como el aparato o dispositivo destinado a esta reparación. En términos generales una prótesis (dentro del ámbito médico), es una extensión artificial que reemplaza una parte faltante del cuerpo. Considerando las definiciones citadas en el párrafo anterior, podemos definir a una prótesis robótica, como un elemento artificial dotado de cierta autonomía e inteligencia capaz de realizar una función de una parte faltante del cuerpo. Dicha autonomía e inteligencia se logra al integrar sensores, procesadores, actuadores, y complejos algoritmos de control. De acuerdo a esta definición, las prótesis de uso cosmético quedan completamente excluidas, como por ejemplo los ojos de vidrios, las piernas de madera, etc.

El uso de prótesis data de tiempos inmemorables. Existen ejemplos como la mano de acero utilizada por Götz von Berlichingen (1504 d.C.), o la pierna de Bronce de Capua (300 a.C.), o lo que hasta la fecha es la prótesis hallada más antigua *el dedo gordo del Cairo*, perteneciente a una momia egipcia (entre 1069 a.C. y 664 a.C.) (figura 7.5(a)). En [64] se puede encontrar de manera mas detallada la evolución e historia de las prótesis. Si bien, el uso de prótesis es consecuencia de accidentes, cáncer, malformaciones, etc., el factor determinante que ha impulsado el desarrollo de las mismas fue la necesidad de mejorar la calidad de vida de los sobrevivientes de Guerras. Sólo durante la Guerra Civil Norteamericana (1861 - 1865) se practicaron 30,000 amputaciones, durante la Segunda Guerra Mundial (1939 -1945) 14,782, y durante el conflicto con Irak se han contabilizado 1,214 amputados hasta agosto del 2008. Sin embargo, en este último conflicto, la relación de amputados con la cantidad de muertes ocurridas en combate, duplica al ocurrido al resto de todas las guerras y conflictos.

El desarrollo de prótesis, involucra la necesidad de fusionar conocimientos de la fisiología y biomecánica humana, mecanizado de materiales y prototipado de mecanismos e interfaz hombre-máquina. En esta sección se hace una revisión de las prótesis actuales y de los últimos desarrollos de prótesis robóticas. En la siguiente sección se presentan algunos argumentos que justifican el desarrollo de prótesis robóticas, y los principales retos que existen para crear prótesis fiables.

7.3.2. Justificación de la robotización

Las prótesis de brazos datan desde 1912, estas prótesis tenían la forma de gancho que podía ser cerrado o abierto encogiendo los hombros y mediante una cuerda que pasaba por la espalda. Después de la segunda guerra mundial, se creó la mano mioeléctrica, sin embargo esto no cambió el hecho de que las prótesis eran difíciles de manejar. Actualmente, las prótesis con forma de gancho han sido mejoradas en apariencia a través de moldes de mano hechos de plástico y con una apariencia casi realista. Sin embargo, estas prótesis no mejoran la capacidad del usuario para controlarlas y realizar tareas mas sofisticadas.

Las prótesis de brazo tienen como mucho tres grados de libertad: se puede abrir y



(a) Dedo gordo del Cairo

(b) Pierna de Capua

(c) Mano de acero

Figura 7.5: Primeras prótesis encontradas

cerrar el gancho, se puede extender y retraer el codo, y con los modelos más sofisticados se puede rotar la muñeca. Aun así, esos movimientos sencillos requieren de entrenamiento, concentración, y esfuerzo, lo cual resulta en un movimiento que no es fluido ni preciso. Esto contrasta con un brazo humano que tiene más de 25 grados de libertad y por lo tanto una mayor destreza, además de la habilidad de determinar si algo está frío o caliente.

Por otro lado, en las prótesis de pierna existen aun algunos problemas, por ejemplo, los amputados que utilizan prótesis mecánicas requieren entre 10-60% más energía metabólica que las personas tienen el miembro real, dependiendo de la velocidad de caminado, el estado físico de la persona, la causa de la amputación, el nivel de amputación, y las características de la prótesis. Además, los amputados caminan entre 10-40% más lento que las personas no amputadas.

Tales problemas clínicos son en parte generados por las prótesis actuales. Las prótesis comerciales actuales comprimen estructuras de resortes que almacenan y liberan energía durante cada periodo de estancia. Debido a su naturaleza pasiva, estas prótesis no pueden generar más energía mecánica que la que es almacenada durante cada paso. Al contrario, el tobillo humano genera trabajo neto positivo y tiene mayor potencia pico durante el periodo de estancia. Por otra parte, en los últimos años se han desarrollado con gran éxito diversas prótesis visuales que han permitido a personas ciegas percibir objetos y determinar su posición. Estas prótesis están aún lejos de devolver la vista a las personas ciegas, sin embargo tienen un enorme potencial, y se irán haciendo más útiles a medida que mejore la tecnología.

7.3.3. Dificultades y desafíos

Hacer una prótesis robótica de una calidad aceptable requiere de un enorme esfuerzo, no solo en el campo de la mecatrónica si no también en neurociencia, ingeniería eléctrica, ciencias cognitivas, procesamiento de señales, diseño de baterías, nano-tecnología y ciencias del comportamiento. Para obtener una prótesis que emule en buena forma la dinámica del miembro amputado es necesario que el diseño satisfaga ciertas especificaciones, como son:

- **Tamaño y masa:** Las dimensiones de la prótesis deben ser las mismas que las del miembro que sustituyen. Por otro lado, la masa debe ser igual o menor a la del miembro amputado para que el portador pueda manipularla con facilidad y no haga esfuerzos extraordinarios que puedan dañar los músculos que soportan la prótesis.
- **Velocidad y torque.** La prótesis debe capturar completamente el comportamiento torque-velocidad del miembro que sustituye.
- **Baterías.** La duración de las baterías de una prótesis robótica debe permitir un funcionamiento de al menos 16 hrs para que el usuario no tenga problemas de insuficiencia de energía durante las actividades diarias.
- **Ancho de Banda del Torque.** El ancho de banda de una prótesis es la frecuencia a la que se debe actualizar el torque aplicado en el mecanismo de accionamiento de tal manera que el caminado sea natural.
- **Realimentación al usuario.** La prótesis debe realimentar al usuario que la prótesis ha tenido contacto con el ambiente y también la intensidad del contacto. Esto podría ser solucionado mediante interfaces hápticas o utilizando señales eléctricas.

Una de las principales limitaciones por la cual el desarrollo de prótesis robóticas comerciales no se ha afianzado es el relativamente pequeño número de personas que lo necesitan. Debido a eso puede resultar que una prótesis de elevado número de grados de libertad (e.g., un brazo de 25 gdl) sea muy costosa y prácticamente incosteable para la mayoría de la gente.

7.3.4. Clasificación de las diferentes actividades

A continuación se proponen una serie de clasificaciones basadas en la función que realiza, los elementos que emplea para realizar su función y la metodología empleada para su control. Considerando la función que realiza y dada la importancia de ellas, se puede hacer una gran distinción entre las prótesis motoras del resto de las demás. Dentro de las prótesis motoras, se puede hablar de prótesis de miembros superiores (hombros, brazos, manos) y prótesis de miembros inferiores (cadera, piernas, pies).

Desde el punto de vista de acción, se pueden clasificar como prótesis Pasivas o Activas. Es decir, si poseen un elemento que requiere de energía adicional para realizar su acción o no, como por ejemplo un motor. Por contrapartida, una prótesis pasiva no posee elementos activos. Por lo general, las prótesis pasivas de locomoción están basadas en el empleo de resortes. Respecto al modo de control, se puede considerar aquellas prótesis que son accionadas mediante el uso de interruptores o comandos preprogramados, o aquellas que responden a la voluntad humana empleando alguna señal biológica (electromiografía, electroencefalografía, entre otras). También se pueden clasificar en aquellas que tienen la capacidad de proveer al usuario realimentación de sensaciones o no, como el tacto, la sensación de frío o calor, etc.

En las siguientes subsecciones se revisan las prótesis comerciales actuales y algunas prótesis que están en fase de investigación. Las compañías más representativas en el desarrollo de prótesis son [Otto Bock](#) y [Ossur](#).

7.3.4.1. Prótesis no robotizadas

Este tipo de prótesis son las más comerciales y de uso común. La actuación proviene de fuerzas ejercidas por el usuario ya sea por el movimiento relativo de algunas partes del cuerpo o por la inercia del mismo.

Gancho (OttoBock). El Gancho se utiliza como herramienta de agarre y es actuado directamente por el usuario mediante un sistema de cables. Los ganchos son útiles en aquellas tareas que demanda una gran precisión para agarrar pequeños objetos como lo son los tornillos o clavos. Los ganchos se abren mediante un arnés que se pone en la parte superior del cuerpo. Se cierran independientemente mediante un resorte o componentes elásticos (figura 7.6(a)).

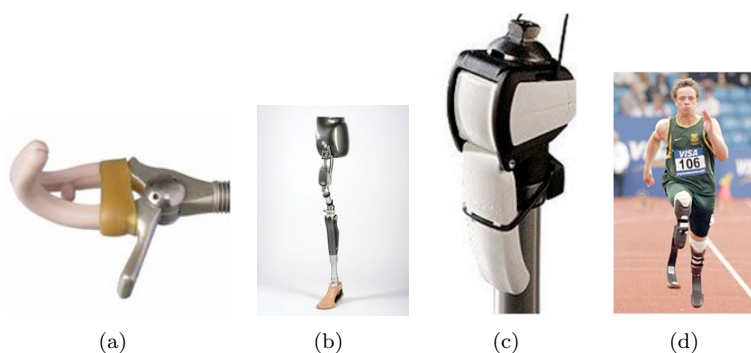


Figura 7.6: Prótesis no articuladas Otto Bock: (a) Gancho, (b) Sistema de articulación de cadera HELIX3D, (c) Articulación modular de rodilla, (d) Prótesis de pie y tobillo.

Sistema de articulación de cadera HELIX3D (Otto Bock). HELIX3D es utilizado por personas con desarticulación de cadera y por personas con hemipelvectomía (figura 7.6(b)). Tiene un control de resorte y sistema hidráulico de la fase de apoyo e impulsión. En general, consigue un movimiento tridimensional de la cadera para compensar la rotación pélvica y favorece un aspecto natural y simétrico de la marcha. Por otro lado posibilita un acortamiento de la pierna en la fase de impulsión con el objetivo de reducir el riesgo de caídas, aumentando con ello la seguridad funcional. Es ligera para no ejercer una carga y trata de lograr una posición adecuada y reducir al mínimo la posición inclinada de la pelvis.

El control de resorte y sistema hidráulico sirve de ayuda al comienzo de la fase de impulsión del usuario de la prótesis mediante resortes integrados de tracción. La energía acumulada en la fase de apoyo se emplea para compensar la ausencia de musculatura de la cadera en el comienzo de la fase de impulsión y reducir el esfuerzo necesario al caminar. También controla el movimiento tridimensional durante el ciclo de la marcha y posibilita un apoyo del pie controlado y amortiguado en la fase de apoyo con una notable reducción de la hiperlordosis (así como una extensión armónica de la articulación de cadera). Es posible una flexión plantar controlada y regular sobre la prótesis bajo una carga completa y permite el ajuste individual de la longitud del paso y el control de los movimientos oscilantes en la fase de impulsión.

Articulación modular de rodilla con bloqueo (Otto Bock). Esta articulación permite bloquear la rodilla para realizar tareas donde es importante mantenerse fijo, el desbloqueo de la articulación debe ser fácil para permitir al usuario caminar después de haber realizado la tarea. Esta articulación soporta a personas con un peso de 100Kg. El sistema de bloqueo asegura de modo automático la articulación en una posición totalmente extendida, encajándose el trinquete de bloqueo de forma audible y perceptible. La articulación puede desbloquearse mediante el cable de tracción, o presionando el trinquete de bloqueo (rótula) (figura 7.6(c)). Las articulaciones EBS tienen un sistema hidráulico optimizado para la fase de impulsión.

Prótesis de pie y tobillo (Ossur). Este tipo de dispositivo consiste de tres partes. La primera es un conector hecho de polipropileno o de materiales compuestos de fibra de carbón que se ajusta sobre la parte baja de la tibia. La segunda parte es una goma de silicona que funciona como interfase entre el pie y el conector para proteger la piel. Finalmente, para remplazar el conjunto tibia y pie se utiliza un elemento curvado de fibra de vidrio de alta rigidez. Este último elemento se encarga de almacenar energía y posteriormente durante la fase de impulsión regresarla al usuario. Esta prótesis fue utilizada por Oscar Pistorius que se convirtió en el primer amputado en romper la barrera de los 22 segundos para los 200 m durante los juegos paralímpicos de 2004 (figura 7.6(d)).

7.3.4.2. Prótesis robotizadas

En esta subsección se presentan algunas prótesis robotizadas que ya son comerciales y otras que están en fase de investigación. Aquí se caracterizan como prótesis robóticas aquellas que tiene una fuente de energía propia, un actuador, y sensores que permiten leer los movimientos deseados por el usuario. Por lo tanto también se requiere un sistema de procesamiento de esas señales (aun en su forma mas básica) para poder convertir esas señales en movimientos de los actuadores. En esta definición no es necesario que el sistema provea de retroalimentación al usuario. Sin embargo en la mayoría de las prótesis la realimentación se realiza a través de la vista.

Mano Mio-eléctrica (Otto Bock). Esta mano tiene una fuerza de agarre (100N) y una velocidad (300 mm/s), se pueden agarrar objetos rápidamente y con precisión. Se puede seleccionar un total de 6 programas diferentes con ayuda del MyoSelect 757T13 y ajustarlos a la indicación del usuario como corresponda. Permiten una adaptación óptima a las necesidades y capacidades del usuario de la prótesis. La supresión del sensor de los pulgares permite agarrar de forma activa y consciente. Los objetos se fijan y se colocan mediante señales musculares, ya que el sistema electrónico de la MyoHand no reajusta automáticamente la fuerza de agarre (figura 7.7(a)). Esta prótesis se recomienda a pacientes activos con un nivel de amputación bajo. Otra ventaja esencial de la mano es que el usuario puede generar de manera activa una fuerza de agarre de hasta 100 N. Gracias a los distintos programas de control puede encontrar una selección perfectamente indicada para el paciente. La velocidad y la generación de la fuerza de agarre pueden adaptarse perfectamente a las necesidades del usuario mediante el MyoSelect 757T13.

Mano electrónica (Otto Bock). Esta mano tiene el Control Dinámico de Modo (DMC en ingles) la velocidad y la fuerza de aprehensión se regulan de forma proporcional a la fuerza de la señal muscular (figura 7.7(b)). Este control también se caracteriza por un nuevo tipo de modo de seguridad: Después de agarrar una vez con la máxima fuerza,

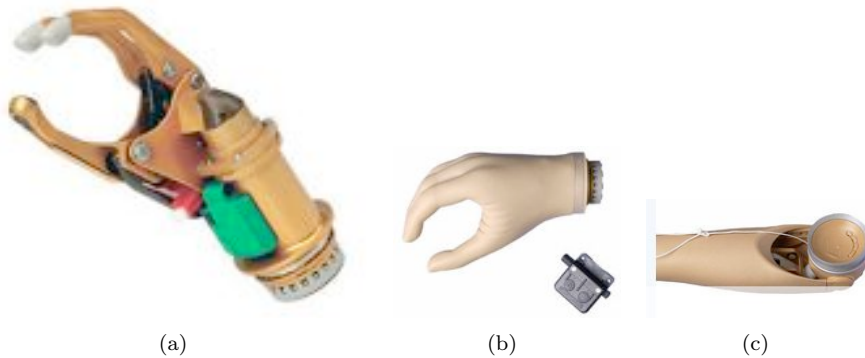


Figura 7.7: Prótesis robotizadas: (a) Prótesis Myohand, (b) Mano eléctrica, (c) Prótesis Codo-Antebrazo

se requiere una señal EMG ligeramente más alta para abrir la mano. Esto evita que la mano se abra debido a una contracción involuntaria del músculo. El control DigitalTwin combina en una sola mano ambos controles clásicos: el digital y el control por doble canal.

Codo-Antebrazo ErgoArm (Otto Bock). ErgoArm[®], ErgoArm[®] plus, ErgoArm[®] Hybrid plus y ErgoArm[®] Electronic plus son cuatro componentes de codo que facilitan el tratamiento mioeléctrico en altos niveles de amputación. Cuanto más alto es el nivel de amputación, mayores son las demandas en la técnica de prototización. El montaje tanto del ErgoArm[®] Electronic plus como del ErgoArm[®] Hybrid facilitan la prototización mioeléctrica en niveles altos de amputación (figura 7.7(c)). El 12K44 ErgoArm[®] Hybrid plus se recomienda para prótesis híbridas con una mano mioeléctrica y una articulación de codo con cable de tracción. Gracias a la conexión fácil “EasyPlug”, todos los cables eléctricos desaparecen en el interior de la prótesis para que pasen desapercibidos y estén protegidos.

Prótesis Bionica I-Limb. La prótesis *I-Limb* es una mano biónica cuyos dedos son controlados independientemente y por lo tanto permiten una gran cantidad de movimientos (figura 7.3.4.2). Esta mano es capaz de hacer agarres de precisión y de potencia de diferentes formas. La mano *I-Limb* ya ha sido implantada en pacientes de varios países.

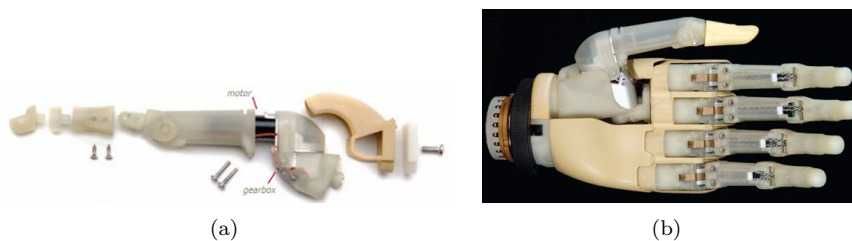


Figura 7.8: (a) Elementos de un dedo, (b) Mano i-Limb

UTAH arm 3 (U3+). El U3+, es una prótesis de brazo basada en el UTAH ARM. La misma consiste en el control del movimiento del codo, la mano y opcionalmente la muñeca (figura 7.8(b)). Provee una gran variedad de entradas de control adecuándose a las necesidades del usuario, como ser señales de EMG, sensores externos, etc. Es compatible con la tecnología i-Limb Hand, y con todos los dispositivos terminales existentes en el mercado. El control de permite realizar movimiento a diferentes velocidades. El codo es capaz de soportar el momento producido por un peso de aproximadamente de 5 kg en la mano



Figura 7.9: (a) UTAH arm, (b) Brazo de Luke

El Brazo de Luke. El brazo de Luke desarrollado por la compañía DEKA ha sido diseñado para que posea cuatro características que sea modular, ligero, ágil y contener múltiples controladores (figura 7.9(b)). El diseño modular permite configurar la prótesis para cada amputado (dependiendo del nivel de amputación). El peso del brazo es el de una mujer promedio. El brazo de Luke tiene 24 grados de libertad. Además, tiene 12 microprocesadores además contiene sensores de realimentación de fuerza para mejorar el control. El brazo se puede mover con señales nerviosas, musculares, o utilizando sensores de presión en la planta del pie. El brazo de Luke es producto de un plan estratégico para el desarrollo de prótesis robóticas iniciado por la DARPA en EU.

Power Knee (Ossur). La Power Knee está fabricada por la compañía Ossur. Es la primera prótesis de rodilla que reemplaza la función muscular perdida a través de una fuente activa de potencia (un actuador eléctrico) que permite generar la propulsión necesaria para el caminado y también en actividades como lo son levantarse de una silla de ruedas o subir las escaleras. La Power Knee contiene un conjunto de sensores, incluyendo giróscopos, células de presión, celdas de cargas, sensores angulares y el Módulo de Propriocepción Artificial en la Pierna de sonido (utiliza un sensor de sonido que permite conocer el ritmo de la pierna sana). El sistema de sensores, habilita al dispositivo para observar el estado general de la interfaz humano-prótesis. El sistema contiene un microprocesador que utiliza la información de los sensores para lograr un caminado similar al humano. La información proveída por el sistema sensorial del sonido lateral permite regenerar la verdadera cinemática del caminado mientras anticipa la función requerida cuando las condiciones de caminado cambian. El Power Knee está limitado para amputados de una sola pierna, que pesen menos de 100 kg. La longitud del suelo a punto

de amputación debe ser al menos de 49 cm y se ha de conectar muy bien al miembro amputado (figura 7.10(a)).

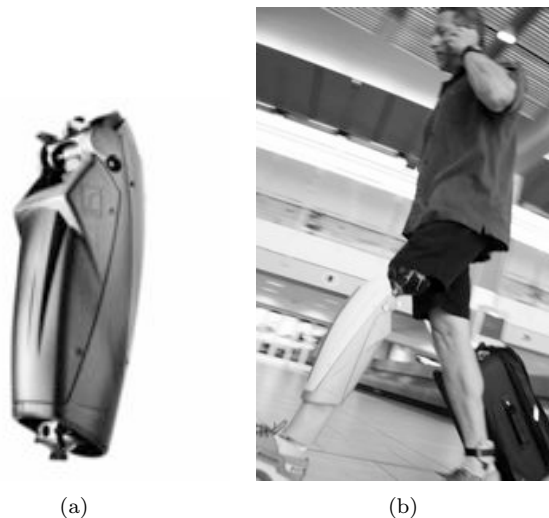


Figura 7.10: PowerKnee instalado

Propio Foot (Ossur). La Propio Foot es el primer módulo de pie inteligente del mundo (figuras 7.11(a) y 7.11(b)). Los acelerómetros miden en tiempo real el movimiento a una velocidad de 1600 ciclos por segundo. Siguiendo la ruta del tobillo a través del espacio, el sistema define las características del caminado y los eventos, incluyendo el golpe del talón y el movimiento para dejar el suelo. Para cada paso, el dispositivo construye su ruta mediante un análisis continuo del movimiento horizontal y vertical, haciendo un trazado del pie como se mueve a través del espacio. Este trazado varía de acuerdo al terreno de tal forma que los algoritmos de reconocimiento del patrón de marcha pueden detectar e identificar cuando el usuario está caminando en una superficie plana o con inclinación o si está subiendo o bajando escaleras, también si el usuario está en posiciones relajadas. Una tarjeta de control recibe un flujo constante de señales del sistema de inteligencia artificial.

El Propio Foot se desarrolló con tecnología previamente desarrollada en el MIT por Hugh Herr. En [8] presentan el diseño de una prótesis de tobillo-pie que soluciona de manera satisfactoria la mayor parte de los problemas que hay en este tipo de prótesis (figura 7.11(c)). La arquitectura básica del diseño mecánico es un resorte, configurado en paralelo con un actuador de alta potencia. La prótesis requiere de gran potencia mecánica como también de un torque de pico alto. El resorte paralelo comparte la carga con el actuador, por lo tanto la fuerza pico del actuador del sistema es disminuida notablemente. Consecuentemente, se puede utilizar un radio de transmisión más pequeño, y también se obtiene un ancho de banda de fuerza mayor. La elasticidad en serie es una característica importante para este tipo de prótesis dado que puede prevenir el daño a la transmisión debido a las cargas de choque cuando el pie choca con el suelo.

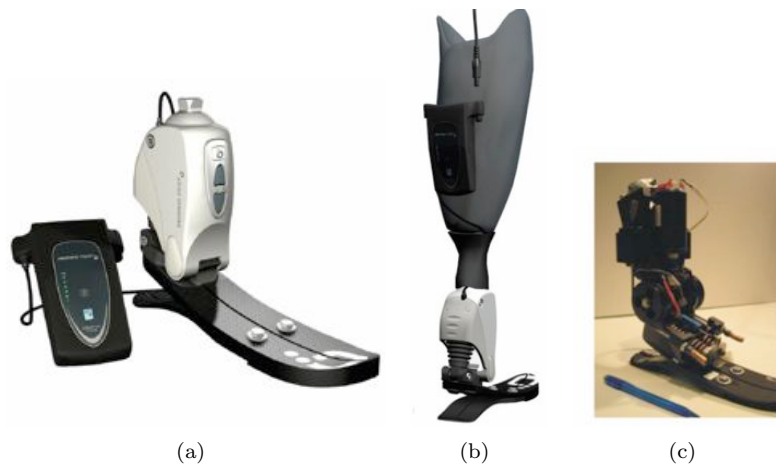


Figura 7.11: (a) Propior Foot, (b) Propior Foot instalado, (c) Prototipo de tobillo

El sistema mecánico del Propior Foot está compuesto de 5 elementos principales: un motor de d.c.; una transmisión; un resorte en serie; un resorte en paralelo unidireccional y una prótesis del pie de plástico. Los tres primeros elementos son combinados forman un sistema llamado Actuador Elástico en Serie (SEA), visto en el apartado 1.4.1. Un SEA, desarrollado previamente para robots caminantes, consiste de un motor de DC en serie con un resorte (o una estructura de resortes) a través de una transmisión mecánica. Los SEA proveen control de fuerza controlando la distancia a la cual los resortes en serie son comprimidos. Usando un potenciómetro, se puede obtener la fuerza aplicada a la carga midiendo la deflexión de los resortes. En esta aplicación se utilizó el SEA para modular la rigidez de la articulación así como también para proveer de un torque constante de offset. El SEA suministra la rigidez durante la Plantar-flexión Controlada (PC), y otra rigidez desde la PC hasta la Plantar-flexión Potenciada (PP). Debido a la demanda de torque y potencia, se incorpora un resorte al SEA, de tal manera que la carga del SEA es ampliamente reducida. Debido a este hecho, el SEA tendrá un gran ancho de banda en fuerza para suministrar el empuje durante la PP. Para evitar obstaculizar el movimiento del pie durante la fase de rotación, el resorte paralelo es implementado de manera unidireccional.

7.4. Dispositivos interfaz hombre-máquina para comunicación de pacientes discapacitados

Usualmente interactuamos con máquinas a través de una interfaz conocidas con el nombre de Interfaz Hombre Máquina. Estos dispositivos permiten la comunicación, e interacción, entre el usuario y cualquier dispositivo mecánico capaz de transmitir o transformar energía destinada a la realización, o asistencia, de tareas humanas.

En los apartados siguientes se describirán las interfases hombre máquina de origen

biónicas, es decir, interfases capaces de simular el comportamiento de los seres vivos por medio de instrumentos mecánicos. Algunos ejemplos son, prótesis activadas por los nervios y robots/exoesqueletos controlados por señales biológicas. Estos sistemas son una combinación de biología, robótica y computación, y pueden ser vistos como cualquier tecnología que usa o monitorea los aspectos biológicos del cuerpo para realizar una función. Las interfases que hacen uso de las señales biológicas estudiadas en el capítulo 4, requieren captar información útil del usuario, lo cual es realizado a través de electrodos. Las interfases hombre maquina que se describirán a continuación usan electrodos para adquirir datos eléctricos (EEG, EMG, EOG y ECG).

7.4.1. Interfases Cerebro Computadora (Brain Computer Interface - BCI)

Desde sus comienzos, el electroencefalograma (EEG) ha sido usado principalmente para estudiar enfermedades neurológicas en el ámbito clínico y la función cerebral en el ámbito de las neurociencias (Travis et al, 1975; Kuhlman, 1978; Elbert et al, 1980; Rockstroh et al, 1989; Rice et al, 1993; Serman, 2000). Sin embargo, desde no hace mucho tiempo, se ha especulado en que el EEG podría ser utilizado para interpretar pensamientos o intenciones. De este modo una persona podría comunicarse, o controlar algún dispositivo sin usar los canales normales tales como nervios periféricos y músculos. Esta idea surgió de la ficción popular, tal como en la película “Firefox” en la cual un aeroplano es controlado, en parte, por la actividad eléctrica cerebral del piloto (Thomas, 1977).

Quizás el primer antecedente fue el trabajo de Dewan, quien experimentó la utilización del ritmo α para producir caracteres en código Morse (Dewan, 1967). En 1988, Farwell y Donchin de la Universidad de Illinois, implementaron un sistema que a partir del potencial P300 permitía seleccionar letras de una pantalla (Farwell y Donchin, 1988). En 1995, Mc Millan en la Wrihth-Patterson Air Force Base, utilizó satisfactoriamente potenciales evocados visuales para controlar la inclinación de un simulador de vuelo (Mc Millan, 1995). Alrededor de 1994, los grupos de Jonathan Wolpaw y Dennis McFarland del Wadsworth Center for Laboratories and Research en New York, USA y el de Pert Pfurtscheller en la University of Technology en Graz, Austria, comenzaron a trabajar en interfases cerebro computadoras (ICC). Estos grupos históricamente se dedicaron al estudio de los ritmos cerebrales, fundamentalmente aquellos relacionados con actividades motoras y posteriormente comenzaron a aplicarlos en la implementación de ICC (Wolpaw, 1994; Kalcher et al, 1996). Estas ICC permiten el movimiento de un cursor sobre la pantalla de un PC en una dimensión. Posteriormente se realizaron esfuerzos, fundamentalmente el grupo de Wolpaw, por extender estas interfases al movimiento del cursor en dos dimensiones (Wolpaw, 1994; McFarland, 1997), con relativo éxito. En 1996, La Course de la Universidad de New Hampshire, USA, combinando la ICC de Wolpaw (1994) con la de Dewan (1967), concibe en forma simple y práctica la primera interfaz que proporciona las funciones de un ratón. Esta ICC permite desplazar el cursor en dos dimensiones en la pantalla y además realizar la activación o ‘click’, permitiendo mediante un esquema de menús, realizar distintas tareas y además ejecutar programas estándares (La Course, 1996). Las ICC construidas por Wolpaw, Pfurtscheller y La Course, podrían hoy considerarse “clásicas”. Estas interfases se basan en una descripción espectral de la señal de

EEG y generan las acciones de control a partir de la energía presente en determinadas bandas. También se ha explorado la utilización de modelos recursivos para caracterizar la señal de EEG. Estas técnicas han sido implementadas en algunas interfaces de Pfurtscheller (Schlogl, 1997; Schlogl, et al, 1997b) y por Penny y Roberts (Penny et al, 1999), de la Universidad de Oxford (Brain Computer-Interface Project) en el Reino Unido.

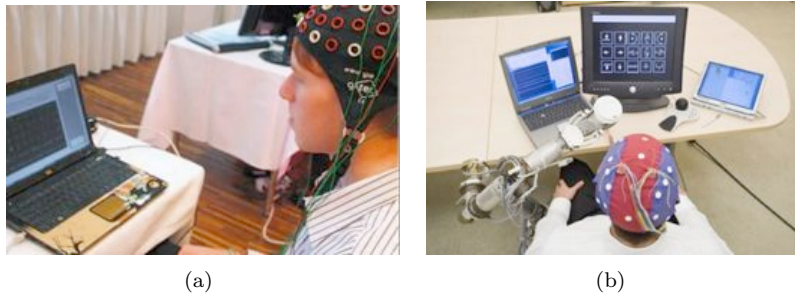


Figura 7.12: (a) Procesador de texto basado en BCI, (b) Brazo robótico controlado por P300

Actualmente las interfaces cerebro computadora están siendo satisfactoriamente empleadas por personas con severas dificultades motoras, proporcionándoles la posibilidad de comunicación y/o control de dispositivos de asistencia (figura 7.4.1). A pesar de esto, las investigaciones destinadas a mejorar el rendimiento y capacidad de transmitir mas información apuntan a incrementar la resolución espacial (y de esta manera la especificidad de la señal de EEG) empleando configuraciones laplacianas en el posicionamiento de los electrodos (Gin-Shin Chena et al, 2005), implementar configuraciones tripolares en la adquisición de la señal de EEG (Besio et al, 2008) y mejorar la impedancia de interfase electrodo-piel (Yi Lu et al, 2008).

7.4.2. Interfaces basadas en la interacción mioeléctrica

La señal de electromiografía (EMG) es el resultado de la sumatoria de todos los Potenciales de Acción de las Unidades Motoras (MUAP) en la región de los electrodos de registro. Desde los años 60 la señal eléctrica de los músculos, ha sido estudiada a fin de utilizarla como alternativa de control, en dispositivos de asistencia dinámicos tales como prótesis y ortesis (Rahman et al, 2001). En el control mioeléctrico, las señales de EMG son analizadas y clasificadas de acuerdo a sus características temporales para luego ser transformadas en comandos de control (Hudgins et al, 1993; Park et al, 1998; Lee et al, 1996). El procesamiento digital aplicado a señales de EMG esta ampliamente desarrollado para su uso en el control mioeléctrico de dispositivos (Reaz et al, 2006). Sin embargo, otros aspectos tales como movimientos, tipos de contracción y músculos involucrados, aun siguen siendo investigados. Estos aspectos muestran que los avances en el control mioeléctrico apuntan fundamentalmente a encontrar, mediante movimientos pre-establecidos, patrones temporales o espectrales que permitan mejorar el control de un sistema.

La implementación de una interfase basada en la interacción mioeléctrica tiene en cuenta los siguientes puntos: a) el número de fibras musculares activas durante una determinada contracción, ya que modifica directamente las características del EMG; b) la distancia entre electrodos de registro en una configuración bipolar, ya que envuelve el número de fibras activas que están siendo observadas; c) el movimiento con cargas externas debido a que involucra el reclutamiento de fibras musculares y modifica las características del EMG (Farfán et al, 2010). En general estas interfases hacen uso de la información de amplitud (por ejemplo, valor RMS) para generar comandos de control, e información de frecuencia (por ejemplo, power spectrum) para monitorear la fatiga.

El EMG, junto con teorías de la mecánica y el procesamiento de señales han sido utilizados para la estimación de la fuerza muscular. Actualmente estos métodos son implementados en prótesis como medio de retroalimentación para el control de estrategias de agarre (por ejemplo, agarrar un huevo sin romperlo). El futuro del control mioeléctrico apunta a imitar en forma precisa la funcionalidad de un miembro perdido (figura 7.4.2)

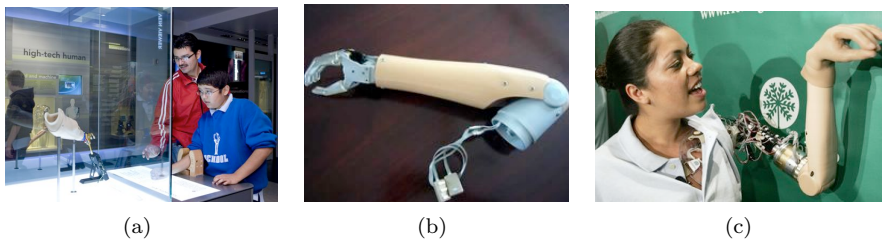


Figura 7.13: Ejemplos de interfases mioeléctricas (a) Prótesis de brazo y mano, (b) dedos y codo, (c) Prótesis de Todd Kulken

7.4.3. Interfases basadas en señales de electrooculografía

Una alternativa para establecer una comunicación hombre-maquina se basa en la detección del movimiento del ojo. Esta detección puede ser realizada a través de electrodos superficiales colocados alrededor del ojo (electrooculografía, EoG).

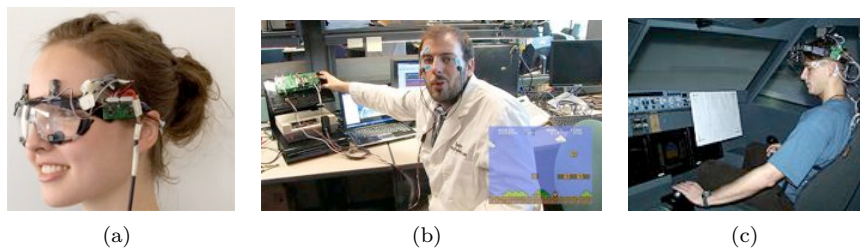


Figura 7.14: Ejemplos de interfases basadas en EoG (a) Interfase desarrollada por el Wearable Computing Lab de la ETH Zurich, (b) control de video juegos con la mirada, (c) Control de la navegación aérea

El método consiste en detectar los movimientos de los ojos mediante el registro de la diferencia de potencial existente entre la córnea y la retina. Dichas señales, convenientemente capturadas y procesadas, permiten obtener información acerca de la dirección de la mirada de la persona (Zhao Lv et al, 2008). Se han utilizado en aplicaciones tales como el control remoto de una TV (Deng et al, 2010). Es una tecnología muy útil para proporcionar apoyo a personas con discapacidades, las cuales restringen el uso de otras interfaces hombre maquina. Otras aplicaciones muy frecuentes son el control de sillas de ruedas (Barea et al, 2002), entretenimiento, manejo de simuladores de vuelo, entre otras (figura 7.4.2).

Actualmente las interfaces basadas en señales biológicas no son perfectas. Es muy difícil conseguir el 100 % de precisión, lo cual es frecuentemente logrado con el uso de múltiples interfaces, por ejemplo combinando EOG y EMG en el control de una silla de ruedas (Tsui et al, 2007), la teleoperación de ojos y brazos robóticos para aplicaciones en seguridad y EMG, EoG y EEG todas juntas para monitorear y analizar los estados del sueño. Estas tecnologías tienen mucho potencial, desde la asistencia a personas discapacitadas a la interacción con el entorno. Estas interfaces forman parte de un área en el desarrollo que podría mejorar la forma de interacción con la tecnología emergente con posibilidades de una perfecta integración en la vida cotidiana.

7.5. ROBOTICA MEDICA APLICADA

7.5.1. Evaluación de los modelos para una prótesis de mano

Para validar los modelos matemáticos obtenidos para la mano se han desarrollado algunos programas bajo el software MatLab/Simulink. El siguiente procedimiento describe un ejercicio para hacer la evaluación, donde se ha utilizado la versión V7.5(R2010a) de Simulink.

1. **Evaluación de agarres cilíndricos.** El modelo geométrico directo de la mano, unido a propiedades geométricas, permite calcular los ángulos que deben describir las articulaciones de los dedos para inscribir un cilindro entre las falanges y la palma de la mano.

Los siguientes algoritmos permiten visualizar el desempeño de la mano para rodear cilindros de diferente radio y determinar de este modo la mejor ubicación del cilindro respecto de la mano en función de las dimensiones de las falanges.

Mediante la función de matlab `manofunc1.m` el usuario puede definir la distancia $d3$ (distancia en centímetros desde la unión entre la palma con las falanges hasta la línea media del cilindro). El usuario también debe definir el radio del cilindro R en centímetros.

En un agarre tipo gancho, el cilindro no se apoya sobre la palma sino sobre las falanges de los dedos. El pulgar no hace contacto con el cilindro. La función de matlab `manofunc.m` corresponde a este caso y solo se diferencia de la descrita antes en que la distancia $d3$ se mide desde la unión de las falanges con la palma hasta la línea media del cilindro que se ubica sobre la falange proximal (como en la figura 7.19).

La figura 7.15 muestra el resultado del algoritmo para un agarre tipo gancho para un radio $R = 5$ cm., mientras que la figura 1 muestra el resultado para un cilindro de radio $R = 8$ cm, donde se muestra el efecto de desplazar la ubicación de la línea media del cilindro respecto a la palma de la mano.

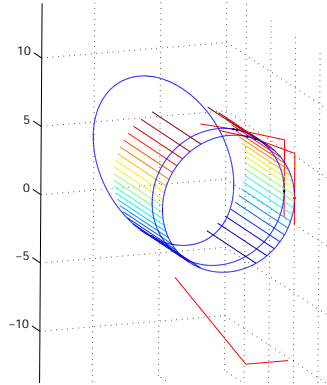


Figura 7.15: Uso de `manofunc.m`, $d3 = 1,93$, $R = 5$ cm.

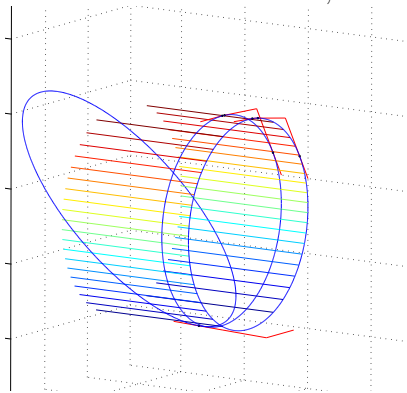


Figura 7.16: Uso de `manofunc1.m`, $d3 = 1,93$, $R = 8$ cm.

El interesado puede usar estos archivos para probar los agarres para diferentes radios o posiciones de $d3$ (líneas 26 y 38 del fichero).

2. **Cálculo de Trayectorias.** Las funciones `trayectorias.m` y `trayectorias1.m` realizan el cálculo de los ángulos para agarres cilíndricos de radios grandes y pequeños respectivamente. Se hace un barrido del radio desde un valor máximo a definir por el usuario (R_{max} en la línea 46) a un valor mínimo (R_{min} en la línea 47) y se guardan las posiciones articulares de los dedos pulgar, medio e índice para cada radio para un total de mil muestras.

El resultado se guarda en los archivos `trayectcil1.mat` o `trayectcil.mat` respectivamente, en las variables `medio`, `indice` y `pulgar` (vectores con los valores del incremento a realizar en las articulares proximal, media y distal) que constituyen

las trayectorias a seguir por un algoritmo de control de posición (valores en radianes, representando cada dato el ángulo que debe girar la articulación en cada paso de simulación).

Se pueden generar diversas trayectorias para esta arquitectura de mano en función de los parámetros geométricos de la mano que se desee probar, o construir archivos `.mat` con otro tipo de trayectoria conservando la misma estructura de datos.

3. **Evaluación del movimiento de la mano en Simulink.** El modelo CAD de la mano robótica diseñada se construyó con la ayuda del software Solid Works y se grabó como un modelo VRML, generando el archivo `mano3.wrl` compatible con el Toolbox de MatLab Simulink 3D Animation. Las trayectorias para un agarre cilíndrico generadas con el algoritmo del punto anterior pueden evaluarse con mayor grado de realismo mediante el archivo de Simulink `manovirtual.mdl`. Las trayectorias se leen mediante la s-función `leertray.m` diseñada para barrer la trayectoria construida hasta un radio R que puede establecer el usuario en el archivo de Simulink. La figura 7.17 muestra el estado final de la animación obtenida con el programa. Para cambiar el archivo de trayectoria a seguir se debe dar doble click a la s-función `leertray.m` y pasar como parámetro adicional, el nombre del archivo de trayectoria (`.mat`) que se desea ejecutar).



Figura 7.17: Evaluación de agarre tipo gancho del modelo `mano3.wrl`
dfigure

4. **Evaluación de un control CTC para el modelo dinámico inverso.** Para evaluar el desempeño de un algoritmo de control de posición, puede usarse el archivo de Simulink `ControlCTC.mdl`. Allí, además del mundo virtual del punto anterior se incluye el bloque `Modelo Dinámico Inverso` en donde se hace uso de s-functions que permiten calcular ese modelo. Como ejemplo, se incluye un control por par calculado (CTC) en el bloque `Controlador`, el cual toma la trayectoria deseada y genera los pares requeridos por el modelo dinámico inverso de la mano robótica para

seguir la trayectoria. El bloque **Error posición** calcula el error de seguimiento de posición obtenido con el controlador. Antes de ejecutar el archivo de Simulink es preciso dar doble click al bloque de inicio de parámetros, `ini1.m`, que define los parámetros del controlador y algunos datos de tiempo. Los valores de las constantes de acción proporcional y derivativa están calculados por ensayo y error, razón por la cual se observa un transitorio de error de posición que se refleja como un temblor de los dedos antes de llegar al valor final. El usuario puede ajustar los parámetros del controlador para obtener una respuesta sin temblor (suavizar el transitorio del error). El usuario puede diseñar y probar otras técnicas de control basadas o no en el modelo dinámico inverso y comparar su desempeño en cuanto a error y esfuerzo de control. La figura 7.18 muestra el esquema en el entorno Simulink.

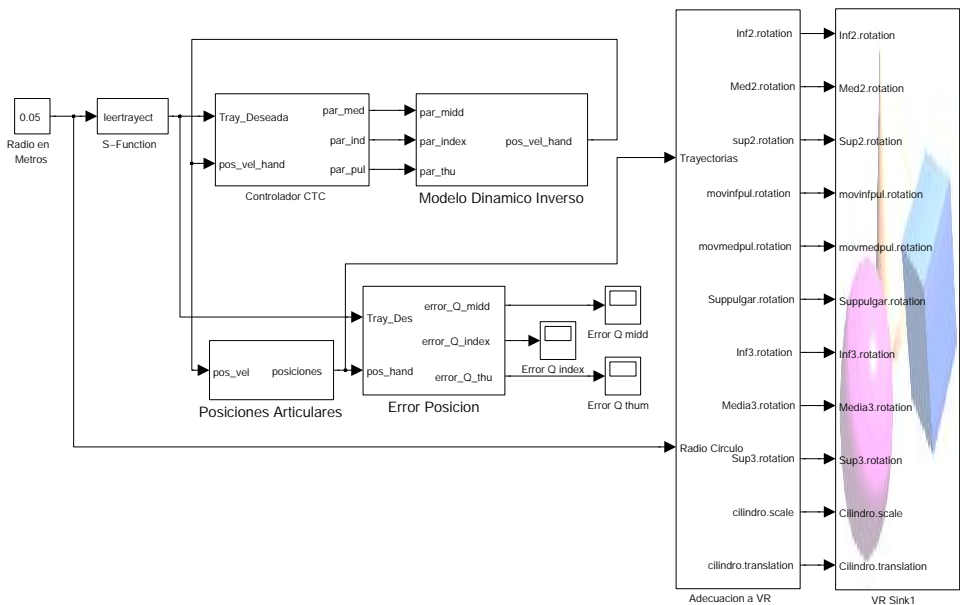


Figura 7.18: Esquema de control de posición

7.5.2. Diseño de una prótesis de mano

7.5.2.1. Aspectos preliminares

La discapacidad por amputación adquiere importancia debido a la gran cantidad de personas afectadas como consecuencia de accidentes, principalmente. Las personas con este tipo de limitación experimentan diversos efectos físicos, psicológicos y sociológicos entre los que se destacan:

- Alteración del equilibrio del organismo que deteriora los grupos musculares asociados a la extremidad.
- Reducción de autoestima.
- Pérdida de independencia para el desarrollo de actividades cotidianas.

La hipótesis planteada para el diseño de la prótesis de mano que se presenta es que el incremento de independencia en una persona con amputación de miembro superior se logra al aumentar las posibilidades de movimiento en la extremidad perdida; este incremento se cree posible mediante la implantación de una prótesis activa de mano. Los sistemas prostéticos de este tipo resultan altamente costosos para la mayoría de la población y aquellos que están al alcance del presupuesto de un ciudadano, en general, resultan poco funcionales.

De otro lado, los sistemas más funcionales resultan de mayor complejidad de operación para el usuario, a tal punto que en algunos casos la persona con discapacidad al poco tiempo de adquirir su prótesis, termina por descartar su uso y continuar su vida sin este tipo de asistencia.

Una alternativa para aliviar la adaptación del paciente a la prótesis activa es el desarrollo de sistemas de entrenamiento basados en prótesis animadas o virtuales. Estos sistemas permiten que un usuario se conecte a un sistema de adquisición de datos y se entrene en la generación de movimientos con la extremidad perdida, teniendo como realimentación básica la animación de una prótesis virtual capaz de simular el comportamiento de una prótesis real.

7.5.2.2. Metodología

Teniendo en cuenta los aspectos anteriores, la metodología adoptada se fundamenta en el desarrollo de tres etapas:

- Modelado, diseño y simulación de una mano virtual controlada a partir de señales electromiográficas.
- Diseño y construcción de un prototipo de prótesis robótica mioeléctrica.
- Construcción de la prótesis activa de mano.

La primera etapa busca un producto denominado *Entrenador virtual*. El sistema Entrenador debe permitir entrenar a un usuario en el desarrollo de posiciones y agarres básicos con la prótesis y en algunos casos interactuar con objetos virtuales. Complementariamente, se busca que el sistema brinde soporte para el desarrollo de terapias de

recuperación de actividad motora básica y procedimientos de evaluación de factibilidad de acople de prótesis en pacientes con amputación de mano.

La segunda etapa busca construir un prototipo mecánico de la prótesis modelada capaz de ser controlado desde señales electromiográficas. En esta fase se diseñan los circuitos electro mecánicos necesarios para poder imitar en parte al sistema biológico humano en los sistemas: motor (mediante los sistemas para la actuación de los motores eléctricos que componen las articulaciones de los dedos de la prótesis), sensorial (mediante la captura y adecuación de señales que permiten tener idea de la posición de articulaciones, contacto con objetos, deslizamiento de objetos, o fuerza en las articulaciones), y planificación o control (mediante la detección de intención de movimiento desde las señales EMG captadas de grupos de músculos sanos en el paciente, seguimiento de trayectorias articulares para la fase de aproximación al objeto, y cálculo de fuerza a aplicar al objeto durante la fase de agarre).

La tercera etapa busca perfeccionar el prototipo de la etapa dos para obtener una prótesis con características cosiméticas adecuada para ser implantada en un paciente. Esta etapa debe considerar los aspectos anatómicos y psicológicos involucrados con la aceptación de la prótesis por parte del paciente, como una parte constitutiva de su propio cuerpo.

En la aplicación descrita en esta sección, se considerarán solamente algunos de los aspectos involucrados en las etapas uno y dos de la metodología ya que tienen relación más estrecha con la aplicación de la robótica en el diseño de la prótesis activa.

7.5.2.3. Selección de la arquitectura de la mano

La prótesis se ha definido como una mano de tres dedos (medio, índice y pulgar). El prototipo final llevará también los dedos anular y meñique, que seguirán fielmente los movimientos del dedo medio. Esto hará que el diseño matemático sea más simple, proporcionando sin embargo las funcionalidades básicas de una mano humana. Como se explicó anteriormente, el número total de grados de libertad es de nueve, tres por cada dedo, utilizando articulaciones de rotación en cada caso. La Figura 7.19 muestra un diagrama esquemático con los parámetros definidos para la construcción del modelo geométrico.

7.5.2.4. Parámetros geométricos

La tabla 7.1 muestra los parámetros geométricos teniendo en cuenta que se trata de una estructura tipo arborescente.

Los valores iniciales de diseño para una mano de tamaño similar a una mano humana promedio son: $D2 = 0,057$; $D3 = 0,039$; $D4 = 0,027$; $D6 = 0,052$; $D7 = 0,036$; $D8 = 0,025$; $D10 = 0,032$; $D11 = 0,039$; $D12 = 0,044$; $R5 = 0,021$; $\theta_7 = 45^\circ$; $d1 = 0,1022$; $d2 = 0,011$ y las longitudes se expresan en metros.

Obsérvese que los parámetros θ_j hacen referencia a las variables de cada articulación de rotación; d_j hace referencia a la longitud de cada falange; α_j al ángulo entre ejes X_j ; b_j la distancia entre los ejes de cada dedo; r_j representa la distancia en tres ejes Z_j ; σ_j define que el tipo de articulaciones utilizadas son todas de rotación; y γ_j representa el ángulo del dedo pulgar respecto a los otros dedos (definido igual a 45°).

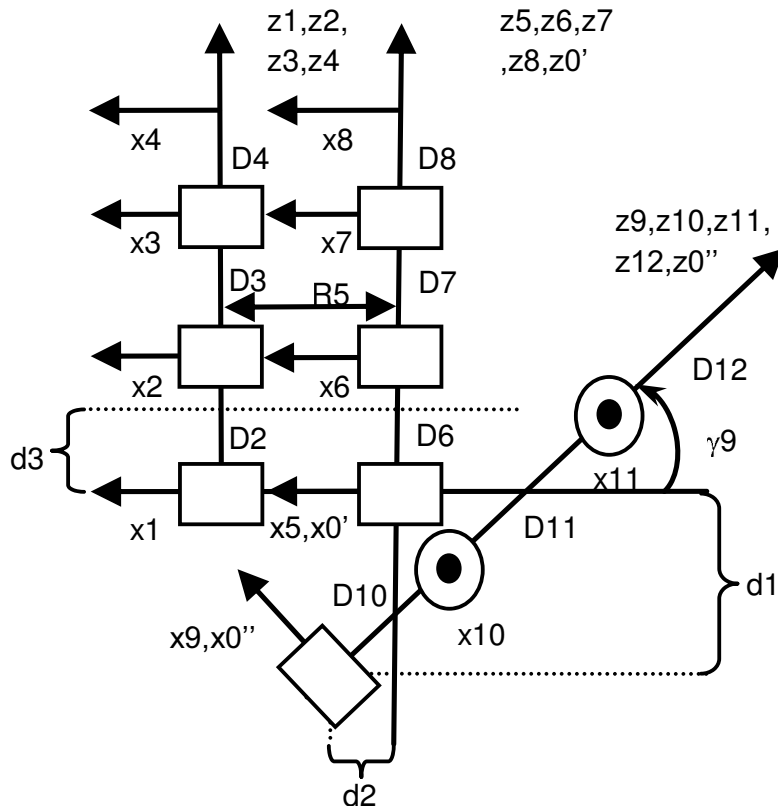


Figura 7.19: Arquitectura de la mano robótica

j	σ_j	γ_j	b_j	α_j	d_j	θ_j	r_j
1	0	0	0	0	0	θ_1	0
2	0	0	0	0	D2	θ_2	0
3	0	0	0	0	D3	θ_3	0
4	0	0	0	0	D4	0	0
5	0	0	0	0	0	θ_5	R5
6	0	0	0	0	D6	θ_6	0
7	0	0	0	0	D7	θ_7	0
8	0	0	0	0	D8	0	0
9	0	γ_9	0	0	0	θ_9	0
10	0	0	0	90°	D10	θ_{10}	0
11	0	0	0	0	D11	θ_{11}	0
12	0	0	0	0	D12	0	0

Tabla 7.1: Parámetros geométricos de la mano

7.5.2.5. Modelo geométrico directo de la mano

El modelo geométrico directo permite encontrar la posición y orientación de la punta de cada dedo, dadas las posiciones articulares de cada articulación. Dada la particular estructura arborescente de la mano compuesta por tres dedos, se obtiene la matriz de transformación entre bases de referencia teniendo en cuenta los parámetros vistos en la tabla 7.1 esta matriz (4×4) es:

$${}^i T_j = \begin{bmatrix} C\gamma_j C\alpha_j & -C\gamma_j S\alpha_j C\theta_j + S\gamma_j S\theta_j & C\gamma_j S\alpha_j S\theta_j + S\gamma_j C\theta_j & C\gamma_j C\alpha_j r_j + S\gamma_j d_j \\ S\alpha_j & C\alpha_j C\theta_j & -C\alpha_j S\theta_j & S\alpha_j r_j + b_j \\ -S\gamma_j C\alpha_j & S\gamma_j S\alpha_j C\theta_j + C\gamma_j S\theta_j & -S\gamma_j S\alpha_j S\theta_j + C\gamma_j C\theta_j & -S\gamma_j C\alpha_j r_j + C\gamma_j d_j \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

De donde, asumiendo de interés la posición de la punta de cada dedo, (E), éstas se pueden describir respecto a las coordenadas de la base, (0), mediante tres cadenas cinemáticas independientes para los dedos índice, medio y pulgar: ${}^0 T_E$, ${}^{0'} T_E$ y ${}^{0''} T_E$ respectivamente mediante:

Medio:

$${}^0 T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C1 & -S1 & 0 \\ 0 & S1 & C1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1 T_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C2 & -S2 & 0 \\ 0 & S2 & C2 & D2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2 T_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C3 & -S3 & 0 \\ 0 & S3 & C3 & D3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^3 T_E = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & D4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Índice:

$${}^{0'} T_5 = \begin{bmatrix} 1 & 0 & 0 & R5 \\ 0 & C5 & -S5 & 0 \\ 0 & S5 & C5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5 T_6 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C6 & -S6 & 0 \\ 0 & S6 & C6 & D6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^6 T_7 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C7 & -S7 & 0 \\ 0 & S7 & C7 & D7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^7 T_E = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & D8 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Pulgar:

$$\begin{aligned}
{}^{0''}T_9 &= \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2}S9 & \frac{\sqrt{2}}{2}C9 & 0 \\ 0 & C9 & -S9 & 0 \\ -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2}S9 & \frac{\sqrt{2}}{2}C9 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^9T_{10} &= \begin{bmatrix} 0 & -C10 & S10 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & S10 & C10 & D10 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
{}^{10}T_{11} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C11 & -S11 & 0 \\ 0 & S11 & C11 & D11 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^{11}T_E &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & D12 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}$$

7.5.2.6. Modelo dinámico de la mano

La forma general de la ecuación dinámica de un robot se expresa de la siguiente manera:

$$\Gamma = \mathbf{A}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{Q}(\mathbf{q}) + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{F}_s \text{sign}(\dot{\mathbf{q}})$$

Donde Γ corresponde a los pares aplicados a cada uno de los motores, \mathbf{A} es la matriz de inercia del robot, \mathbf{C} la matriz de Coriolis y fuerzas centrífugas, \mathbf{Q} el vector de gravedad, \mathbf{F}_v la matriz de rozamientos viscosos y \mathbf{F}_s la matriz de rozamientos secos. Las posiciones, velocidades y aceleraciones articulares están dadas por \mathbf{q} , $\dot{\mathbf{q}}$, y $\ddot{\mathbf{q}}$ respectivamente. El modelo dinámico directo permite realizar la simulación del comportamiento de la mano robot, mientras que el modelo dinámico inverso permitirá la implementación de un controlador basado en el modelo. Un paso previo al cálculo de estos modelos consiste en hallar los parámetros dinámicos de base del robot, esto es, el juego mínimo de parámetros que involucra el modelo con el fin de reducir los cálculos. Como los parámetros dinámicos de un robot son once, seis elementos del tensor de inercia ($XXj, XYj, XZj, YYj, YZj, ZZj$), tres del primer momento de inercia (MXj, MYj, MZj), uno de la masa (Mj) y uno de la inercia del actuador (Iaj), cada dedo dispondría de 33 parámetros diferentes. Pero aplicando las técnicas de agrupamiento de parámetros y ciertas consideraciones especiales (suponer la matriz del tensor de inercia diagonal y los cuerpos de la mano con una forma simétrica), éstos se reducen notablemente. La tabla 7.2 muestra los parámetros dinámicos de base para cada uno de los dedos de la mano propuesta y para cada una de sus tres articulaciones.

En la tabla 7.2 la letra "I" se antepone a alguno de los parámetros indicando que ese término incluye o agrupa otros términos, lo cual permite obtener varios valores nulos en la tabla y por lo tanto simplificar las operaciones. Con el fin de simplificar el modelo, no se consideran los rozamientos ni la matriz de fuerzas de Coriolis y centrífugas que se mostraron en la ecuación dinámica. Es decir, el modelo reducido solo tendría en cuenta la matriz de inercia \mathbf{A} y el vector de gravedad \mathbf{Q} , los dos dependientes de las posiciones articulares. Realizando los cálculos completos para la obtención del modelo dinámico inverso (velocidad de rotación, velocidad de traslación, energía cinética, vector de gravedad), se obtiene este modelo para cada dedo en las ecuaciones siguientes:

Dedo medio											
j	XXj	XYj	XZj	YYj	YZj	ZZj	MXj	MYj	MZj	Mj	Iaj
1	0	0	0	0	0	ZZR1	MX1	MY1	0	0	0
2	XXR2	0	0	0	0	ZZR2	MXR2	MY2	0	0	IA2
3	XXR3	0	0	0	0	ZZ3	MX3	MY3	0	0	IA3
Dedo índice											
j	XXj	XYj	XZj	YYj	YZj	ZZj	MXj	MYj	MZj	Mj	Iaj
4	0	0	0	0	0	ZZR4	MX1	MY4	0	0	0
5	XXR5	0	0	0	0	ZZR5	MXR2	MY5	0	0	IA5
6	XXR6	0	0	0	0	ZZ6	MX3	MY6	0	0	IA6
Dedo pulgar											
j	XXj	XYj	XZj	YYj	YZj	ZZj	MXj	MYj	MZj	Mj	Iaj
7	0	0	0	0	0	ZZR7	MXR1	MY7	0	0	0
8	XXR8	0	0	0	0	ZZR8	MXR2	MY8	0	0	0
9	XXR9	0	0	0	0	0	MX3	MY9	0	0	IA9

Tabla 7.2: Parámetros dinámicos de base

Dedo medio: Matriz de inercia:

$$\begin{aligned}
 A11 &= XXR2 + 2MY2D2S2 + XXR3 + 2MY3D2C2S3 + 2MY3D3S3 + 2MY3D2S2C3 \\
 A22 &= XXR2 + XXR3 + 2MY3D3S3 + IA2 \\
 A33 &= XXR3 + IA3 \\
 A12 &= XXR2 + MY2D2S2 + XXR3 + MY3D2C2S3 + 2MY3D3S3 + MY3D2S2C3 \\
 A13 &= XXR3 + MY3D2C2S3 + MY3D3S3 + MY3D2S2C3 \\
 A23 &= XXR3 + MY3D3S3
 \end{aligned}$$

Vector de gravedad:

$$\begin{aligned}
 Q1 &= -G3MY1C1 - G3MY2C12 - G3MY3C3C12 + G3MY3S3S12 \\
 Q2 &= -G3MY2C12 - G3MY3C3C12 + G3MY3S3S12 \\
 Q3 &= G3MY3S12S3 - G3MY3C12C3
 \end{aligned}$$

Dedo índice: Matriz de inercia:

$$\begin{aligned}
 A44 &= XXR5 + 2MY5D5S5 + XXR6 + 2MY6D5C5S6 + 2MY6D6S6 + 2MY6D5S5C6 \\
 A55 &= XXR5 + XXR6 + 2MY6D6S6 + IA5 \\
 A66 &= XXR6 + IA6 \\
 A45 &= XXR5 + MY5D5S5 + XXR6 + MY6D5C5S6 + 2MY6D6S6 + MY6D5S5C6 \\
 A46 &= XXR6 + MY6D5C5S6 + MY6D6S6 + MY6D5S5C6 \\
 A56 &= XXR6 + MY6D6S6
 \end{aligned}$$

Vector de gravedad:

$$Q4 = -G3MY4C4 - G3MY5C45 - G3MY6C6C45 + G3MY6S6S45$$

$$Q5 = -G3MY5C45 - G3MY6C6C45 + G3MY6S6S45$$

$$Q6 = G3MY6S45S6 - G3MY6C45C6$$

Dedo pulgar: Matriz de inercia:

$$A77 = ZZR8S82 + 2MY8D8S8 + 2MY9D8S89 + 2MY9D9C8S89$$

$$A88 = XXR8 + XXR9 + 2MY9D9S9$$

$$A99 = XXR9$$

$$A78 = 0$$

$$A79 = 0$$

$$A89 = XXR9 + MY9D9S9$$

Vector de gravedad:

$$Q7 = -G3MY7C7 + G3MY8S7S8 + G3MY9S7S8C9 + G3MY9S7C8S9$$

$$Q8 = G3MY8S8 - G3MY8C7C8 + G3MY9S8C9 - G3MY9C7C8C9 + \\ +G3MY9C8S9 + G3MY9C7S8S9$$

$$Q9 = G3MY9C9S9 + G3MY9C7S8S9 + G3MY9S8C9 - G3MY9C7C8C9$$

El modelo dinámico directo que permite la simulación de la prótesis de mano en el computador, se expresa por medio de la siguiente ecuación aplicable a cada uno de los dedos de la mano:

$$\ddot{\mathbf{q}} = \mathbf{A}^{-1}(\mathbf{\Gamma} - \mathbf{Q})$$

Bibliografía para ampliar

Handbook of robotics B. Siciliano, O. Khatib. Springer. Los capítulos 53 “Rehabilitation and Health care robotics” y 54 “Domestic robotics” presentan un completo estado de arte de los robots asistenciales y domésticos.

Winter school on rehabilitation robots, 2008 . [enlace a la documentación de la escuela](#) Este curso, celebrado en Elche en 2008 contó con ponencias y cursillos de los más prestigiosos investigadores en el área de robótica de rehabilitación y asistiva.

Parte II

Guiones y tutoriales del
proyecto OpenSurg

Capítulo 8

Guiones básicos prácticos

“En teoría, teoría y práctica es lo mismo. En la práctica no es así.”
Popular

Indice

8.1. Introducción al proyecto OpenSurg	234
8.2. Instalación básica de Ubuntu 8.10 Intrepid Ibex para OpenSurg	234
8.2.1. Prerrequisitos	234
8.2.2. Instalación	234
8.3. Tutorial básico de operaciones con ITK	237
8.3.1. Prerrequisitos	237
8.3.2. Ejercicio 1. Operaciones básicas	237
8.3.3. Ejercicio 2. Filtrado de imágenes	238
8.3.4. Ejercicio 3. Detección de bordes	239
8.4. Tutorial básico de uso de VTK	240
8.4.1. Ejercicio 1. Hola mundo. El cono de VTK	240
8.4.2. Ejercicio 2. Ejemplo básico de reconstrucción 3D con imagen médica	241
8.4.3. Ejercicio 3. Ejemplo de integración VTK-Qt	241
8.5. Tutorial básico de uso de IgsTK	242
8.6. Ejercicio para la detección y medición de calcificaciones en mamografía 2D	244
8.6.1. Indicaciones para la solución	245
Bibliografía para ampliar	246

8.1. Introducción al proyecto OpenSurg

OpenSurg es un proyecto de colaboración iberoamericano para la realización de guiones docentes para la enseñanza de la robótica médica, la robótica quirúrgica y la robótica de rehabilitación. El objetivo final del proyecto es la preparación de personal cualificado para la investigación biomédica en estas áreas. Como resultados del proyecto se plantea la realización de diferentes simuladores robóticos por parte de los grupos del consorcio. Aunque todas las herramientas software utilizadas son de código abierto, los guiones de **OpenSurg** se escriben para su utilización con Ubuntu 8.10 Intrepid Ibex. OpenSurg se extendió por un periodo de cuatro años (2009-2012), y durante el mismo se fueron añadiendo los resultados parciales y finales obtenidos, así como mejoras y modificaciones de los guiones docentes creados.

El proyecto OpenSurg se fundamenta en los trabajos sobre simuladores médicos con capacidades hapticas que están siendo realizados por los grupos participantes. En concreto, el proyecto se centra en el desarrollo de simuladores en dos campos principales: simuladores para robótica quirúrgica y simuladores para robótica de rehabilitación. Una de las ventajas más importantes de los simuladores por computador es la oportunidad que ofrecen como herramienta de entrenamiento y docencia. Además, permiten disponer de una muy valiosa herramienta software útil para el desarrollo de nuevos procedimientos o dispositivos médicos. OpenSurg trata de coordinar los diferentes desarrollos y aplicaciones de los distintos grupos participantes de forma que utilicen una plataforma software común basada en aplicaciones de software libre. En concreto se pretende fomentar el uso e interrelación de los diferentes proyectos de software libre, y particularizarlo a diferentes aplicaciones dentro de cada grupo.

8.2. Instalación básica de Ubuntu 8.10 Intrepid Ibex para OpenSurg

Resumen. Descripción del proceso seguido para la instalación de Ubuntu 8.10 Intrepid Ibex en los equipos informáticos utilizados en el desarrollo del proyecto OpenSurg. Se recomienda seguir estos pasos para disponer de una misma configuración mínima en todos los equipos.

8.2.1. Prerrequisitos

Se necesita disponer del cd de instalación de Ubuntu 8.10. Se puede obtener de la página oficial de Ubuntu (www.ubuntu.com).

8.2.2. Instalación

A continuación se detallan los pasos a seguir para la instalación básica de Ubuntu y la instalación de los paquetes adicionales necesarios. Estos paquetes son necesarios para los posteriores ejercicios.

- Instalación básica de Ubuntu 8.10.
- Utilizando el gestor de paquetes Synaptic (accesible desde Sistema → Administración → Gestor de paquetes Synaptic) se instalan los siguientes paquetes:
 - *build-essential*. Este paquete incluye todo lo necesario para la compilación de la mayor parte de software que podáis necesitar compilar. (Esto añadirá otros paquetes como son *g++*, *g++4.3*, *dpkg-dev*, *libstdc++ 6-4.3-dev* y *patch*).
 - *Cmake*. Se trata de un sistema make de código abierto y multiplataforma. Cmake se usa para controlar el proceso de compilación del software. También puede instalarse bajando las fuentes desde www.cmake.org., aunque la versión utilizada en OpenSurg es la 2.6.0-4 que viene con Ubuntu.
 - *Cvs*. Version 1:1.12.13-11 Sistema de control de versiones concurrentes.
 - *Subversion*. Version 1.5.1 Sistema de control de versiones similar a cvs.
 - *qt4-dev-tools*. Paquete para la instalación de las librerías Qt de troltech para realizar las interfaces de usuario.
 - *Alien*. Aplicación utilizado para pasar los paquetes .rpm (de RedHat, Suse, etc) a .deb (debian, ubuntu). Necesario para instalar los drivers del dispositivo haptico Phantom. (ver apartado OHAE).
- Aunque con *cvs* se pueden mantener las versiones actualizadas de las librerías, los guiones de OpenSurg se van a redactar considerando las siguientes versiones instaladas de las librerías
 - LIBRERIAS VTK version 5.2.1. Visualization Toolkit.
 - Nos bajamos el fichero *vtk-5.2.1.zip* de la página de www.vtk.org y lo almacenamos en un directorio */VTK* dentro de nuestro directorio de trabajo, (en nuestro caso */home/jsabater/VTK*).
 - Descomprimos este fichero en el directorio */home/jsabater/VTK/vtk-5.2.1*.
 - Creamos el directorio *vtk-built*, donde posteriormente compilaremos las librerías.
 - Desde la consola de terminal, entramos en el directorio *vtk-built*, y tecleamos el siguiente comando:
 - `~/VTK/vtk-built$ cmake ../vtk-5.2.1`
 - nos aparece la interfaz de *cmake*, donde debemos configurar algunos parámetros. Para ello oprimos la tecla [c] de configure, apareciendo unos valores por defecto. Apretamos la tecla [t] para ver los parámetros avanzados y cambiamos el valor de *VTK_USE_GUISUPPORT* a ON, activando así la posibilidad de añadir soporte para QT. Oprimimos [c] de nuevo. Ahora cambiamos el valor de *VTK_USE_QVTK* a ON. Oprimimos [c] de nuevo. Finalmente, cuando todos los valores se hayan configurado correctamente, oprimimos [g] para generar los *makefiles* correspondientes, saliendo de la interfaz de *cmake*.

- `~/VTK/vtk-built$ make`
 - `~/VTK/vtk-built$ sudo make install`
 - Con esto habremos instalado las librerías VTK en nuestro pc.
- LIBRERIAS ITK version 3.8. Insight Segmentation & Registration Toolkit.
- Nos bajamos el fichero `InstightToolkit-3.8.tar.gz` de la página de www.itk.org y lo almacenamos en un directorio `/ITK` dentro de nuestro directorio de trabajo, (en nuestro caso `/home/jsabater/ITK`).
 - Descomprimos este fichero en el directorio `/home/jsabater/ITK/InstightToolkit-3.8.0`
 - Creamos el directorio `itk-built`, donde posteriormente compilaremos las librerías.
 - Desde la consola de terminal, entramos en el directorio `itk-built`, y tecleamos el siguiente comando:
 - `~/ITK/itk-built$ cmake ../InsightToolkit-3.8.0`
 - nos aparece la interfaz de `cmake`, donde debemos configurar algunos parámetros. Para ello oprimimos la tecla `[c]` de `configure`, apareciendo unos valores por defecto. Oprimimos `[c]` de nuevo. Finalmente, cuando todos los valores se hayan configurado correctamente, oprimimos `[g]` para generar los `makefiles` correspondientes, saliendo de la interfaz de `cmake`.
 - `~/ITK/itk-built$ make`
 - `~/ITK/itk-built$ sudo make install`
 - Con esto habremos instalado las librerías ITK en nuestro pc
- LIBRERIAS IgsTK
- Nos bajamos el fichero `IGSTK-4.0.tar.gz` de la página de www.igstk.org y lo almacenamos en un directorio `/IgsTK` dentro de nuestro directorio de trabajo, (en nuestro caso `/home/jsabater/IgsTK`).
 - Descomprimos este fichero en el directorio `/home/jsabater/IgsTK/IGSTK-4-0`
 - Creamos el directorio `IGSTK-built`, donde posteriormente compilaremos las librerías.
 - Desde la consola de terminal, entramos en el directorio `IGSTK-built`, y tecleamos el siguiente comando:
 - `~/IgsTK/IGSTK-built$ cmake ../IgsTK-4-0`
 - nos aparece la interfaz de `cmake`, donde debemos configurar algunos parámetros. Para ello oprimimos la tecla `[c]` de `configure`, apareciendo unos valores por defecto. Ponemos a `OFF` el valor `IGSTK_USE_FLTK` y `ON` el `IGSTK_USE_QT`. También anulamos las opciones de los dispositivos navegadores quirúrgicos externos.
 - `~/IgsTK/IGSTK-built$ make`
 - `~/IgsTK/IGSTL-built$ sudo make install`
 - Con esto habremos instalado las librerías IgsTK en nuestro pc

8.3. Tutorial básico de operaciones con ITK

Resumen. El presente tutorial presenta unos ejercicios básicos de uso de las librerías ITK. Estos ejercicios han sido escritos en C, simplificando la sintaxis con el fin de servir de ejemplos iniciales. En este guión se enseña como trabajar con las imágenes en ITK, realizar operaciones básicas, filtrados, etc. Dado que ITK no incluye herramientas propias para la visualización de las imágenes, los ejercicios de este guión utilizan lectura/escritura de ficheros para generar la salida de los programas que efectuamos.

8.3.1. Prerrequisitos

Se necesita disponer de la instalación de las librerías ITK version 3.8, según lo visto en el punto anterior.

8.3.2. Ejercicio 1. Operaciones básicas

En primer lugar se realiza la lectura y carga de tres imágenes (un .bmp, un .jpg y un .png), para posteriormente realizar una operación puntual con ellas. Concretamente se realiza la diferencia entre cada dos de ellas. Con este ejemplo se observa la diferencia entre ambos formatos debido a la pérdida de información en la compresión de imágenes.

- Se entrega el fichero `g2p1.cxx` que contiene el código fuente y las explicaciones necesarias para su lectura.
- Se creará el fichero `CmakeList.txt` con el siguiente contenido:

```
# This project is designed to be built outside the Insight
# source tree.
PROJECT(g2p1) }

5 # Find ITK.

FIND_PACKAGE(ITK REQUIRED)
IF(ITK_FOUND)
    INCLUDE(${ITK_USE_FILE})
10 ENDFIN(ITK_FOUND)

ADD_EXECUTABLE(g2p1 g2p1.cxx)

TARGET_LINK_LIBRARIES(g2p1 ITKCommon ITKIO)
```

- con ambos ficheros en el mismo directorio, se ejecutará CMAKE de igual forma que se hizo en las instalaciones anteriores. En este caso, dado que estamos en el mismo directorio, se ejecuta como “`ccmake .`”. Normalmente se deberá especificar el directorio donde se compilaron las ITK (`ITK_DIR=/home/jsabater/ITK/itk-built`). Esto generará el `Makefile` correspondiente.

- `make`
- se creará la aplicación `g2p1`.
- Para ejecutar la aplicación es necesario que coloquemos en el directorio las imágenes de entrada: `example.bmp`, `example.jpg` y `example.png`
- ejecutamos la aplicación creada: `./g2p1`
- se crearán las imágenes de salida en las que se han efectuado las operaciones detalladas en el código fuente: `result.bmp`, `result.jpg` y `result.png`, que podremos visualizar desde `nautilus`.

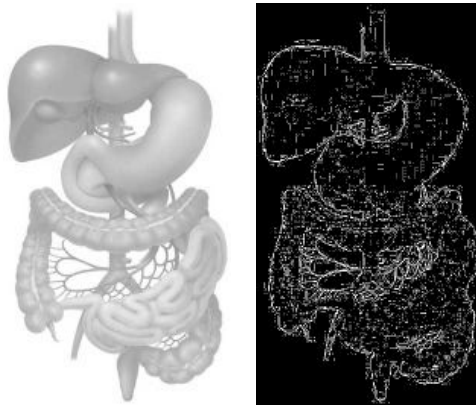


Figura 8.1: Imágenes de entrada y salida de la práctica `g2p1`

NOTA: Se recomienda utilizar otras funciones de ITK para operaciones puntuales, como puedan ser la inversa o una umbralización.

8.3.3. Ejercicio 2. Filtrado de imágenes

El objetivo del ejercicio es realizar un estudio comparativo de diversos algoritmos de filtrado para la reducción del ruido presente en una imagen. Las imágenes utilizadas son en escala de grises, y se encuentran almacenadas en `.jpg`. Se define como ruido toda aquella variación del nivel de gris que sufre un pixel no debida a la aportación lumínica de la escena.

- Se entrega el fichero `g2p2.cxx` que contiene el código fuente y las explicaciones necesarias para su lectura.
- Se creará el fichero `CmakeList.txt` de la misma forma que en el práctica anterior.

- con ambos ficheros en el mismo directorio, se ejecutará CMAKE de igual forma que se hizo en las instalaciones anteriores. Esto generará el `Makefile` correspondiente.
- `make`
- se creará la aplicación `g2p2`.
- Para ejecutar la aplicación es necesario que coloquemos en el directorio la imagen de entrada: `example.jpg`
- ejecutamos la aplicación creada con el número de parámetros de entrada definido en el código, por ejemplo: `./g2p2 1 50 5`
- se crearán la imagen de salida: `result.jpg`, que podremos visualizar desde `nautilus`.

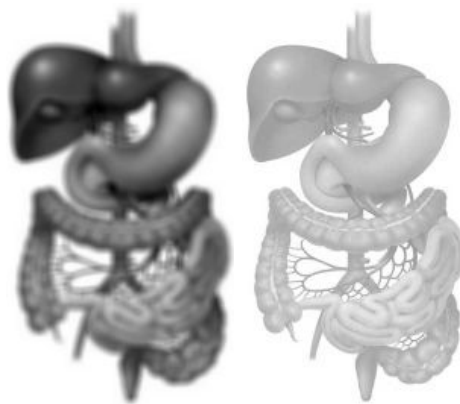


Figura 8.2: Imágenes de entrada y salida de la práctica `g2p2`

8.3.4. Ejercicio 3. Detección de bordes

El objetivo del ejercicio es realizar un estudio comparativo de diversos algoritmos de detección de bordes existentes en una imagen. Las imágenes utilizadas son en escala de grises, y se encuentran almacenadas en `.jpg`.

- Se entrega el fichero `g2p3.cxx` que contiene el código fuente y las explicaciones necesarias para su lectura.
- Se creará el fichero `CmakeList.txt` de la misma forma que en el práctica anterior.
- con ambos ficheros en el mismo directorio, se ejecutará CMAKE de igual forma que se hizo en las instalaciones anteriores. Esto generará el `Makefile` correspondiente.
- `make`

- se creará la aplicación `g2p3`.
- Para ejecutar la aplicación es necesario que coloquemos en el directorio la imagen de entrada: `example.jpg`
- ejecutamos la aplicación creada, por ejemplo: `./g2p3 3 2 10 2 2`
- se crearán la imagen de salida: `result.jpg`, que podremos visualizar desde `nautilus`.

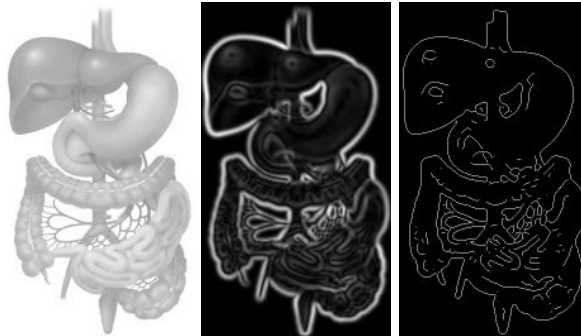


Figura 8.3: Imágenes de entrada y salida de la práctica `g2p3`. Original, Sobel y Canny

Notas adicionales

Se recomienda realizar modificaciones sobre el código ejemplo, y evaluar el resultado de dichos cambios. Se puede practicar con las imágenes de nombre `ruido_X.jpg` disponibles en la página de `OpenSurg`.

8.4. Tutorial básico de uso de VTK

8.4.1. Ejercicio 1. Hola mundo. El cono de VTK

Como primer paso se recomienda realizar los ejercicios ejemplo que vienen en la instalación de VTK. Los siguientes comentarios están relacionados con los ejercicios tutoriales que se encuentran en `$/home/jsabater/VTK/vtk-5.2.1/Examples/Tutorial`.

- Si durante la instalación de VTK se habilitó la opción `BUILD-EXAMPLES`, encontraremos estos ejemplos ya compilados bajo la dirección `$/home/jsabater/VTK/vtk-built/bin`. Sin embargo, en este ejercicio vamos a volver a compilarlos para estudiarlos mejor.
- En un terminal, nos ubicamos en `$/home/jsabater/VTK/vtk-5.2.1/Examples/Tutorial/Step1/Cxx`, y abrimos los ficheros `Cone.cxx` y `CmakeList.txt` para estudiar su contenido.
- Ejecutamos `CMAKE` (`$ ccmake .`). Probablemente tengamos que indicar el directorio donde se encuentran las VTK compiladas (`VTK_DIR = /home/jsabater/VTK/vtk-built`). Oprimimos `[c]` dos veces y generamos el proyecto `[g]`.

- `make`
- ejecutamos el programa con `$./Cone`

Repetimos los pasos para los ejemplos `Step2`, `Step3`, `Step4`, `Step5` y `Step6`.

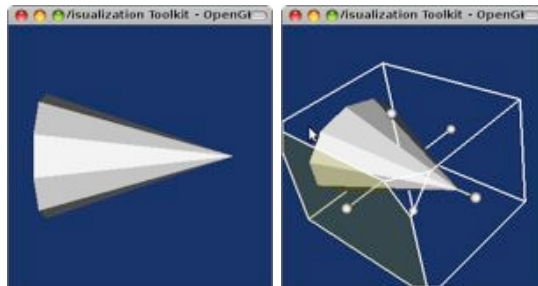


Figura 8.4: Hola Mundo de VTK

8.4.2. Ejercicio 2. Ejemplo básico de reconstrucción 3D con imagen médica

Para realizar este tutorial es necesario bajar el fichero `vtkdata-5.2.1.tar.gz` de la página de VTK. Descomprimos este fichero en el directorio `~VTK/vtk-built`, de forma que tengamos el directorio `~VTK/VTKData`. Se pide ahora realizar los tutoriales que se encuentran en `~VTK/vtk-5.2.1/Examples/Medical/Cxx`. Para ello se actúa como en el ejercicio anterior.

- Se prueba el ejemplo `Medical1`, invocandolo de la siguiente forma: `./Medical1 ../../../../../../VTKData/Data/headsq/quarter`. Modificar posteriormente el valor de 500 por 1150 (línea 68) en el código de `Medical1.cxx`. Esto modifica el valor de la isosuperficie del algoritmo utilizado. Observar el efecto del cambio al volver a compilar y ejecutar el código.
- Se prueba el ejemplo `Medical2`, invocandolo de la siguiente forma: `./Medical2 ../../../../../../VTKData/Data/headsq/quarter`.
- Se prueba el ejemplo `Medical3`, invocandolo de la siguiente forma: `./Medical3 ../../../../../../VTKData/Data/headsq/quarter`.

8.4.3. Ejercicio 3. Ejemplo de integración VTK-Qt

- En este ejemplo se construye la primera aplicación `Qt-Vtk`. En el directorio `~/Examples/GUI/ImageViewer` se puede encontrar el código necesario. En este ejemplo se carga una imagen `.png` en una ventana `Qt`.
- Más interesante es el ejemplo `SimpleView` en el que se utiliza la potencia de `Qt` para crear una interfaz que integre un `vtkWidget` donde se renderiza un cilindro.

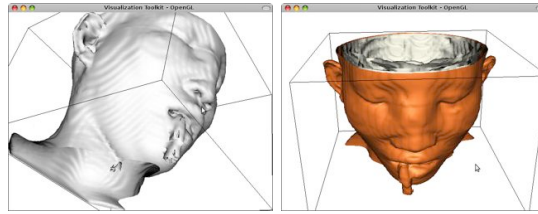


Figura 8.5: Ejemplo básico de reconstrucción 3D con VTK

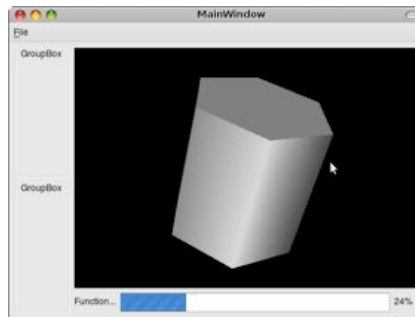


Figura 8.6: Ejemplo básico de integración Qt-VTK

8.5. Tutorial básico de uso de IgsTK

En este tutorial se va a realizar un ejemplo *HolaMundo* con las librerías IgsTK.

La mayoría de los ejemplos con GUI incluidos en el directorio de IgsTK, están compilados con FLTK. Si se incluyó esta librería, una vez compilado el proyecto IgsTK, diríjase al directorio `../IGSTK-4-2_bin/bin/debug`. En la figura 8.5 se presenta la ejecución de las aplicaciones `igstkVersionTest` y `NeedleBiopsy` con la que se pueden analizar las versiones de las librerías instaladas, las funciones de lectura de imágenes DICOM de IGSTK y la aplicación de GUI con FLTK.

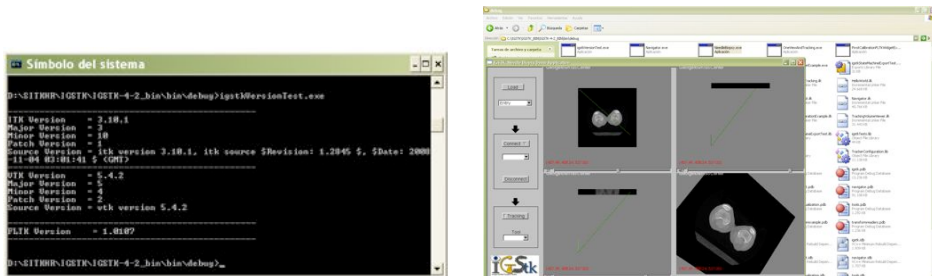


Figura 8.7: (a) `igstkVersionTest` y (b) `needleBiopsy`

Pasos:

Copie en un directorio los archivos que se presentan para un primer ejemplo de IGSTK con QT. Estos archivos corresponden al directorio /Sandbox-4-2/Examples/OneViewAndTrackingUsingQTWidget que también se puede descargar en www.igstk.org, pero se han modificado las directivas de CMakeLists.txt para que se pueda compilar como un proyecto independiente con la opción de adicionar aplicaciones de vtk e itk. Puede comparar los archivos de cada caso para verificar que se adicionaron las líneas:

```
cmake_minimum_required(VERSION 2.6)
PROJECT(OneViewAndTrackingUsingQTWidget)
#
#   FIND IGSTK
5 #
FIND_PACKAGE(IGSTK)
IF(NOT IGSTK_DIR)
    MESSAGE(FATAL_ERROR "Indique donde se encuentra el directorio de
    IGSTK IGSTK_DIR")
10 ENDFIN(NOT IGSTK_DIR)
INCLUDE(${IGSTK_USE_FILE})
#
#   FIND ITK
15 FIND_PACKAGE(ITK)
IF (USE_ITK_FILE)
    INCLUDE (${USE_ITK_FILE})
ELSE (USE_ITK_FILE)
    MESSAGE( FATAL_ERROR "Esta aplicacion requiere ITK. No se
20     encuentran sus componentes. Por favor verifique su configuracion")
ENDIF (USE_ITK_FILE)
#
#   FIND VTK
25 FIND_PACKAGE(VTK)
IF (USE_VTK_FILE)
    INCLUDE (${USE_VTK_FILE})
ELSE (USE_VTK_FILE)
    MESSAGE( FATAL_ERROR "Esta aplicacion requiere VTK. No se
30     encuentran sus componentes. Por favor verifique su configuracion")
ENDIF (USE_VTK_FILE)
#
#   FIND Qt
35 FIND_PACKAGE(Qt4 REQUIRED)
IF(QT_USE_FILE)
    INCLUDE(${QT_USE_FILE})
ELSE(QT_USE_FILE)
    SET(QT_LIBRARIES    ${QT_QT_LIBRARY})
40 ENDFIN(QT_USE_FILE)
```

```

INCLUDE_DIRECTORIES (${QT_INCLUDE_DIR})

INCLUDE_DIRECTORIES (
45   ${IGSTK_SOURCE_DIR}/Utilities/SceneGraphVisualization
   ${IGSTK_BINARY_DIR}/Utilities/SceneGraphVisualization
)

SET(OneViewAndTrackingUsingQTWidget_GUI_SRCS
50   OneViewAndTrackingUsingQTWidgetGUI.ui
)

INCLUDE_DIRECTORIES (
55   ${OneViewAndTrackingUsingQTWidget_SOURCE_DIR}
   ${OneViewAndTrackingUsingQTWidget_BINARY_DIR}
)

QT4_WRAP_UI(OneViewAndTrackingUsingQTWidget_QT_UI_SRCS
60   ${OneViewAndTrackingUsingQTWidget_GUI_SRCS})

SET(OneViewAndTrackingUsingQTWidget_SRCS
   OneViewAndTrackingUsingQTWidget.cxx
   OneViewAndTrackingUsingQTWidgetGUI.cxx
)
65 QT4_AUTOMOC(${OneViewAndTrackingUsingQTWidget_SRCS})

ADD_EXECUTABLE(OneViewAndTrackingUsingQTWidget
70   ${OneViewAndTrackingUsingQTWidget_QT_UI_SRCS}
   ${OneViewAndTrackingUsingQTWidget_SRCS})

TARGET_LINK_LIBRARIES(OneViewAndTrackingUsingQTWidget
IGSTK
${QT_LIBRARIES})

```

Ejecute CMake indicando el directorio de construcción del primer ejemplo. Compile la solución del proyecto, teniendo en cuenta que se haga en modo Release o Debug tal como se compilaban las librerías IGSTK, ITK y VTK. En la figura 8.5 se muestran las ejecuciones del primer ejemplo con Qt. Verifique las funciones de conexión de slots de Qt dadas en el archivo `OneViewAndTrackingUsingQTWidgetGUI.cxx`. Interactúe con las opciones de tipo, tamaño y ubicación de las primitivas de la escena, así como opciones de interacción del ratón (típica de `qvtk` o `tracking`) en `OneViewAndTrackingUsingQTWidget.cxx`.

8.6. Ejercicio para la detección y medición de calcificaciones en mamografía 2D

Uno de los problemas en el seguimiento del historial de mamografías de una paciente es poder conservar los datos de área y volumen de las formaciones calcáreas candidatas

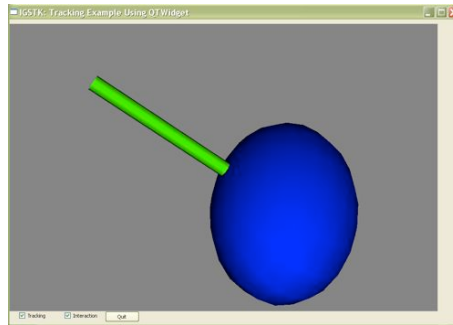


Figura 8.8: Ejemplo de aplicación de IgsTK

a carcinoma. De esta forma, una vez se tiene la mamografía, es interesante ser capaz de obtener de ella los datos del volumen de la formación detectada y almacenar dichos datos junto con la imagen.

El ejercicio propuesto consiste en desarrollar una sencilla aplicación que permita cumplir estos objetivos.

8.6.1. Indicaciones para la solución

El ejercicio consiste en desarrollar una aplicación con las siguientes capacidades:

- ser capaz de abrir y visualizar imágenes mamográficas. En el fichero anexo se han añadido como ejemplo las imágenes `mamografia.jpg` y `mamografia.png`, por lo que se recomienda desarrollar el código para ser capaz de trabajar al menos con ficheros `.jpg` y `.png`.
- Configurar la interacción 2D mediante el uso de un `vtkSmartPointer` de forma que nos permita seleccionar píxeles de la imagen.
- Seleccionar un área en la imagen de la izquierda de forma que podamos tener un dato de área de la formación cálcica.
 - Para este punto, una forma de actuar es utilizar un algoritmo de semillas de forma que el crecimiento de la semilla encuentre los límites de la formación.
 - Otra forma más trivial es aproximar dicha área a un rectángulo de dimensiones dadas por el usuario mediante el ratón. Esta es la solución programada en los ficheros anexos.
- Seleccionar en la imagen de la derecha la tercera dimensión de la formación cálcica. De igual forma, se pueden utilizar algoritmos de segmentación o simplemente obtener este dato mediante la interacción con el ratón.
- Calcular el volumen de la formación cálcica dadas las dimensiones obtenidas previamente. Almacenar una nueva imagen resultado en la que aparezcan marcadas la formación cálcica en las dos imágenes y el dato del volumen obtenido.

Como ejemplo explicativo del funcionamiento pretendido de la aplicación, se muestran a continuación una secuencia de imágenes de uso del programa anexo a este ejercicio:

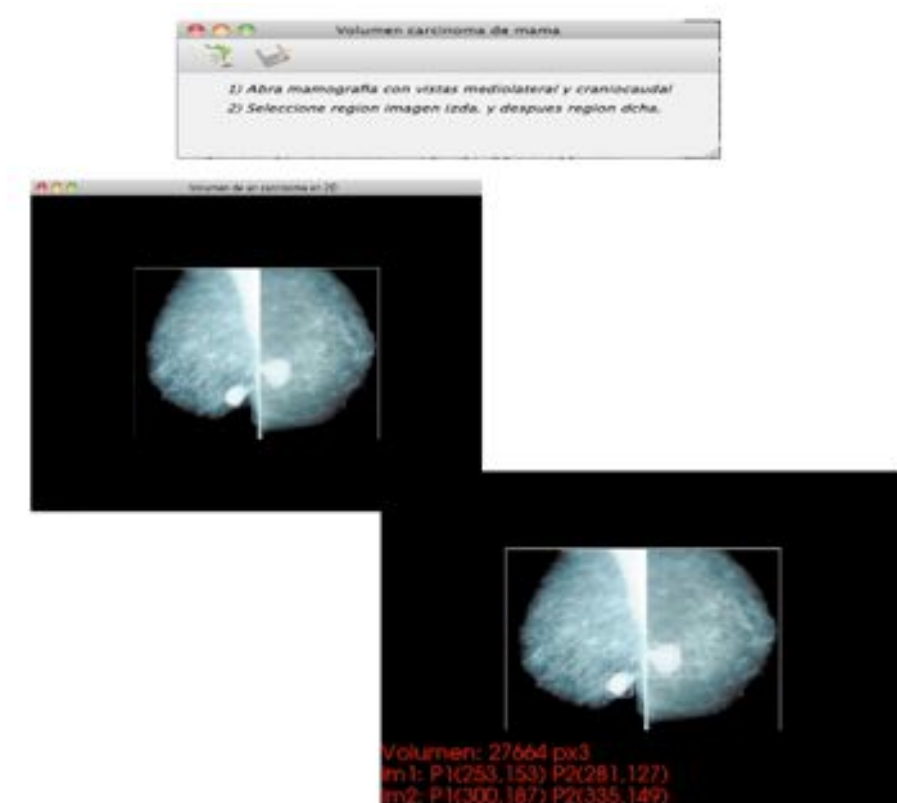


Figura 8.9: Aspecto de la solución del ejercicio

Una posible implementación de este ejercicio se encuentra resuelta en el cd-anexo a este libro.

Bibliografía para ampliar

The Visualization Toolkit Kitware, Inc. 4th edition (December 2006). ISBN-13: 978-1930934191. Libro de texto necesario para la comprensión de las librerías VTK.

VTK Users Guide Kitware, Inc. 11th Edition (March 2010). ISBN-13: 978-1-930934-23-8. Libro de texto necesario para la comprensión de las librerías VTK.

The ITK Software Guide Kitware, Inc. (November 2005). ISBN-13: 978-1930934153. Libro de texto necesario para la comprensión de las librerías ITK.

Webpage de Kitware . <http://www.kitware.com/opensource/opensource.html> donde se pueden encontrar multitud de ejemplos y material adicional para trabajar con sus librerías.

Capítulo 9

Simuladores médicos realizados en OpenSurg

“El hombre que quiera volar deberá antes aprender a andar”
Popular

Indice

9.1. Introducción	249
9.2. Herramientas para la simulación	250
9.2.1. OGRE	250
9.2.2. PhysX de NVIDIA	253
9.2.3. NxOgre	255
9.2.4. Blender	256
9.3. Entrenador de técnicas MIS	256
9.4. Simulador de un robot para tareas de colicestomía	257
9.4.1. Descripción de las trayectorias de colicestomía laparoscópica	257
9.4.2. Estructura funcional del robot diseñado para realizar operaciones de laparoscopia	257
9.4.3. Trayectorias para simulación	259
9.5. Simulador de traumatología. Operación de reemplazo total de rodilla	263

9.1. Introducción

El éxito que ha tenido el entrenamiento para aviación utilizando simuladores ha servido de ejemplo para conducir al desarrollo de simuladores quirúrgicos. Su desarrollo

también se ha dado como una solución a la problemática que tuvo en sus orígenes la cirugía mínimamente invasiva por la falta de entrenamiento apropiado de los cirujanos.

Numerosos estudios han confirmado la capacidad de los simuladores para la enseñanza de las habilidades psicomotoras básicas necesarias. Su ventaja mayor consiste en que puede aportar evaluación objetiva del desempeño del cirujano, la cual puede utilizarse para la retroalimentación durante el entrenamiento y la evaluación de las habilidades y su revalidación.

Los simuladores quirúrgicos interactúan con sistemas digitales de imagen médica, proporcionando una visualización de imágenes en 3D, además constan de un programa específico de entrenamiento para el control del sistema robótico. Estos simuladores ofrecen un impacto positivo en la formación de los cirujanos, mejorando enormemente la enseñanza práctica de la cirugía, ya que son una herramienta objetiva para comprobar y evaluar la destreza y el grado de aprendizaje quirúrgico de los cirujanos en formación.

Entre los requerimientos que deben tener los simuladores es que deben mostrar detalles precisos y ser altamente interactivos. Una simulación realista debe tener fidelidad, entrada sensorial, reactividad, interactividad y guardar las propiedades de los objetos.

En este capítulo del libro se va a presentar de manera reducida algunos de los trabajos realizados dentro de [OpenSurg](#). Por cuestión de espacio, en este capítulo sólo se muestran los resultados de estos trabajos, y las características básicas de los mismos, y se remite al estudiante a los **documentos adicionales** que en cada apartado son referenciados y que podrá encontrar en el **cd-anexo**. En estos documentos adicionales podrá encontrar detalles sobre la implementación software y hardware de las soluciones presentadas.

Por tanto, las siguientes secciones no deben ser leídas de forma secuencial, pues hacen referencia a diferentes trabajos realizados por diferentes grupos.

9.2. Herramientas para la simulación

En primer lugar se van a listar las herramientas comunes utilizadas en los desarrollos que posteriormente se presentan:

9.2.1. OGRE

OGRE (Object-Oriented Graphics Rendering Engine) es un motor de renderizado 3D que aprovecha las posibilidades de las tarjetas gráficas utilizando la aceleración por hardware y da soporte en el uso de **DirectX** y **OpenGL**. Es flexible y escrito en lenguaje C++, además de ser multiplataforma ya que permite la creación de aplicaciones para entornos como Windows, Linux o Mac. Esta herramienta está totalmente orientada a objetos por lo que su interfaz es clara, intuitiva y fácil de usar, diseñada para que el renderizado de la escena 3D requiera poco esfuerzo de manera eficiente. Su arquitectura está basada en plugins para extender las funcionales en caso de ser necesarias, como sonido o física.

El motor es un proyecto de código abierto bajo licencia LGPL (Lesser General Public License), lo que significa que su uso es gratuito y no necesita muchas exigencias para su

uso. También tiene una comunidad muy activa de desarrolladores y existe gran cantidad de tutoriales, ayudas o foros con mucha documentación sobre la API.

Antes de empezar a programar el simulador quirúrgico, hay que conocer los objetos principales que se utilizarán en Ogre. La figura 9.1 muestra un diagrama con algunos de los objetos más significativos.

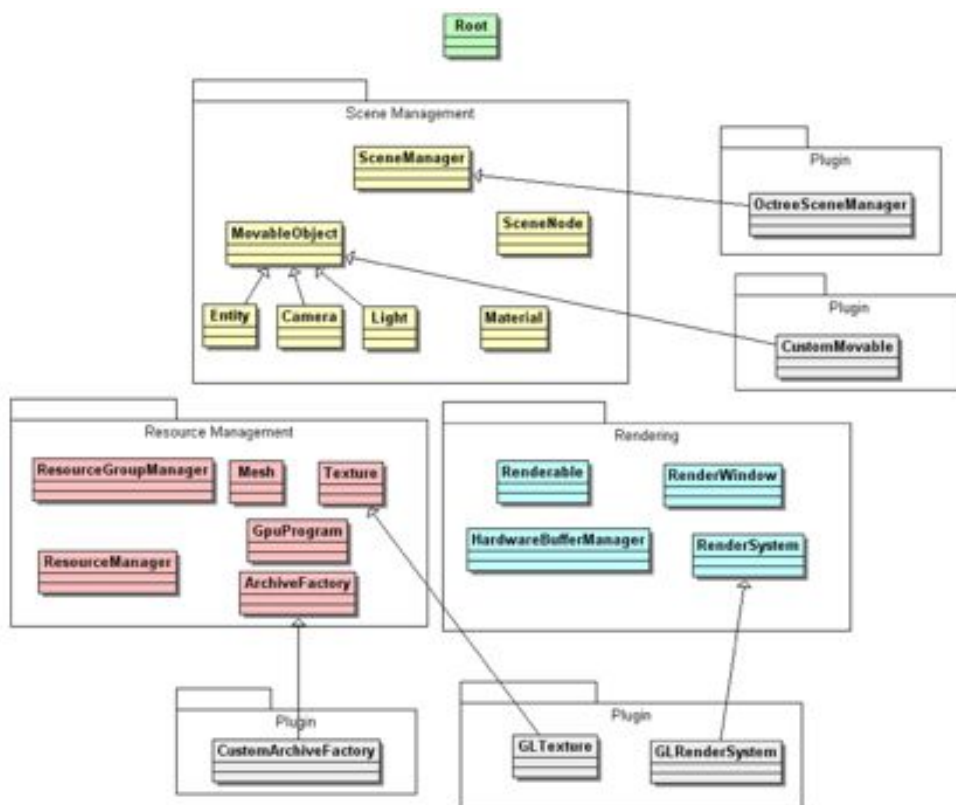


Figura 9.1: Esquema general de componentes Ogre.

En la parte superior está el objeto raíz (*Root*). Es el punto de entrada al sistema Ogre encargado de crear los objetos de nivel superior como los directores de escena (*Scene Managers*), sistemas de renderizado y las ventanas de renderización, carga plugin y todas las cosas fundamentales. El resto de las clases entran en una de las siguientes funciones.

9.2.1.1. Gestión de escenario

Informa sobre el contenido del escenario. Las clases principales son:

Gestor de escena (*SceneManager*): Es el objeto más utilizado por la aplicación porque contiene todas las referencias y propiedades de la escena que va a ser dibujada

por el motor de gráficos. Permite crear y controlar las cámaras, objetos móviles (entidades), luces y materiales (propiedades de la superficie de los objetos), y el mundo geométrico estático (partes inmóviles de la escena). Existen diferentes *SceneManager* especializados para utilizar el más adecuado de acuerdo a la funcionalidad de nuestra escena y así obtener más rapidez a la hora de visualizar la misma.

Entidades (*Entity*): Son los objetos de la escena que están representados por mallas 3D, caracterizadas por una lista de vértices y una lista de caras formadas por índices al listado de vértices. Las entidades no se consideran parte de la escena hasta que no es añadida a un nodo de la misma, por lo que la entidad representa la idea de la visualización del objeto.

Nodo de la escena (*SceneNode*): Los *SceneNode* proporcionan una posición, orientación y escalado a un objeto dentro de la escena, siendo las transformaciones relativas al nodo y no al objeto. Tienen la función de ser contenedores de otros nodos, luces, cámaras o entidades. Además, se pueden crear jerarquías de nodos creando relaciones complejas en forma de árboles y pudiendo realizar una transformación a un nodo padre afectando a todos sus hijos.

Material: Esta clase controla cómo se renderizan los objetos de la escena estableciendo las propiedades de la superficie.

9.2.1.2. Gestión de recursos

Todo el renderizado necesita recursos como la geometría, las texturas, las fuentes, etc. Las clases de este tipo de funcionamiento gestionan la carga, reutilización y la descarga de estos elementos:

ResourceGroupManager: Es la clase encargada de leer todos los grupos de recursos que pueden ser cargados y descargados en cualquier momento (mallas, animaciones, materiales, texturas) desde algún directorio para proporcionar a Ogre los datos necesarios. Hay una serie de *ResourceManagers* que gestionan los distintos tipos de recursos como *TextureManager* o *MeshManager*. De esta manera se garantizan que los recursos sólo se cargan una vez y se gestionan los requisitos de memoria de los recursos a tratar. Existe un fichero de configuración externo llamado `resources.cfg`, en el que se especifican dónde están ubicados los recursos.

Mesh: Este objeto representa un modelo discreto de malla con un conjunto de geometría. Se utiliza para representar objetos móviles dentro de la escena como pueden ser personajes, objetos, etc. Generalmente son creados en herramientas de modelado 3D y luego son exportados a un archivo `.mesh` que utiliza Ogre para recrear el objeto y son la base de los objetos móviles llamados Entidades.

9.2.1.3. Renderización

Para finalizar se ponen las imágenes en la pantalla. Se trata del nivel más bajo del canal de renderizado. La clase más importante es:

RenderSystem: Esta clase es la responsable de enviar las operaciones de renderizado y del establecimiento de todas las diferentes opciones de renderizado. Hace de interfaz entre Ogre y la API de render utilizada (DirectX u OpenGL).

9.2.2. PhysX de NVIDIA

PhysX es un motor físico que enlaza la utilización de cuerpos rígidos dinámicos y la detección de colisiones en una biblioteca fácil de usar. El componente dinámico de cuerpos rígidos permiten al usuario simular objetos con un alto grado de realismo. Hace uso de conceptos físicos como los puntos de referencia, posición, velocidad, aceleración, momento, fuerzas, movimiento rotacional, energía, fricción, impulso. colisiones, restricciones, con el objetivo de construir diferentes tipos de dispositivos mecánicos.

PhysX de NVIDIA permite crear efectos físicos en tiempo real para la simulación en aplicaciones. La simulación de movimiento de objetos en una escena con gravedad y fricción, el cálculo de colisiones y la creación de formas complejas articuladas son algunas de las funcionalidades más importantes. Es de comercialización gratuita pero de código cerrado, con lo que no se puede conocer la implementación de las funciones que componen el motor. Tiene una interfaz ANSI C++ implementada como una jerarquía de clases que contienen funcionalidades accesibles y cálculos específicos al hardware usado.

Las unidades para la masa, el tamaño y el tiempo son por defecto los kilogramos, los metros y los segundos, aunque pueden ser modificados. El motor realiza cálculos de valores en coma flotante de 24 bits, con lo que la precisión del motor también dependerá de que unidades se haya escogido para trabajar. El siguiente diagrama (fig 9.2) muestra la arquitectura del motor:

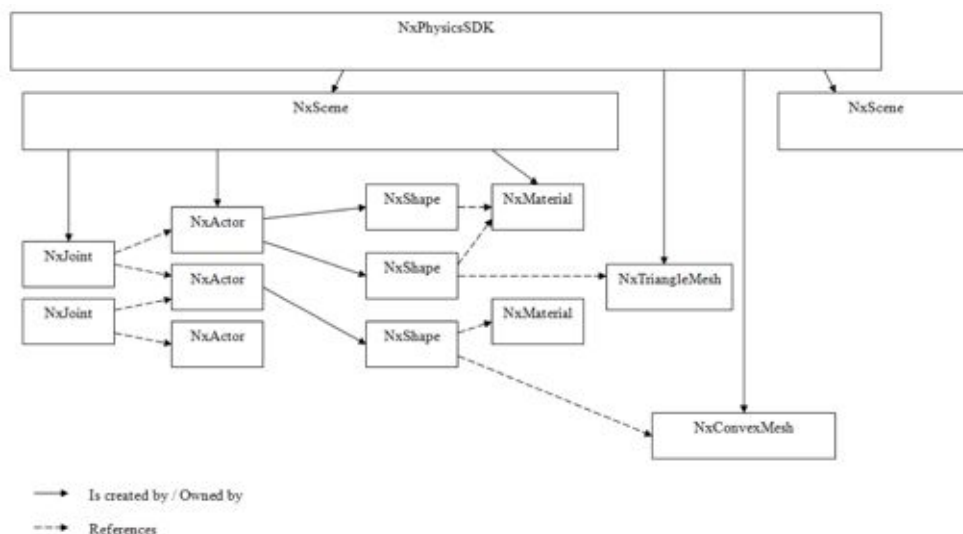


Figura 9.2: Esquema general de componentes PhysX.

Al igual que Ogre, PhysX tiene un punto de entrada para inicializar el motor, la clase *NxPhysicsSDK*. Antes de realizar una simulación, se crea una instancia de la esta clase para inicializar el sistema y poder ajustar parámetros, como poder visualizar los ejes de un actor (XYZ) para saber cómo está orientado o hacia donde se está aplicando una fuerza. Una vez que el sistema se ha inicializado y ajustado, *NxPhysicsSDK* se encarga de crear una instancia de la clase *NxScene*. Puede tener una o más escenas.

NxScene, de la misma forma que Ogre, define una escena donde se colocan los objetos, en este caso se llaman Actores, en un ambiente 3D que permite observar el comportamiento de estos objetos según las condiciones físicas establecidas. La escena es un contenedor de las simulaciones de los actores. Al momento de crear una escena se debe especificar la gravedad que afectará a los objetos en la escena mediante un descriptor de escena (instancia de *NxSceneDesc*). Los descriptores son estructuras muy usadas en PhysX y contienen toda la información sobre las características que se le van a dar al objeto en el momento de crearlo.

Las escenas contienen actores, los cuales son instancias de la clase *NxActor*. Los actores son los objetos de simulación que existen en una escena del motor físico e interactúan unos con otros. La clase *NxScene* es la encargada de crear todos estos actores. PhysX emplea tres tipos de actores:

Actores estáticos: Son objetos fijados en un punto de referencia de la escena, los cuales no se moverán nunca. Su principal propósito es la detección de colisiones y pueden ser utilizados para representar suelos, montañas, intersecciones, etc. Se consideran actores que poseen una masa infinita y no son afectados por fuerzas.

Actores dinámicos: Son actores que se pueden mover y actuar bajo la influencia de fuerzas o velocidades. Tienen una masa finita.

Actores cinemáticos: Es un tipo especial de actores dinámicos que no se mueven directamente por la aplicación de fuerzas, impactos o por acción de la gravedad. Pueden ser posicionados dentro de la escena según convenga.

El tipo de actor debe ser especificado en el momento de su creación, de la misma manera que la escena, mediante la utilización de un descriptor de actor, instancia de la clase *NxActorDesc*. Con este descriptor también se definen otros tipos de características como su forma, densidad o cuerpo. Los objetos dinámicos pueden ser convertidos en cinemáticos y viceversa, mientras que los estáticos no pueden ser modificados de tipo.

Estos actores están formados por un cuerpo rígido (lo que PhysX denomina *Body*), y una forma de colisión (*Shape*). Las formas de colisión representan el espacio físico que el actor ocupa en la simulación y es el núcleo de la detección de colisiones. Es la forma que tomará el actor y se comportará en consecuencia. Toda esta información está contenida dentro del descriptor de actor, y si a un actor no se le asigna un cuerpo rígido (*body*) se considera de tipo estático. La manera de asignar un cuerpo rígido a un actor es a través de la creación de un descriptor de cuerpo, mediante una instancia de la clase *NxBodyDesc*.

La clase *NxShape* define las formas que se pueden utilizar con PhysX, las cuales pueden ser simples como planos, cajas, esferas, capsulas, ruedas o más complejas como mallas de triángulos (*mesh*). Como en casos anteriores, para crear las formas se deben utilizar descriptores de forma, instancias de la clase *NxShapeDesc*, que se encargan de

especificar las características como dimensiones y los materiales que tendrá asociado. Un actor puede tener varias formas asociadas. Tiene algoritmos muy eficientes para algunas formas simples (y para comprobar las colisiones entre ellos), aunque también pueden crearse formas más complejas de mallas de triángulos con un pre-proceso para crear la representación interna a partir de herramientas modeladoras desde donde se pueden exportar a un formato que puede ser leído por PhysX. También se pueden tener uniones de distintos tipos entre dos actores (*NxJoint*).

La clase *NxMaterial* determina las propiedades internas físicas de los objetos (actores o más directamente las formas) como la restitución y propiedades de la superficie como fricciones, mientras que en Ogre los materiales están más relacionados con la apariencia. Los materiales son creados por el objeto de la escena y es necesario ajustar sus propiedades dando valores a su restitución, fricción estática y fricción dinámica. Una forma puede usar un material creado así como también puede dejar de usarlo y cambiarlo por otro en cualquier momento.

El SDK también incluye clases matemáticas para la creación de vectores, matrices, operaciones entre estas y algunas funciones y macros. También incluye un SDK para la gestión de personajes y otro para el *cocinado* de mallas (creación de la estructura interna de una malla). El hecho de elegir este motor físico para el desarrollo del simulador se basa en su potencia de simulación y sus facilidades de uso e integración, destacando por su rapidez y robustez. Otra razón es que posee herramientas que facilitan su compatibilidad con [Ogre](#) trabajando con [NxOgre](#), mientras que existe una gran disponibilidad de tutoriales y foros activos como soporte en el aprendizaje.

9.2.3. NxOgre

[NxOgre](#) es una biblioteca de software que permite integrar PhysX con Ogre 3D envolviendo todas las funciones y clase de PhysX, actuando como intermediario para poder aprovechar la potencia de simulación física de PhysX y la potencia de renderizado de Ogre.

Es una biblioteca de código abierto sobre la licencia LGPL. Como NxOgre es un envoltorio que integra a Ogre con PhysX es posible acceder, a través de él, a ambos motores, gráfico y físico, dando la posibilidad de implementar aplicaciones con el potencial de cada uno. NxOgre está escrito en lenguaje de programación C++ y es orientado a objetos. Posee muchas clases que envuelven a las clases definidas para Ogre y PhysX.

9.2.3.1. Critter

[Critter](#) es la biblioteca que se encarga de manejar el sistema de renderizado de NxOgre. Enlaza los conceptos de renderización de Ogre creando entidades y al mismo tiempo crea un actor para PhysX. De esta manera se realizan simulaciones físicas con el actor y representar esto gráficamente a través de la entidad creada para Ogre.

9.2.3.2. Flour

Para la integración de mallas físicas se hace uso de un programa llamado Flour, que transforma el modelo creado por el usuario con un programa de modelado gráfico, a una

malla convexa o de triángulos (en principio se usa Blender). Convierte diferentes tipos de formatos para poder trabajar con NxOgre. Como el formato de Ogre mesh a mallas de triángulos, esqueletos o cuerpo convexos.

9.2.4. Blender

Todo simulador en 3D requiere de la creación de objetos en tres dimensiones que forman la parte visual. Para crear elementos 3D y exportarlos al motor gráfico de Ogre se ha utilizado el software de diseño [Blender](#). Blender es un programa multiplataforma dedicado al modelado, animación y creación de gráficos tridimensionales, otorgando gran facilidad para crear objetos complejos. Es totalmente gratuito y de código fuente libre, con una interfaz gráfica de usuario muy característica.

Por otra parte, exportar los objetos y animaciones a Ogre desde Blender es muy sencillo y las aplicaciones para esta tarea están muy depuradas, ya que se usan más que las aplicaciones para exportar desde otras herramientas de diseño 3D.

9.3. Entrenador de técnicas MIS

En el cd-anexo se puede encontrar los documentos “*Desarrollo de un simulador virtual para la cirugía de mínimo acceso*” y “*TutorialOgreMedicina version1.3*”, generados por el grupo de la *Universidad Central de la Villas, Cuba* y la empresa *SimPro*. En estos documentos se muestran los detalles de la implementación de un entrenador para técnicas MIS formado por dos dispositivos Phantom Omni adaptados a la configuración típica de las operaciones MIS y un software construido sobre Ogre y Physx que permite adquirir destreza en el manejo de las pinzas quirúrgicas de MIS. La figura 9.3 muestra uno de los posibles ejercicios que se pueden realizar con este simulador.



Figura 9.3: Entorno de simulación, instrumentos y ejercicio para el desarrollo de la coordinación psicomotora

Se recomienda al estudiante visitar la página roboticamedica.umh.es donde puede encontrar vídeos ilustrativos con diferentes ejercicios hechos con este simulador.

9.4. SIMULADOR DE UN ROBOT PARA TAREAS DE COLICESTOMÍA

9.4. Simulador de un robot para tareas de colicestomía

Este trabajo, realizado en la Universidad del Cauca, Colombia, muestra los pasos a realizar en la síntesis de la cinemática de un robot quirúrgico. Los detalles del mismo se encuentran en el documento *Simulador de un robot para tareas de colicestomía* y pueden ser consultados para clarificar lo aquí mostrado.

9.4.1. Descripción de las trayectorias de colicestomía laparoscópica

La colecistectomía laparoscópica es una cirugía abdominal que se realiza bajo anestesia general, donde se hacen cuatro pequeñas incisiones sobre el abdomen del paciente. La primera que generalmente mide 10 mm se practica cercana al ombligo y las tres restantes, denominadas puertos, en la parte superior derecha del abdomen, con medidas que varían entre los 5 y los 10 mm.

Por la primera incisión se introduce el laparoscopio, que es un instrumento que lleva una pequeña cámara de video y una fuente de luz con el objetivo de proyectar lo que sucede en el interior del paciente. También es utilizada esta incisión para insuflar el abdomen (inyectar gas carbónico) con el fin de obtener espacio para realizar la intervención quirúrgica. Las dos incisiones ubicadas a la derecha del abdomen miden 5 mm y se utilizan para introducir pinzas con el fin de mover el hígado, la vesícula y sus conductos para dar mayor visibilidad, mientras que por la cuarta incisión que varía de 5 a 10 mm se introducen los instrumentos para cortar, disecar y coagular, necesarios para esta operación. Generalmente la colecistectomía se divide en cinco pasos o etapas:

1. Exposición del conducto y arteria císticos.
2. Disección del triángulo de Calot.
3. Sección del conducto y la arteria císticos.
4. Disección de la vesícula del lecho hepático.
5. Extracción de la vesícula biliar.

9.4.2. Estructura funcional del robot diseñado para realizar operaciones de laparoscopia

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 en el 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 notar 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. 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 nuevo robot se escogió una estructura de nueve grados de libertad, con seis articulaciones activas que permitirán posicionar y orientar el instrumento quirúrgico en un espacio tridimensional; y tres pasivas, las cuales permiten mantener el punto fijo de movimiento sobre el punto de incisión donde se encuentra el trocar. Con estos tres grados de libertad pasivo 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 9.4 se puede observar la estructura cinemática que se diseñó para el robot, llamado **LapBot**.

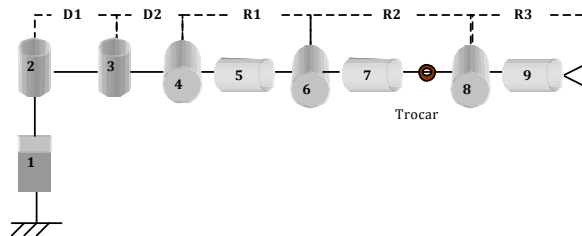


Figura 9.4: Estructura cinemática del robot LapBot.

Las distancias del robot se han etiquetado con letras D y R, la articulación prismática se representa con un prisma y las rotacionales 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 siete y ocho, de tal forma que sólo dos articulaciones quedan dentro de la cavidad abdominal del paciente.

Las tres primeras articulaciones permiten posicionar a LapBot en un espacio tridimensional, la articulación cuatro se fija en un valor determinado con el fin de que el robot se incline y se ubique sobre el 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 corresponden a la orientación del efector final.

9.4. SIMULADOR DE UN ROBOT PARA TAREAS DE COLICESTOMÍA

9.4.3. Trayectorias para simulación

Se realizaron la construcción de las trayectorias cartesianas para las etapas 2, 3 y 4 de la colecistectomía laparoscópica, utilizando videos proporcionados por la Fundación Clínica Valle del Lili de la ciudad de Cali, Colombia. Estos videos fueron enlazados 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 9.5 se puede observar el ambiente tridimensional desarrollado, el video y el código para exportar los datos de Blender a un archivo de texto.

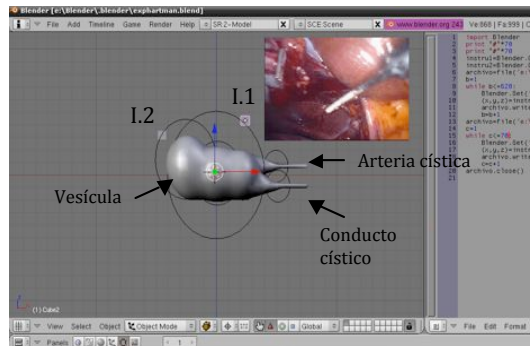


Figura 9.5: Entorno de trabajo en Blender, plano XZ

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.

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). La figura 9.6 muestra la trayectoria. Para I2, el movimiento está definido

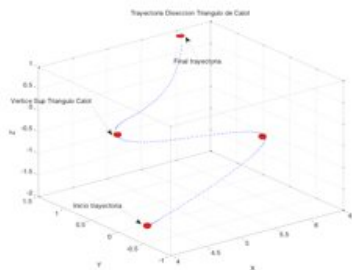


Figura 9.6: Trayectoria del instrumento 1 para la disección del triángulo de Calot.

por la tracción de la vesícula. Es importante resaltar que durante todo el procedimiento quirúrgico este movimiento puede repetirse varias veces. Este movimiento se muestra en la figura 9.7.

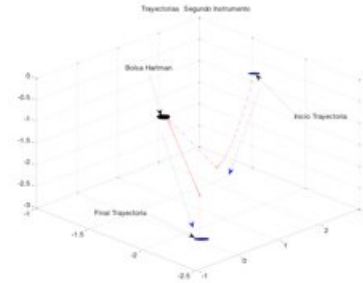


Figura 9.7: Trayectoria del instrumento 2 para la disección del triángulo de Calot.

2. Sección del conducto cístico. En esta etapa I2 repite el movimiento de la 9.8, pero I1 tiene la trayectoria de 70 puntos que se presenta en la figura 9.8.

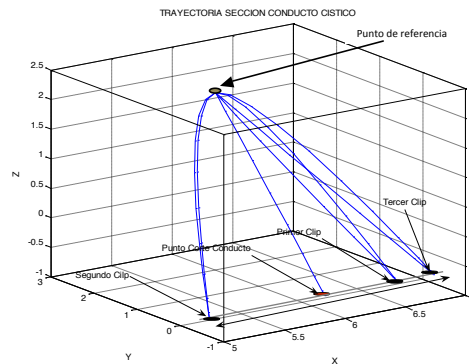


Figura 9.8: Trayectoria del instrumento 1 para el corte del conducto cístico.

3. Sección de la arteria cística. El procedimiento para el corte de la arteria cística es similar al corte del conducto. La 9.9 muestra en rojo el punto de corte y en azul los clips.
4. Disección de la vesícula del lecho hepático. Se ingresa el instrumento I1 con un electrocauterio y simultáneamente I2 realiza la tracción de la 9.10. Luego se realizan 480 puntos de movimiento en zigzag, concluyendo la trayectoria.

9.4. SIMULADOR DE UN ROBOT PARA TAREAS DE COLICESTOMÍA

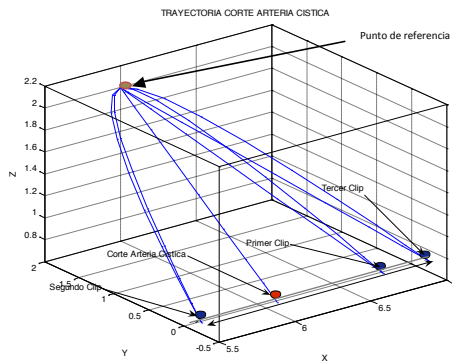


Figura 9.9: Trayectoria del instrumento 1 para el corte de la arteria cística.

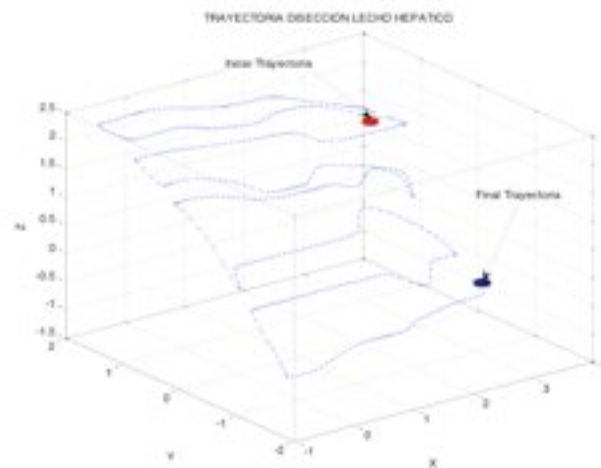


Figura 9.10: Trayectoria del instrumento 1 para la disección de la vesícula del lecho hepático.

9.4.3.1. 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 interfaz permite seleccionar entre las cuatro trayectorias mencionadas en el aparta-

do anterior (Diseción de Calot, Conducto Cístico, Artería Cística, Lecho Hepático), y se adicionó otra que es la Colectomía, la cual contiene todas las trayectorias necesarias para realizar la operación total.

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). 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 y uno de los modelos para hacer la simulación. 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 y el GUI. El ambiente 3D con los dos LapBot, el abdomen, el GUI y los demás elementos se presentan en la 9.11.

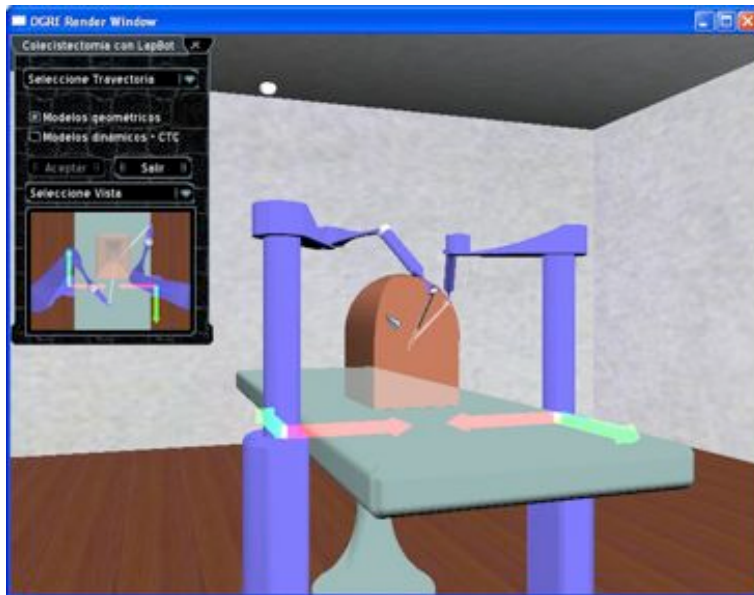


Figura 9.11: Ambiente 3D con dos LapBot y la interfaz de usuario.

Cuando el robot se comienza a mover, se puede observar la trayectoria que está siguiendo dentro del abdomen dado que ésta se remarca con un color azul oscuro, como se observa en la 9.12.

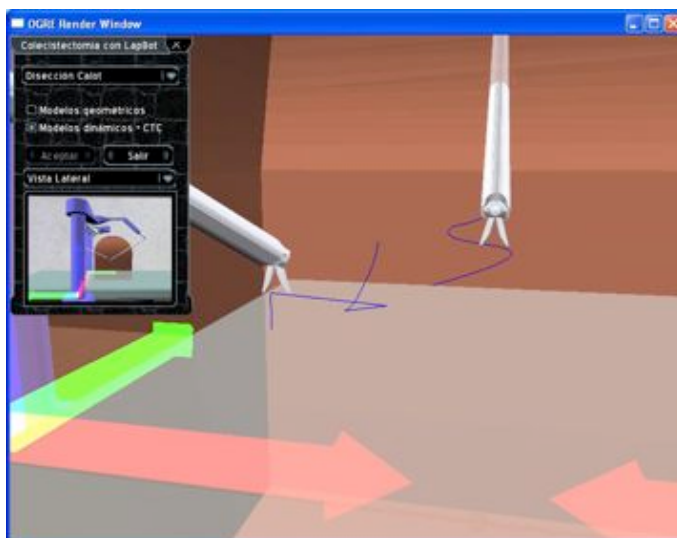


Figura 9.12: Seguimiento 3D de la trayectoria Dissección de Calot.

9.5. Simulador de traumatología. Operación de reemplazo total de rodilla

Este trabajo fue desarrollado en la Universidad Miguel Hernández durante la estancia de Dña. Maria Luisa Pinto, de Colombia. Los detalles de la implementación pueden encontrarse en el artículo publicado en la revista [Trauma](#), de la Fundación Mapfre [65].

El objetivo de esta pasantía fue desarrollar un simulador de cirugía ortopédica para las operaciones de reemplazo total de rodilla (TKR) que sirve además como base para el estudio de otras intervenciones como reemplazo total de cadera (THR).

En el citado trabajo se describe una aplicación basada en software de código abierto que integra el modelado de objetos óseos y tejidos blandos con el procesamiento de imágenes médicas y la tecnología háptica para aumentar las propiedades de inmersión. El sistema está compuesto por dos etapas, primero se analizan imágenes médicas de una articulación, y se establece un plan intraoperatorio para el seccionado de tejidos. En una etapa posterior, el usuario integra la información de los huesos seccionados para interactuar con modelos virtuales de prótesis e instrumental quirúrgico, con la adición de sensaciones de contacto.

La principal contribución de este trabajo es la *generación de un modelo mixto superficial-volumétrico que permite simular el maquinado de un hueso mediante la separación de secciones definidas por el usuario* como: sección superficial sobre la cual no se proponen deformaciones o modificaciones (cortes, taladrado, fresado); y sección volumétrica compuesta por primitivas superficiales (puntos, esferas, o cubos) con propiedades de transformaciones geométricas y reportes de colisión independiente.

Las características de este trabajo denominado **SITKHR** (Simulación de Interven-

ciones TKR y THR) son:

- Visualización de imágenes radiográficas en formato DICOM (Digital Imaging and Communication in Medicine), en planos anatómicos diferentes con marcación de puntos de referencia anatómicos.
- Herramienta para el corte de imágenes en cualquier plano anatómico o una sección específica de acuerdo a las marcas del usuario.
- Herramienta para la obtención de un modelo mixto superficial/volumétrico en cualquier zona, eje y dimensión de la imagen DICOM.
- Asignación morfológica de tamaños para las partículas elementales que constituyen el modelo volumétrico para su aplicación en el renderizado háptico.
- Reconstrucción de tejidos específicos a partir de la imagen DICOM, con la posibilidad de exportarlos en diferentes formatos de visualización gráfica tridimensional.
- Simulación del mecanizado de un hueso a partir de la exploración y reconstrucción de la imagen DICOM de una zona específica de la cadera, el fémur o la rodilla.
- Realimentación de fuerzas en las herramientas de exploración y mecanizado, mediante la integración de la interfaz háptica Phantom de Sensable Inc.
- Desarrollo de simulaciones de cortes e interacción con los módulos de eliminación de partículas y de renderización de fuerzas en función de las superficies que entran en contacto.

La figura 9.13 muestra una imagen del sistema completo en funcionamiento, y las imágenes 9.14 y 9.15 muestran distintas instantáneas del simulador trabajando.



Figura 9.13: Componentes del sistema Sitkhr

El detalle de la implementación, así como el código fuente puede encontrarse en el correspondiente directorio del cd-anexo.

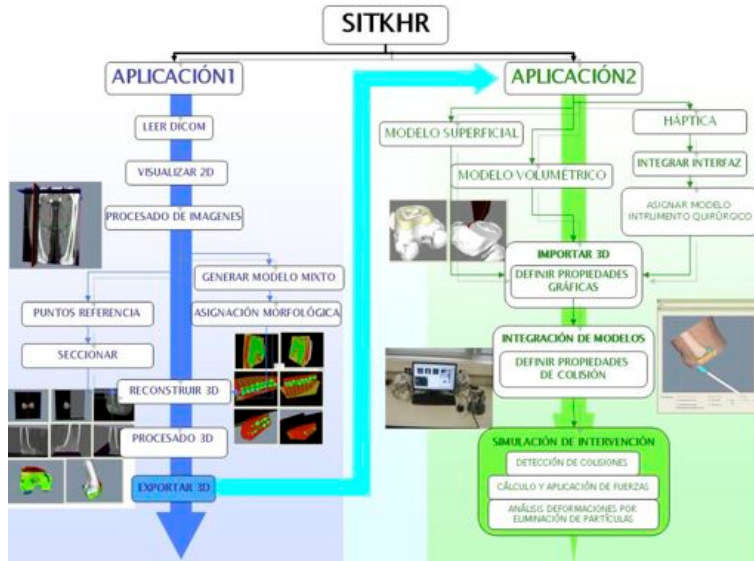


Figura 9.14: Esquema general de los módulos del sistema Sitkhr

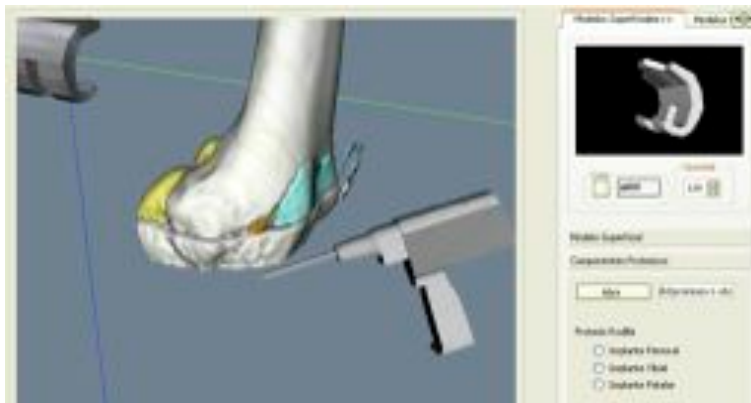


Figura 9.15: Instantánea del sistema Sitkhr

Capítulo 10

Aplicaciones de robótica médica

“Las ciencias aplicadas no existen, sólo las aplicaciones de la ciencia”
Louis Pasteur

Índice

10.1. Introducción	267
10.2. Sistema de bajo coste para rehabilitación de la mano	268
10.3. Guante háptico sensorizado	270
10.4. Robot de 2 grados de libertad para el fresado maxilofacial	271

10.1. Introducción

En este capítulo se van a presentar de manera resumida algunos trabajos que incluyen hardware y que han sido realizados con relación a las tareas que se desarrollaron dentro de **OpenSurg**.

Al igual que en el capítulo anterior de simuladores, en este texto se presentan sólo las características de dichos trabajos, y el detalle de la implementación de los mismos debe ser consultado en los documentos anexos a cada sección. Toda esta información se encuentra en el **cd-anexo** a este libro.

Para completar la información aquí presentada y tener las últimas actualizaciones y nuevos proyectos que puedan surgir fruto de la sostenibilidad de la acción OpenSurg, se recomienda visitar el sitio web roboticamedica.umh.es, donde se mantienen las últimas versiones de éstas y otras aplicaciones open-source.

10.2. Sistema de bajo coste para rehabilitación de la mano

Este trabajo presenta un sistema pensado para la rehabilitación de mano. El trabajo fue realizado por los estudiantes **D. Luis Eduardo Camayo Gómez** y **D. Daniel Ricardo Ramos Tovar**, estudiantes de la Universidad del Cauca durante su estancia de 3 meses en la Universidad Miguel Hernández. Todos los detalles de la implementación del sistema, incluyendo los anexos, se pueden encontrar en el trabajo de maestría presentado por los estudiantes, titulado *“Diseño y simulación de una mano robot con actuadores de Nitinol para rehabilitación”*, y que se encuentra anexo en el cd.

Para la realización de este trabajo se tuvo que hacer un estudio antropométrico y cinemático de la movilidad de la mano humana, modelando las articulaciones con las herramientas vistas en el capítulo 2, prestando especial interés en los ángulos posibles en cada articulación. Posteriormente se hace un estudio experimental del comportamiento de los alambres y muelles de NiTi (nitinol) con objeto de establecer la viabilidad de su uso como sistema de actuación. La figura 10.1 muestra algunos de estos experimentos, en los que se concluyó que un sistema antagonista de muelles permitía el rango de movimiento necesario para esta aplicación.

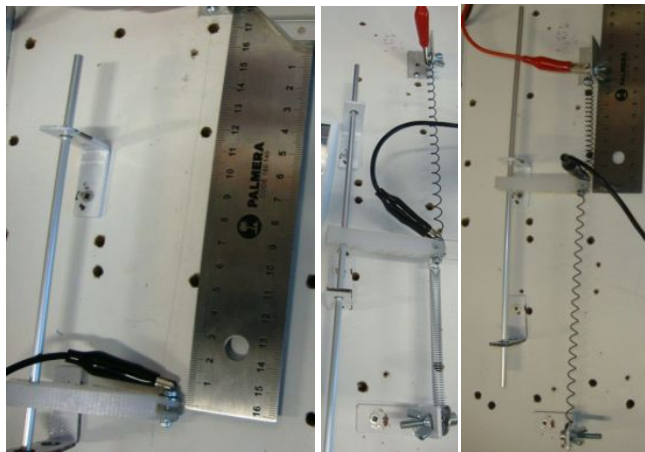


Figura 10.1: Maquetas de experimentación con alambres y muelles de Nitinol

Una vez completados estos pasos se realizó el diseño mecánico de las piezas, el diseño del sistema electrónico y un software de usuario y realidad virtual que permitiera trabajar con el sistema.

Los actuadores utilizados son muelles de Nitinol (un material con memoria de forma) y el control se ha realizado utilizando una tarjeta open-electronics de Arduino. La principal virtud de este prototipo es la capacidad de regular el movimiento de cada dedo de forma independiente y la seguridad que implica el uso de muelles como sistema de actuación. Otra de las características de este dispositivo es la facilidad para adaptarlo a la patología del paciente, de forma que se puede utilizar con casi todas las manos. El sistema incorpora

un software de control y realidad virtual con ejercicios terapéuticos sencillos. Los detalles de la implementación de este sistema se pueden ver en [67].

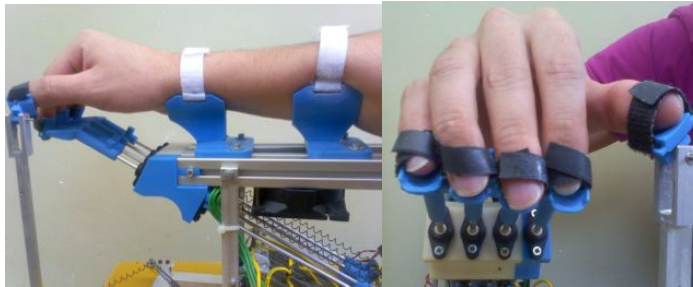


Figura 10.2: Primer prototipo de i-phalanx

La siguiente figura 10.3 muestra una comparación entre el mecanismo real con la mano de un paciente real y la simulación CAD del mecanismo. Se propone al estudiante consultar los vídeos de funcionamiento del sistema en roboticamedica.umh.es o en el cd adjunto.

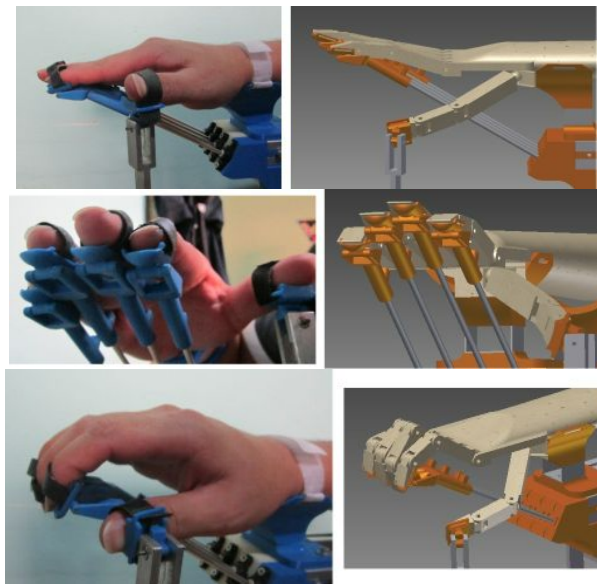


Figura 10.3: Comparación entre el mecanismo real y el modelo CAD

10.3. Guante haptico sensorizado

Este trabajo presenta un sistema pensado para la rehabilitación de mano. El trabajo fue realizado por los estudiantes **D^a. Mónica Rocío Díaz** y **D. Jose Manuel Escobar**, estudiantes de la Universidad del Cauca durante su estancia de 3 meses en la Universidad Miguel Hernández.

Todos los detalles de la implementación del sistema, incluyendo los anexos, se pueden encontrar en los documentos anexos “*Manual de usuario VirtualTouch.pdf*” y “*Virtual Touch.pdf*”.

Se trata de guante sensorizado de bajo coste. El guante incorpora un sensor *flexsensor* para cada dedo que permite estimar la posición del mismo. Además, incorpora unos vibradores de pequeño tamaño en la yema de cada uno de los dedos (2 en los casos del pulgar e índice), que permiten informar al usuario cuando se produce un contacto con un objeto virtual o cuando hay un deslizamiento entre el objeto virtual y los dedos índice y pulgar. Se tiene la capacidad de variar la frecuencia de vibración de estos actuadores.

Una característica interesante de este trabajo es que el sistema completo se ha integrado dentro de **ROS** (Robot Operating System), que es un meta-sistema operativo muy extendido para la investigación con robots, de forma que su manejo es sencillo. Para ello se ha conectado la tarjeta Arduino con el sistema ROS a través del paquete `rosserial_arduino`, tal como muestra la figura 10.4.

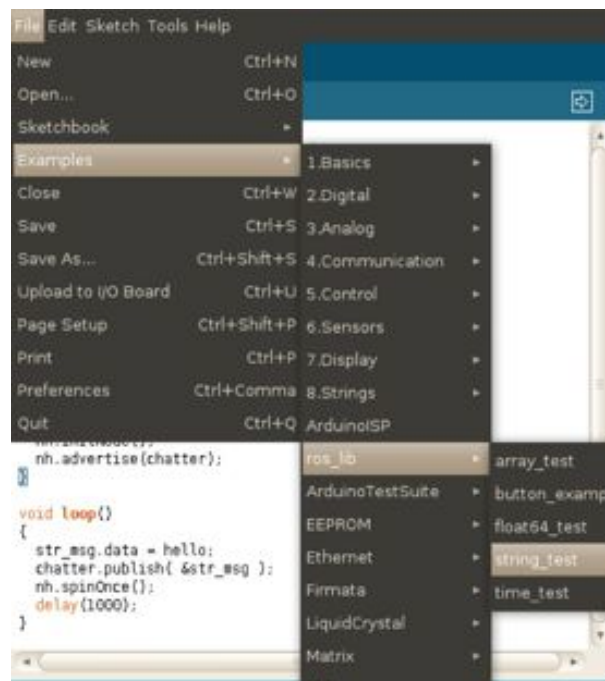


Figura 10.4: Arduino IDE con ROS

Haciendo uso de las capacidades de ROS, se presenta la interacción del guante sensorizado con un modelo de mano robótica (con la Shadow Hand). En la figura 10.5 se observa el primer prototipo de este guante.



Figura 10.5: Primer prototipo de guante sensorizado haptico

10.4. Robot de 2 grados de libertad para el fresado maxilofacial

Este trabajo presenta el desarrollo llevado a cabo entre la Universidad Tecnológica de Panamá, la Universidad Politécnica de Madrid y la Universidad Miguel Hernández con objeto de diseñar un mecanismo robótico de una estructura paralela, que será usado como muñeca en un brazo robótico para cirugías. Los detalles de este trabajo pueden encontrarse en los trabajos de D. José Rolando Serracin como el anexo en el cd (“*robot 2gdl para fresado.pdf*”) o la referencia [75].

La plataforma 2UPS-1U (figura 10.6) es un mecanismo paralelo de dos grados de libertad que será capaz de orientarse y ubicarse sobre la cabeza o cara del paciente, para poder realizar tareas quirúrgicas de gran precisión como lo son:

- incisiones biseladas en la piel, manteniendo siempre una posición perpendicular a la superficie cutánea, con precisión, exactitud y de un solo trazo, evitando pequeños jirones que podrían ocasionar quistes de inclusión o problemas en la cicatrización.
- incisiones no biseladas en zonas pilosas con una suficiente inclinación paralela hacia la dirección del pelo, que respete los folículos pilosos y permita el nacimiento de pelo en la cicatriz.
- acciones de fresado, colocación de tornillos sobre partes óseas, colocación de fijaciones en el hueso maxilar insertadas con un torque que cumpla con las especificaciones del sistema Branemark (entre 35 y 50 Nm), etc.

La figura 10.7 muestra la implementación real del robot y la simulación del mismo. Una característica importante de este sistema es su robustez y la capacidad de estimar

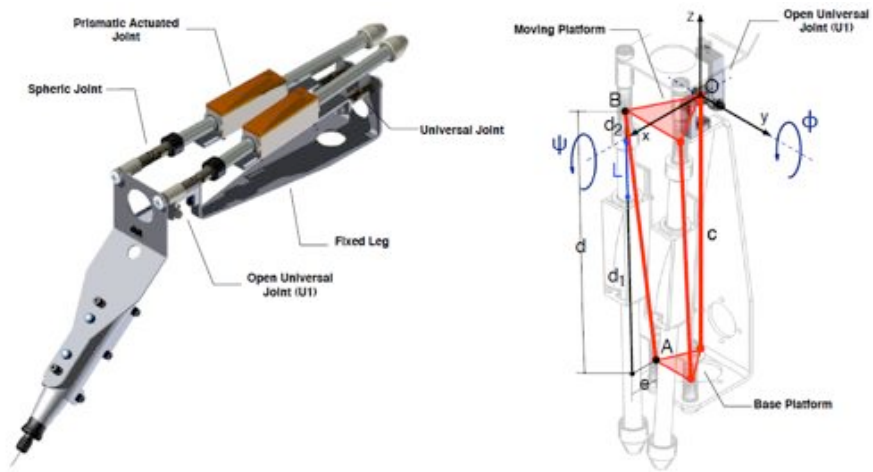


Figura 10.6: Robot 2UPS1U para cirugía maxilofacial.

las fuerzas de mecanizado a partir de las corrientes medidas en cada uno de los motores.

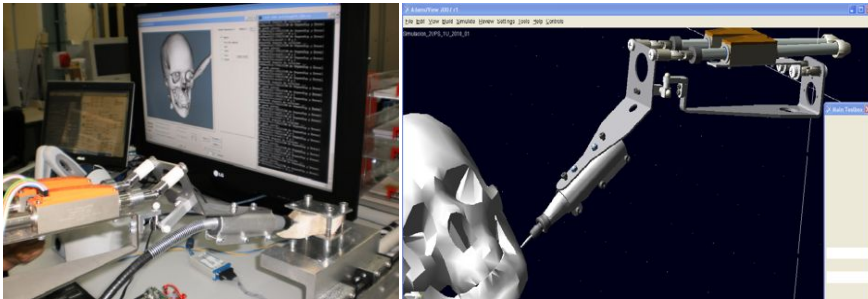


Figura 10.7: Implementación real del robot y simulación del mismo

Bibliografía

- [1] *Technical-related assistance for individuals with disabilities*, Public Law (PL) 100-407, 1988.
- [2] *Design and control of a rehabilitation robot driven by pneumatic swivel modules.*, 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (IEEE, ed.), vol. IEEE Catalog Number: CFP10BRB-DVD, IEEE BioRob, September 2010.
- [3] Bassel Abou-Khalil, *Atlas of eeg and seizure semiology*, no. 0-7506-7513-6, Elsevier, 2005.
- [4] Sunil K. Agrawal and Abbas Fattah, *Theory and design of an orthotic device for full or partial gravity-balancing of a human leg during motion*, **12** (2004), 157 – 165.
- [5] F. Anderson, M. Annett, and W. F. Bischof, *Lean on wii: physical rehabilitation with virtual reality wii peripherals*, *Stud Health Technol Inform* **154** (2010), 229 – 234.
- [6] J. Angeles and C. S. L pez-Caj n, *The design of isotropic manipulator architectures in the presence of redundancies*, *International journal of robotic research* **11** (1992), 196–201.
- [7] ARMIN, *Swiss federal institute of technology, armin system*, <http://www.sms.mavt.ethz.ch/research/projects/armin/armin/>, Last visited, September 2011.
- [8] Samuel K. Au, Jeff Weber, and Hugh Herr, *Biomechanical design of a powered ankle-foot prosthesis*, *Proceedings of the 2007 IEEE 10th International Conference on Rehabilitation Robotics*, June 2007.
- [9] F.J. Badesa, R. Morales, N. Garcia, J. M. Sabater, C. Perez, and E. Fernandez, *Multimodal interfaces to improve therapeutic outcomes in robot-assisted rehabilitation*, *IEEE Transactions on Systems, Man and Cybernetics. Part C* (2012).
- [10] Sai K. Banala, Sunil K. Agrawal, Abbas Fattah, John Scholz, Vijaya Krishnamoorthy, Wei-Li Hsu, J Scholz, and Katherine Rudolph, *Gravity balancing leg orthosis and its performance evaluation*, **6** (2006), 1228 – 1239.

- [11] A. Barrientos, L. F. Peñin, C. Balaguer, and R. Aracil, *Fundamentos de robótica*, Mc Graw-Hill, España, 1997.
- [12] R. Bischoff and V. Graefe, *Machine vision for intelligent robots*, IAPR Works. On Machine Vision Applications (Tokyo, Japan), November 1998, pp. 167–176.
- [13] I. A. Bonev and J. Ryu, *A simple new closed-form solution of the direct kinematics using three linear extra sensors*, IEEE/ASME Int. conf. on advanced intelligent mechanisms. Atlanta 19-23 Sept (IEEE/ASME, ed.), vol. 1, 1999, pp. 526–530.
- [14] Mourad Bouzit, Grigore Burdea, George Popescu, and Rares Boian, *The rutgers master ii. new design force-feedback glove*, IEEE/ASME Transactions on Mechatronics (2002).
- [15] G. Burdea, *Force and touch feedback for virtual reality*, Wiley Interscience. John Wiley and Sons, 1996.
- [16] A. Casals, R. Merchan, E. Portelli, X Cuffi, and J Contijoch, *Capdi: A robotized kitchen for the disable and elderly.*, Assistive Technology at the Thershold of the New Millennium. AAATE 99 (Dusseldorf), November 1999, pp. 346–351.
- [17] J. Dallaway and S. Robin, *Raid – a vocational robotic workstation*, IEEE ICORR, (Keel, UK), September 1992.
- [18] C. Danielsson and L. Holmberg, *Evaluation of the raid workstation*, Proc. RESNA conference (Tennessee, USA.), 1994, pp. 451–453.
- [19] ———, *Evaluation of the raid workstation*, Proc. International Conference on Rehabilitation Robotics (ICORR) (IEEE, ed.), 1994, pp. 7–11.
- [20] H. Van der Loos, *Va/stanford rehabilitation robotics research and development program: Lessons learned in the application of robotics technology to the field of rehabilitation*, IEEE Trans. on Neural Systems and Rehabilitation Engineering (1995), 46–55.
- [21] H. Van der Loos and J. Hammel, *Use rehabilitation robot in a classroom setting.*, Proc. RESNA conference, Nashville (Tennessee, USA.), 1994, pp. 442–444.
- [22] H. Van der Loos, J. Hammel, D. Lees, D. Chang, E. Perakash, and L. Leifer, *A voice-controlled robot system as a quadriplegic programmer’s assistant*, Proc. RESNA conference, (Washington, D.C., USA.), 1990, pp. 129–130.
- [23] J. E. Deutsch, C. Paserchia, C. Vecchione, A. Mirelman, J. E. Lewis, R. Boian, and G. Burdea, *Improved gait and elevation speed of individuals post-stroke after lower extremity training in virtual environments*, JNPT **28** (2004), no. 4, 185 – 186.
- [24] T. Fong, I. Nourbakhsh, and K. Dautenhahn, *A survey of socially interactive robots*, Robotics and Autonomous Systems **42** (2003), 143–166.

- [25] H.J. Freund, H.J. Budingen, and V. Dietz, *Activity of single motr units from human forearm mucle during voluntary isometric contractions*, J Neurophysiol **38** (1975), 993–946.
- [26] J. Fung, C. L. Richards, F. Malouin, B. J. McFadyen, and A. Lamontagne, *A treadmill and motion coupled virtual reality system for gait training post stroke*, Cyber-Psychology Behavior **9** (2006), no. 2, 157 – 162.
- [27] J. García.de.JalÛn and E. Bayo, *Kinematic and dynamic simulation of multibody systems. the real time challenge*, Springer Verlag, 1994.
- [28] R Garnett, M. J. O’Donovan, J.A. Stephens, and A. Taylor, *Motor unit organization of human medial gastrocnemius*, J Physiol **287** (1979), 33–43.
- [29] GBO, *University of delaware, gbo system*.
- [30] GENTLE, *University of reading, gentle/s system*, <http://www.gentle.reading.ac.uk/>, Last visited, September 2011.
- [31] B. Graf, M. Hans, and R. Schraft, *Care-o-bot ii development of a next generation robotic home assistant*, Autonomous Robot **16** (2004), no. 2, 193–205.
- [32] E. Gugliemelli, C. Laschi, G. Teti, R. Fontanelli, and P. Dario, *Robots for the assistance to disabled persons at home and in hospitals*, IARP Int. Workshop on Medical Robots (Viena, Austria), 1996.
- [33] Tore Hägglund, *A friction compensator for pneumatic control valves*, Journal of process control **12** (2002), no. 8, 897–904.
- [34] R. Ham, T. Sugar, B. Vanderborght, K. Hollander, and D. Lefeber, *Compliant actuator designs*, IEEE Robotics and Automation Magazine **16** (2009), no. 3, 81–94.
- [35] K. Harada, E. Susilo, A. Menciassi, and P. Dario, *Wireless reconfigurable modules for robotic endoluminal surgery*, IEEE International Conference on Robotics and Automation, 2009.
- [36] E. Helms, R.D. Schraft, and M. Haegele, *robwork: Robot assistant in industrial environments*, In Proc. of the 11th IEEE Int. Workshop on Robot and Human interactive Communication, ROMAN2002 (Berlin, Germany) (IEEE, ed.), September, 25-27 2002, pp. 399–404.
- [37] E Henneman and L. M. Mendell, *Functional organization of the motoneuron pool and its inputs*, Handbook of physiology: The nervous system, American Physiological Society (1981), 423–507.
- [38] Hocoma, *Hocoma, arneo and lokomat systems*, <http://www.hocoma.com>, Last visited, September 2011.
- [39] M. Holden, E. Todorov, J. Callaban, and E. Bizzi, *Virtual environment training improves motor performace in two patients with stroke: Case report*, Neurology Report **23** (1999), no. 2, 57 – 68.

- [40] H.S. Hong, S.Y. Jung, J.H. Jung, B.G. Lee, J.W. Kang, D.J. Park, and M.J. Chung, *Development of work assistant mobile robot system for the handicapped in a real manufacturing environment*, Proceedings of the 2005 IEEE 9th International Conference on Rehabilitation Robotics (Chicago, USA) (IEEE, ed.), IEEE, July 2005, pp. 197–200.
- [41] P. Hoppenot and E. Colle, *Localization and control of a rehabilitation mobile robot by close human-machine cooperation*, IEEE Trans. On Rehabilitation Engineering **9** (2001), no. 2, 181–190.
- [42] C. Innocenti and V. Parenti-Castelli, *Closed form direct position analysis of a 5-5 parallel mechanism*, ASME Transaction, Journal of mechanical design **120** (1993), 73–99.
- [43] IPAM, *University of leeds, ipam system*, <http://www.personal.leeds.ac.uk/~menaej/>, Last visited, September 2011.
- [44] IPK, *Fraunhofer-institute for production systems and design technology ipk in berlin, hapticwalker system*, <http://www.ipk.fraunhofer.de/>, Last visited, September 2011.
- [45] Heckman C. J and M. D. Binder, *Computer simulation of the effects of different synaptic input system on motor unit recruitment*, J Neurophysiol **70** (1993), 1827–1840.
- [46] D. Jack, R. Boian, A. Merians, S. Adamovich, M. Tremaine, M. Recce, G. C. Burdea, and H. Poizner, *A virtual reality-based exercise program for stroke rehabilitation. neural systems and rehabilitation engineering*, In Proceedings of ACM SIGCAPH ASSETS 2000, 2000, pp. 56 – 63.
- [47] Jiri Jan, *Medical image processing, reconstruction and restoration. concepts and methods*, Taylor and Francis, 2006.
- [48] T. Jones, *Raid – towards greater independence in the office and home environment*, Proc. ICORR 99 (Stanford, US) (IEEE, ed.), 1999, pp. 201–206.
- [49] K. Kawamura, S. Bagchi, M. Iskarous, and M. Bishay, *Intelligent robotic systems in service of the disabled*, IEEE Trans. on Rehabilitation Engineering **3** (1995), no. 1, 14–21.
- [50] Peter Kazanzides, Gabor Fichtinger, Gregory Hager, Allison M. Okamura, Louis L. Whitcomb, and Russell H. Taylor, *Surgical and interventional robotics: part i*, IEEE Robotics and Automation Magazine **15** (2008), no. 2, 122–130.
- [51] ———, *Surgical and interventional robotics: part ii*, IEEE Robotics and Automation Magazine **15** (2008), no. 3.
- [52] ———, *Surgical and interventional robotics: part iii*, IEEE Robotics and Automation Magazine **15** (2008), no. 4, 84–93.

- [53] A.C. Lehman, J. Dumpert, N. A. Wood, L. Redden, A. Visty, S. Farritor, B. Varnell, and D. Oleynikov, *Dexterous miniature robot for advanced minimally invasive surgery*, *Surgical Endoscopic* **23** (2009), 260–266.
- [54] A. Liu, F. Tendick, K. Cleary, and C. Kaufmann, *A survey of surgical simulation: applications, technology, and education*, *Presence* **12** (6) (2003).
- [55] L. F. Lucca, *Virtual reality and motor rehabilitation of the upper limb after stroke: a generation of progress?*, *J Rehabil Med* **41** (2009), 1003 – 1006.
- [56] Brigell. M, Bach. M, Barber. C, Kawasaki. K, and Kooijman. A, *Guideline for calibration of stimulus and recording parameters used in clinical electrophysiology of vision*, *Doc Ophthalmol* **95** (1998), 1–14.
- [57] E. C. Magdalon, Michaelsen S. M., A. Quevedo, and M. F. Levin, *Comparison of grasping movements made by healthy subjects in a 3-dimensional immersive virtual versus physical environment*, *Acta Psychologica* (2011).
- [58] Motek Medical, *Motek, caren system*, <http://www.motekmedical.com>, Last visited, September 2011.
- [59] J. P. Merlet, *Les robots parallèles*, Hermes, Paris, 1997.
- [60] Marmor MF, Holder GE, Porciatti V, Trick GL, and Zrenner E., *Guidelines for basic pattern electroretinography*, Recommendations by the International Society for Clinical Electrophysiology of Vision. *Doc Ophthalmol*, 1996.
- [61] MITMANUS, *Massachusetts institute of technology, mit manus system*, <http://meche.mit.edu/research/controls/>, Last visited, September 2011.
- [62] Motorika, *Motorika, reogo and reoambulator systems*, <http://www.motorika.com>, Last visited, September 2011.
- [63] Neveryd and B. Bolmsj, *Walky, an ultrasonic navigating mobile robot for the disabled*, TIDE Congress (Paris), 1995, pp. 366–370.
- [64] Kim N. Norton, *A brief history of prosthetics*, *In Motion* **17** (2007), no. 7.
- [65] ML Pinto, J.M. Sabater-Navarro, J. Sofrony, N. Garcia, J.M Azorin, and C. Perez, *Sistema de entrenamiento para operaciones de reemplazo total de rodilla*, *Trauma* **22** (2011), no. 3.
- [66] E. Pollack, S. Engberg, J. T. Matthews, S. Thrun, L. Brown, D. Colbry, C. Orosz, B. Peintner, S. Ramakrishnan, J. Dunbar-Jacob, C. McCarthy, M. Montemerlo, J. Pineau, and N. Roy, *Pearl: A mobile robotic assistant for the elderly*, AAAI Workshop on Automation as Eldercare, August 2002.
- [67] D. Ramos, E. Camayo, J. M. Sabater, and O. Vivas, *hand neuro-rehabilitation system using spring nitinol actuators*, *Mechatronics* (2013).

- [68] A. A. Rizzo and G. J. Kim, *A swotanalysis of the field of vr rehabilitation and therapy*, Presence: Teleoperators and Virtual Environments **14** (2005), 119 – 146.
- [69] Nicholas Roy, Gregory Baltus, Dieter Fox, Francine Gemperle, Jennifer Goetz, Tad Hirsch, Dimitris Margaritis, Michael Montemerlo, Joelle Pineau, Jamie Schulte, and Sebastian Thrun, *Towards personal service robots for the elderly*.
- [70] R. Saltaren, L. Puglisi, J.M. Sabater-Navarro, and F.Thomas, *Robotica aplicada utilizando matlab. análisis y diseño de robots.*, vol. 1, (to appear), 2012.
- [71] R. Saltaren, J. M. Sabater, E. Yime, J.M Azorin, and R. Aracil, *Performance evaluation of spherical parallel platforms for humanoid robots*, Robotica **25** (2007), no. 3, 257–267.
- [72] G. Saposnik, *Virtual reality in stroke rehabilitation a meta-analysis and implications for clinicians*, Journal of the American Heart Association **42** (2011), no. 5, 1380 – 1386.
- [73] Ioannis Sarakoglou, Sophia Kousidou, Nikolaos G. Tsagarais, and Darwin G. Caldwell, *Exoskeleton-based exercises for the disabilities of the upper arm and hand*, Itech Education and Publishing, Viena, Austria, 2007.
- [74] M. T Schultheis and A. A Rizzo, *The application of virtual reality technology in rehabilitation*, Rehabilitation Psychology **46** (2011), no. 5, 296 – 311.
- [75] J.R. Serracín, L.J. Puglisi, R. Saltaren, G. Ejarque, J.M. Sabater-Navarro, and R. Aracil, *Kinematic analysis of a novel 2-d.o.f. orientation device*, Robotics and Autonomous Systems **60** (2012), 852–861.
- [76] Oussama Siciliano, Bruno; Khatib (ed.), *Springer handbook of robotics*, Springer, 2008.
- [77] R. Simpson and al., *Navchair: An assistive wheelchair navigation system with automatic adaptation*, Lecture Notes in Artificial Intelligence, 1458 **1458** (1998), 235–255.
- [78] H. Kwee.and C. Stanger, *The manus robot arm*, Rehabilitation Robotics Newsletter **5** (1993), no. 2.
- [79] C.D. Takahashi, L. Der-Yeghiaian, V.H. Le, and S.C. Cramer, *A robotic device for hand motor therapy after stroke*, 9th International Conference on Rehabilitation Robotics, 2005.
- [80] B. Taylor, M.E. Cupo, and S.J. Sheredos, *Workstation robotics: a pilot study of a desktop vocational assistant robot*, Am J Occup Ther. **47** (1993), no. 11, 1009–1013.
- [81] L.W. Tsai, *Robot analysis: the mechanics of serial and parallel manipulators*, Wiley Interscience. John Wiley and Sons, 1999.
- [82] N. Ulrich and V. Kumar, *Passive mechanical gravity compensation for robot manipulators*, vol. 2, April 1991, pp. 1536 – 1541.

- [83] P. Valdastri, R. J. III Webster, C. Quaglia, M. Quirini, A. Menciassi, and P. Dario, *A new mechanism for mesoscale legged locomotion in compliant tubular environments*, IEEE Transactions on Robotics **25** (2009), no. 5.
- [84] A. Vivas, *Diseño y control de robots industriales: teoría y práctica*, elaleph.com, 2012.
- [85] Jonathan R. Wolpaw and al., *Brain-computer interface technology: A review of the first international meeting*, IEEE Transactions on rehabilitation engineering **8** (2000), no. 2, 164–173.
- [86] H. Yanco and Wheellesley, *A robotic wheelchair system: Indoor navigation and user interface*, Lecture Notes in Artificial Intelligence, 1458 **1458** (1998), 256–268.

