

**DESARROLLO DE UNA INTERFAZ  
KINESTÉSICA PARALELA Y  
EXPERIMENTACIÓN EN  
CONTROL DE SISTEMAS  
HÁPTICOS Y TELEOPERADOS**

*José María Sabater Navarro*

*División de Ingeniería de Sistemas y Automática  
Departamento de Ingeniería de Sistemas Industriales  
Universidad Miguel Hernández*

30 de junio de 2003



DEPARTAMENTO DE INGENIERÍA DE SISTEMAS INDUSTRIALES

DIVISIÓN DE INGENIERÍA DE SISTEMAS Y AUTOMÁTICA

Escuela Politécnica Superior de Elche

**DESARROLLO DE UNA INTERFAZ KINESTÉSICA  
PARALELA Y EXPERIMENTACIÓN EN  
CONTROL DE SISTEMAS HÁPTICOS Y  
TELEOPERADOS**

Autor

**José M<sup>a</sup> Sabater Navarro**

Ingeniero Industrial

Directores

**Roque J. Saltarén Pazmiño**

Dr. Ingeniero Industrial

**Rafael Aracil Santonja**

Dr. Ingeniero Industrial

Elche, 2003



# Tribunal

Presidente:

Vocal:

Vocal:

Vocal:

Vocal secretario:



Calificación de la tesis:.....



# Agradecimientos

En primer lugar, me gustaría dedicar este trabajo a todos aquellos que han colaborado en algún momento en mi formación académica y personal.

A mis profesores de la UPV, Rosa, Mercedes y Jose Luis Muñoz-Cobo, por enseñarme a amar la investigación.

En especial quisiera hacer constar mi agradecimiento a los directores de la tesis, los doctores Roque Salterén y Rafael Aracil, por sus acertados consejos y comentarios, por la confianza puesta en mí para la realización de esta tesis y por la amistad que amablemente me han brindado.

También quisiera expresar mi gratitud sincera a todos los miembros de la División de Ingeniería de Sistemas y Automática de la Escuela Politécnica Superior de Elche: los profesores José María Azorín, Nicolás García, Ramón P. Ñeco, Carlos Pérez, César Fernández, Asunción Vicente, Rafael Puerto, Luis Miguel Jiménez y Óscar Reinoso. Gracias a todos por la acogida que me habéis brindado, por estar siempre dispuestos a ayudar y por el clima de trabajo que se disfruta en la división. Gracias también a Miguel Almonacid y a Antonia Montoya por las facilidades dadas en muchos aspectos de mi trabajo. Y por supuesto a Miguel Angel Oliva, Emilio Velasco y todos los que forman el departamento de Ingeniería de Sistemas Industriales.

A Susi, por todo el apoyo mostrado día a día.

Por último, las personas más importantes, quiero agradecer a mis padres toda su enorme paciencia demostrada a lo largo de estos años.

Ah!, y a todos los que me he dejado por despiste, gracias.





# Resumen

La teleoperación comprende el conjunto de tecnologías que permiten a un operador la realización de tareas a distancia, teniendo su aplicación específica en el desarrollo de tareas en entornos que conlleven un riesgo al operador o bien sean inaccesibles por éste. Tanto la teleoperación de robots reales como la exploración y manipulación de entornos virtuales está tomando de nuevo una gran importancia dentro de los grupos de investigación en robótica, gracias, entre otras cosas, a las aportaciones de nuevos algoritmos de control basados en el conocimiento de los modelos dinámicos de los dispositivos involucrados y al incremento de la capacidad de cálculo de los ordenadores actuales.

Sin embargo, la experimentación con estos nuevos esquemas de teleoperación se encuentra reducida a simulaciones con modelos dinámicos sencillos o a sistemas mecánicos sencillos, debido principalmente a la falta de dispositivos “abiertos” a los cuales se les pueda modificar sus esquemas de control.

La presente Tesis pretende cubrir este espacio aportando los conocimientos necesarios para la implantación de plataformas de teleoperación constituidas por dispositivos abiertos, y ahondando en el estudio de los esquemas de control bilateral existentes. Como primer objetivo para disponer de dicha plataforma experimental se ha diseñado, modelado e implementado un dispositivo háptico abierto, de configuración paralela y con cadena cinemática 6-URS. Se han desarrollado los modelos y algoritmos necesarios para el control de dicho dispositivo, de manera que sea capaz de funcionar tanto como display de impedancias o como display de admitancias. La utilización de una formulación elegante basada en la dinámica multicuerpo ha permitido la obtención de los modelos cinemáticos y dinámicos necesarios para el diseño de las diferentes estrategias de control. La interfaz de dicho dispositivo debe cumplir con los requisitos especificados para la plataforma de teleoperación.

Para completar una plataforma de experimentación lo más generalista posible, y a su vez simplificar los problemas de sincronización de las tareas entre el entorno local y el remoto ha sido necesario el desarrollo de un simulador dinámico de robots que permita experimentar en tecnología haptics. Entre las características de este simulador se encuentran la inclusión del modelo dinámico del robot esclavo, el tratamiento de colisiones y fuerzas

generadas y la fácil implementación de esquemas de control.

De igual forma se ha desarrollado el procedimiento necesario para disponer de un robot esclavo en lazo abierto. Se ha procedido a la “apertura” de los servos de un robot comercial y se presenta un procedimiento para la identificación de los parámetros del modelo dinámico de dicho robot serial.

La disponibilidad de dispositivos maestro y esclavo abiertos (reales o virtuales) permite la implementación y experimentación con diferentes esquemas de control bilateral, constituyendo así una interesante herramienta para futuros desarrollos.



# Abstract

Teleoperation includes the integrated technologies that allow the operator to make distance-tasks. This has an specific application on the development of tasks on operator-risk or difficult-access environments. Both the teleoperation of real robots and the exploration and manipulation of virtual environments is renewing robotics researchers interest thanks, among other reasons, to the new control algorithms based on the knowledge of the dynamic models of the devices and of the increasing on the computational capacity of new processors.

Meanwhile, experimentation with these teleoperation schemes is limited to simulations of simplified dynamic models and simple mechanical systems, mainly due to the shortage of “open” devices where the control schemes could be modified.

Present Thesis tries to cover this space giving the necessary knowledge to the implementation of teleoperation platforms built by open devices, and to go deeply into the study of actual bilateral control schemes. Firstly, the design and development of an “open” six degrees of freedom parallel master has been needed. The device has 6-URS kinematic chains and is able to work either as an impedance or as an admittance device. An smart formulation based on the multibody dynamics theory has been developed. These dynamic models have been used for the obtention of the control algorithms of the devices. The interface of this device fits with the specific requirements of the experimental platform.

To complete a general experimental platform, and to solve the synchronization problems between remote and local zone, a dynamic simulator of robots has been developed. This simulator is able to experiment with complete dynamic models of robots and allow to easily implement the growing haptic theory algorithms. Among others, the characteristics of the simulator are a complete implementation of the dynamic model of the slave, a collision algorithm, a force generator module and the easy declaration of different control schemes.

At the same time, the “opening” procedure of the servos of a commercial

slave robot has been developed, and a method for the identification of the inertial parameters is reviewed.

The availability of “open” master and slave devices (real or virtual ones) allow the implementation and experimentation with different bilateral control schemes, being an interesting tool for future developments.



# Índice general

<b>I Tesis</b>	<b>1</b>
<b>1. Introducción</b>	<b>3</b>
1.1. Nuestro “sexto sentido” . . . . .	3
1.2. Definiciones . . . . .	4
1.3. Motivación y génesis de la Tesis . . . . .	5
1.4. Estructura de la Tesis . . . . .	5
1.5. Contribuciones aportadas . . . . .	7
<b>2. Plataformas de teleoperación actuales</b>	<b>9</b>
2.1. Introducción . . . . .	9
2.1.1. Estructura del capítulo . . . . .	12
2.2. Estado del arte de maestros con reflexión de esfuerzos . . . . .	13
2.2.1. Aspectos fisiológicos relativos al diseño de interfaces maestras . . . . .	13
2.2.1.1. Fisiología del tacto . . . . .	15
2.2.1.2. Adaptación sensorial . . . . .	16
2.2.1.3. Control sensorial humano . . . . .	17
2.2.1.4. Amplitud de banda de estímulos y amplitud de banda de control . . . . .	19
2.2.2. Actuadores para interfaces maestras . . . . .	19
2.2.3. Clasificación de las interfaces kinestésicas . . . . .	22
2.2.3.1. Interfaces de escritorio . . . . .	23
2.2.3.2. Interfaces sostenidas en el techo o en un muro	29
2.2.3.3. Interfaces portátiles . . . . .	31
2.3. Estado del arte de esclavos utilizados en teleoperación . . . . .	33
2.3.1. Telemanipuladores en el sector nuclear . . . . .	34
2.3.2. Telemanipuladores para vehículos submarinos . . . . .	35
2.3.3. Telemanipuladores en el sector espacial . . . . .	37

2.3.4.	Telemanipuladores en el sector de servicios . . . . .	38
2.3.5.	Telemanipuladores virtuales. La realidad virtual como ayuda a la robótica . . . . .	39
2.4.	Estado del arte de las comunicaciones: clasificación de arquitecturas de control bilateral . . . . .	43
2.4.1.	Clasificación de los esquemas de teleoperación . . . . .	44
2.4.2.	Analogía del cuadripolo eléctrico . . . . .	45
2.4.2.1.	Estabilidad . . . . .	48
2.4.2.2.	Representación de un sistema bilateral . . . . .	50
2.5.	Plataformas paralelas en teleoperación . . . . .	53
2.6.	Conclusiones . . . . .	58
<b>3.</b>	<b>Diseño y análisis de un dispositivo háptico paralelo 6-URS</b>	<b>59</b>
3.1.	Introducción . . . . .	59
3.1.1.	Display de impedancias vs. display de admitancias . . . . .	61
3.1.2.	Estructura del capítulo . . . . .	61
3.2.	Diseño mecánico del dispositivo . . . . .	62
3.2.0.1.	¿Por qué un dispositivo paralelo? . . . . .	63
3.2.0.2.	¿Por qué un dispositivo sin reductores engranados? . . . . .	63
3.2.0.3.	¿Por qué encoders incrementales? . . . . .	64
3.2.0.4.	¿Por qué seis grados de libertad? . . . . .	64
3.2.0.5.	¿Por qué un paralelogramo articulado? . . . . .	64
3.2.0.6.	¿Por qué está realizado en aluminio? . . . . .	64
3.2.1.	Análisis del dispositivo con software de simulación dinámica . . . . .	65
3.2.1.1.	Prediseño y elección de los actuadores . . . . .	66
3.2.2.	Análisis del dispositivo con software de animación gráfica . . . . .	67
3.2.2.1.	Espacio de trabajo del dispositivo . . . . .	69
3.3.	Modelado cinemático de un sistema multicuerpo . . . . .	72
3.3.0.2.	Bases matemáticas para el movimiento de 1 cuerpo rígido . . . . .	72
3.3.0.3.	Sistemas de múltiples cuerpos . . . . .	80
3.3.0.4.	Desplazamientos virtuales y rotaciones . . . . .	82
3.3.1.	Modelo y algoritmos de cinemática inversa . . . . .	84
3.3.1.1.	Obtención geométrica de la matriz jacobiana . . . . .	89
3.3.1.2.	Software desarrollado . . . . .	90
3.3.2.	Modelo y algoritmos de cinemática directa . . . . .	92

3.3.2.1.	Obtención del vector de restricciones . . . . .	93
3.3.2.2.	Obtención de la matriz jacobiana del vector de restricciones . . . . .	99
3.3.2.3.	Solución numérica . . . . .	101
3.3.2.4.	Software desarrollado . . . . .	102
3.3.3.	Análisis de velocidad y aceleración en coordenadas an- gulares . . . . .	103
3.3.3.1.	Software desarrollado . . . . .	104
3.3.4.	Simulación cinemática con planificación de trayectorias	104
3.4.	Modelado dinámico de un sistema multicuerpo . . . . .	105
3.4.0.1.	Bases matemáticas para el movimiento de 1 cuerpo rígido . . . . .	105
3.4.0.2.	Sistemas de múltiples cuerpos con restricciones	108
3.4.1.	Modelo y algoritmos de dinámica inversa . . . . .	109
3.4.1.1.	Software desarrollado . . . . .	110
3.4.2.	Modelo y algoritmos de dinámica directa . . . . .	112
3.4.2.1.	Software desarrollado . . . . .	113
3.5.	Conclusiones . . . . .	114
<b>4.</b>	<b>Simulador dinámico de arquitecturas de teleoperación</b>	<b>117</b>
4.1.	Introducción . . . . .	117
4.1.1.	Estructura del capítulo . . . . .	118
4.2.	Dinámica computacional . . . . .	118
4.3.	Arquitectura general del simulador . . . . .	120
4.4.	Interfaz del simulador . . . . .	124
4.4.1.	Interfaz genérica para conexión del maestro . . . . .	124
4.4.1.1.	PHANToM . . . . .	125
4.4.1.2.	MAGISTER-P . . . . .	126
4.4.2.	Interfaz de visualización . . . . .	127
4.5.	Red de acople: esquema de control . . . . .	128
4.6.	Simulación de escenas . . . . .	129
4.7.	Algoritmos cinemáticos de mapeado de espacios de trabajo . .	130
4.8.	Conclusiones . . . . .	134
<b>5.</b>	<b>Experimentación con sistemas hápticos de n grados de liber- tad</b>	<b>135</b>
5.1.	Introducción . . . . .	135
5.1.1.	Estructura del capítulo . . . . .	136

5.2.	Implementación del dispositivo háptico; modos de trabajo . . .	137
5.2.1.	Arquitectura hardware . . . . .	137
5.2.1.1.	Tarjeta dSpace DS1103, librerías RTlib y software ControlDesk . . . . .	138
5.2.1.2.	Librerías MBlib de dinámica multicuerpo . . . . .	138
5.3.	Calibración del dispositivo háptico . . . . .	139
5.3.1.	Como display de impedancias... . . . .	141
5.3.2.	Como display de admitancias... . . . .	142
5.3.3.	Resultados experimentales . . . . .	144
5.4.	Conexión del dispositivo háptico . . . . .	146
5.4.1.	Un diseño para la estabilidad del sistema . . . . .	146
5.4.2.	Conexión del MAGISTER-P; Estabilidad del sistema implementado . . . . .	151
5.4.3.	Conexión del MAGISTER-P; Uso de las librerías CLIB . . . . .	154
5.4.4.	Resultados experimentales . . . . .	156
5.5.	Conclusiones . . . . .	156
<b>6.</b>	<b>Aplicación a sistemas de teleoperación abiertos de n grados de libertad</b>	<b>159</b>
6.1.	Introducción . . . . .	159
6.1.1.	Estructura del capítulo . . . . .	160
6.2.	Comunicaciones en los esquemas de teleoperación . . . . .	160
6.2.1.	Comunicaciones con retardos . . . . .	161
6.2.1.1.	Comunicaciones a nivel de scripts . . . . .	161
6.2.1.2.	Comunicaciones a nivel de señales . . . . .	162
6.2.2.	Comunicaciones en tiempo real . . . . .	164
6.3.	Elección de un dispositivo esclavo abierto . . . . .	165
6.3.1.	Justificación de la elección . . . . .	165
6.3.1.1.	Arquitectura de control abierta . . . . .	166
6.3.1.2.	Interfaz para la programación del robot . . . . .	169
6.3.2.	Descripción del brazo esclavo . . . . .	169
6.4.	Modelado de un dispositivo esclavo abierto . . . . .	170
6.4.1.	Modelado cinemático del brazo serie . . . . .	170
6.4.2.	Modelado dinámico del brazo serie . . . . .	172
6.4.2.1.	Modelo dinámico simplificado . . . . .	172
6.4.2.2.	Datos del fabricante . . . . .	173
6.4.2.3.	Identificación de los parámetros dinámicos . . . . .	174
6.5.	Conexión del dispositivo esclavo . . . . .	176



6.5.1.	Apertura del lazo de control del brazo robot . . . . .	176
6.5.1.1.	Interfaz de comunicación en arquitectura abierta . . . . .	178
6.6.	Implementación de un esquema de teleoperación . . . . .	180
6.7.	Conclusiones . . . . .	182
<b>7.</b>	<b>Conclusiones y trabajo futuro</b>	<b>183</b>
7.1.	Conclusiones generales . . . . .	183
7.1.1.	Contribuciones aportadas . . . . .	184
7.2.	Objetivos pendientes . . . . .	185
7.2.1.	En modelado y análisis de robots paralelos... . . . .	186
7.2.2.	En identificación y modelado de robots seriales... . . .	186
7.2.3.	En robótica virtual y simulación dinámica... . . . .	187
7.2.4.	En comunicaciones en tiempo real... . . . .	187
<b>II</b>	<b>Anexos</b>	<b>189</b>
<b>A.</b>	<b>Planos mecánicos del MAGISTER-P</b>	<b>191</b>
A.1.	Planos generales . . . . .	191
A.2.	Planos de fabricación . . . . .	191
<b>B.</b>	<b>Conexiones Eléctricas del MAGISTER-P</b>	<b>197</b>
B.1.	Conexiones eléctricas motor EM2342-012CR/LC3002 . . . . .	197
B.2.	Conexiones LC3002+encoders HP / dSpace ds1103 . . . . .	198



# Índice de figuras

2.1. Principales avances en teleoperación, agrupados según sectores de aplicación . . . . .	10
2.2. Elementos de un sistema teloperado y sistema háptico . . . . .	11
2.3. Espacio útil para el brazo del operador . . . . .	13
2.4. Estructura y localización de los receptores táctiles de la piel . . . . .	16
2.5. Clasificación del agarre de un objeto . . . . .	18
2.6. Amplitud de la banda de estímulo . . . . .	20
2.7. Ratio potencia-masa de distintos actuadores . . . . .	21
2.8. Clasificación de las interfaces hápticas según su soporte . . . . .	22
2.9. Joystick esférico de 2 grados de libertad . . . . .	23
2.10. Force Feedback Master de 4 grados de libertad . . . . .	24
2.11. Dibujo de la patente [US 6,429,849 B1] del Microsoft Force Feedback . . . . .	25
2.12. Joystick cartesiano de 2 grados de libertad HBFD y dibujo de la patente del maestro Escalibur [US 6,046,563] . . . . .	26
2.13. Haptic Master de 6 gdl . . . . .	27
2.14. Pen-based display de la Universidad de Washington . . . . .	28
2.15. PHANToM y dibujo de la patente original . . . . .	28
2.16. WYSIWYF . . . . .	30
2.17. SARCOS Dextrous Arm Master . . . . .	31
2.18. RUTGERS MASTER II . . . . .	33
2.19. Telemanipuladores en el sector servicios: Retina y Trepá . . . . .	39
2.20. Esquema de un sistema de teleoperación para trabajo con retardos . . . . .	43
2.21. Esquema general de plataforma de teleoperación . . . . .	44
2.22. Modelo simplificado de plataforma de teleoperación . . . . .	45
2.23. Control Bilateral posición-posición mediante teoría clásica . . . . .	46
2.24. Control Bilateral fuerza-posición mediante teoría clásica . . . . .	46

2.25. Cuadripolo cargado con otro cuadripolo . . . . .	48
2.26. Cuadripolo cargado con dos bipolos . . . . .	49
2.27. Modelo de un sistema teleoperado según la teoría del cuadripolo	50
2.28. Modelo del operador . . . . .	52
2.29. Cuadripolo de la interfaz kinestésica . . . . .	53
2.30. Posiblemente el primer mecanismo espacial paralelo. Patentado en 1931 (US patent N <sup>o</sup> 1.789.680) . . . . .	54
3.1. Diseño mecánico de una plataforma 6 URS . . . . .	62
3.2. Modelo utilizado con I-deas . . . . .	65
3.3. Curvas de simulación con I-deas . . . . .	66
3.4. Software de animación gráfica . . . . .	68
3.5. Posición del SR pivote del eslabón brazo . . . . .	69
3.6. Estudio del espacio de trabajo con una herramienta de animación gráfica . . . . .	70
3.7. Rotación finita de un punto $P'$ . . . . .	73
3.8. Desplazamiento de un punto material en el espacio . . . . .	79
3.9. Cadena cinemática de dos cuerpos rígidos . . . . .	81
3.10. Modelo espacial del MAGISTER-P . . . . .	84
3.11. Esquema del modelo cinemático propuesto . . . . .	85
3.12. Posiciones relativas de los vectores $\vec{u}_{z\text{-pares}}$ y $s_0^{A_n}$ . . . . .	87
3.13. Software de solución de la Cinemática Inversa . . . . .	90
3.14. Representación VRML de la solución cinemática inversa. . . . .	90
3.15. Subsistemas de Visualización VRML. . . . .	91
3.16. Esquema del algoritmo de solución de la Cinemática Inversa. . . . .	92
3.17. Restricciones básicas espaciales. . . . .	94
3.18. Paralelismo entre pares de cuerpos. . . . .	95
3.19. Representación de una restricción rotacional entre pares de cuerpos. . . . .	96
3.20. Representación de una restricción esférica entre pares de cuerpos. . . . .	97
3.21. Representación de una restricción universal entre pares de cuerpos. . . . .	98
3.22. Representación de una restricción de accionamiento rotacional. . . . .	99
3.23. Estructura de la matriz jacobiana del vector de restricciones. . . . .	101
3.24. Software de solución de la Cinemática Directa . . . . .	102
3.25. Esquema del algoritmo de solución de la Cinemática Directa. . . . .	103
3.26. Esquema del algoritmo de solución del análisis de velocidades y aceleraciones. . . . .	104

3.27. Definición de cuerpo espacial. . . . .	105
3.28. Software de solución de la Dinámica Inversa. . . . .	111
3.29. Esquema del algoritmo de solución de la dinámica inversa. . . . .	111
3.30. Software de solución de la Dinámica Directa. . . . .	113
3.31. Esquema del algoritmo de solución de la dinámica directa. . . . .	114
4.1. Arquitectura del Simulador. . . . .	120
4.2. Ejemplo de AABB en un objeto 2D. . . . .	122
4.3. Jerarquía de clases desarrolladas sobre la librería ODE. . . . .	124
4.4. Jerarquía de las clases de la interfaz maestra. . . . .	125
4.5. Interfaz de visualización del simulador desarrollado. . . . .	128
4.6. Jerarquía de las clases de la escena. . . . .	130
4.7. Mapeado e indexación de espacios de trabajo. . . . .	131
4.8. Casos 1 y 2 de los algoritmos de mapeado e indexación. . . . .	133
4.9. Caso 3 de los algoritmos de mapeado e indexación. . . . .	134
5.1. Arquitectura Hardware del Magíster-P . . . . .	137
5.2. Esquema del experimento 1A . . . . .	139
5.3. Esquema del experimento 1F . . . . .	140
5.4. Esquema del experimento 2A . . . . .	141
5.5. Esquema del experimento 2C. Display de Impedancias . . . . .	142
5.6. Esquema del experimento 2D. Display de Admitancias . . . . .	143
5.7. Experimentos con el dispositivo maestro. . . . .	144
5.8. Espacio de trabajo con orientación del efector final constante. . . . .	145
5.9. Implementación del esquema fuerza-posición . . . . .	150
5.10. Red de acople en esquema fuerza-posición . . . . .	151
5.11. Maestro MAGISTER-P . . . . .	152
5.12. Inserciones con esquema fuerza-posición . . . . .	157
6.1. Plataforma de experimentación a nivel de scripts . . . . .	162
6.2. Plataforma de experimentación a nivel de scripts con el maestro PHANToM . . . . .	162
6.3. Plataforma de experimentación a nivel de señales con retardos variables . . . . .	163
6.4. Plataforma de experimentación a nivel de señales . . . . .	164
6.5. Plataforma de experimentación haptics . . . . .	165
6.6. Estructura de capas del Mitsubishi PA10 . . . . .	167
6.7. Arquitectura abierta del Mitsubishi PA10 . . . . .	168

6.8. Brazo mecánico del Mitsubishi PA10 . . . . .	170
6.9. Arquitectura de los servos del Mitsubishi PA10 . . . . .	176
6.10. Modos de comunicación de los servos del PA10 . . . . .	177
6.11. Graphics User Interface de comunicaciones . . . . .	181
6.12. Implementación de un esquema de teleoperación . . . . .	182
A.1. Modelo Mechanical Desktop . . . . .	192
A.2. Brazo mecánico . . . . .	192
A.3. Despiece de un brazo . . . . .	193
B.1. Dimensiones de los motores EM2342-012CR . . . . .	198
B.2. Conector DIN41612 . . . . .	198



# Índice de tablas

2.1. Características de los receptores táctiles. . . . .	17
3.1. Características Mecánicas del dispositivo . . . . .	71
3.2. Modelo del dispositivo maestro . . . . .	86
3.3. Resumen de Restricciones primitivas . . . . .	101
4.1. Variables de los algoritmos de indexación y mapeado . . . . .	132
5.1. Experimento 2D.- Parámetros PD . . . . .	146
5.2. Experimentos con MAGISTER-P como dispositivo individual .	147
5.3. Parámetros de simulación estable: fuerza-posición . . . . .	154
6.1. Esquemas en plataforma 1 (MAGISTER-P) . . . . .	162
6.2. Esquemas en plataforma 1 (PHANToM) . . . . .	163
6.3. Esquemas en plataforma 2 (MAGISTER-P) . . . . .	163
6.4. Parámetros Denavit-Hartenberg Mitsubishi PA-10 robot . . .	173
6.5. Parámetros máxicos del robot Mitsubishi PA-10 . . . . .	173
6.6. Configuración de los jumpers de la tarjeta ARC-AT. . . . .	179
6.7. Funciones miembro de la clase CServo. . . . .	180
B.1. Características motor-driver . . . . .	197
B.2. Conexiones analógicas . . . . .	199
B.3. Conexiones encoders . . . . .	200





# Parte I

## Capítulos





# CAPÍTULO 1

## INTRODUCCIÓN: Interacción Hombre-Robot

---

*“Algunas clases de animales tienen todos los sentidos, algunas otras sólo algunos de ellos, y otras sólo uno, el más indispensable, el tacto.”*

Aristóteles, 350 a.C.

### 1.1. Nuestro “sexto sentido”

Aristóteles describía nuestras cinco facultades básicas: vista, oído, tacto, olor y gusto [22]. De ellas, tan sólo las dos primeras están siendo explotadas con eficacia dentro del campo de la interacción hombre-robot. Las imágenes visuales, bien obtenidas por una cámara o bien generadas por ordenador, y en menor medida las señales sonoras obtenidas por micrófonos o generadas como señales de alarma, son una gran ayuda para el operador que está utilizando un robot.

Sin embargo, la utilización de nuestro sentido del tacto todavía se encuentra en sus primeros desarrollos. Los modernos estudios fisiológicos dan la razón a la aseveración de Aristóteles, otorgando una amplia capacidad a nuestro sentido táctil para percibir señales dentro de un rango de frecuencias y magnitudes muy amplio. El sentido del tacto implica muy variados tipos

de receptores cutáneos, que detectan variaciones en los niveles de presión, temperatura, vibraciones o incluso dolor.

Además, y estrechamente relacionado con el sentido del tacto se encuentra nuestro “sexto sentido” que Aristóteles olvidó, y que encuentra su símil en los sensores internos de los robots; nuestra capacidad kinestésica de conocer la propia posición de nuestro cuerpo en el espacio. Esta información nos la proporcionan nuestros propios sensores internos; músculos, articulaciones y tendones. Cuando entramos en contacto o manipulamos un objeto, nuestro sentido del tacto y nuestro sentido kinestésico funcionan de manera conjunta, permitiéndonos realizar la operación deseada.

Por tanto, es nuestro sentido kinestésico el que nos permite conocer las fuerzas, pares o movimientos aplicados sobre nuestro cuerpo. Esta reflexión kinestésica y su aplicación práctica a la teleoperación de robots constituyen la motivación de esta Tesis.

## 1.2. Definiciones

La palabra “haptics” hace referencia a nuestro sentido kinestésico. Los interfaces hápticos permiten al operador mantener la sensación de interacción con el entorno remoto. En ocasiones, y gracias a la diferencia en la frecuencia de las señales kinestésica, visual y sonora, la realimentación háptica se combina fácilmente con realimentación óptica o acústica.

Para obtener un buen comportamiento del sistema completo, se deben considerar tres factores que afectan principalmente a la sensación de interacción:

- La naturaleza de la señal realimentada, que puede provenir de un entorno físico real, un entorno virtual generado por computador o una combinación de ambos.
- El control del propio dispositivo háptico.
- Las características mecánicas del propio dispositivo.

Los sistemas de teleoperación con realimentación son esquemas bilaterales que incluyen una zona remota, una zona local y una comunicación entre ambas, de forma que el operador, colocado en la zona remota, pueda interactuar con los objetos de la zona remota.

De las definiciones anteriores se desprende que la única diferencia entre un sistema háptico y los sistemas teleoperados reales afecta a la generación de la señal de realimentación. En el primer caso, la zona remota no existe físicamente, y es simulado via software, mientras que en el segundo caso, la zona remota está compuesta por un robot manipulador, su controladora y el propio entorno con sus leyes físicas.

### 1.3. Motivación y génesis de la Tesis

La teleoperación comprende el conjunto de tecnologías que permiten a un operador la realización de tareas a distancia, teniendo su aplicación específica en el desarrollo de tareas en entornos que conllevan un riesgo al operador o bien son inaccesibles por éste.

Un sistema de teleoperación con control bilateral realimenta hacia el dispositivo maestro señales con información acerca del esclavo y de su interacción con el entorno remoto (bien posiciones y velocidades, bien fuerzas y momentos), con la finalidad de que el operador posea las mismas sensaciones que si estuviera realizando físicamente la tarea en el entorno remoto.

Existen multitud de referencias de plataformas experimentales de sistemas teleoperados de  $n$  grados de libertad, pero la implementación real de todos estos sistemas se fundamenta en el supuesto que el maestro y el esclavo son estables en el rango de impedancias de las condiciones de trabajo especificadas. Esto implica que se utiliza el maestro y el esclavo con comandos de alto nivel, propios de las librerías de control de cada dispositivo, lo cual limita los experimentos que se pueden realizar en estas plataformas.

Sin embargo, no se encuentran en la bibliografía muchas referencias explícitas de experimentación en control bilateral con entornos reales de varios grados de libertad (si las hay con sistemas remotos simulados) y que trabajen con maestros de arquitectura abierta sobre los cuales se puedan implementar diferentes reguladores y esquemas de control.

### 1.4. Estructura de la Tesis

La Tesis se enmarca dentro del campo de la experimentación de control bilateral de sistemas de  $n$  grados de libertad. Para poder llevar a cabo este estudio se ha tratado de analizar las tres componentes principales de un

esquema teleoperado:

- Zona local: que abarca entre otros aspectos el dispositivo maestro como interfaz mecánica y la interfaz local del operador.
- Comunicaciones: o la configuración y naturaleza de la arquitectura de control establecida entre la zona local y la zona remota.
- Zona remota: que engloba el dispositivo esclavo y su interacción con su entorno. En esta Tesis se ha trabajado tanto con entornos remotos reales como entornos dinámicos simulados mediante ordenador.

En función de esta división, los capítulos de la Tesis se estructuran de la siguiente forma:

En el capítulo 2 se presenta una visión general de las plataformas de experimentación en teleoperación que actualmente pueden encontrarse en diferentes laboratorios de robótica. Para ello se ha seguido la misma estructura temática que en la tesis, abarcando aspectos relativos a la zona local, la zona remota y las diferentes posibilidades de interconexión entre las mismas. También se añade un breve estado de arte de robots paralelos en teleoperación, dado que uno de los resultados de la tesis ha sido el diseño de un maestro paralelo como interfaz local.

El capítulo 3 engloba los aspectos de diseño y modelado del citado maestro paralelo de arquitectura de control abierta. Gracias a la estructura cinemática 6-URS, y a los algoritmos de control basados en los modelos desarrollados, el dispositivo diseñado presenta una serie de características que lo hacen apropiado para su utilización como dispositivo kinestésico. Se justifica la necesidad de dicho maestro y se realiza un análisis del mismo basado en la dinámica multicuerpo.

El capítulo 4 presenta un breve estudio de la dinámica computacional y las herramientas necesarias para el desarrollo de un simulador de tareas teleoperadas sobre el cual se puedan implementar diferentes esquemas de control bilateral. La obtención de este simulador ha permitido la consecución del objetivo inicial de la tesis, poder experimentar diferentes arquitecturas de control entre diferentes dispositivos robóticos.

El capítulo 5 muestra la experimentación llevada a cabo con la plataforma háptica desarrollada. Se implementan esquemas de teleoperación clásicos

entre los diferentes dispositivos, y se solucionan los diferentes problemas que aparecen en la experimentación en teleoperación. Para ello se hace uso de los resultados anteriores.

El capítulo 6 muestra una aplicación sobre un entorno remoto real de un esquema de teleoperación. Para ser capaces de experimentar a bajo nivel con diferentes esquemas de control bilateral, ha sido necesario “acceder” a los servos del propio robot esclavo, de forma que no nos encontráramos limitados a enviar comandos de alto nivel a la controladora del robot esclavo. El capítulo muestra el modelado del robot serial seleccionado y desarrolla brevemente el aspecto tecnológico que permite el acceso a los servos del mismo.

Finalmente, el capítulo 7 presenta una serie de conclusiones obtenidas durante el desarrollo de la tesis y un plan de trabajo que pretende ampliar las prestaciones de las herramientas desarrolladas.

## 1.5. Contribuciones aportadas

El presente trabajo contiene algunas contribuciones originales al mundo de la teleoperación. Algunas de ellas en orden de aparición son:

- Se desarrolla un estudio desde el punto de vista fisiológico y ergonómico del diseño de interfaces mecánicas de operador.
- Se presenta un estado de arte desde un punto de vista tecnológico de los sistemas de teleoperación actuales.
- Se desarrollan algoritmos de análisis de un robot paralelo de cadena cinemática tipo 6-URS, y se muestra la implementación de diferentes algoritmos de control basados en las soluciones de dichos algoritmos.
- Se ha obtenido un dispositivo háptico de 6 grados de libertad y estructura paralela capaz de funcionar como interfaz kinestésica en cualquier tipo de implementación de un esquema de control bilateral (es capaz de reflejar fuerzas y/o posiciones).
- Se muestra la implementación hardware y software de una plataforma para la experimentación háptica con arquitecturas de control bilateral.
- Se presentan los requisitos necesarios de cualquier dispositivo mecánico para poder “conectarse” a la plataforma experimental desarrollada.

- Se dispone de un método de identificación y obtención de los parámetros inerciales válidos para el modelo dinámico de un robot serial, y del procedimiento de “apertura” de la controladora del robot seleccionado como esclavo en una aplicación de teleoperación real.





# CAPÍTULO 2

## PLATAFORMAS DE TELEOPERACIÓN ACTUALES. Perspectiva histórica

---

*“Claramente, cualquiera que desee construir un dispositivo para comunicar una sensación de contacto remoto a un usuario debe tener en cuenta el rango dinámico de los receptores de contacto, con particular énfasis en la adaptación a ciertos estímulos”*  
*Kalawsky, 1993*

### 2.1. Introducción

Para obtener una visión global de las plataformas de teleoperación actuales, es conveniente disponer, en primer lugar, de unas referencias históricas acerca del desarrollo de la teleoperación. El gráfico de la figura 2.1 muestra los principales logros de la teleoperación en las últimas décadas. Estos avances se encuentran clasificados según los sectores en los que han sido aplicados, lo

que da una perspectiva global de la evolución de la teleoperación, así como de las aplicaciones que la han impulsado. En los últimos años se han entrelazado los avances tecnológicos producidos en los diferentes sectores, a la vez que han surgido nuevos campos de aplicación.

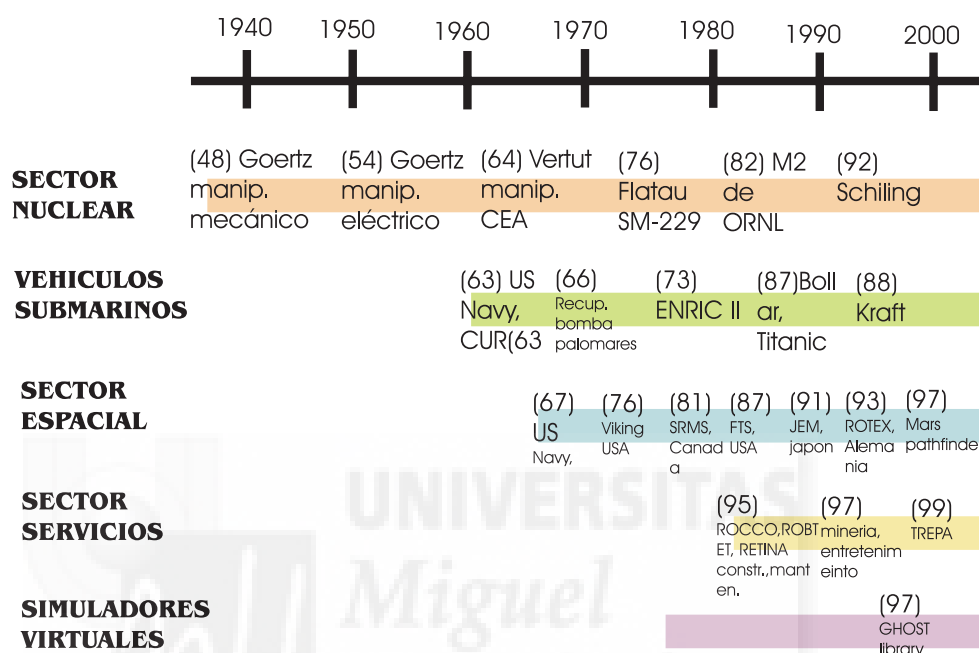


Figura 2.1: Principales avances en teleoperación, agrupados según sectores de aplicación

En cualquier esquema de teleoperación (figura 2.2) se distinguen necesariamente los siguientes elementos:

- Zona local; en la cual el operador interactúa con un dispositivo maestro que en el caso de control bilateral debe ser capaz de proporcionar información del entorno remoto hacia el operador. De los múltiples aspectos de la investigación en diseño de entornos locales [80], la presente Tesis se centra en el diseño de una interfaz mecánica de 6 grados de libertad con capacidad para la reflexión de esfuerzos.
- Zona remota; en la cual el dispositivo esclavo interactúa con el entorno remoto. Cualquier dispositivo robótico puede ser utilizado como esclavo

en un esquema de teleoperación, si bien la gran parte de los mismos disponen de una arquitectura de control cerrada que impide el acceso a bajo nivel de los servos del robot. Esta característica conduce a que la mayoría de plataformas de teleoperación se encuentran implementadas a nivel de comandos del robot.

- Comunicaciones; Las diferentes posibilidades de conexión entre la zona local y la zona remota así como la naturaleza y configuración de dichas conexiones proporciona un criterio de clasificación de las plataformas de teleoperación. En la literatura se encuentran ejemplos que van desde plataformas que utilizan protocolos estándar de comunicaciones (como TCP/IP) [138],[13] entre tareas de tiempo real que se ejecutan en la zona remota y en la zona local, hasta soluciones en las cuales una única tarea de tiempo real controla maestro y esclavo que se encuentran comunicados eléctricamente [28], [113], [170].

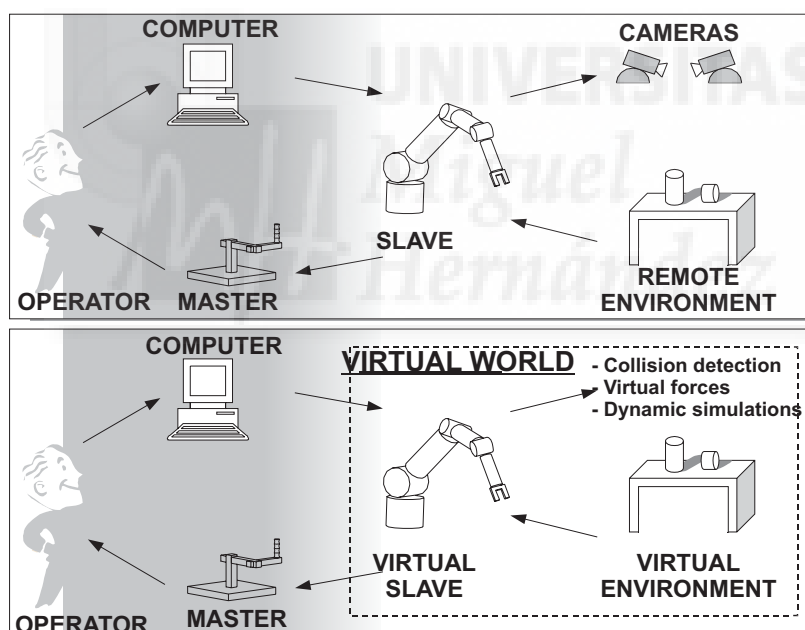


Figura 2.2: Elementos de un sistema teloperado y sistema háptico

Además de los elementos anteriores, las dificultades esbozadas, relativas a los problemas de sincronización de las tareas de control en el entorno local

y remoto, unidas a la creciente capacidad de cómputo y modelado de sistemas mecánicos están motivando un gran interés acerca de los simuladores virtuales, bien como sustitutos de la zona remota, bien como ayuda a las tareas de teleoperación.

### **2.1.1. Estructura del capítulo**

Este capítulo se estructura presentando un estado del arte que abarque los elementos constitutivos del esquema anterior:

En primer lugar se repasan los aspectos concernientes a los dispositivos maestros utilizados en teleoperación. Para poder realizar un análisis formal que incluya aspectos de diseño de dichos dispositivos, se deben tener en consideración los aspectos fisiológicos relacionados con la tarea desempeñada, de forma que se incluya al operador que va a hacer uso del maestro. En relación con la magnitud y fase de las fuerzas que intervienen en dicha tarea, se repasan brevemente las diferentes tecnologías de actuadores utilizados en teleoperación. Finalmente se presenta una clasificación de los diferentes dispositivos kinestésicos que pueden encontrarse actualmente, realizando una comparativa entre los mismos.

El segundo elemento constitutivo de cualquier esquema de teleoperación lo constituye el entorno remoto. En la presente Tesis se ha trabajado indistintamente utilizando un entorno remoto real que incluya un dispositivo robótico abierto como con un entorno remoto simulado. Por tanto, se presenta tanto un estudio de los dispositivos remotos utilizados en teleoperación real como un breve resumen de los simuladores robóticos y sus aplicaciones a la robótica.

El tercer pilar del estudio previo lo constituye la propia plataforma de teleoperación, capaz de integrar los dispositivos maestro y esclavo. Se presenta una clasificación de los esquemas clásicos de control bilateral y se repasan los conceptos de estabilidad apoyándose en la analogía del cuadripolo eléctrico.

Finalmente, y debido al diseño elegido para el maestro construido en el capítulo 3, se realiza un estado del arte de robots paralelos dentro del campo de teleoperación.

## 2.2. Estado del arte de maestros con reflexión de esfuerzos de esfuerzos

### 2.2.1. Aspectos fisiológicos relativos al diseño de interfaces maestras

*Tocar* puede definirse como la sensación provocada cuando la piel se estimula por medios mecánicos, térmicos, químicos o eléctricos [55]. Hablar sobre la fisiología del tacto es complicado debido a una serie de factores, como por ejemplo las características temporales y espaciales del receptor. Su saturación, adaptación, etc., tienen influencia en su capacidad de percepción o en la complejidad del sistema nervioso central. Los aspectos tratados en este apartado tienen una especial importancia a la hora de diseñar una interfaz háptica adecuada para una tarea.

En literatura especializada se encuentran varios estudios sobre la ergonomía de los dispositivos hápticos y la fatiga que producen [3]. Estos estudios modelan el brazo humano como un robot serie redundante, y se aplican técnicas de optimización sobre funciones de coste definidas para caracterizar la comodidad de utilización por parte del operador. Los resultados de estos estudios muestran un espacio tridimensional en el cual el operador puede maniobrar con comodidad (figura 2.3).

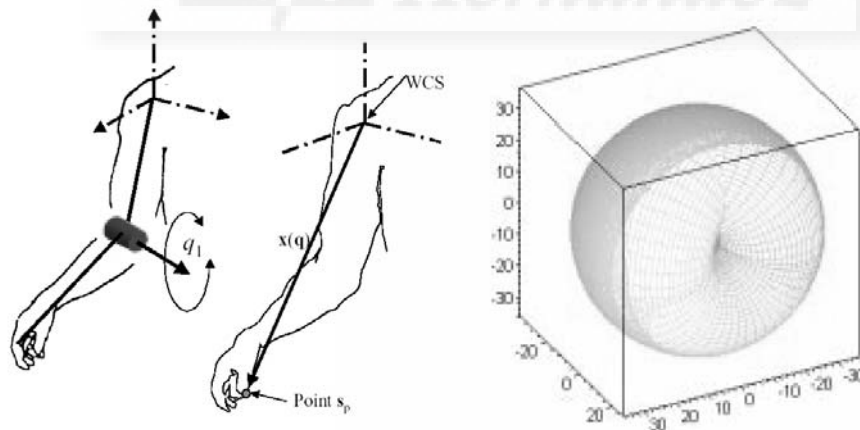


Figura 2.3: Espacio útil para el brazo del operador

Otro aspecto importante es la capacidad sensora del operador, y en particular su capacidad táctil. En particular, las manos son lo más frecuentemente usado en las interfaces hápticas, y por tanto, sus propiedades son muy importantes cuando diseñamos una nueva interfaz. En concreto, los dedos son una de las partes más sensitivas de la piel, teniendo hasta un máximo de 135 sensores por centímetro cuadrado en la yema de un dedo. Además, el dedo es capaz de sentir una vibración de hasta 10,000 Hz cuando siente texturas y fuerzas de hasta 230 Hz. A partir de 320 Hz los dedos no pueden distinguir entre dos señales de fuerzas, y por tanto se sienten como vibraciones. Las fuerzas sobre dedos individuales deberían ser de menos de 30-50 N en total. Para el “usuario promedio”, el dedo índice puede ejercer 7 N, el dedo medio 6 N y el dedo anular 4.5 N sin experimentar molestia o fatiga [47].

Diversos estudios muestran que hay una fuerte relación entre las sensaciones sentidas por una mano humana, tal como un objeto deslizándose, y los movimientos que la mano estaba haciendo para adquirir ese conocimiento, tal como sostener un aparato experimental [179]. El sistema háptico humano está construido de dos subsistemas, el subsistema motor y el subsistema sensorial. Hay una fuerte relación entre los dos subsistemas. A diferencia del sistema visual, no solamente es importante lo que el sistema sensorial detecta sino qué movimientos fueron usados para obtener esa información.

En [75] se usa un dispositivo para determinar cómo reaccionaban los operarios cuando sentían que un objeto que sostenían empezaba a deslizarse. El dispositivo consistió en un solenoide pegado a una placa de metal la cual se podía deslizar cuando el solenoide estaba apagado. Ninguno de los sujetos fue “engañado” con la creencia que los objetos se estaban realmente deslizando. Todos ellos notaron que “algo estaba equivocado con el objeto”, pero ninguno comentó que el objeto se comportaba como si se estuviera deslizando.

Las personas usamos dos diferentes formas de exploración háptica: la activa y la pasiva. La **exploración háptica activa** es cuando el usuario controla sus propias acciones. La **exploración háptica pasiva** es cuando la mano o el dedo del usuario es guiado por otra persona. En el caso de exploración en dos dimensiones el error más común es perder el contorno y, en ese caso, el usuario debe gastar una gran cantidad de esfuerzo para permanecer en contacto con el contorno. Sin embargo, cuando el sujeto es guiado, su atención completa puede ser dedicada a identificar el objeto representado.

Muchas formas pueden ser identificadas más rápidamente con exploración háptica pasiva. Algunos experimentos que comparan la exactitud de las estimulaciones táctiles activas contra las pasivas muestran que las hápticas

pasivas son más exactas en identificar formas como un todo. Cuando el dedo de un sujeto era guiado alrededor de un objeto de dos dimensiones, tal como el perfil de un cisne, la probabilidad de identificar el objeto era mucho mayor. Algunos estudios sugieren que los observadores activos cometen “errores” por distracción y pueden tener dificultad al diferenciar entre las trayectorias de exploración erróneas y correctas.

Cuando el operador se enfrenta a una tarea multidimensional, tal como mover un objeto en un espacio de tres dimensiones, [41] demuestra que los usuarios descomponen la tarea en una serie de problemas de una o dos dimensiones. En este caso moverían un objeto en el plano  $x - y$  antes de moverlo a su posición final en la dirección  $z$ .

Se presentan a continuación una serie de datos empíricos relacionados con la fisiología humana. Las características más importantes de los órganos receptores del sentido del tacto se presentan en la tabla 2.1.

#### 2.2.1.1. Fisiología del tacto

La sensación del tacto se centraliza principalmente en la zona de piel libre de vello, como la palma de la mano y los dedos. Los dedos tienen una alta densidad de receptores táctiles. La piel tiene principalmente cinco tipos de receptores; receptores libres (terminaciones nerviosas), corpúsculos de Meissner, discos de Merkel, corpúsculos de Panician y corpúsculos de Ruffini (figura 2.4). La piel con vello contiene además otro tipo de receptor en la base los pelos que detecta movimiento en la superficie de la piel.

Las terminaciones nerviosas están localizadas cerca de la superficie de la piel. Estas terminaciones libres responden a sensaciones de dolor, mientras que el resto de corpúsculos responden a estímulos mecánicos. Los corpúsculos de Meissner representan el 40 % de los receptores táctiles y se localizan justo debajo de la epidermis. Funcionan como sensores de velocidad, ya que se mueven junto con la piel y detectan este movimiento. Los discos de Merkel representan el 25 % de los receptores en la mano y tienen forma de disco, de ahí su nombre. Estos receptores detectan la presión y la vibración. Los corpúsculos de Panician son los de mayor tamaño y representan el 13 % de los receptores en la mano. Están localizados a mayor profundidad, en la dermis y funcionan como sensores de aceleración. Detectan los contactos ligeros y las vibraciones (son más sensibles a unos 250 Hz). Finalmente los corpúsculos de Ruffini tienen una estructura fusiforme y ocupan el 19 % de los receptores de la mano. Detectan la presión en la piel así como los cambios de temperatura.

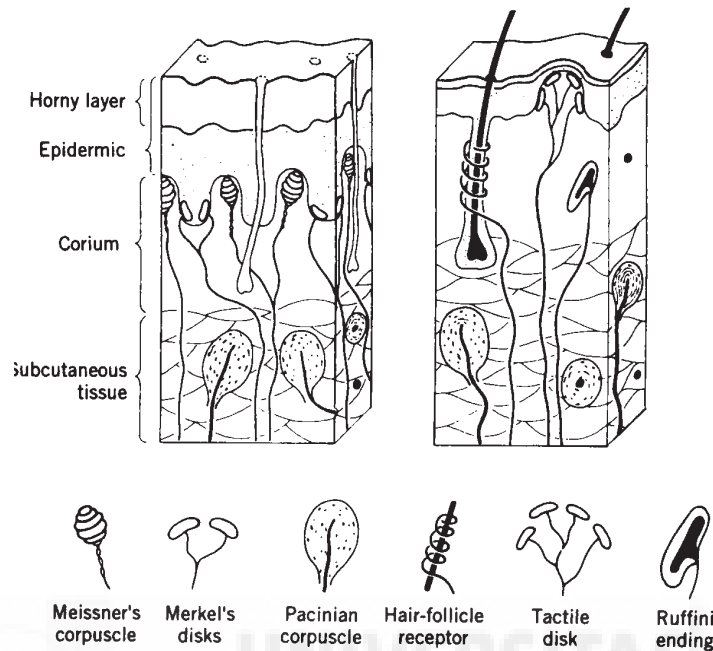


Figura 2.4: Estructura y localización de los receptores táctiles de la piel

### 2.2.1.2. Adaptación sensorial

La característica de adaptación sensorial cuantifica la variación temporal de la descarga de un receptor en respuesta a un estímulo constante. Algunos sensores tienen un tasa de decaimiento lenta (medida en impulsos por segundo); son los llamados sensores *slow adapting* (SA).

Los receptores *rapidly adapting* (RA) tienen un decaimiento de la tasa de impulsos mayor, tal que en un espacio corto de tiempo los estímulos dejan de detectarse. Un ejemplo de cómo actúan estos sensores son los guantes o las gafas, al tiempo de llevarlas dejan de detectarse y la gente no los siente sobre su piel.

Los discos de Merkel son receptores SA de tipo I, y producen una tasa de descargas larga pero irregular en respuesta a las fuerzas aplicadas en la piel. Los corpúsculos de Ruffini son de tipo SA-II y producen una tasa de descargas larga y regular. Los corpúsculos de Meissner son de tipo RA-I, mientras que los de Pacinian se descargan una única vez para cada aplicación del estímulo,



luego no pueden ser sensibles a la presión constante sobre la piel. Se utilizan como sensores de aceleración o vibración. Varios experimentos muestran que responden mejor alrededor de 250 Hz.

Otra característica de los receptores de tacto es el área sobre la cual actúa cada receptor. Cada receptor es sensible para un área alrededor de él. Esta área puede variar desde 1-2 mm hasta 45  $cm^2$  dependiendo del tipo de receptor y de su localización en el cuerpo.

La tabla 2.1 muestra un resumen de las características de estos sensores.

Tabla 2.1: Características de los receptores táctiles.

Tipo de receptor	Adaptación sensorial	Frecuencia de estímulo	Área de aplicación	Función
Discos de Merkel	SA-I	0-10	Pequeña	Detectar cantos
Corp. de Ruffini	SA-II	0-10	Grande	Fuerza estática
Corp. de Meissner	RA-I	20-50	Pequeña	Velocidad
Corp. de Panician	RA-I	100-300	Grande	Aceleración vibración

### 2.2.1.3. Control sensorial humano

Las tareas de exploración táctil, al igual que las tareas de modificación del entorno, están regidas por un lazo de control sensorial. Un ejemplo de esto es la operación de agarre de un objeto deslizante. Aquí la fuerza aplicada depende tanto de la masa del objeto como de su coeficiente de fricción superficial. Las fuerzas de contacto dependen de cómo se cogen los objetos. Según esto la operación de agarrar un objeto se puede clasificar en dos categorías. Por una parte está el agarre fuerte y por otra el agarre de precisión. El agarre fuerte permite una mayor estabilidad y fuerza ya que se utiliza toda la palma de la mano, sin embargo, tiene una menor “destreza” que el agarre preciso, en el que sólo se utilizan los dedos. La figura 2.5 muestra un esquema de esta clasificación.

Varios estudios han mostrado que la máxima fuerza controlable que un hombre puede ejercer es de 400N para los hombres y 228N para las mujeres.

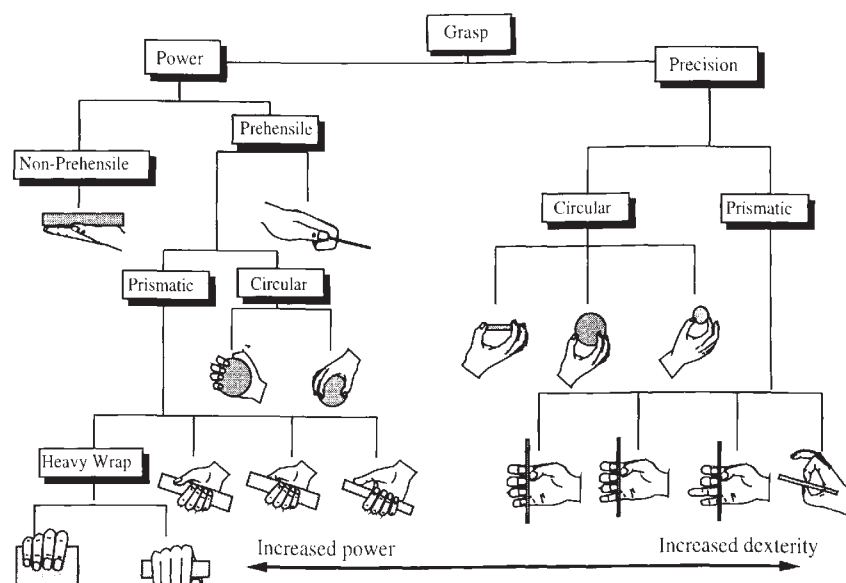


Figura 2.5: Clasificación del agarre de un objeto

Existen varios estudios que muestran la fuerza máxima ejercida por cada uno de los dedos en función de los ángulos que forman las falanges. Estos experimentos muestran que la máxima fuerza para los dedos pulgar e índice es de 50 N mientras que para el resto es de 40 N.

También es importante conocer la máxima fuerza continua que se puede ejercer para periodos largos de tiempo. La fatiga de los músculos afecta tanto al valor pico de fuerza que puede ejercerse como al control de esta fuerza. Algunos estudios [189] han tratado de relacionar la fatiga del lazo de control sensorial con la magnitud de la fuerza ejercida, la duración y el estado de la tarea. Estos estudios muestran que la fuerza continua no debe exceder del 20 ~ 25 % de la fuerza máxima.

Otra variable muy importante a la hora de diseñar una interfaz háptica es la habilidad de distinguir la dureza y la viscosidad del objeto real o virtual manipulado. En la literatura especializada existen varios artículos en los que se puede encontrar resultados necesarios para un correcto diseño de la interfaz [181].

Relacionado directamente con las variables que acaban de citarse está la impedancia del dedo humano. La impedancia relaciona la fuerza aplicada y

el desplazamiento de la mano. La impedancia del operador juega un papel crucial en lo referente a la sensibilidad y estabilidad de las interfaces hombre-máquina.

#### 2.2.1.4. Amplitud de banda de estímulos y amplitud de banda de control

La amplitud de banda de estímulos hace referencia a la frecuencia de los estímulos táctiles, kinestésicos o propioceptivos que son percibidos, mientras que la amplitud de banda de control se refiere a la rapidez con que el operador puede reaccionar a estos estímulos.

Los lazos de recepción de estímulos y de control motriz son asimétricos, es decir que la amplitud de banda de los estímulos (entrada) es mucho mayor que la correspondiente al control motriz (salida).

La habilidad de la mano y los dedos para ejercer fuerzas tiene una amplitud de banda de control de 5-10 Hz, mientras que los estímulos kinestésicos están del orden de 20-30 Hz y las vibraciones táctiles pueden ir desde 0 hasta 400 Hz. El reconocimiento de texturas y rugosidades puede ocupar una banda desde 5000 hasta 10000 Hz. La figura 2.6 esquematiza estos datos.

#### 2.2.2. Actuadores para interfaces maestras

Es obvio que para que exista realimentación de fuerzas las interfaces hápticas deben disponer de actuadores. Sin ellos tan solo podrían ser dispositivos de entrada de datos. Para ello, los actuadores hápticos deben disponer de una serie de características que permitan su uso en las interfaces. La tecnología actual de los actuadores y su desarrollo está íntimamente ligada al desarrollo de nuevas y mejores interfaces. Tanto los actuadores clásicos, como son los eléctricos, los frenos magnéticos, las aleaciones con memoria de forma, los accionamientos hidráulicos o neumáticos como los que actualmente están en investigación, como los basados en fluidos reomagnéticos, actuadores piezoeléctricos, geles poliméricos o actuadores magnéticos deben cumplir una serie de requisitos para poder ser utilizados en la tecnología haptics.

Los actuadores para interfaces portátiles deben ser ligeros para minimizar la fatiga, pero a su vez deben ser lo suficientemente potentes para aplicar la fuerza requerida. Es lo que se conoce como *ratio potencia-masa*.

También se debe de tener en cuenta el *ratio potencia-volumen* debido a que el gran número de grados de libertad que tiene la mano obliga a un gran

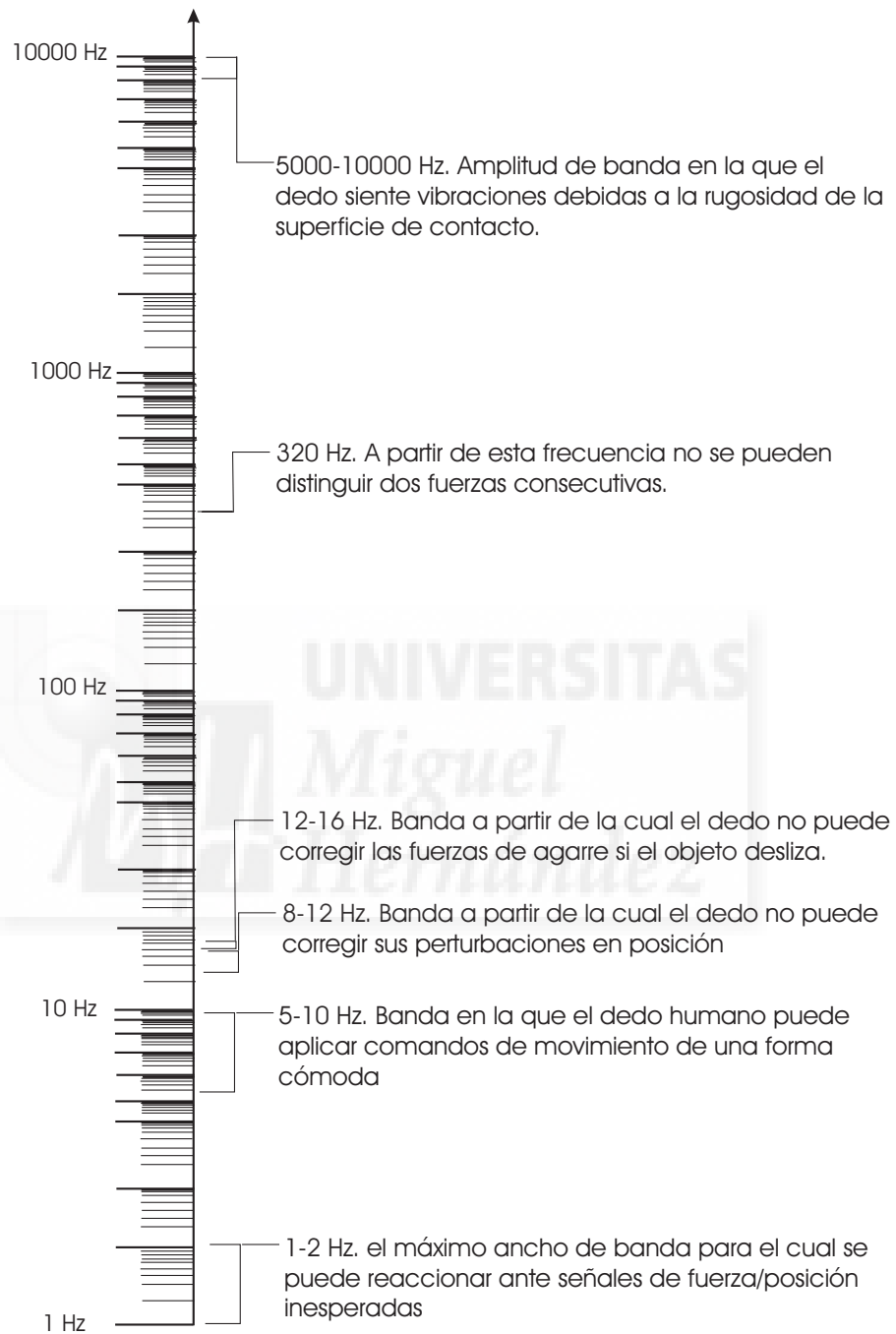


Figura 2.6: Amplitud de la banda de estímulo

número de actuadores en la interfaz. Estos actuadores deben localizarse a poder ser en las articulaciones, para evitar pérdidas en las transmisiones.

Además deben producir fuerzas no mayores del 25 % de la máxima admisible para evitar la fatiga. Por ejemplo el PHANToM produce fuerzas de 8.5 N, apenas el 20 % del máximo admisible por la mano humana, con lo cual evita en gran parte la fatiga producida por un uso continuo.

Idealmente, la interfaz debe ser transparente, lo cual quiere decir que cuando no hay interacciones con el modelo real o virtual, no deben de aparecer fuerzas aplicadas sobre el usuario. Esto significa que los actuadores deben ser capaces de seguir el movimiento del usuario rápidamente y sin oposición. Esta característica se conoce como *backdrivability*. Por tanto se requiere un coeficiente de fricción estática mínimo y una inercia baja para los actuadores. Muchos interfaces calculan vía software un efecto que minimiza la fricción e inercia de sus actuadores y eslabones de manera que el usuario no sienta sus efectos. La relación entre la fuerza máxima del actuador y su fricción se define como su *rango dinámico*. Lo ideal es disponer de actuadores con un rango dinámico y un ancho de banda de control lo mayores posibles.

Es fácil darse cuenta de la importancia que tiene la elección de los actuadores en un diseño de una interfaz háptica. Las investigaciones en nuevos actuadores permitirán obtener interfaces con mayores prestaciones. La figura 2.7 muestra una figura comparativa de diferentes actuadores en función de su ratio potencia-masa.

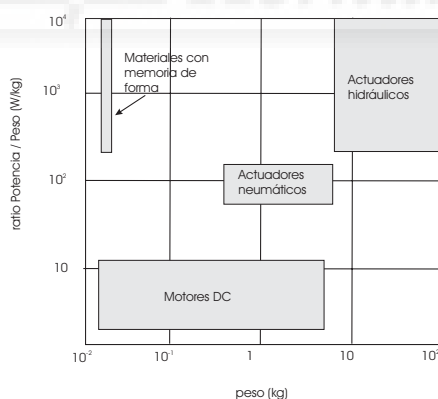


Figura 2.7: Ratio potencia-masa de distintos actuadores

### 2.2.3. Clasificación de las interfaces kinestésicas

Después de revisar la tecnología de los actuadores, se revisan diferentes arquitecturas hardware de lo que se ha llamado “interfaces kinestésicas” o interfaces de reflexión de fuerzas. Estas interfaces se diseñan para reflejar las fuerzas generadas en la simulación o en el propio entorno remoto. Existen varias clasificaciones de las interfaces hápticas en función del tipo de actuador utilizado (eléctrico, hidráulico, neumático, etc.), el tipo de control, o su cadena cinemática. La mayoría de las interfaces actuales son interfaces de escritorio, es decir se soportan sobre una mesa, como los joysticks o aquellos maestros cuyo agarre es un eslabón en forma de lápiz, pero también hay otras estructuras como los exoesqueletos o los manipuladores que se sostienen suspendidos en el techo. La figura 2.8 muestra un esquema de las diferentes estructuras. Otra clasificación de las interfaces se hace en atención a su portabilidad. Los sistemas de escritorio, o sostenidos sobre el techo o un muro no son sistemas portátiles, mientras que los exoesqueletos y las interfaces que actúan sobre la mano del operador (como los guantes) suelen ser portátiles.

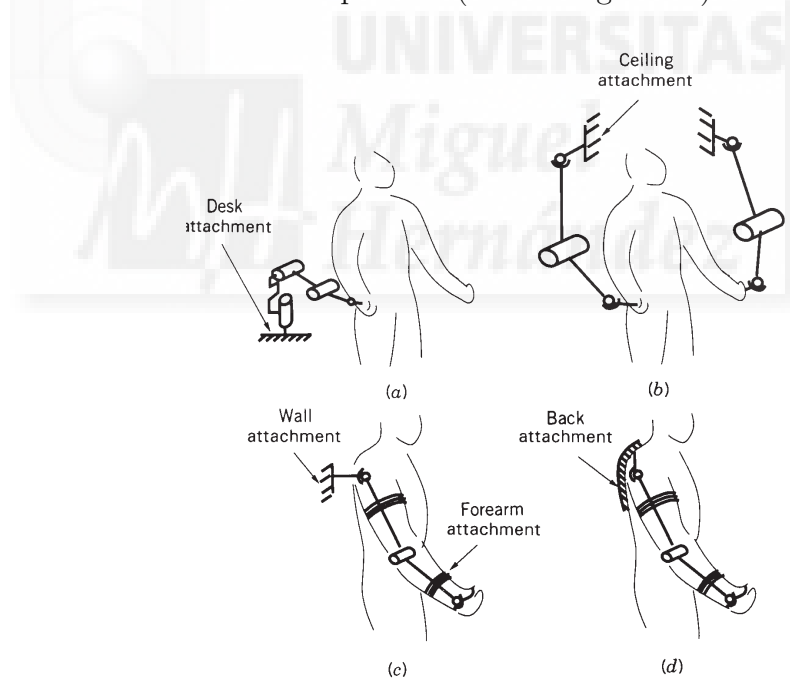


Figura 2.8: Clasificación de las interfaces hápticas según su soporte

### 2.2.3.1. Interfaces de escritorio

Se denominan así las interfaces que se sitúan sobre algún tipo de mesa, soportando ésta el peso de la interfaz. Son interfaces más compactas y económicas que otro tipo de maestros. Al estar constituidas por eslabones pequeños y tener que considerar únicamente el efector final, los problemas de inercia propia de los eslabones son menos acusados que en otro tipo de interfaces como los exoesqueletos.

Se puede distinguir entre joysticks (esféricos, cartesianos o basados en plataformas de Stewart, etc.), interfaces cuyo efector final tiene forma de lápiz como el PHANTOM<sup>®</sup> [127] o las interfaces construidas con cables como el SPIDAR [104].

#### Joysticks

Cuando se piensa en una interfaz de escritorio se visualiza rápidamente un joystick, debido a que este es históricamente la interfaz más utilizada para la entrada de fuerzas en todo tipo de aplicaciones, tanto industriales como de entretenimiento. La figura 2.9 muestra el esquema básico del funcionamiento de un joystick de 2 grados de libertad donde los potenciómetros miden la orientación de los ejes y unos muelles devuelven al efector final a la posición vertical neutra. Para dotar al joystick de la capacidad de realimentar fuerzas se deben realizar una serie de modificaciones. En primer lugar los muelles de retorno deben eliminarse, pues confunden la fuerza reflejada y sustituirse por actuadores. El tipo de actuadores de los joysticks es normalmente eléctrico.

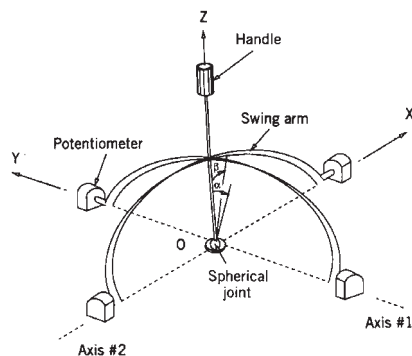


Figura 2.9: Joystick esférico de 2 grados de libertad

### joysticks de configuración esférica

Se basan en la estructura de la figura 2.9. Un ejemplo es el joystick desarrollado por Adelstein y Rosen [6] para el estudio de los temblores de la mano. En él los actuadores son motores dc de imán permanente que proporcionan una fuerza continua máxima de 20N y están controlados mediante PWM. Unos encoder con una resolución de  $0.044^\circ$  informan de la posición del efector final. La amplitud de banda de este joystick es de 48 Hz. Las fuerzas  $F_x$  y  $F_y$  en el efector final se relacionan con los pares proporcionados por los actuadores mediante:

$$\begin{aligned} F_x &= \left[ \frac{\cos \beta \sqrt{1 - \sin^2 \alpha \sin^2 \beta}}{R_0 \cos \beta} \right] \tau_\beta \\ F_y &= - \left[ \frac{\cos \alpha \sqrt{1 - \sin^2 \alpha \sin^2 \beta}}{R_0 \cos \beta} \right] \tau_\alpha \end{aligned} \quad (2.1)$$

donde  $R_0$  es la longitud del efector final y  $\alpha$  y  $\beta$  los ángulos en las dos rotaciones. Es importante notar el desacople cinemático en el plano  $x/y$ .

Otro ejemplo comercial de estos joysticks es el Force Feedback Master 4DOF de EXOS para simulación quirúrgica. La figura 2.10 muestra un dibujo de este joystick. Tres grados de libertad se obtienen sobre el punto de pivote, y el cuarto es lineal permitiendo a la herramienta deslizarse sobre el eje del joystick. Una posible aplicación de esta interfaz es la simulación de cirugía mínima invasiva.

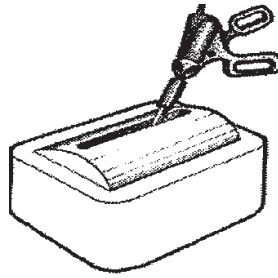


Figura 2.10: Force Feedback Master de 4 grados de libertad

Otro ejemplo comercial son los joysticks desarrollados por Immersion Cor-



poration. Recientemente han aparecido joysticks desarrollados inicialmente para aplicaciones de ocio que incorporan realimentación de esfuerzos, como el Microsoft Force Feedback [US 6,429,849 B1]. Estos joysticks, junto con las librerías de desarrollo de Microsoft<sup>®</sup>, constituyen una opción para la implantación de arquitecturas de teleoperación a nivel de comandos de los dispositivos.

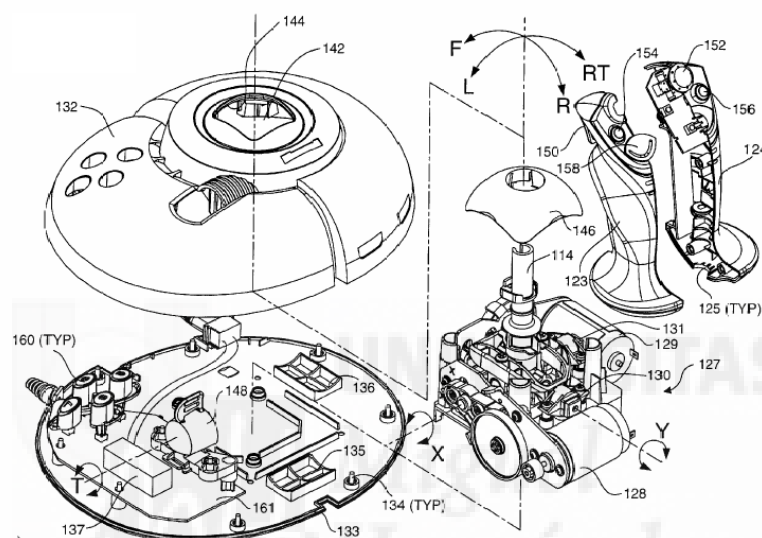


Figura 2.11: Dibujo de la patente [US 6,429,849 B1] del Microsoft Force Feedback

### joysticks de configuración cartesiana.

En la Universidad de Washington se desarrolló una investigación con una interfaz de 2 grados de libertad una serie de experimentos sobre la estabilidad de interfaces de gran amplitud de banda. Estos experimentos dieron lugar a la construcción del Escalibur [US 6,046,563], un maestro de 3 grados de libertad con capacidad para trabajar como display de admitancias (figura 2.12).

Los joysticks comerciales cartesianos más populares son los PER-Force (“Programmable Environment Reality through Force”) desarrollados por Cybernet Systems. Los actuadores son servomotores dc brushless con 5N de

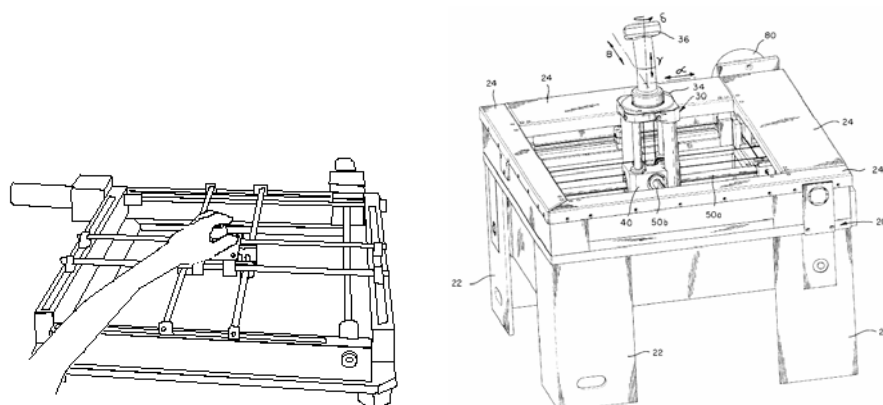


Figura 2.12: Joystick cartesiano de 2 grados de libertad HBFD y dibujo de la patente del maestro Escalibur [US 6,046,563]

fuerza translacional. La amplitud de banda es de 100Hz. Los PER-Force tienen un asa similar a la de los aviones que puede estar dotada de botones que proporcionan grados de libertad adicionales. Estos dispositivos están controlados por un pc 486 y se entrega con librerías de desarrollo en C y válidas para X windows.

### joysticks basados en plataformas de Stewart.

La plataforma de 6 grados de libertad diseñada por Stewart en el año 1965 [183] sirve como base para otro tipo de interfaces de reflexión de fuerzas. La cinemática de esta estructura permite una alta capacidad de carga, un posicionamiento exacto y una repetibilidad alta. Al ser una estructura paralela, los problemas de inercia asociados a los eslabones de una cadena serial se minimizan. Los inconvenientes de este tipo de dispositivos son su pequeño espacio de trabajo y las dificultades computacionales para su control, así como la presencia de configuraciones singulares. Un ejemplo de estos joysticks es la interfaz desarrollada por Millman [140] en la universidad de Northwestern.

Otro ejemplo de joysticks basados en plataformas de Stewart es el desarrollado por Iwata en la universidad de Tsukuba [105],[108]. Actualmente este joystick está comercializado por la firma Nissho Electronics. Se trata de una plataforma de Stewart redundante en la que los cilindros lineales se sustituyen por tres pantógrafos cada uno de ellos accionado por tres motores

de controlados mediante PWM. La figura 2.13 muestra el producto comercial de este joystick.



Figura 2.13: Haptic Master de 6 gdl

### Dispositivos cuyo efector final tiene forma de lápiz.

Con la intención de reflejar la interacción de un apuntador o cualquier otra herramienta en un entorno virtual se han desarrollado una serie de interfaces en las que el efector final tiene forma de lápiz para que sea fácilmente agarrado por el usuario.

Ejemplos de este tipo de interfaces son los desarrollos de Buttolo y Hannaford [51] en los que se utiliza una configuración serie-paralelo con un actuador redundante con el fin de obtener la mínima fricción e inercia, tal y como muestra la figura 2.14. El dispositivo tiene 3 grados de libertad estando el último grado de libertad en una junta rotacional en el efector final. El pico de fuerza aplicado en el efector final es de 13.5 N, mientras que la fuerza continua es de 4.5 N. El espacio de trabajo es pequeño, apenas  $1.5 \text{ cm}^2$  en el plano y 1 cm vertical.

Pero sin lugar a dudas la interfaz háptica más conocida y que cuenta incluso con un grupo de usuarios y desarrolladores de aplicaciones es el Personal Haptic Interface Mechanism; PHANTOM de Sensable Technologies, Inc. [US 6,405,158]. Tal y como se observa en la figura 2.15 la interfaz es un brazo robotizado con morfología serie que finaliza con un dedal donde el usuario puede introducir el dedo. De los seis grados de libertad de la interfaz, los tres grados de posición son activos y corresponden a las primeras

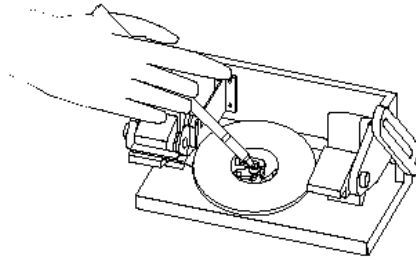


Figura 2.14: Pen-based display de la Universidad de Washington

articulaciones del brazo y los tres últimos del dedal son pasivos. El PHANToM utiliza 3 motores dc brushless con encoders ópticos. Las transmisiones se realizan con cables y poleas, con un mecanismo de reducción que mezcla dos motores capstan con un único cable. La arquitectura del PHANToM permite un desacople de los tres actuadores, simplificando los cálculos para el control. El PHANToM fue diseñado inicialmente en el MIT por Massie y Salisbury [127], y actualmente se comercializan varias versiones de este dispositivo (<http://www.sensable.com>) junto con un software de desarrollo (GHoST) en C++ que permite la implementación de espacios virtuales.

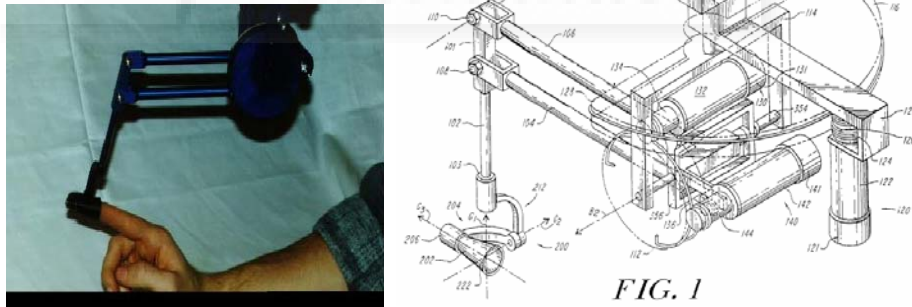


Figura 2.15: PHANToM y dibujo de la patente original

Otro dispositivo es el mecanismo de 6 grados de libertad diseñado por Iwata [107] en el que basándose en el mecanismo de 4 barras del PHANToM se unen los efectores finales de dos PHANToM situados en paralelo mediante

un eslabón en serie conectado a un *stylus* final.

### Interfaces basadas en cables.

Este tipo de interfaces utiliza cables de acero muy finos para reflejar las fuerzas en un efector final. Se distinguen de los anteriores en que los actuadores no mueven un eslabón sino que están situados lejos del efector final y todos aplicados sobre él mediante una estructura paralela. La ventaja de esta configuración es el bajo peso y pequeña inercia en relación al gran espacio de trabajo que proporciona. Sin embargo tiene el inconveniente de que un cable solo trabaja a tracción, dando lugar a fuerzas unidireccionales y a la necesidad de un gran número de muelles y actuadores para poder reflejar fuerzas tridimensionales.

Un ejemplo de estos interfaces es el Space Interface Device for Artificial Reality (SPIDAR) desarrollado por Ishii y Sato [104] en el Instituto de Tecnología de Tokio. En esta interfaz, el usuario inserta sus dedos en un dedal sujetado por cuatro cables. Utilizando un sistema de poleas y motores los cables se utilizan para medir la posición de los dedos y los cables tensado se utilizan para reflejar las fuerzas calculadas. En el prototipo inicial de un solo dedal la resolución de los encoders era de 0.503 mm (con poleas de 16mm). La actual versión del SPIDAR, el SPIDAR II proporciona fuerzas a los dedos pulgar e índice y está utilizándose con éxito en experimentos con operaciones de “pick&place”.

#### 2.2.3.2. Interfaces sostenidas en el techo o en un muro

Dentro de esta categoría se incluyen las interfaces basadas en brazos robot de varias configuraciones, plataformas de movimiento y exoesqueletos que se encuentran anclados al techo. En general estas interfaces son más pesadas, complejas y caras que las interfaces de escritorio. Debido a que suelen tener la capacidad de producir fuerzas de salida mayores, la seguridad es especialmente importante en su diseño [1].

### Brazos robot

Los brazos robot industriales también pueden utilizarse con un control adecuado como interfaces hápticas. Normalmente el extremo del robot soporta una pequeña interfaz háptica a la que dota de ciertos movimientos que

permiten una interacción más realista. Se pueden encontrar varios ejemplos en las investigaciones que actualmente se están llevando a cabo. Un ejemplo es el WYSIWYF (What You can See Is What You can Feel). Se trata de una interfaz visual y háptica que utiliza un brazo robot comercial (PUMA 560) como interfaz háptica (figura 2.16). Otro ejemplo es el “Palmtop display for detrous manipulation with haptic sensation PDDM” en el que se utiliza un pequeño LCD situado sobre una interfaz háptica que permite visualizar el mundo virtual.

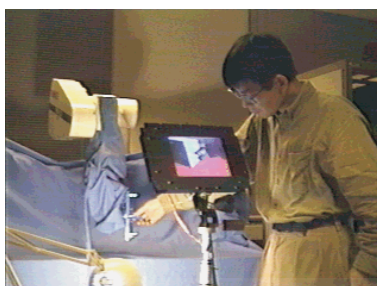


Figura 2.16: WYSIWYF

### Exoesqueletos sujetos al suelo

Los exoesqueletos que están anclados al techo o a algún muro tiene algunas diferencias sobre los exoesqueletos portátiles que se describen más adelante. En los exoesqueletos, las fuerzas se aplican sobre todo el brazo y la muñeca del usuario, luego se necesita un número elevado de grados de libertad y un nivel de seguridad ante fallos muy alto.

El SARCOS (figura 2.17) fue desarrollado por Sarcos Co. [170] para aplicaciones submarinas de teleoperación. La cinemática del brazo se asemeja a la del brazo humano. La muñeca incorpora una unidad modular donde el usuario introduce su mano. Este módulo para la mano tiene menos grados de libertad que la mano y tan solo utiliza los dedos pulgar e índice para reflejar operaciones de agarre de piezas. En total el ancho de banda del brazo varía entre los 10Hz del hombro y los 100Hz de la muñeca. El par máximo que puede reflejar es de 97.7 N-m para el hombro, 50 N-m para el codo, 22 N-m para la muñeca y 5.5 N-m para la mano.

El FREFLEX (Force REFlecting Exoskeleton Master) es un prototipo

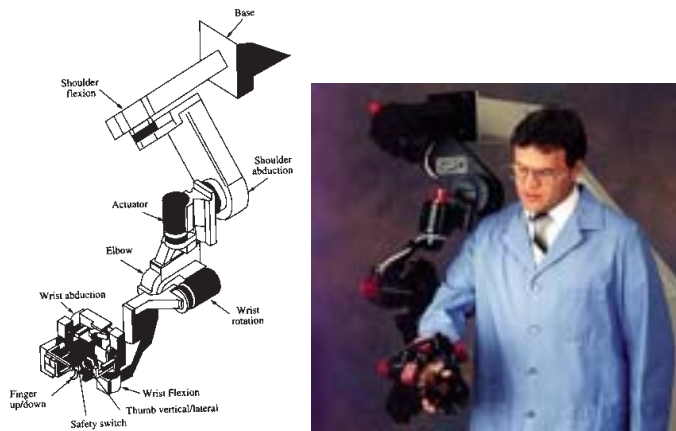


Figura 2.17: SARCOS Dextrous Arm Master

eléctrico desarrollado por el ejército para aplicaciones militares. Dispone de 7 grados de libertad. Los actuadores se encuentran en una consola vertical amarrada al suelo. Otro ejemplo es el PHI system en el que utilizan actuadores neumáticos. La estructura del exoesqueleto es muy ligera, y está hecha de aluminio, pero todas las válvulas de control se encuentran sujetas al respaldo de una silla sobre la que se sienta el operador.

### 2.2.3.3. Interfaces portátiles

Las interfaces portátiles se distinguen de las anteriores en que el soporte sobre el que se apoya todo el peso de la interfaz es el propio usuario. Esta característica obliga a diseñar interfaces de bajo peso y volumen, para evitar problemas de fatiga. Los actuadores deben disponer de un alto ratio potencia-peso y potencia-volumen para poder ser utilizados en estas interfaces. Se clasifican en dos grandes grupos, los *exoesqueletos para el brazo* y los *maestros para la mano* (tipo guantes).

#### Exoesqueletos portátiles

La característica de portabilidad en las interfaces exoesqueletos requiere un compromiso en el número de grados de libertad. La restricción de peso impide un gran número de actuadores, pero a la vez se necesitan un mínimo de 7 grados de libertad para poder reflejar las fuerzas sobre el brazo del



operario. Los ejemplos que se muestran a continuación son exoesqueletos de 5 grados de libertad, debido a que los grados de libertad de la muñeca se implementan en un maestro para la mano que normalmente se añade al exoesqueleto.

Un ejemplo es el proyecto ESPRIT europeo GLAD-IN-ART Arm Exoskeleton [35]. Este proyecto continua en evolución con el exoesqueleto PERCRO. Otro ejemplo es el EXOS Force ArmMaster, desarrollado por EXOS Co. Esta interfaz dispone de 5 grados de libertad al igual que la anterior. Una estructura suspendida por encima del hombro proporciona 3 grados de libertad a la parte superior del brazo, mientras que los otros 2 grados de libertad se sitúan en la parte baja del brazo.

### **Maestros para la mano**

Los maestros para la mano son interfaces hápticas que aplican fuerzas sobre la mano del usuario (y en ocasiones sobre la muñeca) durante la simulación de tareas de exploración y manipulación. La mayoría de las interfaces de mano actuales colocan los actuadores fuera de la propia mano, bien en el antebrazo o bien en la espalda del usuario, utilizando transmisiones por cables y poleas, lo que aumenta los problemas de fricción y reducen la capacidad de seguir el movimiento del usuario (backdriveability). Actualmente se han diseñado interfaces en las que los actuadores se sitúan en la propia mano, reduciendo así el problema de backdriveability [47]. Las interfaces para la mano se pueden clasificar en exoesqueletos para la mano y guantes sensibles (sensing gloves).

El primer ejemplo de exoesqueleto para la mano es la interfaz de Iwata en la universidad de Tsukuba [106]. La interfaz, de tan solo 0.25 kg de peso, produce realimentación de fuerzas a los dedos índice y pulgar.

Otro ejemplo es el LPR Hand Master del Laboratorio de Robótica de Paris [40], [58] que ofrece realimentación a todos los dedos de la mano. Al igual que la interfaz de Iwata, el LPR Hand Master posee una estructura exoesquelética de plástico y aluminio en la parte trasera de la mano y transmisiones mediante cables y poleas.

Otro ejemplo es el SAFIRE (Sensing and Force Reflecting Exoskeleton) diseñado por EXOS Co. Esta interfaz es capaz de aplicar fuerzas a los dedos pulgar e índice y a la muñeca del operador. En total posee 8 grados de libertad activos, 3 en cada dedo y 2 más en la muñeca. El SAFIRE está accionado por motores dc y conectado mediante cables y engranajes a los eslabones de la



estructura. SARCOS Co. también ha desarrollado en colaboración con la universidad de Uath y el MIT una interfaz adaptable a su brazo exoesquelético con flexión en cuatro dedos.

El último ejemplo que se muestra es el RUTGERS MASTER [47], que se diferencia de los anteriores en que los actuadores se sitúan en la parte frontal de la mano. El diseño del RUTGERS MASTER permite que los actuadores desarrollen su potencia directamente sobre la mano del usuario, evitando los problemas originados por las transmisiones. Los actuadores son micro-cilindros de tipo neumático y se colocan formando una “L” sobre la palma de la mano.

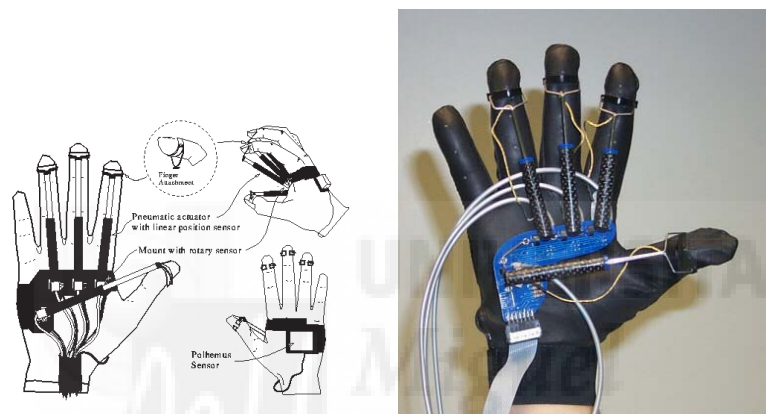


Figura 2.18: RUTGERS MASTER II

## 2.3. Estado del arte de esclavos utilizados en teleoperación

Cualquier dispositivo robótico puede ser utilizado como esclavo en una arquitectura de teleoperación. Sin embargo, es importante resaltar las diferencias que existen en cuanto al control de los mismos, (y que serán revisadas en el siguiente apartado), desde el mismo momento en el que la figura del operador aparece en el esquema de control. Esto ha hecho que los robots utilizados como esclavos hayan sido en su mayoría diseñados específicamente para dicho propósito. Sin embargo recientemente aparecen artículos de experimentación con robots industriales a los cuales primeramente ha habido que

acceder a su sistema de control [28]. Además, robots industriales están siendo utilizados en arquitecturas de teleoperación en las cuales la comunicación se realiza a nivel de comandos del robot [13],[138].

Utilizando los diferentes sectores industriales que aparecen en la figura 2.1, se muestra a continuación un breve resumen de los telemanipuladores utilizados en diferentes campos de la teleoperación.

### **2.3.1. Telemanipuladores en el sector nuclear**

Tal y como se muestra en la primera fila de la figura 2.1, a finales de los 40 se desarrollaron los primeros mecanismos que permitían trabajar sobre elementos peligrosos a distancia, garantizándose de este modo la seguridad del operador. Estos trabajos tuvieron su origen en el riesgo que encerraba la manipulación de elementos radiactivos, [88] y [124]. Los primeros desarrollos de interés fueron realizados en 1948 por Raymond Goertz, en el Argonne National Laboratory (ANL), [87]. El primer prototipo realizado puede ser considerado como un manipulador mecánico avanzado. Dicho dispositivo constaba de diferentes articulaciones que permitían transmitir los movimientos del operador, que se encontraba en un extremo, hasta el otro extremo en el que se hallaba una pinza, que manipulaba directamente los objetos. La continuación de estos trabajos dio como resultado, en 1954, el primer telemanipulador maestro-esclavo de accionamiento eléctrico y con reflexión de fuerza. Este prototipo sí se considera un verdadero telemanipulador. Constaba de un brazo articulado de accionamiento eléctrico, denominado esclavo, cuyo controlador recibía órdenes de movimiento para cada una de sus articulaciones. Las órdenes eran generadas por el operador a través de un maestro, sobre el que se reflejaba la fuerza que era realizada durante la manipulación de los objetos. Posteriormente, se desarrollaron nuevos dispositivos más sofisticados, que disponían de realimentación visual mediante cámaras de televisión, dando lugar a los modelos E3 en 1958, y E4 en 1963.

También durante los sesenta fue desarrollado el telemanipulador Mascot, realizado por ENEA en Italia, con una configuración similar a la del E3. Avances sobre este prototipo dieron lugar a modelos de altas prestaciones, como el Mascot IV, utilizado por el Jet Propulsion Laboratory para diferentes tareas de mantenimiento [160].

En 1954, Ralph Mosher desarrolló un nuevo telemanipulador, en General Electric, denominado Handyman, con una unidad de control de accionamiento hidroeléctrico, [142]. Constaba de dos brazos articulados en cuyos extremos

se localizaban unas pinzas de cuatro grados de libertad cada una. El maestro utilizado, con un aspecto muy aparatoso, tenía forma de exoesqueleto rodeando los brazos del operador.

A partir de estos desarrollos aparecieron nuevos manipuladores con mejores características, que poco a poco incorporaron el uso de microprocesadores aumentando significativamente sus prestaciones. En Francia, Jean Vertut y Carl Flatau, desarrollaron durante los años 70 los telemanipuladores MA-22 y MA-23 [184]. Estos trabajos fueron realizados por la Agencia de la Energía Atómica francesa, CEA (Commissariat à l'Énergie Atomique). En un principio fueron ideados para la manipulación de materiales radiactivos, y seguidamente se aplicaron en vehículos submarinos. Posteriormente, Carl Flatau, en TeleOperator Systems Corporation, desarrolló un nuevo telemanipulador con características mejoradas sobre el MA-23 denominado SM-229 [83], con el que se pretendía conseguir mejores prestaciones en cuanto a movilidad y habilidad, combinación difícilmente alcanzada hasta entonces.

En los 80 el ORNL (Oak Ridge National Laboratory), en colaboración con Central Research Laboratories, realizó mejoras sobre el E4 del ANL, desarrollando el primer telemanipulador totalmente digital denominado M2 [95]. El cual constaba de una avanzada y compleja interfaz para el operador, en la que destacaba la realimentación visual mediante cámaras de visión y la reflexión de fuerza sobre los maestros. Este telemanipulador y los siguientes derivados de él han sido diseñados para la manipulación de los residuos radiactivos de las centrales nucleares, dentro del programa de gestión de residuos de combustible nuclear del Departamento de Energía de EE.UU.

En los últimos años han sido desarrolladas nuevas familias de telemanipuladores con avanzadas prestaciones. Ejemplo de ello es el telemanipulador Titanic de Schiling, desarrollado en los noventa, y muy utilizado en aplicaciones para la manipulación de materiales radiactivos. Otro ejemplo, en Europa, es la empresa Telerob que desde principios de los noventa fabrica telemanipuladores y vehículos para la gestión de residuos nucleares.

### **2.3.2. Telemanipuladores para vehículos submarinos**

Numerosos vehículos submarinos han sido desarrollados para la realización de tareas submarinas. La aplicación de robots teleoperados en estos vehículos se debe a los riesgos que tiene para el hombre la realización de trabajos submarinos, debido a las extremas condiciones de presión y temperatura, corrientes de agua, etc. Todo ello ha potenciado el desarrollo continuo

de sistemas teleoperados, que poco a poco han mejorado sus prestaciones, a la vez que se han especializado en la realización de determinados tipos de trabajos submarinos.

El origen de los primeros vehículos submarinos, incorporando telemanipuladores, fue debido a la Armada de EE.UU., los cuales eran guiados desde la superficie por un buque. Sus primeras misiones fueron la recogida de los torpedos. Posteriormente, pasaron a ser utilizados para la recogida de restos de accidentes, como la bomba caída en Palomares en 1966, o el resto de naufragios de barcos o accidentes aéreos. El programa desarrollado para la realización de estos vehículos comenzó en los años 60 [188]. Los primeros en aparecer fueron denominados Snoopy, y SCAT (Sumersible Cable-Actuated Teleoperator - Teleoperador Sumergible Accionado por Cable). Portaban un solo manipulador con dos o tres grados de libertad, e incorporaban una o dos cámaras de vídeo. Tenían como misión la recogida de torpedos y la inspección del fondo marino. A continuación, aparecieron los vehículos del programa CURV (Cable Controlled Underwater Recovery Vehicle - Vehículo de Recuperación Submarina Controlado por Cable). Estos alcanzaban una mayor profundidad de trabajo, hasta 600 metros, y disponían de hasta dos manipuladores, con más grados de libertad que los anteriores, lo que facilitaba la realización de los trabajos submarinos. También, se desarrollaron vehículos de uso específico, como los MNV (Mine Neutralization Vehicle) para la recogida de minas, y los NP (Nozzle Plug) para la recogida de los cohetes de la NASA. Durante los años 80, se lanzó un programa más ambicioso por el que se pretendía el desarrollo de vehículos capaces de hacer trabajos de cierta complejidad en el fondo del mar, WSP (Work Systems Package - Conjunto de Sistemas de Trabajo). A estos se les dotó de dos telemanipuladores, en algunos casos un tercer brazo auxiliar, sistemas avanzados de visión, y diferentes tipos de herramientas para la realización de trabajos submarinos.

En Francia, también fueron desarrollados vehículos submarinos para la observación y realización de trabajos submarinos por la Armada francesa. Estos trabajos comenzaron en los años 70 con el ERIC I (Engin de Reconnaissance et d'Intervention à Cable), y el ERIC II [54]; incorporando este último dos telemanipuladores MA-23, capaces de trabajar a una profundidad máxima de 6000 m.

En la actualidad, se están desarrollando numerosos vehículos para diferentes tipos de aplicaciones. Como ejemplo de ello están los sistemas destinados al mantenimiento de tuberías submarinas. La empresa japonesa Tokio Electric Power Inc., junto a Mitsubishi Heavy Industries Ltd., desarrolló un

robot para la limpieza de tuberías submarinas de centrales eléctricas. En el Reino Unido, un consorcio de empresas relacionadas con las perforaciones petrolíferas del Mar del Norte, está desarrollando un vehículo submarino con un telemanipulador guiado por un maestro para la realización de trabajos de inspección y limpieza de tuberías. Una aplicación más conocida de estos vehículos es su utilización en la búsqueda y rescate de buques y aviones hundidos, como fue la inspección del Titanic en 1985 por R. Bollard.

Al igual que el caso anterior de aplicaciones nucleares, en el sector de los trabajos submarinos también han surgido telemanipuladores que posteriormente han sido utilizados diferentes tipos de sistemas teleoperados. Entre estos desarrollos destaca el telemanipulador Grips de Kraft Telerobotics. Sus primeras aplicaciones fueron para trabajos submarinos; pero posteriormente mejoró sus prestaciones incluyendo reflexión de fuerzas en el maestro, aplicándose en la actualidad para la realización de diferentes tipos de trabajos teleoperados.

### **2.3.3. Telemanipuladores en el sector espacial**

Los avances en el sector espacial han favorecido significativamente la evolución de la teleoperación, dando lugar al desarrollo de nuevos telerobots, y numerosos trabajos de investigación. Los principales aspectos estudiados son la aplicación de robots teleoperados a tareas espaciales [141] y la influencia que tienen los retardos en las comunicaciones en la estabilidad de los sistemas teleoperados.

Numerosos han sido los trabajos realizados en torno a la influencia de los retardos en la teleoperación de robots. Uno de los primeros fue el de W. Ferrel y T. Sheridan, [81], que propusieron un modo de trabajo denominado control supervisado, en lugar de las estrategias, habitualmente utilizadas, de mover-y-esperar. Bernard J. Schroer [172], en 1988 desarrolló un simulador predictivo que permitía al operador conocer la evolución de los trabajos remotos. Un nuevo simulador, que también tenía en cuenta los tiempos de retraso, fue desarrollado en el Jet Propulsion Laboratory [34], mejorando significativamente el rendimiento de las tareas teleoperadas.

El primer uso de manipuladores teleoperados tuvo lugar en 1967 por la Unión Soviética, los cuales recogieron muestras de la superficie lunar durante la misión del Surveyor, existiendo un retraso de unos pocos segundos en la comunicación. Una misión de la NASA, en 1977, durante la misión del Viking hizo la misma operación en Marte [65], pero en este caso el retraso en las

órdenes enviadas desde la Tierra hasta la nave espacial era del orden de minutos.

Durante los años setenta fueron desarrolladas las lanzaderas espaciales, y con ellas surgió una nueva aplicación de los telemanipuladores. En este caso no se pretendía teleoperar desde la Tierra, sino que el propio astronauta desde la nave trabaja con un telemanipulador que se encuentra en el exterior. Son numerosos los retos que se plantean, tales como la falta de gravedad, el diseño de manipuladores ligeros, gran campo de acción, etc. Uno de los primeros trabajos fue llevado a cabo por la empresa canadiense SPAR para la NASA, que desarrolló un robot de grandes dimensiones, denominado SRMS (Shuttle Remote Manipulator System - Sistema de Manipulación Remoto para Lanzadera). Su objetivo era realizar diferentes tipos de tareas en el exterior de la nave, principalmente ensamblado de estructuras y recogida de satélites. Otra aportación realizada por la NASA, ha sido la definición de una arquitectura propia para el control de robots teleoperados, denominada NASREM, [7].

A principios de los noventa, los japoneses comenzaron el desarrollo del Módulo Experimental Japonés (JEM, Japanese Experimental Module), como parte integrante de la estación orbital que están diseñando.

En Abril de 1993, fue utilizado por primera vez un telerobot sobre la lanzadera Columbia [98]. Dicho telerobot, denominado ROTEX, fue desarrollado en Alemania. Consta de numerosos sensores que le permiten trabajar de forma autónoma y manual; siendo controlado bien por astronautas, o bien desde tierra. El guiado del telerobot se realizó desde tierra a través de un ratón de 6 gdl sin reflexión de fuerza. A su vez, al operador se le mostraba una imagen estereoscópica del entorno remoto. El sistema de control del robot se encontraba en la nave, y ejecutaba comandos referentes a la tarea a realizar. Dichos comandos hacían uso de la información suministrada de los sensores, lo cual le permitía adaptarse al entorno de trabajo. El retraso de las comunicaciones en las pruebas realizadas fue en torno a 6 segundos, realizándose con éxito tareas de agarre de piezas que flotaban, conexión y desconexión de enchufes, y ensamblado de estructuras.

#### **2.3.4. Telemanipuladores en el sector de servicios**

La teleoperación ha entrado con fuerza en otros sectores a los que en principio no estaba enfocada. Entre éstos se pueden citar los siguientes: aplicaciones de construcción y minería [82], mantenimiento de líneas eléctricas



[15], mantenimiento de instalaciones [120], intervención en desastres naturales [77] o aplicaciones militares [156]. Especial mención merecen los esfuerzos



Figura 2.19: Telemanipuladores en el sector servicios: Retina y Trepas

nacionales en el desarrollo de telemanipuladores para aplicaciones concretas, como la inspección de líneas de alta tensión [16], la excavación y movimiento de tierras [18] o el mantenimiento de palmerales [17] (figura 2.19).

También cabe citar que recientemente se ha fortalecido de forma importante la aplicación de las tecnologías de teleoperación al sector médico. Desde los primeros desarrollos de prótesis o dispositivos de asistencia a discapacitados hasta la más novedosa de la telecirugía, o el telediagnóstico, aunque éste no pertenezca estrictamente al sector de la teleoperación [68].

### 2.3.5. Telemanipuladores virtuales. La realidad virtual como ayuda a la robótica

*“La realidad virtual es la manipulación de los sentidos humanos (siendo actualmente el tacto, la visión y la audición) por medio de entornos tridimensionales sintetizados por computadora en el que uno o varios participantes acoplados de manera adecuada al sistema de computación interactúan de manera rápida e intuitiva de forma que la computadora desaparece de la mente del usuario dejando como real el entorno generado”*

La tecnología de realidad virtual aplicada a la robótica en general, ofrece una serie de posibilidades para nuevas aplicaciones como son el aprendizaje

fuera de línea, la planificación de trayectorias, el ensamblaje de prototipos, etc. Recientemente están apareciendo muchas publicaciones relacionadas con el tema, en las cuales se revisa la situación actual de la tecnología [49]. Se toman como características básicas de un sistema de realidad virtual las siguientes:

- interacción: Rasgos que permiten al usuario manipular el curso de la acción dentro de una aplicación de realidad virtual, permitiendo que el sistema responda a los estímulos de la persona que lo utiliza; creando interdependencia entre ellos. Un aspecto de la interacción es la dinámica del ambiente, que no es más que las reglas de cómo los componentes del mundo virtual interactúan con el usuario para intercambiar energía o información.
- inmersión: Esta palabra significa bloquear toda distracción y enfocarse selectivamente solo en la información u operación sobre la cual se trabaja.
- imaginación (sensación tridimensional): Esta es una característica básica para cualquier sistema llamado de realidad virtual, y tiene que ver directamente con la manipulación de los sentidos del usuario, principalmente la visión.

Durante mucho tiempo las comunidades de investigadores en robótica y en realidad virtual han estado trabajando por separado, sin embargo los beneficios de la realidad virtual han sido reconocidos por científicos e ingenieros para muchas aplicaciones.

Aparte de aplicaciones como el diseño CAD [100], [8], la programación de robots, la simulación de procesos de planta, etc, la aplicación de la realidad virtual a la teleoperación aporta nuevas e importantes características. Problemas clásicos de teleoperación como son un entorno con una pobre realimentación visual o con retrasos en la realimentación obtienen una considerable mejora cuando se emplea realidad virtual y haptics en ellos.

Actualmente empiezan a aparecer varios artículos en los que problemas clásicos de teleoperación se abordan utilizando la tecnología haptics. En todos de los laboratorios de investigación robótica existen proyectos en los que se tratan de resolver este tipo de problemas. Se citan a continuación algunos ejemplos en los que la colaboración realidad virtual-teleoperación está obteniendo importantes avances.



### Supervisión y control cooperativo

La teleoperación clásica requiere un operador controlando un único robot esclavo. Actualmente el papel del operador ha pasado a ser un mero supervisor que controla el robot indirectamente a través de una interfaz. En estos casos, la realidad virtual puede servir como un previsor de los comandos de movimiento, antes de que éstos sean enviados al robot remoto (el real). En la Universidad de California, Blackmon y Stark [36] desarrollaron un modelo de control supervisado utilizando un “script de secuencias de trabajo”. El sistema consistía en una estación gráfica de trabajo en la que había un simulador del entorno remoto. Una segunda estación de trabajo colocada en el lugar remoto controlaba un robot RM501 de Mitsubishi de 5 grados de libertad. El modelo virtual del robot era controlado por el operador con un par de joysticks de 2 grados de libertad, con los que el operador realizaba la tarea evitando potenciales colisiones. Cada script de la tarea era realizada en primer lugar en el simulador y enviada inmediatamente al entorno real. Los experimentos demostraron que este sistema daba mejor resultado que la manipulación directa.

Los símbolos abstractos usados en los simuladores permiten que el maestro se represente como una mano virtual en lugar de cómo un robot. Estos símbolos se utilizan en investigaciones sobre el control cooperativo de un único robot. Un ejemplo de control cooperativo ha sido realizado en el Laboratory of Human and Machine Haptics del MIT por Basdogan y Srinivasan [32].

### Teleoperación con Realimentación Visual Pobre

Un problema presente en todos los sistemas de teleoperación es la degradación de la información sensorial del entorno remoto debido a una pobre realimentación visual. Este caso es muy común en teleoperación, dado que en muchas ocasiones se trabaja en entornos con poca visibilidad, un campo de visión limitado o una escala microscópica.

Oyama [154] desarrolló un sistema de teleoperación en el que la realimentación visual se degradaba con humo. El operador controlaba un brazo maestro cinemáticamente equivalente al robot esclavo de 6 grados de libertad.

Otro ejemplo de teleoperación con una pobre realimentación visual es la operación de robots en entornos submarinos. Incluso los operadores más experimentados pueden desorientarse trabajando en aguas turbulentas que

contengan una red compleja de tuberías. Esto puede ser fuente de accidentes o pérdidas de equipo. En muchos casos se utilizan técnicas de realidad aumentada o simuladores en los que se observa en la misma pantalla el robot simulado y la imagen proporcionada por la cámara remota

### Teleoperación con Retardos Temporales

Otra dificultad asociada a la teleoperación es la presencia de retardos en las comunicaciones debido a largas distancias. Este caso ocurre tanto en aplicaciones espaciales como terrestres en las que por cualquier razón la comunicación es lenta. En un esquema de teleoperación clásico, el usuario debe adoptar una estrategia de “ordena y espera”, lo cual conduce a tiempos inaceptables en tareas de larga duración.

Una solución a este problema es el robot fantasma (phantom robot) propuesto por Bejczy en el Jet Propulsion Laboratory [34]. Este robot fantasma consiste en una representación gráfica de un robot remoto que se superpone a la imagen de video del robot real. El robot virtual se mueve instantáneamente en respuesta a los comandos de movimiento del operador, de tal forma que se utiliza como previsor de la operación remota y permite una teleoperación más rápida y eficaz. Los experimentos que se hicieron con un robot Puma permitieron observar una reducción del 50% en el tiempo necesario para completar la tarea en sistemas con retardos de hasta 4 segundos.

Los retardos temporales afectan a otras modalidades sensoriales además de la realimentación visual. Normalmente la realimentación de fuerzas es beneficiosa para la realización de tareas debido a que el operador tiene una mejor sensación de las fuerzas de contacto, la estabilidad y otras características físicas de los objetos manipulados. Sin embargo, retardos tan pequeños como 0.1 segundo pueden hacer que la realimentación de fuerzas tenga más efectos negativos que positivos, debido a que genere inestabilidades en el sistema. La solución a este problema está actualmente en investigación en varios centros de investigación. En el MITI de Japón se intentó resolver el problema con un esquema basado en la aproximación de Bejczy. En este caso se trabajó con retrasos de 0.5 segundos. Se utilizó un maestro con la misma dinámica que el esclavo. Los comandos de posición introducidos por el operador llegaban al esclavo real con retardo, un simulador virtual utiliza estas mismas entradas para mover el modelo del robot. Las fuerzas de contacto generadas en el simulador son instantáneamente realimentadas al maestro. De momento se han conseguido realizar operaciones muy simples como bor-

dear una pared rígida en las que el esclavo se modelaba como un punto y las fuerzas de contacto se consideraban sin fricción. El modelo utilizado, tanto del esclavo como del entorno y sus interacciones, es muy simplificado, y se reduce a entornos estructurados.

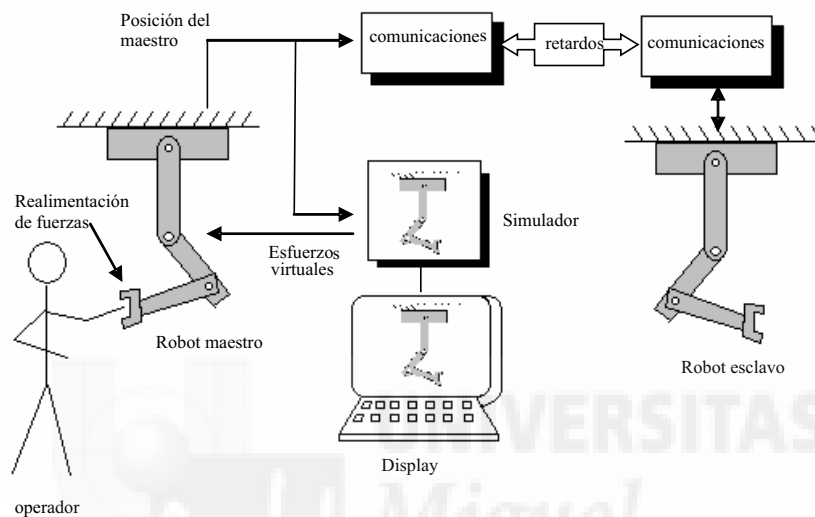


Figura 2.20: Esquema de un sistema de teleoperación para trabajo con retardos

## 2.4. Estado del arte de las comunicaciones: clasificación de arquitecturas de control bilateral

La naturaleza de las comunicaciones y la configuración de las conexiones entre la zona local y la zona remota dan lugar a diferentes arquitecturas de control en teleoperación. Diferentes clasificaciones pueden encontrarse en [158],[2], [21].

### 2.4.1. Clasificación de los esquemas de teleoperación

Basándose en el esquema de 1 grado de libertad de la figura 2.21 Peñin [158] realiza un amplio análisis de los esquemas clásicos de teleoperación, utilizando para ello teoría de control clásica. En este estudio se utiliza el reconocido modelo general de un sistema de teleoperación con control bilateral [190] mostrado en la figura 2.22, en la cual el modelo del operador ha sido simplificado a una masa  $m_{op}$ , con una rigidez  $k_{op}$  y un rozamiento viscoso  $b_{op}$  asociado a su movimiento. El modelo viene descrito por las ecuaciones 2.2 a 2.5.

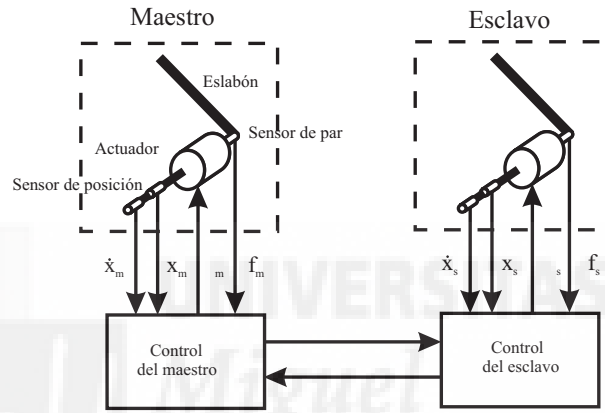


Figura 2.21: Esquema general de plataforma de teleoperación

$$\tau_m + f_m = m_m \ddot{x}_m + b_m \dot{x}_m \quad (2.2)$$

$$\tau_s - f_s = m_s \ddot{x}_s + b_s \dot{x}_s \quad (2.3)$$

$$f_s = m_e \ddot{x}_e + b_e \dot{x}_e + k_e x_s \quad (2.4)$$

$$\tau_{op} - f_m = m_{op} \ddot{x}_m + b_{op} \dot{x}_m + k_{op} x_m \quad (2.5)$$

El citado estudio utiliza los esquemas clásicos posición-posición (figura 2.23) y fuerza-posición (figura 2.24) para presentar un exhaustivo método de análisis de arquitecturas de teleoperación que fundamenta su estudio en:

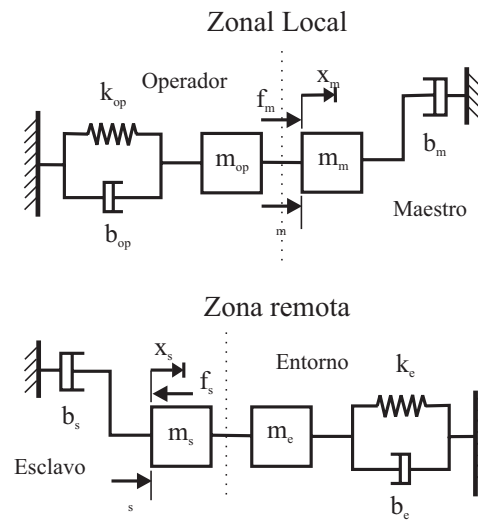


Figura 2.22: Modelo simplificado de plataforma de teleoperación

- Análisis de estabilidad
- Estudio de respuesta temporal
- Análisis en frecuencia (ancho de banda del sistema)
- Inclusión de reguladores PD
- Capacidad de realimentación kinestésica

En este tipo de análisis la naturaleza de las comunicaciones se estudia como un factor que afecta a la estabilidad del sistema en forma de retardos en las comunicaciones [27].

En [21] Arcara realiza una clasificación más extensa de los sistemas bilaterales y desarrolla un controlador basado en los conceptos de pasividad, el IPC (Intrinsically Passive Controller). Este esquema es una generalización del concepto de acople virtual de Colgate [63] a la teleoperación real.

### 2.4.2. Analogía del cuadripolo eléctrico

Otro método de análisis de arquitecturas de teleoperación utiliza una analogía con la teoría del cuadripolo eléctrico para obtener una formulación

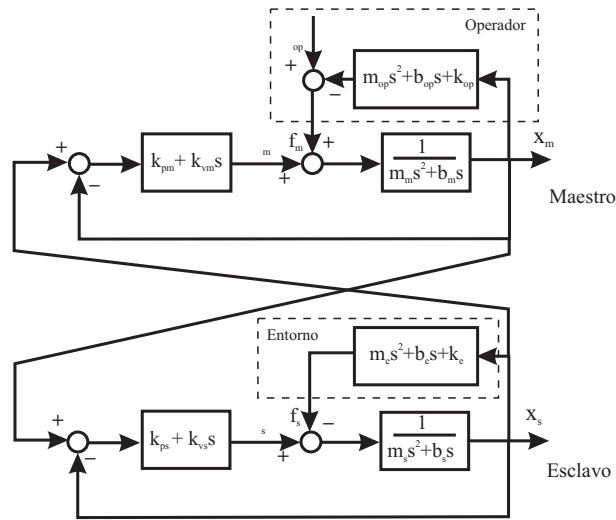


Figura 2.23: Control Bilateral posición-posición mediante teoría clásica

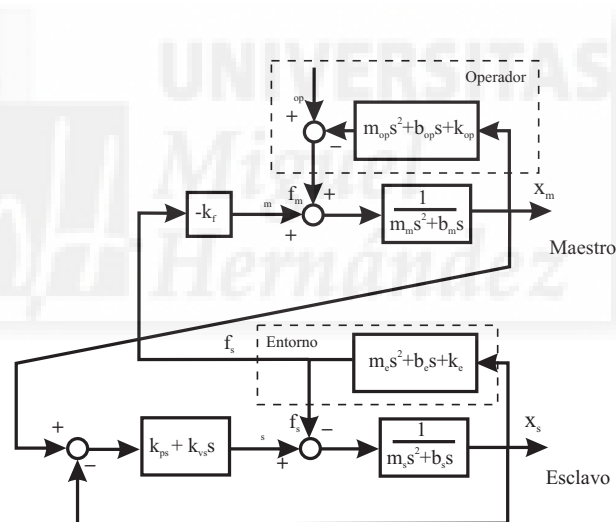


Figura 2.24: Control Bilateral fuerza-posición mediante teoría clásica

de la transferencia energética que se produce entre la zona remota y la zona local.

En 1987 los investigadores del Massachusetts Institute of Technology (MIT), con T.B. Sheridan a la cabeza, plantearon por primera vez la posi-

bilidad de utilización de la teoría del cuadripolo eléctrico para el estudio de sistemas con control bilateral. Dos años después se presentó a la comunidad científica los primeros resultados de esta teoría [161], siendo acogida de forma muy favorable por parte de múltiples investigadores en teleoperación [90], [11], [190], [115], adoptando el nombre inglés de *two-port approach*.

La aparición de esta teoría se basa en la necesidad de contar con un único lenguaje de descripción que permita describir la interacción entre los distintos elementos eléctricos, mecánicos y fisiológicos de un sistema de teleoperación. Se realiza entonces la identificación de todas las variables del sistema en dos tipos de variables generalizadas denominadas de tipo esfuerzo (tensión, fuerza, presión, etc.) y tipo flujo (intensidad, velocidad, caudal, etc.). Este tipo de caracterización es muy habitual para analizar el intercambio de energía entre dos sistemas, como puede ser entre un sensor y la planta sobre la que se realiza la medida [29]. El cociente entre la variable esfuerzo y su variable flujo asociada en un determinado lugar de un sistema se denomina **impedancia** y su inversa **admitancia**, así como el producto de ambas es la potencia en circulación. Si se está hablando de intercambio de energía de un sistema, se hablará de potencia e impedancia de entrada o de salida del mismo.

Se utiliza entonces la teoría del cuadripolo eléctrico [56], también transformada mediante el uso de variables generalizadas. Utilizando esta herramienta, el comportamiento energético global de cualquier sistema (eléctrico, mecánico, etc.) puede ser caracterizado en función del valor que toman las variables esfuerzo y flujo (tensión e intensidad) en cuatro polos, que dos a dos representan los terminales por donde la energía entra y sale del sistema. Realizando el símil inverso de variables generalizadas a variables eléctricas, se podrá decir que el cuadripolo podrá ser descrito como un conjunto de fuentes de esfuerzos y flujos (tensión e intensidad) y de impedancias y admitancias dependientes de los variables esfuerzo (tensión) y flujo (intensidad) existentes en los terminales de entrada y salida.

Sheridan en el MIT [174] utilizó la teoría del cuadripolo y la teoría de pasividad para obtener un método de diseño de las ganancias de los reguladores. Si se fuerza a la matriz de impedancias a ser real positiva, el criterio de Sheridan garantiza la estabilidad del sistema para cualquier entorno remoto pasivo y un operador pasivo. Una generalización de esta teoría fue usada por Adams [2] en su estudio de estabilidad de sistemas discretos.

En el JPL, Hannaford utilizó el cuadripolo pasivo para clasificar las diferentes arquitecturas de teleoperación en función de la matriz híbrida (ecuación

2.10). Estos estudios demostraban que la matriz híbrida caracteriza eficazmente el comportamiento del esquema bilateral.

La pasividad y el cuadripolo eléctrico también ha sido utilizado por Robert Anderson para el estudio de retardos en las comunicaciones [11]. Los estudios de Anderson garantizaban la estabilidad del sistema forzando a la línea de comunicaciones a ser una línea de transmisión de energía sin pérdidas.

Todos estos estudios asumen el modelo del operador como una impedancia pasiva [99]. Colgate utiliza teorías de control robusto para obtener un criterio de estabilidad que no exija la pasividad del operador. Estos estudios aportan al problema la naturaleza mixta continuo-discreto debida a la interacción hombre-computador discreto [61], [63].

#### 2.4.2.1. Estabilidad

Los problemas de estabilidad en un sistema bilateral constituyen un aspecto todavía abierto, y de gran importancia para el ingeniero de control. Considérese el cuadripolo  $N$  lineal e invariante con el tiempo de la figura 2.25, que a su vez se carga con otro segundo cuadripolo  $N_L$ .

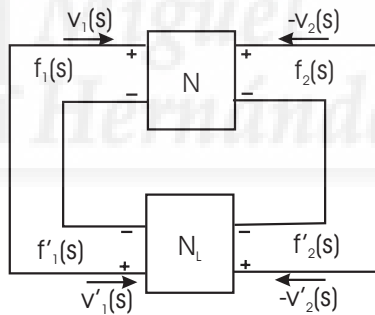


Figura 2.25: Cuadripolo cargado con otro cuadripolo

La estabilidad del sistema se define en términos de las raíces de la ecuación característica del sistema en lazo cerrado.

*Definición:* Un cuadripolo lineal es estable, si y solo si la ecuación característica no tiene raíces en la parte derecha del plano complejo  $s$  y sólo raíces simples en el eje imaginario.



Sin embargo, en sistemas teleoperados la carga  $N_L$  no es conocida, y se necesita algo que caracterice al cuadripolo  $N$  independientemente de la carga.

*Definición:* Un cuadripolo lineal  $N_L$  con energía inicial  $E(0)$  es *pasivo*, si y solo si:

$$\int_0^t (f'_1(\tau)v'_1(\tau) + f'_2(\tau)(-v'_2(\tau))) d\tau + E(0) \geq 0 \quad \forall t \geq 0 \quad (2.6)$$

para todas las fuerzas admisibles  $(f'_1, f'_2)$  y velocidades  $(v'_1, v'_2)$ .

Lo cual significa que  $N_L$  es incapaz de proporcionar energía que no haya sido almacenada previamente. El problema de estabilidad se reduce a conocer las condiciones bajo las cuales  $N$  es estable al conectarse a un cuadripolo pasivo  $N_L$ . Colgate [62] encontró que la condición necesaria y suficiente para que se cumpla esto es que la matriz de immitancias  $\mathbf{P}$  que caracteriza a  $N$  sea positiva y real.

*Definición:* Una matriz de immitancias  $\mathbf{P}$  es *positiva y real* si y solo si no tiene polos en la mitad derecha del plano  $s$ , solo polos simples en el eje imaginario y además:

$$\begin{aligned} \operatorname{Re}(p_{11}) &\geq 0, \quad \operatorname{Re}(p_{22}) \geq 0, \\ \operatorname{Re}(p_{11})\operatorname{Re}(p_{22}) - \left|\frac{p_{21}+p_{12}}{2}\right|^2 &\geq 0, \quad \forall \omega \geq 0 \end{aligned} \quad (2.7)$$

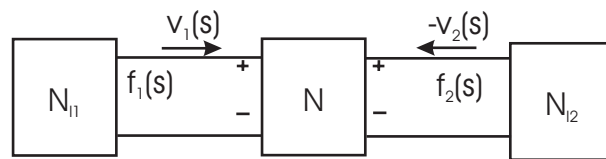


Figura 2.26: Cuadripolo cargado con dos bipolos

Un caso particular de cuadripolo es aquel en el que no hay cruce entre las cargas de sus extremos (figura 2.26). Si se considera que los dos bipolos que representan las cargas son redes pasivas, la estabilidad del sistema puede definirse según el criterio de Llewellyn [121].

*Definición:* Una red caracterizada por una matriz de immitancias  $\mathbf{P}$  es *absolutamente estable* si y solo si no tiene polos en la mitad derecha del plano  $s$ , solo polos simples en el eje imaginario y además:

$$\begin{aligned} \operatorname{Re}(p_{11}) &\geq 0, \\ 2\operatorname{Re}(p_{11})\operatorname{Re}(p_{22}) &\geq |p_{12}p_{21}| + \operatorname{Re}(p_{12}p_{21}), \quad \forall \omega \geq 0 \end{aligned} \quad (2.8)$$

lo que implica que  $\operatorname{Re}(p_{22}) \geq 0$ .

*Definición:* Una red caracterizada es *potencialmente inestable* si no es *absolutamente estable*.

#### 2.4.2.2. Representación de un sistema bilateral

En el caso de un sistema de teleoperación, los dos polos de entrada representan el mando maestro, y los dos polos de salida el extremo del esclavo que interacciona con el entorno (Figura 2.27). Las variables flujo en la entrada y en la salida serán la velocidad del maestro  $\dot{x}_m$  y del esclavo  $\dot{x}_s$  respectivamente, mientras que las variables esfuerzo de entrada y de salida son la fuerza  $f_m$  actuante sobre el maestro y la fuerza  $f_s$  que desarrolla el esclavo respectivamente. El cuadripolo como tal representa la relación existente entre estas cuatro variables, es decir, el tipo de control bilateral que ha sido implantado en el sistema. Se ha agrupado maestro y esclavo en un bloque, que es el cuadripolo, mientras que con la teoría clásica de control, lo que se hace es separarlos y analizar el control que los une.

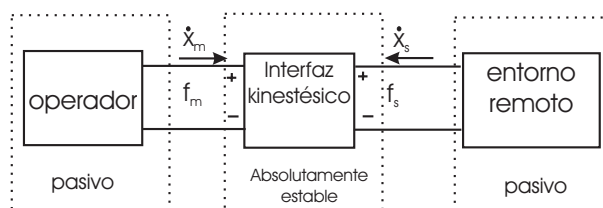


Figura 2.27: Modelo de un sistema teleoperado según la teoría del cuadripolo

Tal y como es habitual en el cuadripolo eléctrico, es posible representar cada uno de los dos terminales mediante un equivalente Thevenin (impedancia compleja en serie con una fuente de tensión), y unir ambos terminales

haciendo que las respectivas fuentes de tensión dependan de los parámetros del otro terminal.

Para caracterizar la interfaz kinestésica, y por tanto su comportamiento, el cuadripolo obtenido podrá expresarse mediante una matriz  $\mathbf{Z}$  de dimensión 2x2 denominada matriz de impedancias [161]:

$$\begin{bmatrix} f_m \\ f_s \end{bmatrix} = \begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{bmatrix} \begin{bmatrix} \dot{x}_m \\ \dot{x}_s \end{bmatrix} \quad (2.9)$$

Aunque dependiendo de cuales son las variables que se realimentan entre esclavo y maestro, pueda ser más interesante considerar otro tipo de matriz. En el caso del control posición-posición las variables que se sensorizan son las posiciones (velocidades) de maestro y esclavo y por tanto es interesante la representación en matriz  $\mathbf{Z}$ . Pero en control fuerza-posición es la posición del maestro y al fuerza del esclavo las variables que se realimentan, por lo que es más interesante utilizar la llamada matriz híbrida  $\mathbf{H}$ , expresada mediante [90]:

$$\begin{bmatrix} f_m \\ \dot{x}_s \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} \dot{x}_m \\ f_s \end{bmatrix} \quad (2.10)$$

El estudio del comportamiento del sistema se hará analizando la matriz de immitancias  $\mathbf{P}$  (impedancias o híbrida, según el caso), en la que previamente se ha realizado una identificación de sus componentes con las impedancias y parámetros del cuadripolo diseñado.

Los terminales del cuadripolo se conectarán por un lado al operador y por otro al entorno. El operador vendrá definido como una fuente de tensión (fuerza)  $\tau_{op}$  en serie con una impedancia compleja  $Z_{op}$  de valor:

$$Z_{op} = m_{op}j\omega + b_{op} + \frac{k_{op}}{j\omega} \quad (2.11)$$

En algunos casos, presentar el operador como una impedancia pasiva puede ser demasiado conservativo, pues admite una impedancia cero (no hay contacto con el operador) o infinito (un contacto rígido). Algunos autores [2] presentan un modelo del operador en el que asume que el contacto con la interfaz no puede romperse, existiendo por tanto una  $Z_{min}$ , y además que la interfaz no es capaz de reflejar contactos de rigidez infinita (existe una  $Z_{max}$ ), obteniendo el modelo de la figura 2.28.

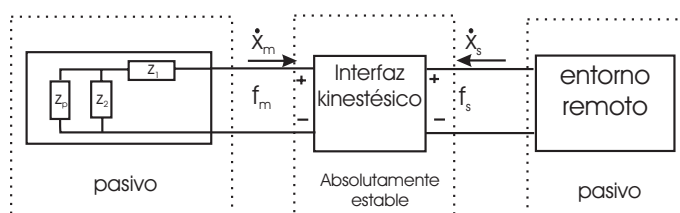


Figura 2.28: Modelo del operador

donde  $Z_1 = Z_{min}$  y  $Z_2 = Z_{max} - Z_{min}$ . Por otro lado, y al haber considerado el entorno como pasivo, este vendrá definido por una impedancia compleja de expresión:

$$Z_e = m_e j\omega + b_e + \frac{k_e}{j\omega} \quad (2.12)$$

Es importante resaltar la naturaleza discreta de las variables que intervienen en el puerto del entorno remoto, debido a los procesos digitales de captación de las señales del entorno remoto, tanto en el caso de un entorno real como simulado. La naturaleza de los entornos locales permite clasificarlos en *entornos de impedancia*, que aceptan velocidades (o posiciones) y generan fuerzas, o *entornos de admitancia*, que aceptan fuerzas y devuelven velocidades (o fuerzas).

En cuanto al interfaz kinestésico, para estudiar la estabilidad del cuadripolo, es común separar la interfaz en dos sub-redes en serie (figura 2.29) [61]. De esta forma se separa la interfaz mecánica (que se puede clasificar como *displays de impedancia* o *displays de admitancia*) de un cuadripolo (llamado en haptics Virtual Coupling Network) cuya estructura interna y valores de sus parámetros se calculan de forma que garanticen la estabilidad absoluta según la ecuación 2.8. Por tanto, esta red de acople tiene como función hacer el sistema estable independientemente del operador y del entorno remoto, y se trata de un sistema digital, implementado en software. La primera implementación de este acople [61] se trataba de un conjunto muelle-amortiguador en paralelo que se conectaba a un display de impedancias y un entorno remoto de admitancias, y se ajustaban los valores de  $k$  y  $b$  de forma que el sistema fuera absolutamente estable.

La teoría del cuadripolo permite tratar diversos sistemas de control bilateral de una forma genérica, definidos por las componentes de una matriz

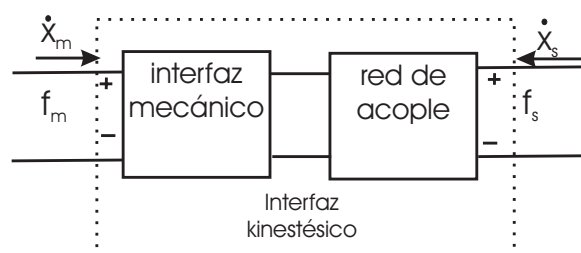


Figura 2.29: Cuadripolo de la interfaz kinestésica

cuadrada de dimensión dos. La notación resultante es elegante y compacta. Existe además la posibilidad de identificar los diversos parámetros (impedancias, fuentes) con elementos de circuitos eléctricos [12] y realizar el análisis del comportamiento del sistema simulando el propio circuito eléctrico resultante mediante simuladores comerciales de circuitos eléctricos.

## 2.5. Plataformas paralelas en teleoperación

El incremento de la capacidad de cálculo de los ordenadores actuales ha renovado el interés de los investigadores acerca de las plataformas robóticas paralelas. Las características especiales de esta clase de robots los hacen especialmente indicados para ciertas tareas, como por ejemplo máquinas de control numérico, robots manipuladores [103] o robots trepadores[17]. En el campo de la teleoperación robótica en general, y en particular en el diseño de interfaces mecánicas, los robots paralelos muestran ciertas ventajas que los hacen adecuados para trabajar como dispositivos maestro.

La estructura cinemática de tipo paralelo de esta clase de robots es conocida desde hace mucho tiempo. Por ejemplo hacia 1800 Cauchy estudió la rigidez de un octaedro articulado [53]. Más recientemente Gough [183] usó un mecanismo similar para el ensayo de neumáticos. Estas estructuras fueron redescubiertas en los años sesenta como la solución más práctica para la construcción de simuladores de vuelo activos. En 1965 Stewart [183] mostró el uso de una estructura paralela para un simulador de vuelo. Desde este momento, a pesar de que el mecanismo de Stewart es menos representativo que el de Gough, los robots paralelos son llamados frecuentemente plataformas de Stewart. En realidad existe una gran variedad de configuraciones mecánicas

para este tipo de robots.

La elección de este tipo de mecanismo para la plataforma de un simulador se justifica por una ventaja muy obvia: la buena relación carga admisible / peso propio. En efecto: en una plataforma de Gough (6 grados de libertad) distribuye su carga de modo que cada uno de sus 6 actuadores soporta aproximadamente 1/6 de los esfuerzos totales. Además los esfuerzos sobre los actuadores son principalmente de tracción y compresión lo cual proporciona gran rigidez al mecanismo. La figura 2.30 muestra el que posiblemente fuera primer dispositivo paralelo espacial.

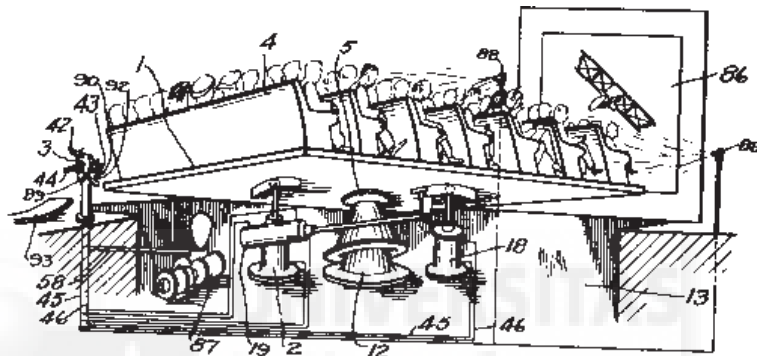


Figura 2.30: Posiblemente el primer mecanismo espacial paralelo. Patentado en 1931 (US patent N° 1.789.680)

#### Ventajas de los robots paralelos

Además de la excelente relación carga admisible / peso propio, los robots paralelos presentan otras características interesantes. McCallion en 1979 [128] usó un manipulador paralelo para una célula de ensamblaje robotizada, fundamentalmente porque la posición del efector final es mucho menos sensible a los errores que los sensores articulares de los robots en serie. La gran rigidez asegura que las deformaciones de los actuadores sean mínimas y esta característica contribuye en gran manera a la precisión en el posicionamiento del manipulador.

Otra característica importante de los robots paralelos es la posibilidad de implementar en ellos 6 sensores de esfuerzos en sus actuadores. De esta manera se pueden calcular las fuerzas y momentos que actúan en la plataforma

móvil.

Son posibles muchos diseños diferentes de robots paralelos y la literatura científica al respecto es muy rica. Todos tienen en común el bajo costo de sus componentes que en su mayoría son estándar, aunque hay que ensamblarlos de una forma cuidadosa.

#### Líneas actuales de investigación

Como la arquitectura de los robots paralelos es muy diferente a la de los robots en serie, la mayoría de los problemas teóricos se tienen que reconsiderar. De hecho existe una extraña dualidad entre los mecanismos en serie y los mecanismos en paralelo: un problema difícil (simple) para una clase es resuelto fácilmente (con dificultad) para la otra clase [191].

#### Diseño mecánico

Un problema interesante es encontrar un método para diseñar la arquitectura de un robot paralelo dado el número y el tipo de grados de libertad. Una aproximación a este problema basada en la teoría de grupos fue presentada por Hervé. Usando esta aproximación el autor ha sido capaz de diseñar un nuevo robot Star [97] con tres grados de libertad traslacionales. Sin embargo queda trabajo por hacer, especialmente en el diseño de robots que combinen grados de libertad traslacionales y rotacionales.

#### Cinemática

Hay que distinguir dos aspectos: la cinemática inversa y la cinemática directa. El problema cinemático inverso, es decir, encontrar las longitudes de los actuadores dada la postura de la plataforma móvil (un problema difícil para los robots en serie) es sencillo para los robots paralelos. Por lo tanto su control es muy simple. Por otra parte la cinemática directa es mucho más complicada. En general este problema tiene más de una solución, por ejemplo, la plataforma de Gough tiene 16 posturas diferentes dadas las 6 longitudes de sus actuadores [134],[131], [132].Lazard [116] demostró que en el caso más general no hay más de 40 soluciones. Algunos algoritmos se han presentado para resolver casos especiales [134],[14],[10], pero soluciones analíticas solo se han encontrado en arquitecturas muy especiales [118]. En la práctica los procedimientos numéricos se usan sin problemas.

Otra forma práctica de resolver la cinemática directa es implementar los apropiados sensores de posición y de orientación para calcular la postura de la plataforma móvil [134], [19].

### Dinámica

En los robots paralelos rápidos es necesario resolver el cálculo de modelo cinemático para llegar a un control satisfactorio. Todavía no se han establecido soluciones analíticas, pero, afortunadamente, se pueden hacer ciertas simplificaciones que permiten simular el comportamiento dinámico de estas plataformas.

### Control con realimentación de esfuerzos

Un robot paralelo puede equiparse con 6 sensores de esfuerzo y llevar a cabo tareas que impliquen contacto con el entorno (ensamblaje, seguimiento de superficies), en este caso es necesario una realimentación de los esfuerzos en el sistema de control. Se han resuelto con éxito experimentos con un manipulador paralelo trabajando en solitario [164], [130], [19], pero quedan problemas interesantes por resolver cuando un macro y un micro manipulador paralelo interactúan.

Otro tema interesante en los robots paralelos es, a pesar que son muy rígidos, obtener una cierta flexibilidad usando actuadores neumáticos o añadiendo elementos elásticos en los anclajes de los actuadores. En este caso una adecuada elección de la situación de los sensores permite calcular con precisión la postura de la plataforma a pesar de la elasticidad de los actuadores.

### Configuraciones singulares

De forma análoga a los robots en serie, los robots paralelos pueden adoptar configuraciones en las que las fuerzas articulares no pueden equilibrar los esfuerzos sobre la plataforma móvil. Es importante determinar estas configuraciones en cuya vecindad las fuerzas articulares tienden a infinito y el robot puede colapsar. Estas disposiciones singulares están caracterizadas por la anulación del determinante de la matriz jacobiana inversa. A pesar de que esta matriz sea conocida, en la mayoría de los casos la computación simbólica de este determinante no conduce a soluciones analíticas, por lo que hay que recurrir a procedimientos numéricos [71] o, más eficientemente, geométricos



para encontrar las configuraciones singulares [131]. Un problema que queda por resolver es determinar, de una forma simultanea, si existen configuraciones singulares dentro del espacio de trabajo de un robot paralelo.

#### Espacio de trabajo

En contraste con los robots en serie más comunes, con tres ejes intersecantes en la muñeca, en los robots paralelos el espacio de trabajo no se puede desacoplar en dos espacios 3D caracterizando los movimientos de traslación y de orientación. El espacio de trabajo de los robots paralelos está embebido en la misma plataforma móvil y no hay forma legible de representarlo. Sin embargo si se pueden representar algunas proyecciones de espacio de trabajo completo. Una forma usual es representar las posibles traslaciones de la plataforma móvil a una altura y orientación constantes, mediante un procedimiento de discretización [19] o, más eficientemente, por un algoritmo geométrico [134] que tenga en cuenta los límites de los actuadores, sus interferencias y las restricciones de las juntas. Otra posibilidad es la representación de todas las posibles rotaciones alrededor de un punto considerado fijo en la plataforma móvil. Recientemente se ha resuelto la verificación de la completa inclusión de una trayectoria recta en el espacio de trabajo [135]. Finalmente hay que hacer notar que la determinación del espacio de trabajo de un robot paralelo planar ha sido profundamente estudiada [136].

Otro problema interesante es determinar las dimensiones de un robot para que abarque un espacio de trabajo dado [137].

#### Calibración

La precisión de un robot paralelo no solo depende del buen control de las longitudes de sus actuadores sino también del buen conocimiento de sus características geométricas. En función de las tolerancias de fabricación muchos factores intervienen en la precisión del robot.

#### Aplicaciones prácticas

A pesar de lo reciente que es el concepto de robots paralelos, varios laboratorios han propuesto prototipos muy interesantes [114], [102], [93], [19]. Deben ser mencionados los robots paralelos desarrollados por MEL en Japón, uno de los cuales es un micro manipulador cuyos actuadores lineales tienen

un rango de trabajo de pocos micrómetros y permiten realizar movimientos de nanómetros [20] y otro es un gigantesco manipulador para la industria minera [145].

El primer robot paralelo comercial, el “Gadfly”, un manipulador de 6 grados de libertad para el ensamblado de componentes electrónicos, fue diseñado por Marconi [122]. Posteriormente esta compañía diseñó un gran robot híbrido serie-paralelo, el “Tetrabot” [123]. Un robot paralelo rápido de 3-4 GDL, el “Delta” ha comenzado recientemente su comercialización por la compañía Demareux. Este manipulador está diseñado para labores muy rápidas de pick&place, y las cualidades de su diseño le permite alcanzar altas velocidades y aceleraciones. El Hexa, actualmente en fase de desarrollo, es un diseño similar con 6 GDL. Hughes ha propuesto un nuevo prototipo denominado SmarTee.

Especial mención merecen los prototipos de Merlet para la cirugía oftálmica [133] y el uso de una clásica plataforma de Gough para la manipulación de cargas de hasta 230 kg, con alta repetitibilidad, en el European Synchrotron Radiation Facility (ESRF).

## 2.6. Conclusiones

La teleoperación comprende el conjunto de tecnologías que permiten a un operador la realización de tareas a distancia, teniendo su aplicación específica en el desarrollo de tareas en entornos que conllevan un riesgo al operador o bien son inaccesibles por éste. Para desarrollar una plataforma experimental con la cual probar diferentes esquemas de teleoperación se han de distinguir los diferentes elementos constitutivos de cualquier esquema de teleoperación. El estudio por separado de estos elementos permite obtener un concepto más claro de la problemática que implica la interconexión que se establece entre los mismos dentro de un esquema de teleoperación.

La presente Tesis aporta el diseño de un dispositivo maestro, de arquitectura paralela y 6 grados de libertad; también estudia el modelado e implementación de un robot comercial que puede ser tratado como dispositivo mecánico “abierto” desde el punto de vista del control, y también presenta un simulador dinámico de robots con capacidad para modelar cualquier arquitectura mecánica basándose en la dinámica multicuerpo. Este capítulo ha aportado las bases y un resumen de los estudios previos que fueron necesarios para abordar los citados retos.

# CAPÍTULO 3

## DISEÑO Y ANÁLISIS DE UN DISPOSITIVO HÁPTICO PARALELO 6-URS CON ARQUITECTURA ABIERTA

---

*“¿Por qué esta magnífica tecnología científica, que ahorra trabajo y nos hace la vida más fácil, nos aporta tan poca felicidad? La respuesta es esta, simplemente: aún no hemos aprendido a usarla con tino.”*

Albert Einstein.

### 3.1. Introducción

A tenor de lo expuesto en el capítulo anterior, se hace patente la necesidad de disponer de un dispositivo robótico con arquitectura abierta, que pueda ser utilizado como maestro en cualquiera de los esquemas de teleoperación citados anteriormente.

Con la meta puesta en este objetivo, se ha procedido al diseño, análisis e implementación de un dispositivo maestro de arquitectura abierta. Los resultados de dicho estudio se muestran en el presente capítulo.

En primer lugar se eligió realizar el diseño de un maestro con arquitectura paralela, justificando principalmente dicha elección en dos características de los mecanismos paralelos que los hacen especialmente indicados para trabajar como maestros en teleoperación; su baja inercia, que permite una interacción más “limpia” y su elevado ratio potencia/carga, lo cual permite que con actuadores relativamente poco potentes, se puedan reflejar fuerzas considerables. La principal desventaja de los mecanismos paralelos, su reducido espacio de trabajo, se ha solucionado bajo distintos enfoques; en primer lugar, en la etapa de diseño del mecanismo se utilizaron técnicas CAD con ayuda de programas de animación gráfica que permitieron realizar un estudio paramétrico del espacio de trabajo. Posteriormente, en la etapa de implementación (capítulo 4), se desarrolló un algoritmo de indexación y mapeado de espacios de trabajo, que basándose en las características del punto de vista utilizado en la interfaz de usuario, permite la reindexación del entorno remoto.

Siguiendo el espíritu de generalidad que motiva esta tesis se optó por desarrollar el maestro con 6 grados de libertad, lo cual permite abordar problemas de teleoperación complejos. Sin embargo, el mecanismo se puede utilizar perfectamente como un dispositivo de menos grados de libertad, permitiendo también su utilización en esquemas de teleoperación experimentales con un menor número de grados de libertad.

La elección de esta estructura mecánica implica el desarrollo de los modelos cinemáticos y dinámicos de un dispositivo paralelo URS. Para ello se ha hecho uso de los elegantes desarrollos de las teorías de dinámica multicuerpo. El desarrollo de estos algoritmos bajo un enfoque de dinámica computacional unido a un “aceptable” conocimiento de los parámetros cinemáticos y dinámicos del mecanismo ha permitido disponer de todas las herramientas necesarias para trabajar con un dispositivo robótico abierto, a la vez que se ha podido experimentar con diferentes esquemas de control considerando dicho mecanismo como un dispositivo robótico aislado.

El uso de una tarjeta de control de ejes de propósito general permite la implementación de los algoritmos anteriores a la vez que facilita el desarrollo de la interfaz necesaria para la inserción del dispositivo en una plataforma de teleoperación.

Debido a la propia configuración mecánica del dispositivo y a la arquitectura completamente abierta, este maestro puede trabajar tanto como *display de impedancias*, o como *display de admitancias*.

### 3.1.1. Display de impedancias vs. display de admitancias

Los displays de impedancias se definen como dispositivos capaces de proporcionar información sobre su posición y velocidad a la vez que pueden reflejar hacia el operador un vector de fuerzas generalizadas.

Análogamente, los displays de admitancias se definen como dispositivos capaces de leer el vector de fuerzas generalizadas proporcionado por el operador y reflejar al mismo la información de posición proveniente del entorno remoto.

Por tanto, para utilizar un dispositivo mecánico como display de impedancias se deben cumplir los siguientes requisitos: para leer la posición se requiere disponer de sensores de posición y del modelo cinemático directo que proporcione la posición y orientación del efector final en coordenadas del espacio cartesiano. En el caso de un mecanismo paralelo, la resolución en tiempo real de la cinemática directa es una de las aportaciones de esta tesis. Para reflejar un vector de fuerzas generalizadas se requiere disponer de la matriz jacobiana del mecanismo y de la posibilidad de comandar directamente pares a los actuadores en un esquema en lazo abierto.

Para utilizar un dispositivo mecánico como display de admitancias se deben cumplir los siguientes requisitos: para leer el vector de fuerzas generalizadas se requiere disponer de un sistema capaz de obtener las fuerzas. Típicamente estos sistemas pueden ser un bote sensor de fuerzas o utilizar la intensidad de los motores para obtener los pares de cada actuador y utilizando la jacobiana obtener el vector de fuerzas. Para reflejar la posición del entorno remoto se necesita disponer del modelo dinámico del mecanismo de forma que se puedan hallar los reguladores necesarios para establecer un control en posición. Además es obvio que la estructura mecánica debe ser suficientemente robusta para vencer las fuerzas ejercidas por el operador.

### 3.1.2. Estructura del capítulo

El presente capítulo sintetiza el proceso de diseño, análisis y modelado de un dispositivo de 6 grados de libertad con estructura paralela destinado a trabajar como maestro en una plataforma robótica de teleoperación.

En primer lugar se justifican las decisiones tomadas en el proceso de diseño, apoyándose en los resultados obtenidos con paquetes de simulación comerciales. Posteriormente se presenta un estudio del modelo cinemático y

dinámico del mecanismo. Para este estudio se ha utilizado los desarrollos de la mecánica multicuerpo particularizándolos en un robot de estructura paralela tipo URS. La implementación de estos desarrollos permite la generación de los algoritmos de control necesarios para el desarrollo de la interfaz de conexión del dispositivo a la plataforma de teleoperación.

### 3.2. Diseño mecánico del dispositivo

El problema del diseño de un mecanismo suele plantear serias dificultades al ingeniero de control, sin embargo la aparición de nuevas herramientas computacionales y el continuo desarrollo de la potencia de cálculo de los simuladores actuales proporcionan una ayuda inestimable a la hora de abordar dicho reto. La creciente simplicidad en la utilización de estos programas invita a los ingenieros a explorar las capacidades de los mismos para solucionar el problema del “papel en blanco”. La figura 3.1 muestra el diseño final del dispositivo maestro.

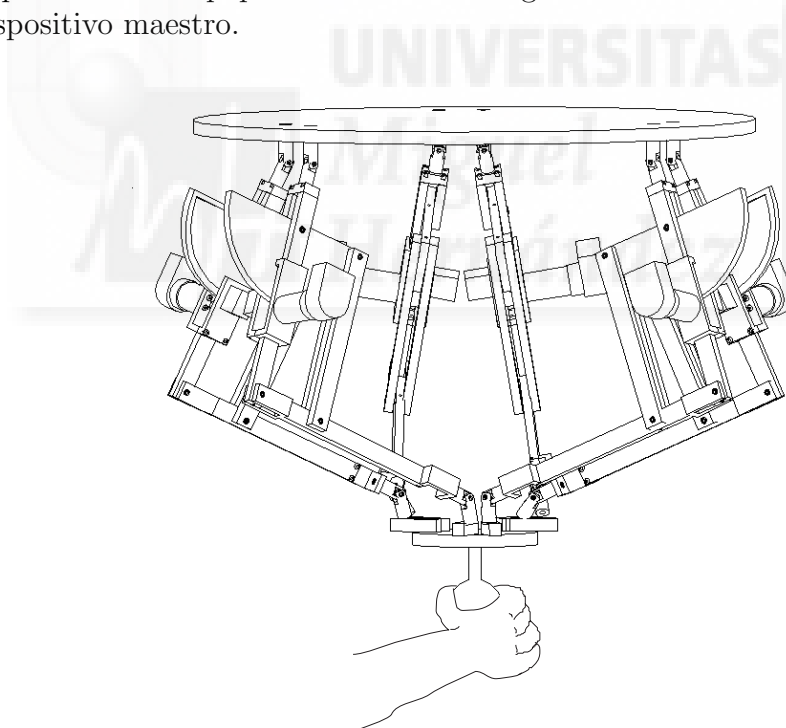


Figura 3.1: Diseño mecánico de una plataforma 6 URS

En este apartado se presentan las justificaciones de las decisiones de diseño del dispositivo paralelo presentado en este capítulo, a la vez que se realiza una breve explicación del uso de programas de propósito general como apoyo a estas decisiones.

### **3.2.0.1. ¿Por qué un dispositivo paralelo?**

Las estructuras paralelas proporcionan unas características que las hacen especialmente indicadas para su uso como dispositivos maestros. En primer lugar la baja inercia que reflejan constituye una gran ayuda en su utilización como display de impedancias, dado que es deseable que el operador no sienta el propio mecanismo maestro, sino que tan solo sienta las fuerzas generadas en el entorno remoto. Por otro lado, el elevado ratio potencia/carga que presentan estas estructuras es esencial en la utilización como display de admitancias, permitiendo el uso de actuadores relativamente poco potentes. La utilización de un mecanismo serial implicaría el uso de actuadores mucho mayores.

Adicionalmente el uso de estructuras paralelas presenta algunas dificultades, las cuales han sido abordadas en esta tesis como retos a superar, a saber:

- Pequeño espacio de trabajo, sobretodo a orientaciones variables
- Dificultad en la obtención matemática de los modelos cinemáticos y dinámicos
- Dificultad en el control en posición de dichos dispositivos

### **3.2.0.2. ¿Por qué un dispositivo sin reductores engranados?**

Los dispositivos maestros de teleoperación deben disponer de un rango de impedancias lo mayor posible, permitiendo así la reflexión más nítida de esfuerzos en un ancho de banda lo más amplio posible. El uso de dispositivos reductores implica la aparición de holguras y backlash, comprometiendo así la veracidad de la información reflejada.

Por otra parte, una actuación directa sobre el eje de rotación compromete el logro de la utilización de actuadores de pequeña potencia, a la vez que complica su elección, pues deberían ser motores que trabajaran a muy

bajas revoluciones y con un par aceptable. La transmisión por cable permite solucionar dicho problema, obteniendo una reductora que carece de las desventajas citadas más arriba.

### **3.2.0.3. ¿Por qué encoders incrementales?**

La disposición invertida del dispositivo proporciona una posición de mínima energía que permite obtener una posición de equilibrio estable que marque el cero absoluto. Esta ventaja permite el uso de encoders incrementales mucho más económicos que otros sensores de posición a la vez que evita al operador colocar el dispositivo en una posición determinada antes de conectar la alimentación.

### **3.2.0.4. ¿Por qué seis grados de libertad?**

Se pretende obtener un maestro lo más general posible.

### **3.2.0.5. ¿Por qué un paralelogramo articulado?**

La plataforma conocida como plataforma de Gough o de Stewart consiste en una plataforma de 6 grados de libertad con seis cadenas cinemáticas independientes de tipo UPS (Universal-Prismatic-Spheric). En este caso la articulación prismática ha sido sustituida por una rotacional que genera un movimiento lineal mediante un paralelogramo articulado. Al evitar la restricción de la longitud natural del elemento prismático se proporciona un mayor espacio de trabajo al mecanismo. Además, permite una diferente orientación de la articulación esférica, que si bien teóricamente no afecta, en el dispositivo real, la articulación esférica posee limitaciones de movimiento que ven mejorado su comportamiento al colocarlas en distinta orientación.

### **3.2.0.6. ¿Por qué está realizado en aluminio?**

Todas las piezas del dispositivo han sido realizadas en un pequeño centro de control numérico del Departamento de Ingeniería de Sistemas Industriales de la UMH. El requisito principal era una simplicidad en la programación de la secuencia de control numérico y el menor peso posible en las mismas. Los programas de control numérico se encuentran en el cd anexo de la presente tesis.



### 3.2.1. Análisis del dispositivo con software de simulación dinámica

En este apartado se muestran algunos de los resultados obtenidos de las simulaciones realizadas con el software I-deas para el diseño paramétrico del dispositivo mecánico. El modelo utilizado en estas simulaciones previas es el mostrado en la figura 3.2.

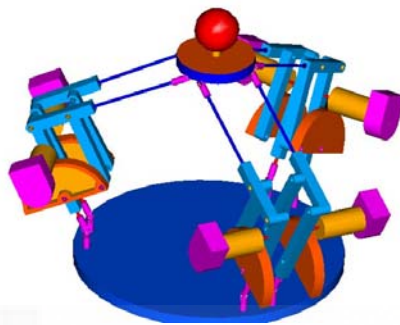


Figura 3.2: Modelo utilizado con I-deas

La finalidad de estas simulaciones era obtener los datos necesarios para realizar un correcto dimensionamiento de los actuadores. Para ello los pasos realizados consisten en:

- obtener un modelo lo más parecido posible al mecanismo final
- definir la geometría y los datos másicos de todos los eslabones utilizando las herramientas CAD del programa.
- definir la jerarquía de elementos y las relaciones entre los mismos utilizando los *markers* o sistemas de referencia de los eslabones.
- definir movimientos relativos entre los eslabones del mecanismo utilizando los *motion generators* del programa.

Las simulaciones realizadas consistieron en definir movimientos relativos que llevaran al mecanismo hasta posiciones extremas, y graficar las fuerzas que aparecían en las articulaciones. Como ejemplo se muestra a continuación la gráfica (figura 3.3) obtenida para la simulación de un giro en x de los pares

necesarios en las articulaciones disco ( $\sim 80$  Nmm), a partir de las cuales, y teniendo en consideración el factor de reducción que introduce la transmisión por cable, se pudo realizar una elección de los actuadores.

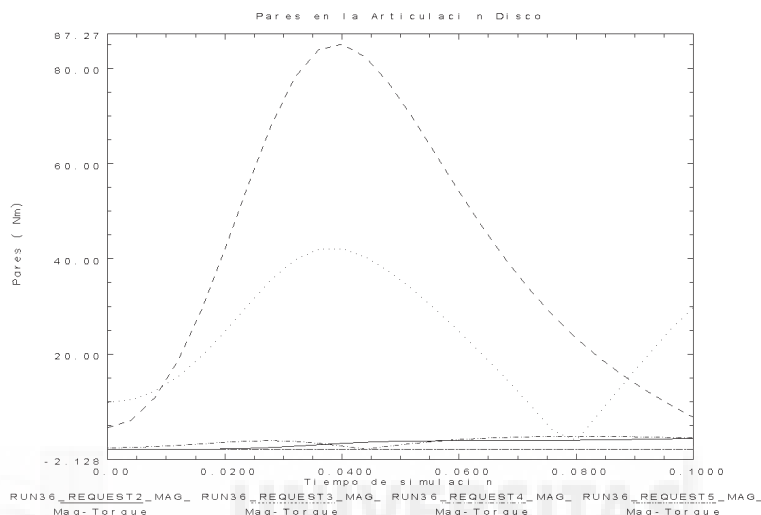


Figura 3.3: Curvas de simulación con I-deas

Tanto las simulaciones como los diferentes modelos utilizados en el desarrollo de esta tesis se encuentran en el cd anexo.

### 3.2.1.1. Prediseño y elección de los actuadores

A la vista de los resultados de las simulaciones anteriores (requisitos de 10 mNm de par) y de las consideraciones realizadas en el capítulo 2 acerca del ancho de banda necesario en los actuadores para la reflexión de fuerzas (a partir de 320 Hz del dedo humano es incapaz de discriminar entre dos señales de fuerza consecutivas), se procede a la elección de unos actuadores para el MAGÍSTER-P.

Las opciones barajadas en este apartado consistieron en:

- SMA's
- Frenos magnéticos
- Actuadores neumáticos

- DC coreless
- DC brushless
- Motores paso a paso
- Switching reluctance motors
- Motores híbridos

Sin entrar en otras consideraciones se estima conveniente el uso de los motores DC coreless debido a los requisitos de ancho de banda y facilidad de control. En concreto se utilizan los motores EM2342 012CR de la casa Faulhaber GMBH. junto con los drivers LC 3002.

### **3.2.2. Análisis del dispositivo con software de animación gráfica**

La continua aparición de nuevas capacidades en el software de animación gráfica unido a la creciente simplicidad de manejo de los mismos, invita al ingeniero a explorar estas nuevas características en búsqueda de herramientas de apoyo en el proceso de diseño. La inclusión de algoritmos de resolución cinemática basados en sistemas de referencia solidarios a los eslabones proporciona una potente utilidad para el pre-diseño de mecanismos. La capacidad de poder manejar un cuerpo sólido en el espacio mediante simples movimientos del ratón a la vez que se visualizan los efectos de estos movimientos sobre el resto de los eslabones del mecanismo constituye un adelanto en el tiempo de desarrollo de un mecanismo móvil, a la vez que reduce los posibles errores que se puedan cometer en esta etapa de diseño.

En el desarrollo de un dispositivo robótico en general, y más particularmente en un dispositivo con estructura paralela cobra especial importancia la capacidad de definir cadenas cinemáticas con restricciones entre los eslabones de las mismas. La solución interactiva de la cinemática inversa de estas cadenas proporciona una inestimable ayuda en nuestro trabajo.

En concreto, se pueden encontrar varios programas dedicados a la animación gráfica, como por ejemplo el 3DMAX<sup>®</sup> de DISCREET<sup>®</sup> en los cuales se encuentran implementados algoritmos de solución de la cinemática inversa de cadenas seriales. En este apartado se va a explicar brevemente cómo se ha

logrado utilizar esta característica para el estudio de mecanismos paralelos tipo plataforma URS.

Como se ha mencionado anteriormente, estos programas permiten definir cadenas cinemáticas seriales, pero no permiten la resolución cinemática de lazos cerrados. Sin embargo, nada impide que se definan varias cadenas seriales acabando cada una de ellas en un eslabón ficticio. En la simulación, todos los eslabones ficticios son movidos en conjunto, todos referidos a un mismo sistema de referencia, como si constituyeran un único elemento. De esta forma se resuelven todas las cadenas cinemáticas seriales simultáneamente, permitiendo la visualización del movimiento del mecanismo paralelo.

Sin entrar en detalle del manejo del programa de animación utilizado, se muestran a continuación unas figuras ilustrativas del proceso explicado arriba.

En primer lugar, la figura 3.4 muestra una vista general del software, en la que se observan los distintos eslabones que constituyen el citado mecanismo. Se puede observar en la pestaña de la derecha la definición de los parámetros de las cadenas cinemáticas.

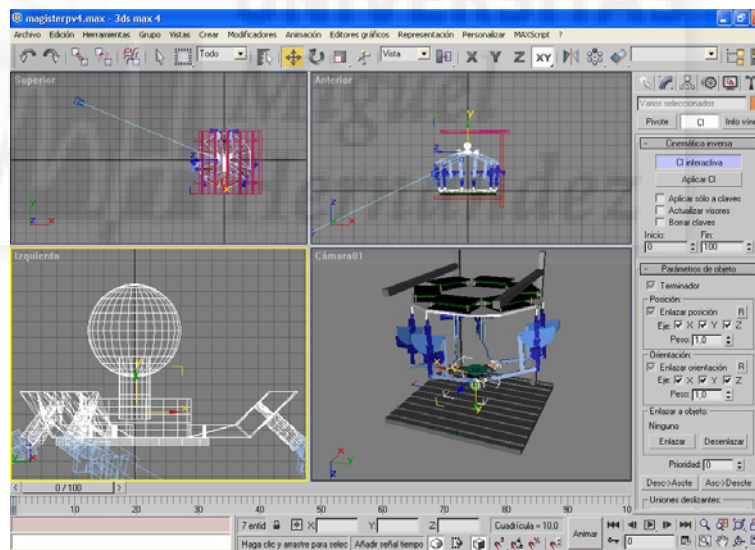


Figura 3.4: Software de animación gráfica

El primer paso para definir estas cadenas (en este caso número de cadenas = 6) es definir los sistemas de referencia de cada uno de los eslabones del

mecanismo. Las operaciones booleanas a partir de las cuales se ha definido la geometría de cada eslabón genera un sistema de referencia que no tiene porque coincidir con el deseado para el eslabón, sin embargo, este sistema (pivote) puede ser posicionado y orientado posteriormente, para hacerlo coincidir con la definición de las articulaciones, tal y como muestra la figura 3.5.

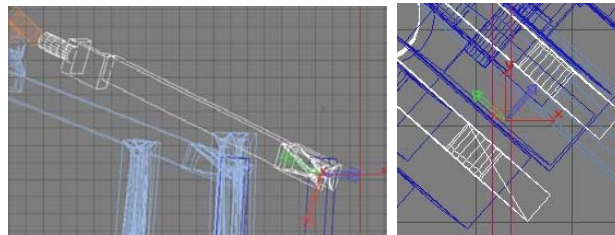


Figura 3.5: Posición del SR pivote del eslabón brazo

El siguiente paso es definir las cadenas cinemáticas. Para ello se han utilizado las herramientas de herencia que proporciona el software para construir seis cadenas cinemáticas independientes cuya única ligazón es que todas parten del elemento *base\_inferior*.

Finalmente es necesario definir el tipo de relación existente entre cada par de eslabones que forman la cadena. Para ello los programas suelen tener librerías de articulaciones rotacionales y prismáticas o simplemente se bloquea el movimiento en los ejes correspondientes. Para el dispositivo diseñado se definieron 6 cadenas tipo URS (Universal-Rotational-Spherical) limitando los movimientos relativos de cada eslabón.

### 3.2.2.1. Espacio de trabajo del dispositivo

A partir del modelo desarrollado, se puede iniciar un proceso rápido de “prueba y error” basado en mover el mecanismo hasta sus posiciones extremas con diferentes orientaciones del elemento *base\_sup* que permite realizar un estudio previo del espacio de trabajo.

Las conclusiones a las que se puede llegar con este estudio han permitido dimensionar tanto la base inferior como la base superior con objeto de intentar maximizar el espacio de trabajo del joystick. Además se ha podido realizar un estudio de los ángulos obtenidos en las articulaciones de tipo hook, que

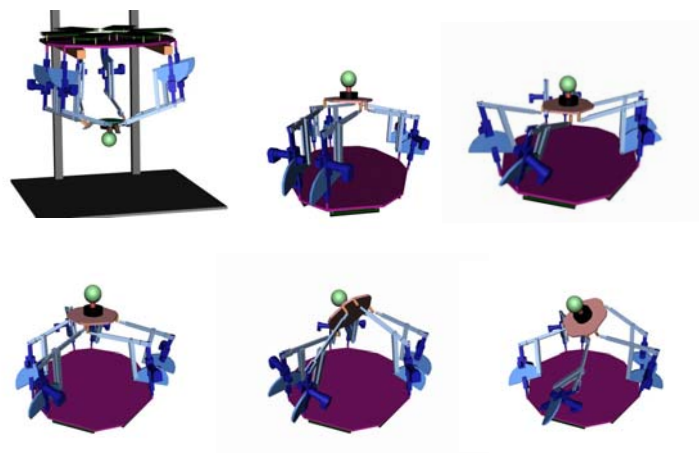


Figura 3.6: Estudio del espacio de trabajo con una herramienta de animación gráfica

si bien teóricamente no tienen limitaciones, en la realidad las articulaciones utilizadas tienen limitado su recorrido. Este hecho fue solucionado en la etapa de diseño mediante unas pequeñas piezas que posicionaban la articulación con un giro de  $45^\circ$  sobre la base superior.

A pesar de esto, como es bien sabido, el estudio del espacio de trabajo de una plataforma paralela no es un problema trivial. La naturaleza de los mecanismos paralelos propicia la aparición de las llamadas singularidades estáticas o de fuerza, en las cuales las fuerzas que actúan en alguna dirección en el efector final no pueden ser soportadas por los pares generados en los actuadores, es decir, se vuelve incontrolable. Es obvio que este tipo de singularidades no pueden ser estudiadas con una herramienta que resuelva exclusivamente la cinemática de cadenas seriales, por lo que en ningún caso se puede dar por concluido un estudio del espacio de trabajo a través de las simulaciones de “prueba y error”. Para el estudio de las singularidades de fuerza es necesario disponer de un modelo completo de la cinemática, como se verá más adelante.

Sin embargo, las singularidades cinemáticas debidas a limitaciones del espacio de trabajo si son detectables mediante este procedimiento, lo que justifica su utilización como etapa de la fase de diseño.

Como resumen del diseño mecánico del dispositivo, la tabla 3.1 presenta las principales características mecánicas del prototipo desarrollado:

Tabla 3.1: Características Mecánicas del dispositivo

<b>datos geométricos</b>	
Radio de la base superior	250 mm
Radio de la base inferior	100 mm
Ángulo de separación entre las articulaciones esféricas	15 °
Ángulo de separación entre las articulaciones universales	20 °
Longitud de los eslabones pares	188 mm
Longitud de los eslabones impares	220 mm
Relación de transmisión del actuador	13:1
Ángulo máximo de extensión del paralelogramo	135 °
Ángulo mínimo de extensión del paralelogramo	40 °
<b>datos dinámicos</b>	
Masa del eslabón base inferior	0.29 Kg
Masa reducida de los eslabones pares	0.06 Kg
Masa reducida de los eslabones impares(+motor)	0.45 Kg
Inercias principales de la base inferior	$kgmm^2$ $I_x$ 236.16
	$I_y$ 236.17
	$I_z$ 431.85
Inercias principales de los eslabones pares	$kgmm^2$ $I_x$ 42.04
	$I_y$ 139.21
	$I_z$ 146.10
Inercias principales de los eslabones impares	$kgmm^2$ $I_x$ 2246.35
	$I_y$ 2206.94
	$I_z$ 1139.79

### 3.3. Modelado cinemático de un sistema multicuerpo

En este apartado se utilizan los elegantes desarrollos presentes en la literatura especializada de análisis cinemático de sistemas multicuerpos. La elección de estos desarrollos se encuentra en la necesidad de encontrar un método sistemático de análisis de mecanismos de cadena cerradas y de abandonar las rígidas restricciones del análisis clásico de robots. Se realiza a continuación una explicación de las bases matemáticas necesarias para comprender los algoritmos desarrollados más adelante, si bien una explicación más detallada puede encontrarse en [173], [89], [168].

#### 3.3.0.2. Bases matemáticas para el movimiento de 1 cuerpo rígido

La descripción de un sistema mecánico con múltiples cuerpos parte de la definición de un sistema inercial de referencia  $XYZ$ , y un sistema de referencia móvil  $X'Y'Z'$  cuyo origen  $O'$  esta rígidamente atado al cuerpo  $i$ . En el caso de la descripción espacial es necesario la posición del origen  $O'$  y los tres ángulos de rotación de Euler  $Z, X', Z''$  (313, u otras rotaciones de Euler), lo que da como resultado un vector de coordenadas generalizadas,  $q = [x, y, z, \psi, \theta, \phi]$ . Los sistemas mecánicos en general se configuran para aplicaciones específicas y, dependiendo de éstas se definen los grados de libertad, así como el tipo de cinemática de las articulaciones. Usualmente en robótica, las articulaciones son del tipo rotacional y prismático y las diferentes morfologías de los robots son una combinación de las mismas.

Un sistema de múltiples cuerpos con restricciones está compuesto por una serie de eslabones enlazados entre sí por juntas y/o fuerzas que imponen restricciones sobre su movimiento relativo o sobre la dinámica de su interacción. La configuración de las uniones también determina su clasificación como mecanismos de cadena cerrada, en el caso de que su configuración se cierre en un lazo sobre alguna referencia determinada, o de cadena abierta cuando no existen lazos cerrados sobre su configuración.

#### Matriz de rotación

Para obtener la transformación que define la orientación relativa entre dos sistemas de coordenadas  $X'Y'Z'$  y  $XYZ$ , se asume, sin pérdida de generalidad, que los orígenes y los ejes de ambos sistemas de referencia coinciden.



Posteriormente se ampliará esta suposición a un desplazamiento general.

Sea  $s'$  el vector de posición del punto  $P'$  cuyas coordenadas se asumen fijas en  $X'Y'Z'$ . La figura 3.7 muestra la rotación de un ángulo  $\theta$  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  $XYZ$  se denomina  $s$ . El vector  $\Delta s$  define el cambio el vector de posición del punto  $P'$ .

$$s = s' + \Delta s \quad (3.1)$$

donde

$$\Delta s = b_1 + b_2 \quad (3.2)$$

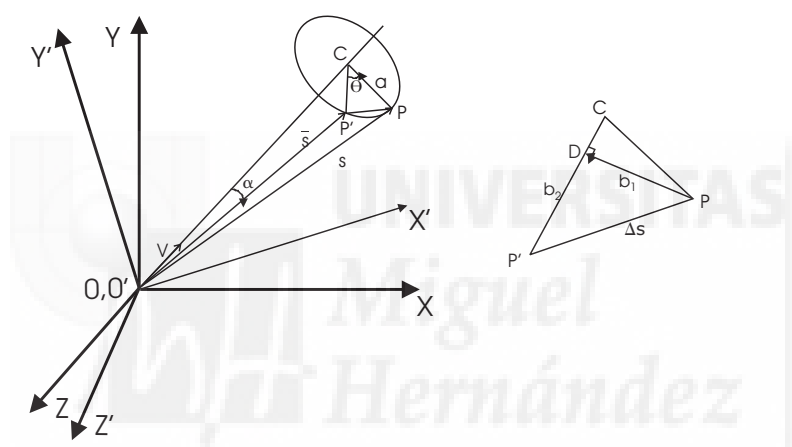


Figura 3.7: Rotación finita de un punto  $P'$

Donde el vector  $b_1$  es perpendicular al plano  $OCP'$  y por tanto tiene la dirección de  $(v \times s')$  donde  $v$  es el vector unitario a lo largo del eje de rotación  $\overline{OC}$ . La magnitud del vector  $b_1$  viene dada por:

$$|b_1| = a \sin \theta \quad (3.3)$$

donde  $a = |s'| \sin \theta = |v \times s'|$ , luego:

$$b_1 = a \sin \theta \frac{v \times s'}{|v \times s'|} = (v \times s') \sin \theta \quad (3.4)$$

El vector  $b_2$  de la ecuación (3.2) tiene una magnitud dada por:

$$|b_2| = a - a \cos \theta = (1 - \cos \theta)a = 2a \sin^2 \left( \frac{\theta}{2} \right) \quad (3.5)$$

Y es perpendicular tanto a  $v$  como a  $\overline{DQ}$ , por tanto

$$b_2 = 2a \sin^2 \left( \frac{\theta}{2} \right) \frac{v \times (v \times s')}{a} = 2[v \times (v \times s')] \sin^2 \left( \frac{\theta}{2} \right) \quad (3.6)$$

luego

$$s = s' + (v \times s') \sin \theta + 2[v \times (v \times s')] \sin^2 \left( \frac{\theta}{2} \right) \quad (3.7)$$

Utilizando la definición de matriz Skew, lo que da lugar a la identidad  $v \times s' = \tilde{v}s' = -\tilde{s}'v$ , donde:

$$\tilde{v} \equiv \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & -v_1 & 0 \end{bmatrix} \quad \tilde{s}' \equiv \begin{bmatrix} 0 & -s'_3 & s'_2 \\ s'_3 & 0 & -s'_1 \\ -s'_2 & -s'_1 & 0 \end{bmatrix} \quad (3.8)$$

Siendo  $v_1, v_2$  y  $v_3$  las componentes del vector unitario  $v$  y  $s'_1, s'_2, s'_3$  las componentes del vector  $s'$ . La ecuación 3.7 puede describirse como

$$s = s' + \tilde{v}s' \sin \theta + 2(\tilde{v})^2 s' \sin^2 \left( \frac{\theta}{2} \right) = \left[ I + \tilde{v} \sin \theta + 2(\tilde{v})^2 s' \sin^2 \left( \frac{\theta}{2} \right) \right] s' \quad (3.9)$$

Donde  $I$  es una matriz identidad 3x3

En general se definirá una transformación de un vector algebraico como:

$$s = As' \quad (3.10)$$

donde  $s'$  es un vector descrito respecto del sistema de referencia atado al cuerpo  $i$  y  $A$  viene expresada por la conocida *fórmula de Rodriguez*.

$$A = \left[ I + \tilde{v} \sin \theta + 2(\tilde{v})^2 \sin^2 \left( \frac{\theta}{2} \right) \right] \quad (3.11)$$

La matriz de rotación  $A$  es una matriz que describe el vector  $s'$  respecto del sistema de referencia absoluto inercial, y puede expresarse igualmente como matriz de los cosenos directores [173].

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = [f, g, h] \quad (3.12)$$

donde  $f, g, h$  son los vectores cosenos directores de la matriz de rotación  $A$ .

### Parámetros de Euler

Utilizando la identidad trigonométrica  $\sin \theta = 2 \sin \left(\frac{\theta}{2}\right) \cos \left(\frac{\theta}{2}\right)$ , se rescriba la ecuación 3.11 como:

$$A = I + 2\tilde{v} \sin \left(\frac{\theta}{2}\right) \left[ I \cos \left(\frac{\theta}{2}\right) + \tilde{v} \sin \left(\frac{\theta}{2}\right) \right] \quad (3.13)$$

Y, utilizando los parámetros de Euler, definidos como:

$$\begin{aligned} e_0 &= \cos \left(\frac{\theta}{2}\right) & e_1 &= v_1 \sin \left(\frac{\theta}{2}\right) \\ e_2 &= v_2 \sin \left(\frac{\theta}{2}\right) & e_3 &= v_3 \sin \left(\frac{\theta}{2}\right) \end{aligned} \quad (3.14)$$

Expresando  $e = [e_1, e_2, e_3]^T$ , la matriz de rotación  $A = [f, g, h]$ , se puede escribir en términos de los parámetros de Euler como:

$$A = (2e_0^2 - 1)I + 2(ee^T + e_0\tilde{e}) = 2 \begin{bmatrix} e_0^2 + e_1^2 - 1/2 & e_1e_2 - e_0e_3 & e_1e_3 - e_0e_2 \\ e_1e_2 - e_0e_3 & e_0^2 + e_2^2 - 1/2 & e_2e_3 - e_0e_1 \\ e_1e_3 - e_0e_2 & e_2e_3 - e_0e_1 & e_0^2 + e_3^2 - 1/2 \end{bmatrix} \quad (3.15)$$

si se asume que  $e_0 \neq 0$ , e indiferentemente del signo

$$\text{tr} A = a_{11} + a_{22} + a_{33} = 2(3e_0^2 + e_1^2 + e_2^2 + e_3^2) = 4e_0^2 - 1 \quad (3.16)$$

las siguientes ecuaciones relacionan la matriz de cosenos directores con la matriz de los parámetros de Euler.

$$e_0^2 = \frac{\text{tr} A + 1}{4} \quad (3.17)$$

$$e_1^2 = \frac{1 + 2a_{33} - \text{tr} A}{4} \quad (3.18)$$

$$e_2^2 = \frac{1 + 2a_{22} - \text{tr}A}{4} \quad (3.19)$$

$$e_3^2 = \frac{1 + 2a_{11} - \text{tr}A}{4} \quad (3.20)$$

Por conveniencia se define un vector de parámetros de Euler,

$$P = [e_0 e^T] = [e_0, e_1, e_2, e_3]^T \quad (3.21)$$

La expresión 3.22 indica que al menos uno de los términos debe ser diferente de cero.

$$P^T P = e_0^2 + e_1^2 + e_2^2 + e_3^2 = 1 \quad (3.22)$$

Para determinar los signos algebraicos de los parámetros de Euler, se deben usar las siguientes expresiones;

si  $e_0$  es diferente de cero

$$\begin{aligned} e_1 &= \frac{a_{32} - a_{23}}{4e_0} \\ e_2 &= \frac{a_{21} - a_{12}}{4e_0} \\ e_3 &= \frac{a_{13} - a_{31}}{4e_0} \end{aligned} \quad (3.23)$$

si  $e_0 = 0$ , los signos se deben determinar de las siguientes expresiones de acuerdo a la condición de que P es diferente de cero.

$$\begin{aligned} a_{21} - a_{12} &= 4e_1 e_2 \\ a_{31} - a_{13} &= 4e_1 e_3 \\ a_{32} - a_{23} &= 4e_2 e_3 \end{aligned} \quad (3.24)$$

### Propiedades de la matriz de rotación

Es importante anotar que mientras  $\tilde{v}$  es una matriz skew ( $\tilde{v}^T = -\tilde{v}$ ),  $\tilde{v}^2$  es una matriz simétrica. De hecho se obtienen las siguientes identidades [24]

$$(\tilde{v})^{2n-1} = (-1)^{n-1} \tilde{v}; \quad (\tilde{v})^{2n} = (-1)^{n-1} (\tilde{v})^2 \quad (3.25)$$

Y por tanto, a partir de la ecuación 3.11, se obtiene:

$$AA^T = \left[ I + \tilde{v} \sin \theta + 2(\tilde{v})^2 \sin^2 \left( \frac{\theta}{2} \right) \right] \left[ I + \tilde{v} \sin \theta + 2(\tilde{v})^2 \sin^2 \left( \frac{\theta}{2} \right) \right] = A^T A \quad (3.26)$$

De donde utilizando 3.25 se prueba que:

$$AA^T = I; \quad A^T = A^{-1} \quad (3.27)$$

Por otra parte, los parámetros de Euler tienen una serie de propiedades de importancia básica para el modelado cinemático. Las identidades de los parámetros Euler facilitan la partición lineal de la matriz de rotación A, cuyos términos altamente no lineales, dificultan el manejo algebraico de las diferentes relaciones cinemáticas. Un contenido mas amplio sobre las propiedades de los parámetros de Euler se puede consultar en [173] y [148]. Se definen las matrices 3.28 y 3.29 de forma que se cumpla la ecuación 3.31

$$E = [-e, \tilde{e} + e_0 I] = \begin{bmatrix} -e_1 & e_0 & -e_3 & e_2 \\ -e_2 & e_3 & e_0 & -e_1 \\ -e_3 & -e_2 & e_1 & e_0 \end{bmatrix} \quad (3.28)$$

$$G = [-e, -\tilde{e} + e_0 I] = \begin{bmatrix} -e_1 & e_0 & e_3 & -e_2 \\ -e_2 & -e_3 & e_0 & e_1 \\ -e_3 & e_2 & e_1 & e_0 \end{bmatrix} \quad (3.29)$$

Algunas propiedades de las identidades E y G de los parámetros de Euler se indican abajo.

$$\begin{aligned} EP &= -e_0 e + \tilde{e} e + e_0 e = 0 \\ GP &= 0 \\ EE^T &= GG^T = I \\ E^T E &= I_4 - PP^T \\ G^T G &= I_4 - PP^T \end{aligned} \quad (3.30)$$

donde  $I_4$  es una matriz de identidad de 4x4, además es importante la siguiente identidad.

$$EG^T = A = (2e_0^2 - 1)I + 2(ee^T + e_0\tilde{e}) \quad (3.31)$$

Además, diferenciando 3.10 respecto del tiempo, se obtiene

$$\dot{s} = A\dot{s}' + \dot{A}s' \quad (3.32)$$

Donde  $(\cdot)$  denota la derivada respecto del tiempo. Esta ecuación implica que la variación temporal de un vector se debe a dos contribuciones. La primera parte es  $A\dot{s}'$  que representa la variación de la longitud del vector  $s'$  mientras que la segunda es  $\dot{A}s'$  debida al cambio de orientación del vector  $s$ .

Dado que  $A$  depende solamente de los cuatro parámetros de Euler, utilizando la regla de la cadena se obtiene

$$\dot{A}s' = \sum_{k=0}^3 (A_{e_k} s') \dot{e}_k = A_{e_0} s' \dot{e}_0 + A_{e_1} s' \dot{e}_1 + A_{e_2} s' \dot{e}_2 + A_{e_3} s' \dot{e}_3 \quad (3.33)$$

Donde  $A_{e_k}$  es la derivada parcial de  $A$  con respecto al parámetro  $e_k$

Si se definen los siguientes vectores columna:

$$B_0 = A_{e_0} s'; \quad B_1 = A_{e_1} s'; \quad B_2 = A_{e_2} s'; \quad B_3 = A_{e_3} s'; \quad (3.34)$$

La ecuación 3.33 puede ser rescrita como

$$\dot{A}s' = B_0 \dot{e}_0 + B_1 \dot{e}_1 + B_2 \dot{e}_2 + B_3 \dot{e}_3 \quad (3.35)$$

O, utilizando 3.21 de una forma más compacta como

$$\dot{A}s' = B\dot{P} \quad (3.36)$$

### Derivadas temporales de los parámetros de Euler

Utilizando las ecuaciones 3.21, 3.28, 3.29 y 3.36 y derivando, se obtienen las siguientes relaciones [168]:

$$\begin{aligned} P^T \dot{P} &= \dot{P}^T P = 0 \\ E\dot{P} &= -\dot{E}P \\ G\dot{P} &= -\dot{G}P \\ EG^T &= -\dot{E}G^T \\ (G\dot{P} &= G\dot{G}^T) \end{aligned} \quad (3.37)$$

y para la matriz de rotación  $A$ .

$$\dot{A} = 2EG^T \quad (3.38)$$

### Desplazamiento general

En general un punto en el espacio se define como:

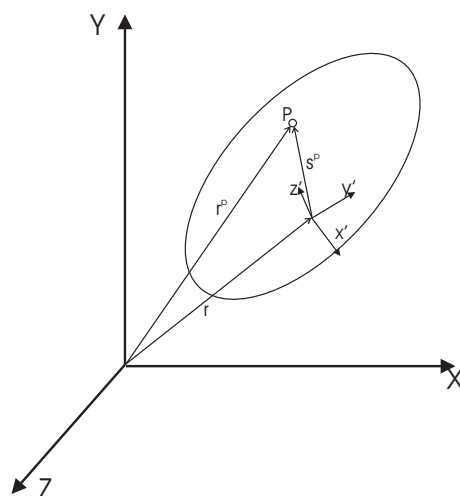


Figura 3.8: Desplazamiento de un punto material en el espacio

$$r^P = r + A s'^P \quad (3.39)$$

El vector que describe la velocidad del punto  $P$  indicado en la figura 3.8, es

$$\dot{r}^P = \dot{r} + \dot{A} s'^P \quad (3.40)$$

considerando que  $s'^P = A^T s^P$ , y sustituyendo esta expresión en la ecuación 3.39, tenemos.

$$\dot{r}^P = \dot{r} + \dot{A} A^T s^P \quad (3.41)$$

por analogía  $\tilde{\omega} = \dot{A} A^T$  o también  $\dot{A} = \tilde{\omega} A$ , por lo que la ecuación 3.41 también se puede escribir como

$$\dot{r}^P = \dot{r} + \tilde{\omega} A s^P \quad (3.42)$$

donde  $\tilde{\omega}$ , es la matriz Skew del vector de velocidad angular  $\omega = [\omega_x, \omega_y, \omega_z]^T$ . En el caso de la aceleración, derivando la expresión 3.40, se tiene

$$\ddot{r}^P = \ddot{r} + \ddot{A} s'^P \quad (3.43)$$

la aceleración angular tiene las siguientes relaciones

$$\ddot{A} = \dot{\tilde{\omega}}A + \tilde{\omega}\dot{A} \quad (3.44)$$

Expresiones a partir de las cuales se demuestran las siguientes relaciones entre los parámetros de Euler y las velocidades angulares [173].

### Relaciones entre los parámetros de Euler y las velocidades angulares

Siendo  $\omega'$  la velocidad angular relativa

$$\begin{aligned} \tilde{\omega}' &= A^T \dot{A} = 2GE^T E \dot{G}^T \\ \tilde{\omega}' &= 2GG^T = 2(\tilde{G}P) \\ \omega' &= 2G\dot{P} \\ \dot{P} &= 1/2G^T \omega' \end{aligned} \quad (3.45)$$

Y  $\omega$  la velocidad angular absoluta de forma particular para la velocidad angular absoluta

$$\begin{aligned} \omega &= A\omega' = 2EG^T G\dot{P} \\ \omega &= 2E\dot{P} \end{aligned} \quad (3.46)$$

y finalmente:

$$\dot{P} = 1/2E^T \omega \quad (3.47)$$

#### 3.3.0.3. Sistemas de múltiples cuerpos

Un sistema de múltiples cuerpos, es una cadena cinemática de eslabones atados entre sí a través de restricciones cinemáticas en las articulaciones o por fuerzas de restricción.

$$r_i + A(\phi_i) s_i'^P = r_j + A(\phi_j) s_j'^P \quad (3.48)$$

$$\phi(q, t) = r_i + A(\phi_i) s_i'^P - r_j + A(\phi_j) s_j'^P = 0 \quad (3.49)$$

Para el ejemplo ilustrado en la figura 3.9, los cuerpos  $i$  y  $j$  están atados a través de la restricción que ejerce la articulación en el punto P. Por lo tanto, la articulación P debe tener la misma descripción vista desde cualquiera de los cuerpos  $i$  o  $j$ . En este caso la restricción es de carácter geométrico, donde



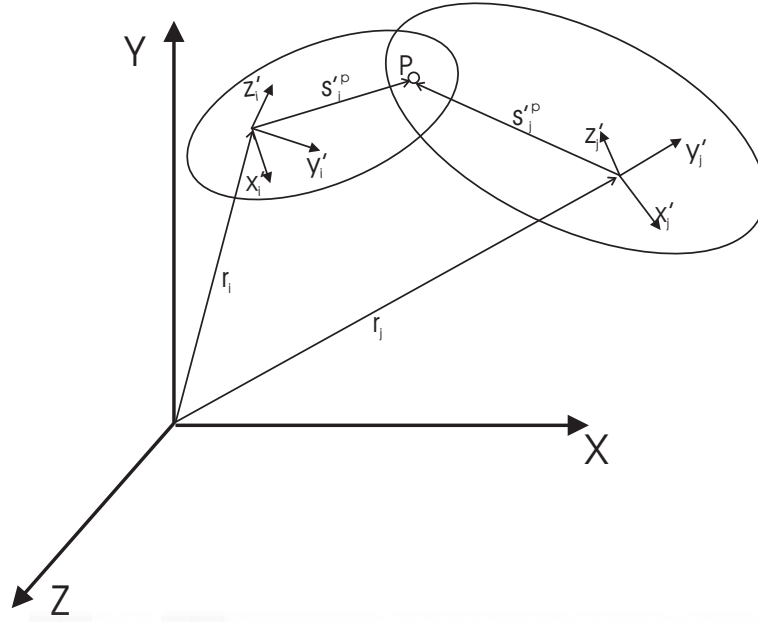


Figura 3.9: Cadena cinemática de dos cuerpos rígidos

$\phi(q, t)$  es el vector de restricciones en el que se pueden incluir restricciones que sean función explícita del tiempo (ej. accionamientos con funciones  $\phi = \omega t$ ), siendo  $q$  el vector de coordenadas generalizadas.

La derivada temporal del vector de restricciones  $\phi(q, t)$  da como resultado interesantes propiedades. Al derivar respecto del tiempo el vector de restricciones  $\phi(q, t)$ , que incluye términos que no son función explícita del tiempo, es necesario la aplicación de la regla de la cadena

$$\dot{\phi}(q, t) = \phi_q \dot{q} + \phi_t = 0 \quad (3.50)$$

$$\phi_q \dot{q} = -\phi_t = \nu \quad (3.51)$$

donde  $\dot{q} = [\dot{x}_i \dot{y}_i \dot{z}_i \dot{\alpha}_i \dot{\beta}_i \dot{\gamma}_i]^T$  es el vector de velocidades, y  $\phi_t$  es el vector de restricciones derivadas respecto del tiempo.

Derivando por segunda vez respecto del tiempo el vector de restricciones, se tiene.

$$\ddot{\phi}(q, t) = \dot{\phi}_q \dot{q} + \phi_q \ddot{q} + \phi_t = 0 \quad (3.52)$$

$$\phi_q \ddot{q} = -(\dot{\phi}_q \dot{q}) - \phi_{tt} = \gamma \quad (3.53)$$

### 3.3.0.4. Desplazamientos virtuales y rotaciones

Volviendo a la ecuación 3.39 y a la figura 3.8, se define un desplazamiento virtual  $\delta r^P$  de un punto P como un desplazamiento infinitesimal cuando el tiempo se mantiene fijo. Para un punto P que está atado al cuerpo o equivalentemente fijo respecto de su sistema de referencia  $x'y'z'$  se puede describir un desplazamiento virtual como el desplazamiento virtual de su origen  $x'y'z'$ , más la variación de sus cosenos directores.

El vector  $\delta r^P$  depende de  $r$  y de  $A$ , si el sistema es perturbado ligeramente  $r$  cambia a  $r + \delta r$  y  $A$  cambia a  $A + \delta A + 0(\delta A^2)$ . Además, dado que  $A$  es ortogonal, se cumple que:

$$\begin{aligned} (A + \delta A + O(\delta A^2))(A + \delta A + 0(\delta A^2))^T &= \\ &= AA^T + A\delta A^T + \delta AA^T + 0(\delta A^2) = I \end{aligned} \quad (3.54)$$

y, por lo tanto

$$A\delta A^T + \delta AA^T = 0(\delta A^2) \quad (3.55)$$

Igualando los términos lineales de  $\delta A$  en ambos lados, se tiene  $A\delta A^T + \delta AA^T = 0$ , y se observa que  $(\delta AA^T)^T = A\delta A^T = -\delta AA^T$ . Por lo que la matriz  $\delta AA^T$ , es una matriz Skew y se puede representar como

$$\begin{aligned} \delta AA^T &= \delta \tilde{\pi} \\ \delta A &= \delta \tilde{\pi} A \end{aligned} \quad (3.56)$$

que se puede extender a un desplazamiento virtual considerando que  $\delta \pi = A\delta \pi'$ .

$$\delta r^P = \delta r + \delta A s'^p = \delta r + \delta \tilde{\pi} A s'^p \quad (3.57)$$

así mismo como  $\delta A = A\delta \pi'$  y  $\delta p' = A^T \delta A$ , entonces también se puede escribir la ecuación 3.57 como

$$\delta r^P = \delta r + A\delta \tilde{\pi}' s'^p \quad (3.58)$$

lo que hace evidente que una rotación virtual es una cantidad vectorial.

Para ilustrar la potencia de combinar las propiedades vectoriales de los vectores y desplazamientos virtuales, se consideran sus efectos sobre una restricción colineal entre dos sistemas de referencia coincidentes  $x'_i, y'_i, z'_i$  y  $x'_j, y'_j, z'_j$ .

$$\phi = s_i^T s_j = 0 \quad (3.59)$$

$$\delta\phi = s_j^T \delta s_i + s_i^T \delta s_j = 0 \quad (3.60)$$

considerando además  $\delta s = \delta A s' = A \delta \tilde{\pi}' s' = -A \tilde{s}' \delta \pi'$ , la expresión 3.60 se puede escribir como:

$$\delta\phi = - \left( s_j'^T A_j^T A_i \tilde{s}_i' \delta \pi_i' + s_i'^T A_i^T A_j \tilde{s}_j' \delta \pi_j' \right) = 0 \quad (3.61)$$

Por lo que se puede notar que es fácil derivar expresiones diferenciales de las restricciones, lo cual facilitará más adelante la determinación de los términos del jacobiano espacial.

### Relaciones entre los parámetros de Euler y las rotaciones virtuales

El objetivo de esta sección es plantear las relaciones entre los parámetros de Euler y las rotaciones virtuales  $x'y'z'$ . A partir de la ecuación 3.37 se obtiene

$$P^T \delta P = 0 \quad (3.62)$$

y en términos de la matriz de rotación expresada en 3.31

$$\delta A = \delta E G^T + E \delta G^T \quad (3.63)$$

de la ecuación

$$\delta \pi' = A^T \delta A \quad (3.64)$$

sustituyendo  $\delta A$  por la expresión 3.63

$$\delta \pi' = 2G \delta P = 2E \delta P \quad (3.65)$$

despejando  $\delta P$

$$\delta P = 1/2G^T \delta \pi' = 1/2E^T \delta \pi' \quad (3.66)$$

Expresión que se utilizará para generar la librería de restricciones diferenciales.

### 3.3.1. Modelo y algoritmos de cinemática inversa

En este apartado se detalla el análisis cinemático inverso del dispositivo maestro MAGISTER-P (MAestro Genérico para Interfaz en eStructuras TEleopeRadas). El citado mecanismo está basado en una modificación de la conocida plataforma de Stewart o plataforma de Gough 6UPS, en la que la articulación P (prismatic) de las cadenas cinemáticas ha sido substituida por una articulación R (rotational) que transmite un movimiento lineal a través de un paralelogramo articulado. Un modelo espacial del mecanismo de 6 grados de libertad se muestra esquemáticamente en la figura 3.10.

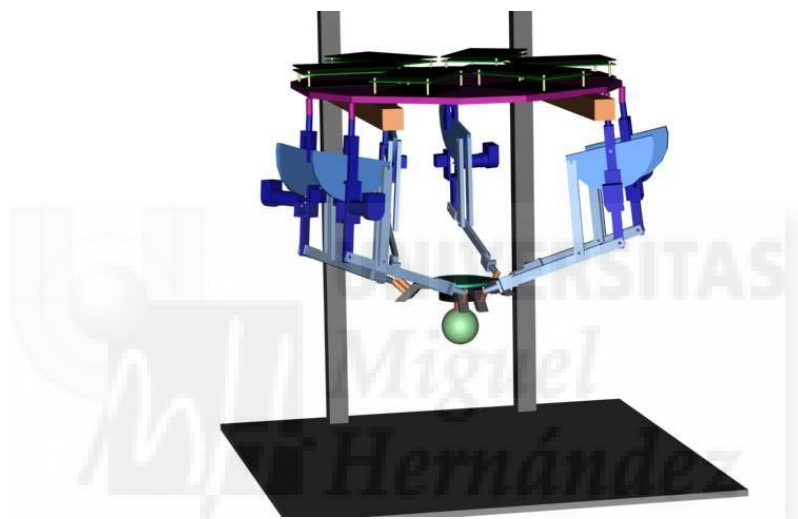


Figura 3.10: Modelo espacial del MAGISTER-P

Para el análisis cinemático se han considerado únicamente 13 eslabones dado que la inclusión del paralelogramo articulado no proporciona ninguna información adicional al no modificar la cinemática del mecanismo y complica los cálculos. Esta simplificación tendrá que ser revisada en el análisis dinámico. La figura 3.11 muestra el modelo de 13 eslabones utilizado.

Si utilizamos los parámetros de Euler para representar la orientación, cada cuerpo necesita siete coordenadas generalizadas, lo que contabiliza un total de 91 coordenadas generalizadas para los 13 cuerpos. Las restricciones impuestas por las articulaciones esféricas, rotacional y universales, junto con las restricciones de normalización de los parámetros de Euler contabilizan un

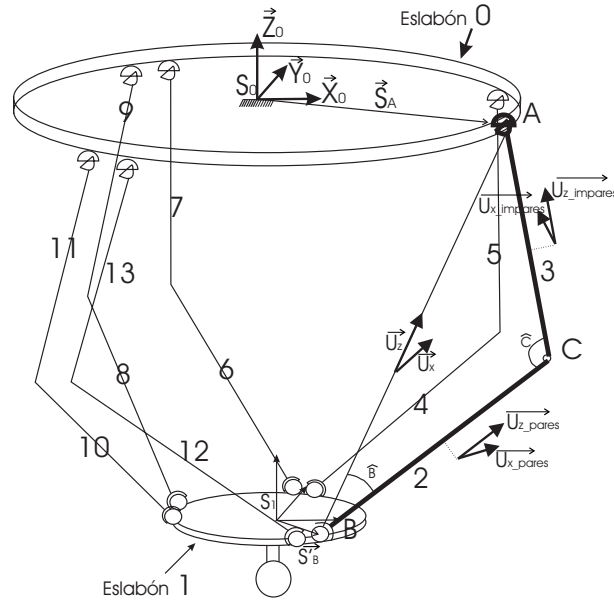


Figura 3.11: Esquema del modelo cinemático propuesto

total de 85 ecuaciones de restricción. La diferencia proporciona los grados de libertad del sistema.

El movimiento del sistema vendrá determinado por la variación temporal del ángulo  $\hat{C}$  entre los eslabones pares e impares. Esto es equivalente a imponer restricciones en los accionamientos con lo que los seis grados de libertad del sistema quedarán determinados.

Como primer paso para resolver la cinemática del mecanismo se requiere asignar un sistema de referencia a cada eslabón del dispositivo.

La base superior, que se encuentra fija, se denomina eslabón 0 y su sistema de referencia  $S_0$  (considerado inercial). Referenciados en  $S_0$  se encuentran los vectores que localizan los puntos de aplicación de las articulaciones universales  $A_n$ .

$$\vec{S}_0^{A_n} = \left[ r_{\text{base}} \cos \left( (2i) \frac{\pi}{3} \pm \frac{\delta_0}{2} \right) \quad r_{\text{base}} \sin \left( (2i) \frac{\pi}{3} \pm \frac{\delta_0}{2} \right) \quad 0 \right]^T \quad (3.67)$$

$$i = 0, \dots, 2$$

Tabla 3.2: Modelo del dispositivo maestro

<b>Coordenadas generalizadas</b>	
13 cuerpos	$13 \times 7 = \mathbf{91}$
<b>restricciones</b>	
Esféricas	$6 \times 3 = 18$
Rotacionales	$6 \times 4 = 24$
Universales	$6 \times 5 = 30$
Normalización de parámetros de Euler	$13 \times 1 = 13$
<b>Suma:</b>	<b>85</b>
grados de libertad	$91-85 = 6$

Donde  $\delta_0$  define la separación entre articulaciones contiguas y  $r_{base}$  representa la separación de las articulaciones respecto del origen del sistema de referencia.

La base inferior o joystick, se denomina eslabón 1 y su sistema de referencia  $S_1$ . Referenciados en  $S_1$  se encuentran referenciados los vectores  $s_1^{\vec{B}_n}$  que localizan los puntos de aplicación de las articulaciones esféricas  $B_n$ .

$$\vec{s}_1^{B_n} = \left[ r_{\text{joystick}} \cos \left( (2i-1) \frac{\pi}{3} \pm \frac{\delta_1}{2} \right) r_{\text{joystick}} \sin \left( (2i-1) \frac{\pi}{3} \pm \frac{\delta_1}{2} \right) 0 \right]^T$$

$$i = 0, \dots, 2$$
(3.68)

Donde  $\delta_1$  y  $r_{\text{joystick}}$  son análogos a los parámetros del eslabón 0.

Para asignar los sistemas de referencias de los eslabones pares e impares se obtiene en primer lugar un sistema de referencia auxiliar formado por los vectores  $\vec{u}_z$  y  $\vec{u}_x$ .

$$\vec{u}_z^n = \frac{r_{ABn}}{\|r_{ABn}\|}, \quad \vec{u}_x^n = \frac{u_z^n \times s_0^{An}}{\|u_z^n \times s_0^{An}\|}$$
(3.69)

Y se resuelven los ángulos  $B_n$  y  $C_n$  de cada brazo aplicando el teorema del coseno.

Los vectores  $\vec{u}_{z\text{-pares}}$  forman parte de los sistemas de referencia de los eslabones pares; 2,4,6,8,10 y 12 se obtienen mediante una rotación del vector

$\vec{u}_z$  alrededor de  $\vec{u}_x$  un ángulo  $B_n$ , calculándose a partir de ellos los vectores  $\vec{u}_{x\_pares}$  y  $\vec{u}_{y\_pares}$ .

$$\vec{u}_{z\_even}^n = (u_x \times u_x^T \times (1 - \cos B_n) + \begin{bmatrix} \cos B_n & -u_x(3) \sin B_n & u_x(2) \sin B_n \\ u_x(3) \sin B_n & \cos B_n & -u_x(1) \sin B_n \\ -u_x(2) \sin B_n & u_x(1) \sin B_n & \cos B_n \end{bmatrix}) \times u_z \quad (3.70)$$

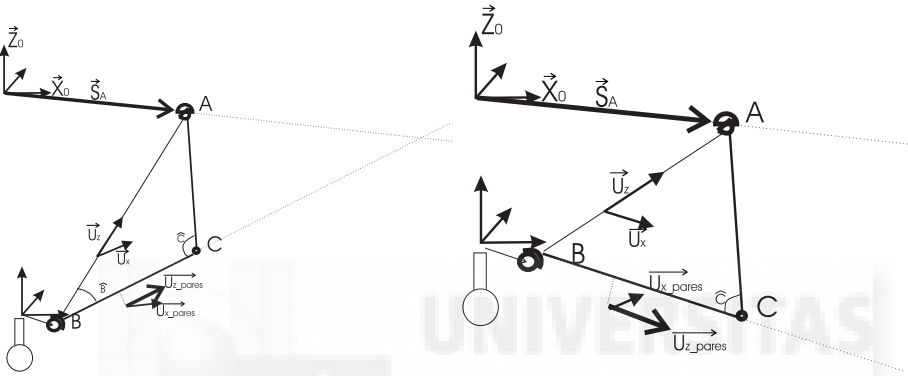


Figura 3.12: Posiciones relativas de los vectores  $\vec{u}_{z\_pares}$  y  $s_0^{A_n}$ .

Para la obtención del vector  $\vec{u}_{x\_pares}$  se debe estudiar la casuística del dispositivo. En la figura 3.12 se puede observar que el vector  $\vec{u}_{z\_pares}$  puede tener su tercera componente positiva o negativa, indicando que el vector  $\vec{u}_{z\_pares}$  queda por encima o por debajo del vector  $s_0^{A_n}$  (cuya tercera componente es nula). En consecuencia la forma de calcular el vector  $\vec{u}_{x\_pares}$  queda expresada por:

$$\vec{u}_{x\_pares}^n = \frac{\vec{u}_{z\_pares}^n \times s_0^{A_n}}{\|\vec{u}_{z\_pares}^n \times s_0^{A_n}\|}, \text{ si } \vec{u}(3)_{z\_pares} \geq 0 \quad (3.71)$$

o bien

$$\vec{u}_{x\_pares}^n = \frac{s_0^{A_n} \times \vec{u}_{z\_pares}^n}{\|s_0^{A_n} \times \vec{u}_{z\_pares}^n\|}, \text{ si } \vec{u}(3)_{z\_pares} < 0 \quad (3.72)$$

$$\vec{u}_{y\_pares}^n = \vec{u}_{z\_pares}^n \times \vec{u}_{x\_pares}^n \quad (3.73)$$

Los vectores  $\vec{u}_{z\_impares}$  forman parte de los sistemas de referencia de los eslabones impares; 3,5,7,9,11 y 13 se obtienen mediante una rotación del

vector  $\vec{u}_{z\_pares}$  alrededor de  $\vec{u}_{x\_pares}$  un ángulo  $(\pi - C_n)$ , calculándose a partir de ellos los vectores  $\vec{u}_{x\_impares}$  y  $\vec{u}_{y\_impares}$ .

$$\vec{u}_{z\_impares}^n = (\text{Rot}(\vec{u}_{x\_pares}, (\pi - C_n))) \times \vec{u}_{z\_pares} \quad (3.74)$$

$$\vec{u}_{x\_impares}^n = \frac{\vec{u}_{z\_impares}^n \times s_0^{A_n}}{\|\vec{u}_{z\_impares}^n \times s_0^{A_n}\|} \quad (3.75)$$

$$\vec{u}_{y\_impares}^n = \vec{u}_{z\_impares}^n \times \vec{u}_{x\_impares}^n \quad (3.76)$$

NOTA: Por comodidad en el uso de la toolbox de Virtual Reality utilizada para visualizar los resultados de los algoritmos desarrollados, se ha optado por no respetar la convención clásica en la que el vector  $z$  cae sobre el eje de la articulación, permaneciendo a lo largo del eje de los cilindros utilizados en VRML para la representación gráfica. Esta particularidad debe ser tomada en consideración en los desarrollos posteriores.

Una vez obtenidos los sistemas de referencia de cada eslabón se resuelve el problema de la cinemática inversa.

Dadas la posición y orientación del eslabón 1 (joystick) en un vector  $q_1 = [r_1, p_1]^T$  donde  $r_1 = [x_1, y_1, z_1]^T$  representa la posición y  $p_1 = [e_0, e_1, e_2, e_3]^T$  representa la orientación en términos de los parámetros de Euler, o bien  $q_1 = [r_1, \alpha_1, \beta_1, \gamma_1]^T$  si la orientación la representamos en términos de los ángulos 313 de Euler. Se observa que se puede calcular fácilmente la distancia entre las articulaciones universales y esféricas como:

$$\vec{r}_{ABn} = \vec{r}_1 + A_1 \vec{s}_1^{Bn} - \vec{s}_0^{Bn} \quad (3.77)$$

Donde  $A_1$  es la matriz de rotación que representa la orientación del eslabón 1.

Calculando la norma de  $r_{ABn}$  se resuelve el ángulo  $\hat{C}_n$  mediante el teorema del coseno.

$$\text{norm}(r_{ABn}) = r_{ABn}^2 = [\vec{r}_1 + A_1 \vec{s}_1^{Bn} - \vec{s}_0^{Bn}]^T [\vec{r}_1 + A_1 \vec{s}_1^{Bn} - \vec{s}_0^{Bn}] \quad (3.78)$$

$$\hat{C}_n = \arccos \left( \frac{(\overline{BC})^2 + (\overline{AC})^2 - \text{norm}(r_{ABn})^2}{2(\overline{BC})(\overline{AC})} \right) \quad (3.79)$$



Y, una vez calculado el ángulo  $\hat{C}_n$  aplicando la relación de transmisión proporcionada por el paralelogramo articulado se calcula la variable de comando del motor.

Finalmente también puede calcularse el vector de coordenadas generalizadas  $q_e$  a partir de los sistemas de referencia de cada eslabón, expresado en términos de los parámetros de Euler.

$$\vec{r}_i = \vec{r}_1 + A_1 s_i'^{\vec{B}_n} - A_{ij} s_i'^{\vec{B}_n} \quad (3.80)$$

$$\vec{r}_j = s_0^{\vec{B}_n} - A_{ij} s_i'^{\vec{A}_n} \quad (3.81)$$

$$q_e = [q_1, q_2, \dots, q_{13}]_{91 \times 1}^T \quad (3.82)$$

### 3.3.1.1. Obtención geométrica de la matriz jacobiana

Si  $q_1 = [r_1, \alpha_1, \beta_1, \gamma_1]$  es la posición cartesiana del efector final, dada la orientación según los ángulos 313 de Euler, la matriz de rotación se obtiene mediante  $A = R(\alpha_1)R(\beta_1)R(\gamma_1)$ .

Utilizando los sistemas de referencia de la figura 3.11, si  $[f, n]^T$  son las fuerzas y pares del eslabon 1, y  $N_n$  son los pares generados por los actuadores en las articulaciones rotacionales.

$$f = \sum_{n=1}^6 \vec{u}_{y\text{-pares}}^n \times \overline{BC} \times N_n \quad (3.83)$$

$$n = \sum_{n=1}^6 (s_1^{Bn} \vec{u}_{y\text{-pares}}^n) \times \overline{BC} \times N_n \quad (3.84)$$

Escribiendo las ecuaciones 3.83 y 3.84 en forma matricial y por analogía con el concepto de matriz jacobiana, se obtiene la ecuación 3.85 que muestra un método geométrico de obtención de la matriz inversa de la matriz jacobiana traspuesta.

$$\begin{bmatrix} f \\ n \end{bmatrix} = (J^T)^{-1} F = \begin{bmatrix} \vec{u}_{y\text{-pares}}^1 & \vec{u}_{y\text{-pares}}^2 & \dots & \vec{u}_{y\text{-pares}}^n \\ s_1^{B1} \times \vec{u}_{y\text{-pares}}^1 & s_1^{B2} \times \vec{u}_{y\text{-pares}}^2 & \dots & s_1^{Bn} \times \vec{u}_{y\text{-pares}}^n \end{bmatrix} F \quad (3.85)$$

Este método se utiliza para mapear las fuerzas cartesianas sobre los actuadores del Magister-P o viceversa.

### 3.3.1.2. Software desarrollado

Aprovechando las capacidades de Matlab+Simulink y de la toolbox de Virtual Reality se han desarrollado los algoritmos anteriores sobre esta plataforma, obteniendo finalmente la herramienta de simulación y visualización mostrada en las figuras 3.13 y 3.14. Es importante anotar el cambio de sistema de

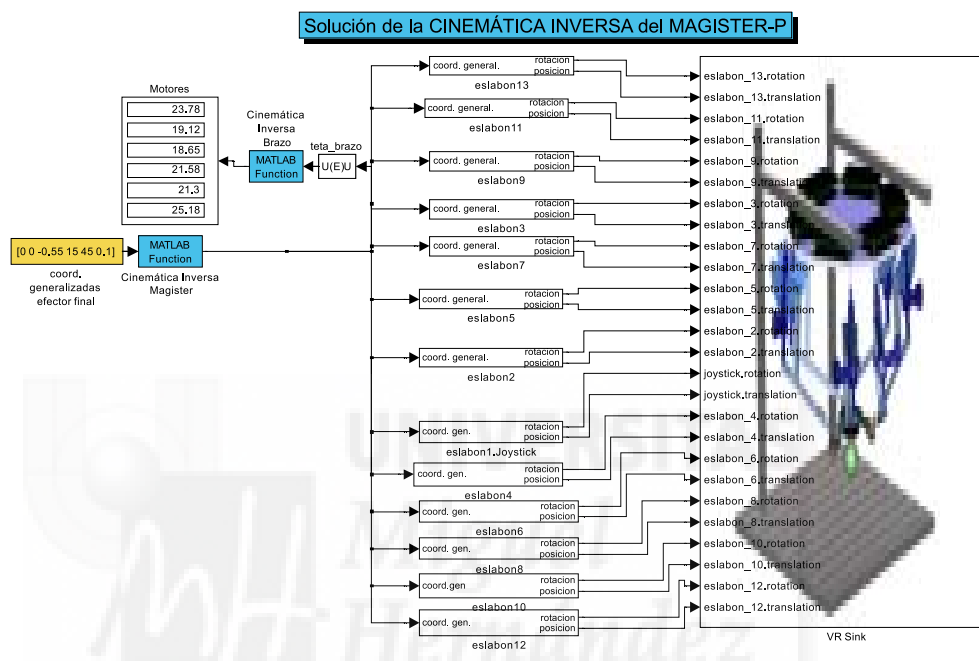


Figura 3.13: Software de solución de la Cinemática Inversa .

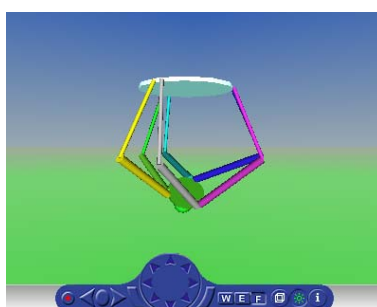


Figura 3.14: Representación VRML de la solución cinemática inversa.

referencia que se produce entre VRML y Matlab, y que está contabilizado

dentro de los bloques selector de Simulink.

Como ejemplo se muestra a continuación una ejecución en Matlab:

```
>> poef=[0 0 -0.4 30 0 0]';
>> [teta_brazo,qe0] = cinematica_inversa_magister(poef);
>> help cinematica_inversa_magister
```

CINEMATICA\_INVERSA\_MAGISTER Solución de la Cinemática Inversa

Sintaxis

```
[teta_brazo,qe] = cinematica_inversa_magister(poef)
```

poef: vector 6x1 que representa la posición y orientación del efector final del robot expresado en terminos de angulos 313 de Euler.

teta\_brazo: angulo C entre los eslabones del brazo, (6x1).

qe: vector de coordenadas generalizadas 91x1 con la configuracion final del robot.

```
>> teta_brazo
```

```
teta_brazo = 1.1371 1.2655 1.1371 1.2655 1.1371 1.2655
```

Los subsistemas de Simulink mostrados en la figura 3.13 son los siguientes:

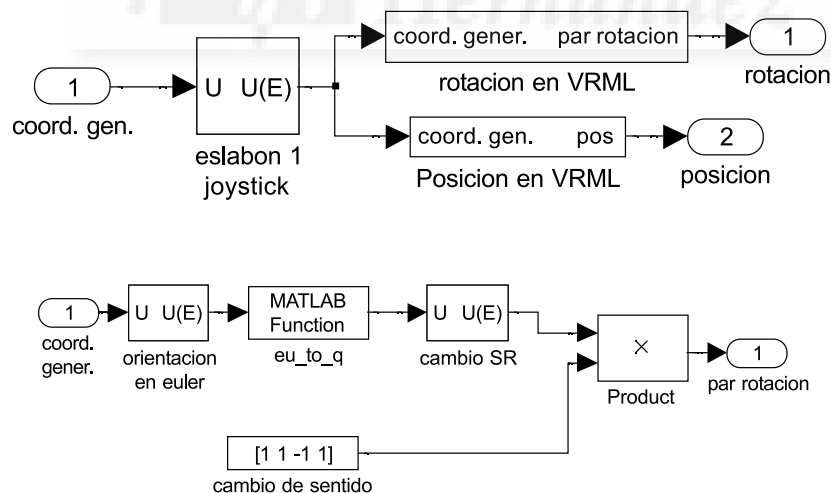


Figura 3.15: Subsistemas de Visualización VRML.

Como ayuda para la comprensión del código anexo en el cd adjunto, se muestra a continuación un sencillo esquema del funcionamiento del código.

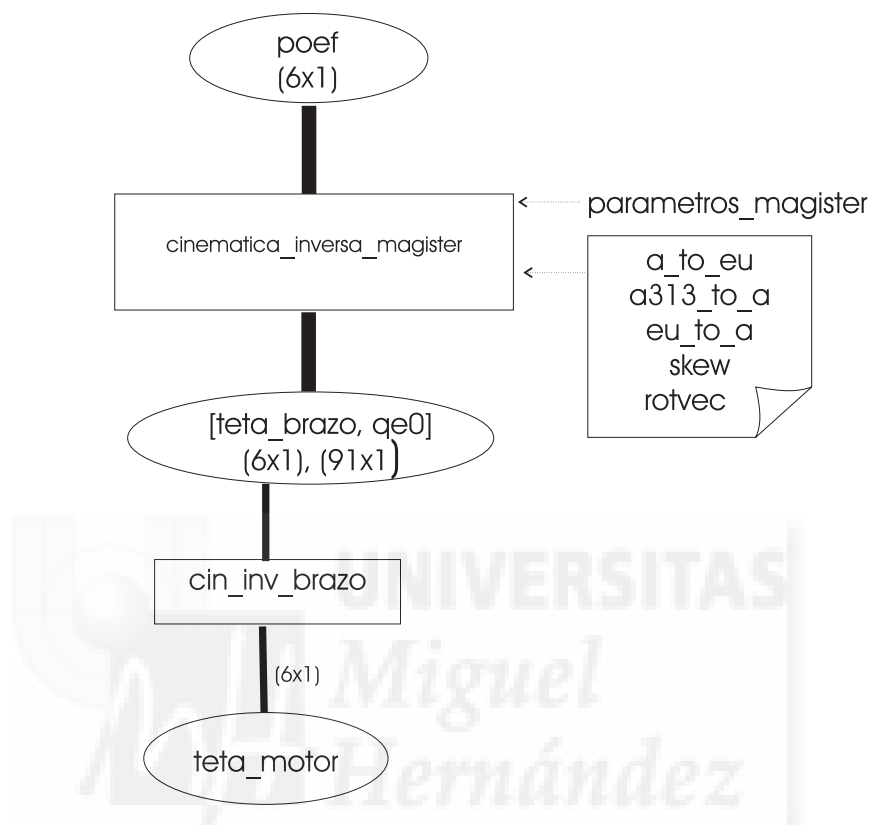


Figura 3.16: Esquema del algoritmo de solución de la Cinemática Inversa.

### 3.3.2. Modelo y algoritmos de cinemática directa

El problema cinemático directo consiste en calcular la posición y orientación del eslabón 1 conocidas las variables articulares de cada actuador. En el caso de los robots paralelos, la solución a este problema es mucho más compleja que en los robots seriales de cadena abierta. Existen en la literatura varios métodos para el cálculo geométrico de la cinemática directa de plataformas paralelas de 6 gdl, algunos de estos métodos permiten hallar las soluciones posibles a través del uso de polinomios que resultan del modelado geométrico de las cadenas cinemáticas de la plataforma, por ejemplo en [134], se calculan las 16 soluciones para una plataforma 6-3 de 6 gdl. Nair

[146] propone un método sistemático para hallar las ecuaciones polinomiales mínimas para ciertos casos de plataformas paralelas, como por ejemplo para una plataforma de 3 gdl, con su método se obtiene una solución de polinomios de grado ocho, aunque para el caso general de un robot de 6 gdl con el método de Nair se llegaría a polinomios de grado 144. Por otro lado, Dasgupta [69] propone una aproximación que reduce el problema a tres ecuaciones más una ecuación de restricciones que debe ser solucionada por métodos numéricos. Husty [101] presenta un método para la formulación polinomial de orden 40 para plataformas de 6 gdl, usando matrices de rotación basadas en cuaternios.

En esta sección se expone un método numérico basado en la estimación inicial del vector de coordenadas generalizadas,  $q_e = [q_1, q_2, \dots, q_{13}]_{91 \times 1}^T$  donde  $q_i$  corresponde al sistema de coordenadas del eslabón 1 y  $q_2, \dots, q_{13}$  corresponden al conjunto de coordenadas generalizadas asociadas a los sistemas de referencia de los eslabones pares e impares de la figura 3.11. En general cada eslabón se describe por un sistema de coordenadas generalizadas  $q_i = [r_{ABi}, P_i]^T$  donde: con  $r_{ABi} = [x_i, y_i, z_i]$ , y los parámetros de Euler  $P = [e_0^i, e_1^i, e_2^i, e_3^i]$ .

La descripción cinemática del robot se basa en una generalización de la ecuación 3.49 a un sistema con múltiples cuerpos ( $n > 2$ ), para el cual se define el vector de restricciones como:

$$\phi(q, t) = \begin{bmatrix} \phi^k(q) \\ \phi^D(q, \hat{C}(t)) \\ \phi^P(q) \end{bmatrix}_{91 \times 1} = 0 \quad (3.86)$$

Donde  $\phi^k(q)_{72 \times 1} = 0$  es el vector de las 72 restricciones holonómicas impuestas por las articulaciones esféricas (3), rotacionales (5) y universales (4).  $\phi^D(q, t)_{6 \times 1}$  es un vector de 6 restricciones impuestas por los accionamientos, y  $\phi^P(q)_{13 \times 1}$  es un vector de 13 restricciones para la normalización de los parámetros de Euler.

### 3.3.2.1. Obtención del vector de restricciones

En el caso espacial, las restricciones que se pueden plantear para varios tipos de articulaciones tales como esféricas, rotacionales, universales etc. Se construyen mediante la combinación de unas librerías básicas que comprenden relaciones cinemáticas de puntos y desplazamientos [168].

Para los dos cuerpos  $i$  y  $j$ , que se ilustran en la figura 3.17, se plantearan las restricciones básicas de ortogonalidad y distancia entre dos vectores  $a_i$  y

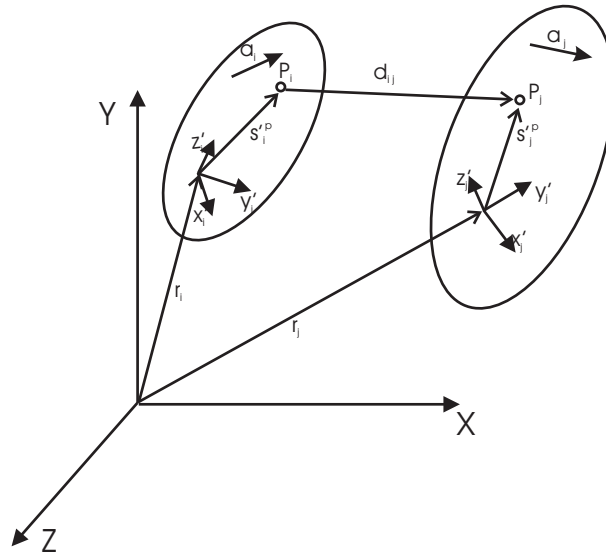


Figura 3.17: Restricciones básicas espaciales.

$a_j$ . La condición necesaria y suficiente para que dos vectores  $a_i$  y  $a_j$ , sean ortogonales es que su producto punto sea cero, y se define como:

$$\phi^{d1}(a_i, a_j) \equiv a_i^T a_j = 0 \quad (3.87)$$

donde d1 es el primer caso de producto vectorial.

$$\phi^{d1} = a_i^T A_i^T A_j A_j' = 0 \quad (3.88)$$

$A_i$  y  $A_j$  son matrices de rotación absolutas (referidas al sistema de referencia absoluto).

Un segundo tipo de producto escalar se usa para describir la ortogonalidad de un vector  $a_i$  fijado al cuerpo  $i$  y un vector  $d_{ij}$  entre los cuerpos  $i$  y  $j$ . La aplicación inmediata de este tipo de restricción se puede ver en articulaciones con desplazamiento lineales.

$$\phi^{d2}(a_i, d_{ij}) \equiv a_i^T d_{ij} = 0 \quad (3.89)$$

donde  $d_{ij} = r_j + A_j s_j'^P - r_i + A_i s_i'^P$

A menudo también se requiere una restricción que imponga la coincidencia de dos puntos  $p_i$   $p_j$ , tendremos una restricción de punto o esférica que retira tres grados de libertad

$$\phi^s(p_i, p_j) = r_j + A_j s_j'^P - r_i + A_i s_i'^P = 0 \quad (3.90)$$

la restricción de distancia o restricción esférica-esférica es un caso particular de la anterior librería.

$$\phi^{ss}(p_i, p_j) = d_{ij} d_{ij}^T - C^2 = 0 \quad (3.91)$$

Por ejemplo en el caso del paralelismo de dos vectores  $h_i$  y  $h_j$  en una articulación como se ilustra en la figura 3.18

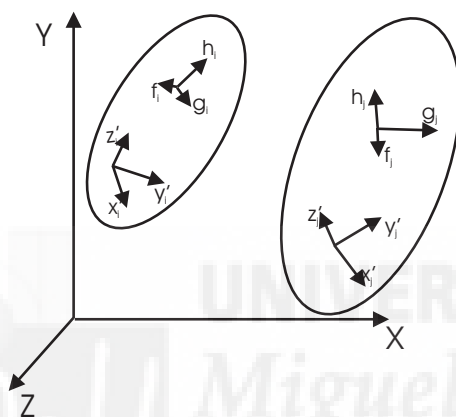


Figura 3.18: Paralelismo entre pares de cuerpos.

se plantea una restricción de ortogonalidad de producto punto entre pares de vectores, en este caso  $h_i$  y  $h_j$ , del siguiente tipo.

$$\phi^{P1}(h_i, h_j) = \begin{bmatrix} \phi^{d1}(f_i, h_j) \\ \phi^{d1}(g_i, h_j) \end{bmatrix} = 0 \quad (3.92)$$

A partir de estas restricciones básicas entre vectores se plantea construir la librería de las restricciones existentes entre los cuerpos que forman parte del MAGISTER-P.

### Restricción rotacional

Este tipo de conexión restringe cinco grados de libertad. Para el modelado, los ejes  $z_i''$  y  $z_j''$  de los sistemas de referencia auxiliares (definidos para las articulaciones) se alinean a lo largo del eje de la articulación, de tal manera

que se ajusten a las conocidas descripciones de Denavit-Hartenberg. Aunque no necesariamente a todas sus reglas, debido a la generalidad de las cadenas cinemáticas abiertas o cerradas involucradas en el modelado de un sistema mecánico.

La alineación de los ejes  $z''$  articulares permiten imponer la restricción primitiva de paralelismo entre los vectores  $h_i$  y  $h_j$  ecuación 3.92, como se ilustran en la figura 3.19. Además también deben coincidir los puntos  $P_i$  y  $P_j$ , (ecuación 3.90).

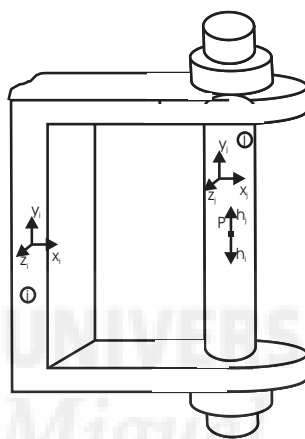


Figura 3.19: Representación de una restricción rotacional entre pares de cuerpos.

Para modelar la restricción se definen los vectores unitarios y ortogonales,  $f'_i = [1, 0, 0]^T$ ,  $g'_i = [0, 1, 0]^T$  y  $f'_j = [1, 0, 0]^T$ ,  $g'_j = [0, 1, 0]^T$ .

$$\phi^{R(i,j)} = \begin{bmatrix} \phi^s(p_i, p_j) \\ \phi^{p1}(h_i, h_j) \end{bmatrix} = \begin{bmatrix} r_j + A_j s_j^{P'} - r_i + A_i s_i^{P'} \\ f_i'^T A_i^T A_j h' \\ g_i'^T A_i^T A_j h' \end{bmatrix}_{5 \times 1} = 0 \quad (3.93)$$

Con objeto de simplificar las operaciones de visualización en VRML, los sistemas de referencia de los eslabones están de forma que su eje  $z$  esté alineado con el eje del elemento cilindro, por lo tanto en el software los ejes alineados con la articulación son  $\vec{u}_{x\_pares}$  y  $\vec{u}_{x\_impares}$ , modificándose en consiguiente la ecuación 3.93.



### Restricción esférica

Las restricciones esféricas restringe 3 grados de libertad, por lo que estas se pueden describir como se muestra en la figura 3.20 y en la siguiente ecuación:

$$\phi^{s,3}(p_i, p_j)_{3 \times 1} = r_j + A_j s_j'^P - r_i + A_i s_i'^P = 0 \quad (3.94)$$

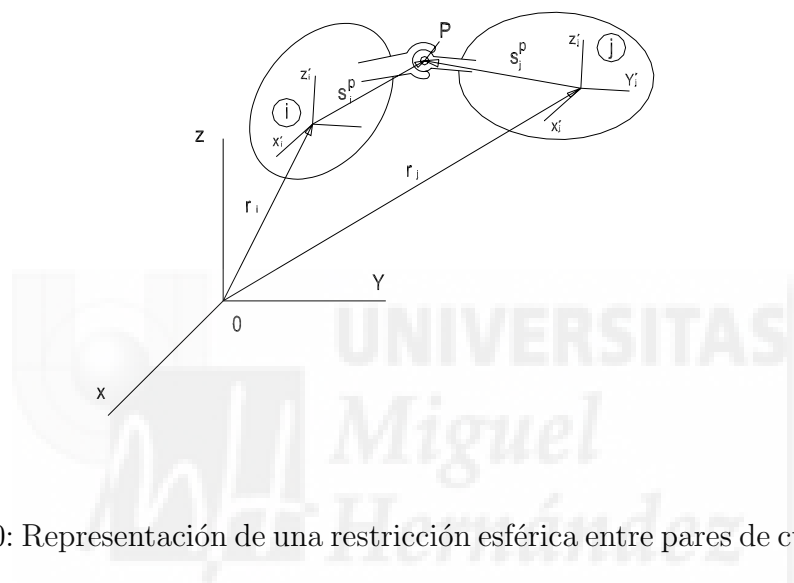


Figura 3.20: Representación de una restricción esférica entre pares de cuerpos.

Este tipo de restricción se define describiendo un punto común de la articulación esférica, desde los sistemas de coordenadas  $[x'_i, y'_i, z'_i]$  y  $[x'_j, y'_j, z'_j]$  de los cuerpos i, j.

### Restricción universal

La restricción universal se muestra en la figura 3.21. Este tipo de restricción restringe 4 grados de libertad, por lo que su descripción se basa en la combinación de una restricción esférica y el producto escalar de los vectores unitarios ortogonales  $h_i$  y  $h_j$ , (con  $h_i = A_i h'_i$ ), por lo que:

$$\begin{aligned} \phi^s(p_i, p_j)_{3 \times 1} &= 0 \\ \phi^{d,1}(h_i, h_j)_{1 \times 1} &= h_i^T h_j = h_i'^T A_i^T A_j h'_j = 0 \end{aligned} \quad (3.95)$$

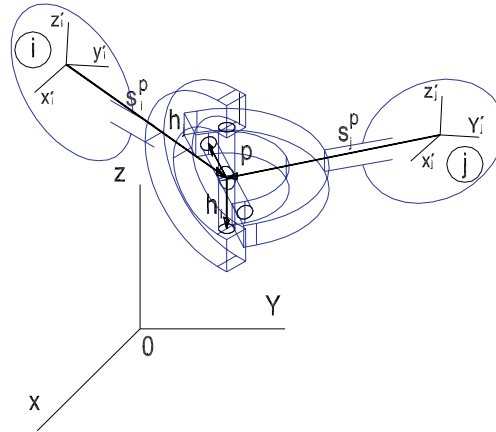


Figura 3.21: Representación de una restricción universal entre pares de cuerpos.

### Restricción de accionamiento rotacional

Como se ha comentado anteriormente se considera al ángulo  $\hat{C}$  como la variable articular.

Este tipo de generador del movimiento rotacional entre dos elementos se basa en la determinación del ángulo  $\theta$  relativo entre los ejes  $x'_i$  y  $x'_j$ , que se ilustran en la figura 3.22.

De la definición del producto escalar y vectorial, se tiene

$$h_i^T h_i = \cos \theta \quad (3.96)$$

$$\tilde{h}_i^T h_j = \sin \theta$$

de las anteriores expresiones se plantea

$$\cos \theta = h_i'^T A_i^T A_j h_j \quad (3.97)$$

$$\sin \theta = g_i'^T A_i^T A_j h_j$$

A partir de donde se obtiene el ángulo  $\theta$  utilizando la función atan2, dado que la articulación no puede tener revoluciones acumuladas, definiendo la librería de accionamiento como:

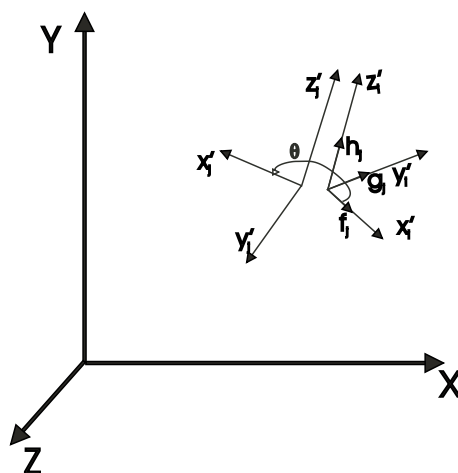


Figura 3.22: Representación de una restricción de accionamiento rotacional.

$$\phi^{rot d} = \theta - f(t) = 0 \quad (3.98)$$

### Restricción de normalización de los parámetros de Euler

La descripción de los sistemas de referencia multicuerpo en términos de los parámetros de Euler, obliga a establecer restricciones para la normalización de estos parámetros, y se escriben como:

$$\phi_i^p = p_i^T p_i - 1 = 0 \quad (3.99)$$

#### 3.3.2.2. Obtención de la matriz jacobiana del vector de restricciones

Además del vector de restricciones dado por la ecuación 3.86, para la obtención de la solución cinemática directa, se necesita calcular la matriz derivada del vector de restricciones respecto de las coordenadas generalizadas, es decir, la matriz jacobiana.

La matriz jacobiana del robot de la figura 3.11 se plantea como:

$$\phi_q(q) = \begin{bmatrix} \phi_q^k \\ \phi_q^D \\ \phi_q^P \\ \phi_q^p \end{bmatrix}_{91 \times 91} \quad (3.100)$$

Donde  $\phi_q^K$  representa los términos diferenciales de las restricciones holonómicas,  $\phi_q^D$  representa los términos diferenciales de las restricciones impuestas por los actuadores, y  $\phi_q^P$  representa los términos diferenciales de la normalización de los parámetros de Euler obtenidos todos ellos a partir de las ecuaciones 3.101 [168].

$$\begin{aligned}\phi_{qi}^k &= [\phi_{ri}^k, \phi_{pi}^k] = [\phi_{ri}^k, 2\phi_{\pi i}^k G_i] \\ \phi_{qi}^D &= [\phi_{ri}^D, \phi_{pi}^D] = [\phi_{ri}^D, 2\phi_{\pi i}^D G_i] \\ \phi_{qi}^P &= [0, 2p_i^T]\end{aligned}\quad (3.101)$$

Los términos anteriores se calculan a partir de la diferenciación de las restricciones entre pares de cuerpos, y de las relaciones diferenciales de la velocidad y la aceleración. Debido a las particularidades anotadas anteriormente a causa del carácter no integrable de las velocidades angulares, es necesario plantear las relaciones diferenciales cinemáticas, a través de los desplazamientos virtuales lineales y angulares de las restricciones. El diferencial de la primera restricción básica  $\phi^{d1}(a_i, a_j)$  se plantea a continuación usando las relaciones entre los desplazamientos virtuales y rotaciones al igual que se hizo en la ecuación 3.61.

$$\delta\phi^{d1}(a_i, a_j) = - \left( a_j'^T A_j^T A_i \tilde{a}_i' \delta\pi_i' + a_i'^T A_i^T A_j \tilde{a}_j' \delta\pi_j' \right) \quad (3.102)$$

para el caso de la restricción  $\phi^s(P_i, P_j)$ , el planteamiento diferencial es

$$\delta\phi^s(P_i, P_j) = \delta r_j - A_j \tilde{s}_j'^P \delta\pi_j' - \delta r_i + A_i \tilde{s}_i'^P \delta\pi_i' \quad (3.103)$$

En general todas las restricciones diferenciales se pueden escribir como:

$$\delta\phi = \phi_{ri} \delta r_i + \phi_{\pi_i} \delta\pi_i' + \phi_{r_j} \delta r_j + \phi_{\pi_j} \delta\pi_j' = 0 \quad (3.104)$$

con lo que se puede concluir que la expresión 3.104, refleja el planteamiento general para la construcción de los términos del jacobiano de las restricciones cinemáticas. En la siguiente tabla se resumen los términos diferenciales para calcular las primitivas de las restricciones que aparecen en el MAGÍSTER-P.

La estructura de la matriz jacobiana construida a partir de las primitivas anteriores se puede observar utilizando la función *spy* de Matlab donde los ceros en la matriz se representan como espacios en blanco (figura 3.23).

Tabla 3.3: Resumen de Restricciones primitivas

Función de restr.	$\phi_{ri}$	$\phi_{rj}$	$\phi_{\pi'_i}$	$\phi_{\pi'_j}$
$\phi^{d1}(a_i, a_j)$	0	0	$a_j'^T A_j^T A_i \tilde{a}_i'$	$a_i'^T A_i^T A_j \tilde{a}_j'$
$\phi^s(P_i, P_j)$	-I	I	$A_i \tilde{s}_i'^T P$	$-A_j \tilde{s}_j'^T P$

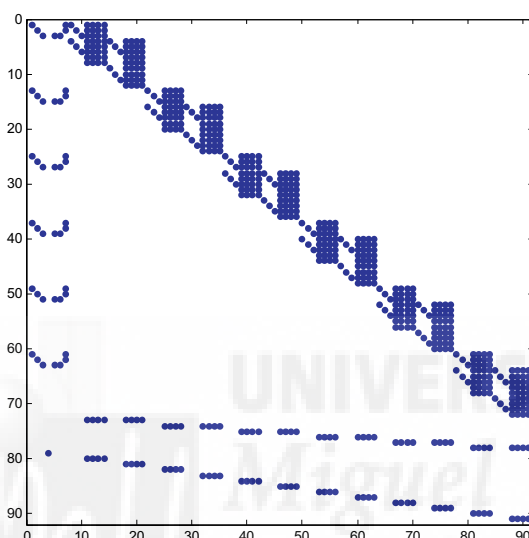


Figura 3.23: Estructura de la matriz jacobiana del vector de restricciones.

### 3.3.2.3. Solución numérica

Como se mencionaba antes, para calcular la solución cinemática directa se parte de un vector de coordenadas generalizadas aproximado  $q_i$ , y los valores de las variables articulares correspondientes al ángulo comprendido entre dos eslabones consecutivos (que para el caso de la solución cinemática directa depende únicamente de la variable de comando). Para estos efectos es común usar el método iterativo de Newton-Raphson.

$$\phi_q \Delta q^{(i)} = -\phi(q^{(j)}, t) \quad (3.105)$$

$$q^{(j+1)} = q^{(j)} + \Delta q^{(j)} \quad (3.106)$$

donde  $\phi_q$  es el jacobiano del vector de restricciones descrito en 3.100 y  $q^{j+1}$  es la solución cinemática directa, cuando  $\Delta q^{(j)} \approx 0$ .

### 3.3.2.4. Software desarrollado

Se presenta a continuación la implementación en Matlab, Simulink y Virtual Reality Toolbox de los algoritmos anteriores. La ejecución en Matlab del

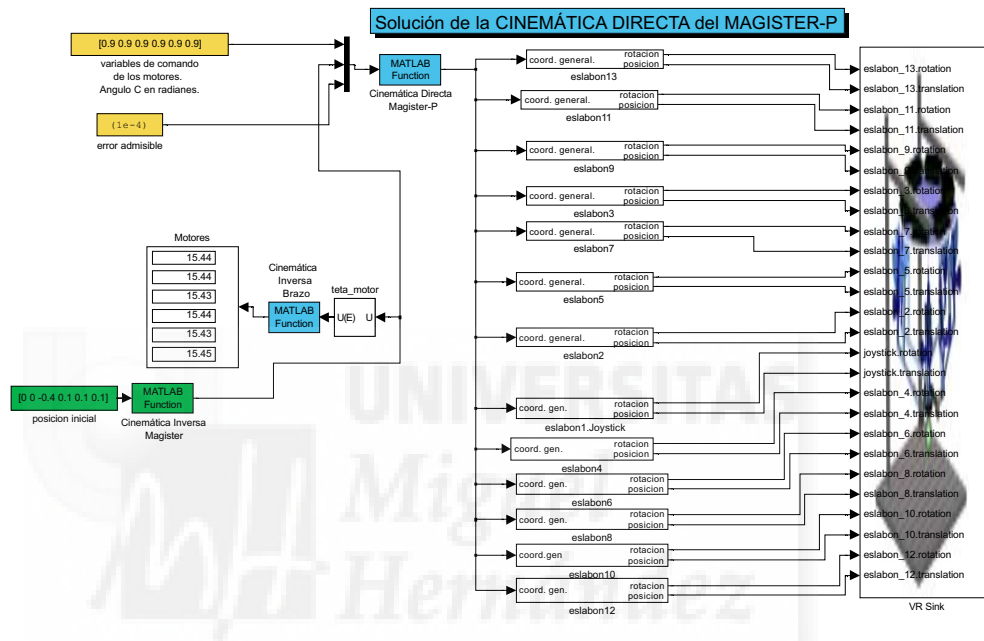


Figura 3.24: Software de solución de la Cinemática Directa .

problema cinemático directo se realiza de la siguiente forma:

```
>> help cinemática_directa_magister
```

```
CINEMATICA_DIRECTA_MAGISTER  Calcula la cinemática directa
Sintaxis
```

```
[qe] = cinemática_directa_magister(teta_brazo,qe_inicial,error)
```

teta\_brazo: vector 6x1 que representa el ángulo entre eslabones  
 qe\_inicial: vector de coordenadas generalizadas 91x1 que representa la configuración inicial del robot.

qe: vector 91x1 con la nueva configuración

>>

Como ayuda para la comprensión del código anexo en el cd adjunto, se muestra en la figura 3.25 un sencillo esquema del funcionamiento del código.

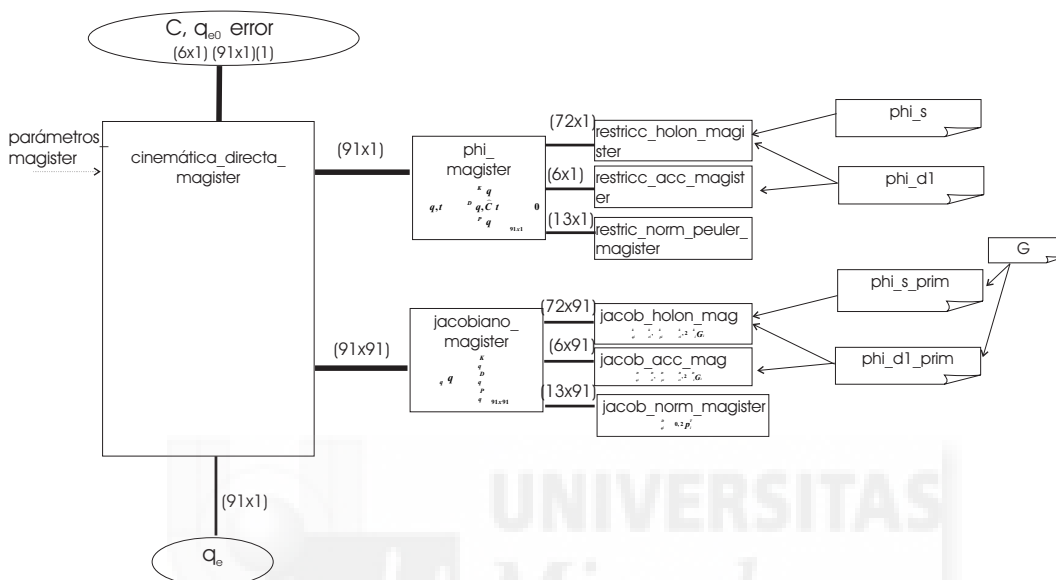


Figura 3.25: Esquema del algoritmo de solución de la Cinemática Directa.

### 3.3.3. Análisis de velocidad y aceleración en coordenadas angulares

Después de calcular el vector de posición exacto de coordenadas generalizadas  $q$ , se puede determinar la velocidad y aceleración de la plataforma URS de acuerdo a las siguientes expresiones las cuales son función de la velocidad angular  $\omega'$  y  $\dot{\omega}$ . Se puede demostrar que para los propósitos del cálculo de la velocidad y aceleración en términos de las velocidades y aceleraciones angulares, no se requieren las restricciones de la normalización de los parámetros de Euler.

$$\sum_{i=1}^{nb} \left( \begin{bmatrix} \phi_{ri}^k \\ \phi_{ri}^D \end{bmatrix} \dot{r}'_i + \begin{bmatrix} \phi_{\pi i}^k \\ \phi_{\pi i}^D \end{bmatrix} \omega'_i \right) = \begin{bmatrix} -\phi_t^k \\ \phi_t^D \end{bmatrix} \equiv \begin{bmatrix} \nu^k \\ \nu^D \end{bmatrix} = \begin{bmatrix} 0 \\ \dot{C}(t) \end{bmatrix} \quad (3.107)$$

$$\sum_{i=1}^{nb} \left( \begin{bmatrix} \phi_{ri}^k \\ \phi_{ri}^D \end{bmatrix} \ddot{r}'_i + \begin{bmatrix} \phi_{\pi i}^k \\ \phi_{\pi i}^D \end{bmatrix} \dot{\omega}'_i \right) = \begin{bmatrix} -\phi_{tt}^k \\ \phi_{tt}^D \end{bmatrix} - \sum_{i=1}^{nb} \left( \begin{bmatrix} \dot{\phi}_{ri}^k \\ \dot{\phi}_{ri}^D \end{bmatrix} \dot{r}'_i + \begin{bmatrix} \dot{\phi}_{\pi i}^k \\ \dot{\phi}_{\pi i}^D \end{bmatrix} \dot{\omega}'_i \right) \equiv \begin{bmatrix} \gamma^k \\ \gamma^D \end{bmatrix} \quad (3.108)$$

Donde  $\dot{C}(t)$  es el polinomio de velocidades que se obtiene a partir de la planificación de trayectorias y los términos  $\gamma^k$  y  $\gamma^D$  se pueden determinar de las referencias [148], ó [89].

### 3.3.3.1. Software desarrollado

Como ayuda para la comprensión del código anexo en el cd adjunto, se muestra en la figura 3.26 un sencillo esquema del funcionamiento del código.

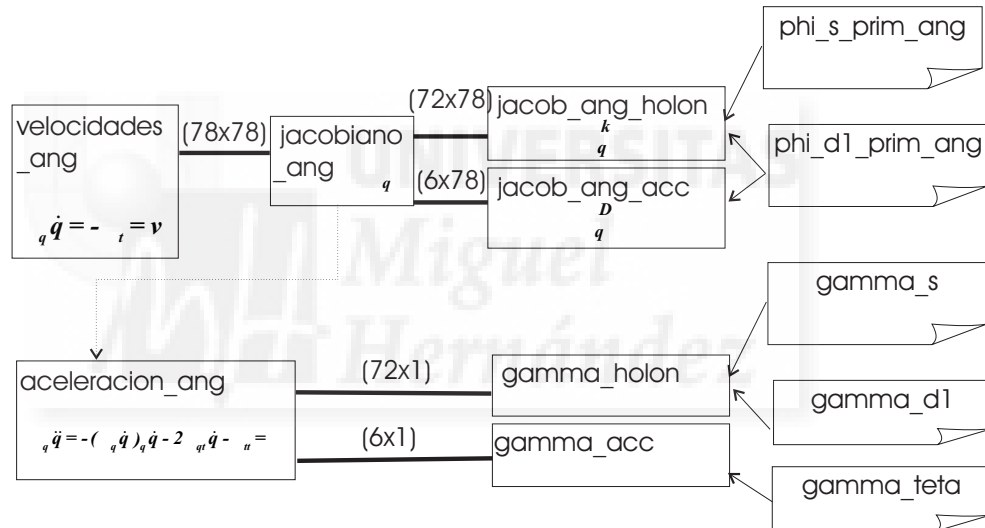


Figura 3.26: Esquema del algoritmo de solución del análisis de velocidades y aceleraciones.

### 3.3.4. Simulación cinemática con planificación de trayectorias

Aprovechando la potencia de Simulink y trabajos previos realizados en “Prácticas de robótica utilizando Matlab” [169], se han utilizado los polinomios interpoladores 4-3-4 [84] para realizar una planificación de trayec-



torias articular teniendo en consideración los parámetros de los actuadores. Estas simulaciones permitieron validar los algoritmos cinemáticos desarrollados. Dichas simulaciones se encuentran en el cd adjunto.

### 3.4. Modelado dinámico de un sistema multicuerpo

El análisis dinámico de un mecanismo comprende la determinación de su modelo cinemático y dinámico, la definición de sus condiciones iniciales, fuerzas, inercias, y propiedades de los componentes del sistema.

#### 3.4.0.1. Bases matemáticas para el movimiento de 1 cuerpo rígido

En consideración de la dinámica Newtoniana, definiremos un cuerpo rígido como se muestra en la figura 3.27, a través de un vector  $r$  y la orientación del sistema de referencia  $x'y'z'$  atado al cuerpo. Definiremos también un punto  $P$  y un vector  $s^P$  fijo respecto del sistema  $x'y'z'$ , que describe un diferencial de masa  $dm(P)$ .

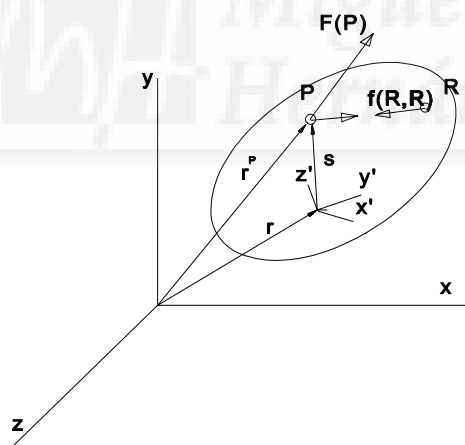


Figura 3.27: Definición de cuerpo espacial.

La fuerza que actúa sobre el diferencial de masa en el punto  $P$  comprende la fuerza externa  $F(P)$  por unidad de masa, y las fuerzas de acción y reacción  $f(P, R)$  y  $f(R, P)$ . Estas fuerzas internas se refieren únicamente a los efectos

de interacción gravitacional y restricciones de distancia. Para el diferencial de masa  $dm(P)$ , la segunda ley de Newton es:

$$\ddot{r}^P dm(P) - F(P)dm(P) - \int_m f(P, R)dm(R)dm(P) = 0 \quad (3.109)$$

donde el tercer término de la anterior ecuación es cero por la tercera ley de Newton. Aunque esto no es estrictamente cierto para fuerzas internas no lineales como campos eléctricos o magnéticos, los cuales deben ser considerados como fuerzas externas. Para la expresión 3.109, un desplazamiento virtual queda como:

$$\int_m \delta r^{PT} \ddot{r}^P dm(P) = \int_m \delta r^{PT} F(P)dm(P) = 0 \quad (3.110)$$

Por otra parte, un desplazamiento virtual, se escribe para el caso espacial como:

$$\delta r^P = \delta r - A \tilde{s}'^P \delta \pi' \quad (3.111)$$

Y, de forma similar la aceleración se escribe como:

$$\ddot{r}^P = \ddot{r} + A \dot{\tilde{\omega}} \tilde{s}'^P + A \tilde{\omega}' \tilde{\omega}' s'^P \quad (3.112)$$

reemplazando estos términos en 3.110, obtendremos la siguiente ecuación

$$\begin{aligned} \int_m (\delta r^T - \delta \pi'^T \tilde{s}'^T A^T) (\ddot{r} + A \dot{\tilde{\omega}} \tilde{s}'^P + A \tilde{\omega}' \tilde{\omega}' s'^P) dm(P) \\ - \int_m (\delta r^T - \delta \pi'^T \tilde{s}'^T A^T) F(P)dm(P) = 0 \end{aligned} \quad (3.113)$$

y expandiendo estas integrales.

$$\begin{aligned} \delta r^T \ddot{r} \int_m dm(P) + \delta r^T (A \dot{\tilde{\omega}} + A \tilde{\omega} \tilde{\omega}) \int_m s'^P dm(P) \\ + \delta \pi'^T \int_m s'^P dm(P) A^T \ddot{r} + \delta \pi'^T \int_m \tilde{s}'^T \dot{\tilde{\omega}} s'^P dm(P) \\ + \delta \pi'^T \int_m \tilde{s}'^P \tilde{\omega}' \tilde{\omega}' s'^P dm(P) - \delta r^T \int_m F(P)dm(P) \\ - \delta \pi'^T \int_m \tilde{s}'^P F'(P)dm(P) = 0 \end{aligned} \quad (3.114)$$

para todo desplazamiento virtual  $\delta r$  y  $\delta \pi'$  que sea consistente con las restricciones, la fuerza externa total de un cuerpo es:

$$F = \int_m F(P) dm(P) \quad (3.115)$$

y el momento es

$$n' = \int_m \tilde{s}'^P F'(P) dm(P) \quad (3.116)$$

la cuarta integral se escribe como:

$$\int_m \tilde{s}'^P \tilde{\omega}' \tilde{\omega}' s'^P dm(P) = - \left( \int_m \tilde{s}'^P dm(P) \right) \dot{\omega}' = J' \dot{\omega}' \quad (3.117)$$

Donde por definición la inercia de masas es un tensor de 3x3

$$J' \equiv - \int_m \tilde{s}'^P \tilde{s}'^P dm(P) \quad (3.118)$$

donde cada termino se integra por separado. Los términos de la diagonal son los momentos de inercia y los términos por fuera de la diagonal son los productos de inercia.

La integral del quinto integrando de la ecuación 3.114 se expande como:

$$\tilde{s}'^P \tilde{\omega}' \tilde{\omega}' s'^P = - \tilde{\omega}' \tilde{s}'^P s'^P \tilde{\omega}' \quad (3.119)$$

e integrando a ambos lados y reemplazando las ecuaciones 3.115 a 3.118 en 3.114, se obtiene la ecuación variacional del movimiento de Newton-Euler.

$$\delta r^T [m\ddot{r} - F] + \delta \pi' [J'\omega + \tilde{\omega}' J' \omega' - n'] = 0 \quad (3.120)$$

Que debe cumplirse para todo desplazamiento virtual  $\delta r$  y rotacional  $\delta \pi'$  del sistema centroidal  $x'y'z'$  que sean consistentes con las restricciones que actúan sobre el cuerpo. Si no existen restricciones como en el caso del movimiento de una partícula en el espacio, entonces:

$$m\ddot{r} = F \quad (3.121)$$

$$J'\omega = n' - \tilde{\omega}' J' \omega'$$

### 3.4.0.2. Sistemas de múltiples cuerpos con restricciones

Considerando un sistema de nb cuerpos el conjunto de coordenadas generalizadas del sistema es:

$$r = [r_1^T, r_2^T, \dots, r_{nb}^T]^T \quad (3.122)$$

$$P = [P_1^T, P_2^T, \dots, P_{nb}^T]^T$$

Las ecuaciones del movimiento en primera instancia se derivan en términos de las aceleraciones angulares, y posteriormente en términos de las segundas derivadas de los parámetros de Euler.

Para el sistema de nb cuerpos la ecuación 3.120 se escribe como:

$$\delta r^T [M\ddot{r} - F] + \delta \pi' [J'\omega + \tilde{\omega}'J'\omega - n'] = 0 \quad (3.123)$$

ecuación que debe cumplirse para todo desplazamiento virtual  $\delta r$  y rotacional  $\delta \pi$  que cumpla con las restricciones del movimiento.

Las fuerzas y pares que actúan sobre el sistema pueden partitionarse en fuerzas y pares aplicados  $F^A$  y  $n'^A$ , y fuerzas y pares internos  $F^C$  y  $n'^C$  los cuales también deben ser consistentes con los desplazamientos y rotaciones virtuales, por lo que aplicando el teorema de los multiplicadores de Lagrange se debe cumplir:

$$\delta r^T [M\ddot{r} - F^A + \phi_r^T \lambda] + \delta \pi' [J'\omega + \tilde{\omega}'J'\omega' - n'^A + \phi_{\pi'}^T \lambda] = 0 \quad (3.124)$$

Expresión que se puede particionar como:

$$M\ddot{r} + \phi_r^T \lambda = F^A \quad (3.125)$$

$$J'\omega + \phi_{\pi'}^T \lambda = n'^A - \tilde{\omega}'J'\omega'$$

para completar las ecuaciones del movimiento se deben asociar las ecuaciones de las aceleraciones y velocidades.

$$\phi_r \dot{r} + \phi_{\pi'} \omega' = \nu \quad (3.126)$$

$$\phi_r \ddot{r} + \phi_{\pi'} \dot{\omega}' = \gamma$$

ecuaciones que se pueden integrar en una ecuación matricial

$$\begin{bmatrix} M & 0 & \phi_r^T \\ 0 & J' & \phi_{\pi'}^T \\ \phi_r & \phi_{\pi'} & 0 \end{bmatrix} \begin{bmatrix} \ddot{r} \\ \dot{\omega}' \\ \lambda \end{bmatrix} = \begin{bmatrix} F^A \\ n'^A - \tilde{\omega}'J'\omega' \\ \gamma \end{bmatrix} \quad (3.127)$$

La ecuación anterior, es una ecuación mixta diferencial algebraica de primer orden para la velocidad y la variable algebraica de los multiplicadores de Lagrange. Pero no puede ser considerada una ecuación diferencial de segundo orden debido a que la velocidad angular no es integrable, por lo que el sistema debe ser aumentado con la ecuación de la velocidad en términos de los parámetros de Euler.

Para el modelado se deben definir también las condiciones iniciales para la posición, velocidad y aceleración. Debido a que la posición de un cuerpo se indican en términos de los parámetros de Euler, las condiciones deben ser definidas como:

$$\phi^I(r, P, t_0) = 0 \quad (3.128)$$

donde  $r$  y  $P$  deben satisfacer el vector de restricciones  $\phi(r, P, t)$  y la normalización de los parámetros de Euler.

Si se usa la velocidad angular para formular las ecuaciones del movimiento, entonces la velocidad inicial debe cumplir la siguiente ecuación.

$$B_r r^I \dot{r} + B_\omega^I \omega' = \nu^I \quad (3.129)$$

Las ecuaciones 3.127 y 3.128 se pueden expresar en función de los parámetros de Euler, [168], quedando de la siguiente forma:

$$\begin{bmatrix} M & 0 & \phi_r^T & 0 \\ 0 & 4G^T J' G & \phi_P^T & \phi_P^{PT} \\ \phi_r & \phi_P & 0 & 0 \\ 0 & \phi_P^P & 0 & 0 \end{bmatrix} \begin{bmatrix} \ddot{r} \\ \ddot{P} \\ \lambda \\ \lambda^P \end{bmatrix} = \begin{bmatrix} F^A \\ 2G^T n'^A + 8\dot{G}^T J' \dot{G} P \\ \gamma \\ \gamma^P \end{bmatrix} \quad (3.130)$$

si bien en esta tesis se ha utilizado la ecuación 3.127 y posteriormente se han pasado a parámetros de Euler.

### 3.4.1. Modelo y algoritmos de dinámica inversa

Para encontrar el modelo dinámico del maestro, se debe recordar que estamos trabajando sobre un modelo geométrico simplificado, en el que el paralelogramo articulado de cada brazo ha quedado reducido a dos eslabones con una articulación rotacional. Sin embargo, los efectos másicos e inerciales de estos eslabones si deben considerarse dentro del modelo dinámico.

Para la realización de esta tesis, las piezas disco y horquilla del paralelogramo articulado han sido simplemente adicionadas al eslabón correspondiente, tomando en consideración su masa y la distribución de la misma, si

bien para obtener un modelado más correcto se debería construir el vector de restricciones del modelo sin simplificar. Los datos utilizados han sido los mostrados en la tabla 3.1.

La obtención de las fuerzas requeridas en los actuadores para obtener un determinado movimiento en el efector final se conoce como problema dinámico inverso. La solución del problema dinámico inverso se resuelve planteando la ecuación 3.127 de forma reducida como:

$$\begin{bmatrix} M & \phi_q^T \\ \phi_q & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ \lambda \end{bmatrix} = \begin{bmatrix} Q^A \\ \gamma \end{bmatrix} \quad (3.131)$$

Donde el determinante de la matriz jacobiana debe ser diferente de cero,

$$|\phi_q(q, t)| \neq 0 \quad (3.132)$$

no singular. De acuerdo al teorema de la función implícita, la ecuación (3.131) y conociendo la aceleración por la solución del problema cinemático, entonces las fuerzas de restricción se calculan de la siguiente expresión:

$$\lambda = (\phi_q^T)^{-1} [Q^A - M\ddot{q}] \quad (3.133)$$

Donde  $\lambda$  son las fuerzas articulares, correspondiendo a las restricciones de accionamiento del vector de restricciones, que además determinan los valores de las fuerzas y pares de accionamiento de reacción necesarias para calcular los accionamientos y las fuerzas y pares de reacción en las articulaciones (importantes para los análisis estructurales).

#### **3.4.1.1. Software desarrollado**

La figura 3.28 muestra la herramienta software desarrollada para la solución de la dinámica inversa del dispositivo. Como ayuda para la comprensión del código anexo en el cd adjunto, se muestra en la figura 3.29 un sencillo esquema del funcionamiento del código.

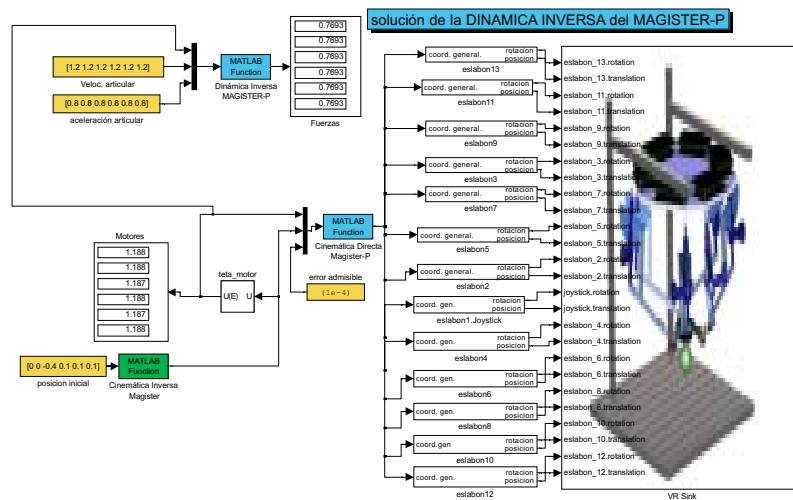


Figura 3.28: Software de solución de la Dinámica Inversa.

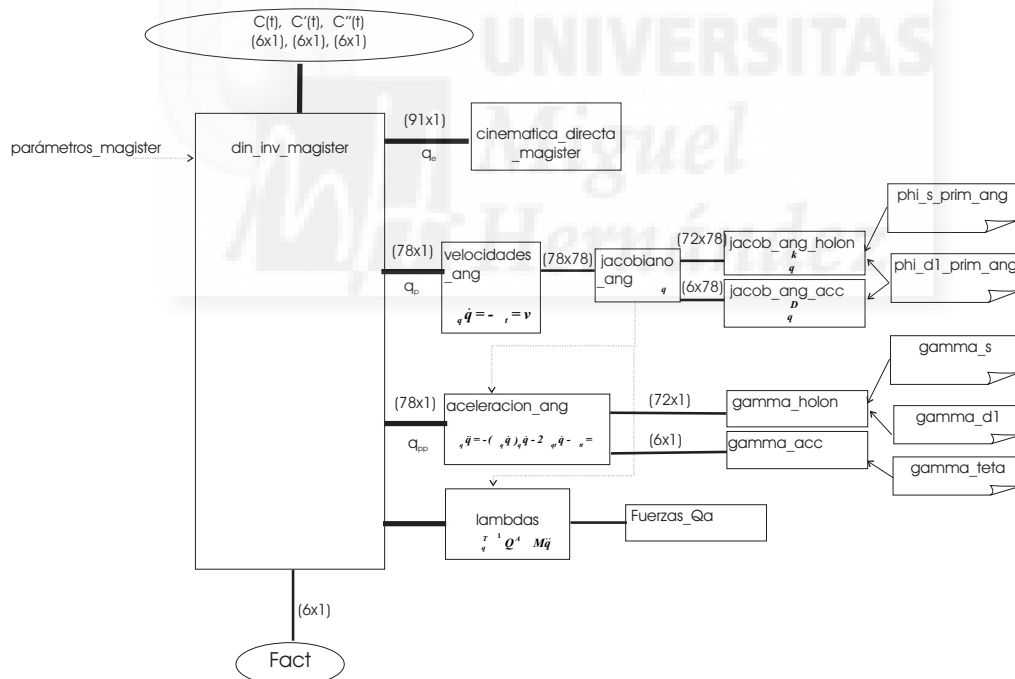


Figura 3.29: Esquema del algoritmo de solución de la dinámica inversa.

### 3.4.2. Modelo y algoritmos de dinámica directa

La solución del problema dinámico consiste en la integración de la ecuación de aceleración de un sistema mecánico con restricciones. En términos numéricos, el problema es la integración de una serie de ecuaciones diferenciales algebraicas con condiciones iniciales.

$$\begin{bmatrix} M & \phi_q^T \\ \phi_q & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ \lambda \end{bmatrix} = \begin{bmatrix} Q^A \\ \gamma \end{bmatrix} \quad (3.134)$$

Donde  $Q^A$  es el vector de fuerza generalizadas externas. En el caso espacial puede depender de las coordenadas generalizadas y de los términos no lineales de los efectos de Coriolis.

Para el modelado deben cumplirse las restricciones del sistema y las ecuaciones del movimiento

$$\phi(q, t) = 0 \quad (3.135)$$

donde se presume que la ecuación (3.135) es continua y derivable por lo menos dos veces respecto de sus argumentos. Así también debe cumplirse la ecuación de la velocidad.

$$\phi_q \dot{q} = \nu \quad (3.136)$$

Las tres ecuaciones anteriores (3.134) a (3.136) constituyen una serie mixta de ecuaciones algebraicas diferenciales, cuya solución debe considerar las condiciones iniciales de posición y velocidad.

$$\begin{aligned} \phi^I(q(t_0), t_0) &= 0 \\ B^I \dot{q}(t_0) &= \nu^I \end{aligned} \quad (3.137)$$

En el problema dinámico directo es esencial que las ecuaciones (3.137) tengan una solución única. Si se considera que  $|\phi_q| \neq 0$ , teóricamente es posible solucionar las ecuaciones (3.137). La integración de las anteriores ecuaciones no es un problema trivial, ya que se debe satisfacer las ecuaciones (3.134) a (3.136), además el planteamiento de coordenadas generalizadas implica un gran número de ecuaciones dependientes e independientes que deben ser separadas, para la solución del problema dinámico directo.

$$\begin{aligned} \ddot{q} &= f(q, \dot{q}, t) \\ \lambda &= g(q, \dot{q}, t) \end{aligned} \quad (3.138)$$



Para resolver numéricamente el problema de valor inicial planteado, existen en la bibliografía especializada diferentes métodos. En este caso se ha utilizado el método híbrido de Park [157] que comprende los métodos de partición de coordenadas independientes [187] y el método de estabilización de restricciones [33].

### 3.4.2.1. Software desarrollado

Para solucionar el problema de la dinámica directa se ha optado por hacer uso de la potencia de cálculo de Simulink, permitiendo un cómodo cambio entre las diferentes opciones para la integración del modelo dinámico. Se ha utilizado el integrador ode45 de Runge-Kutta (figura 3.30). Como ayuda para

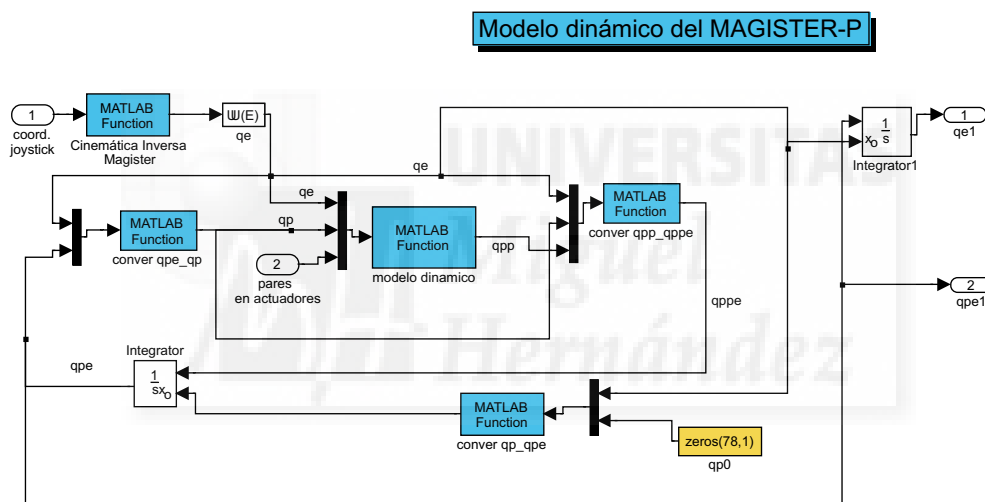


Figura 3.30: Software de solución de la Dinámica Directa.

la comprensión del código anexo en el cd adjunto, se muestra en el esquema de la figura 3.31 el funcionamiento del mismo.

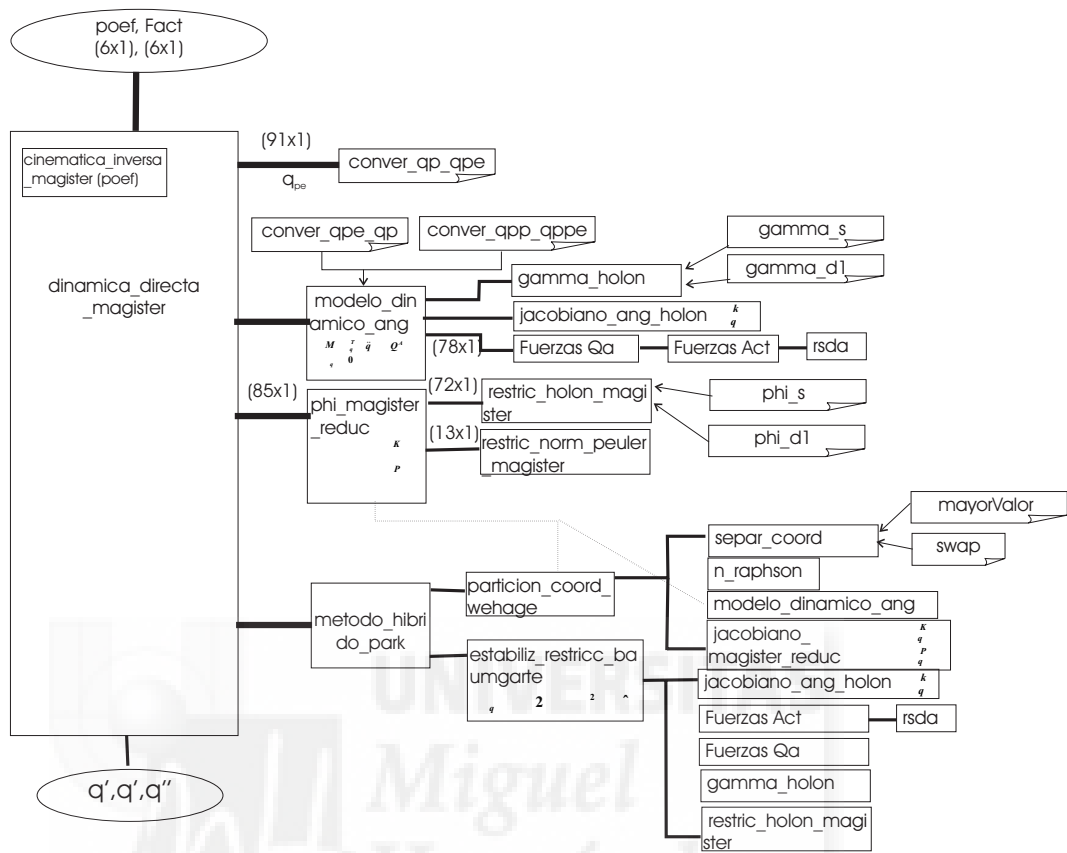


Figura 3.31: Esquema del algoritmo de solución de la dinámica directa.

### 3.5. Conclusiones

En el presente capítulo se ha desarrollado un maestro general para interfaces de teleoperación capaz de trabajar tanto como display de impedancias como de admitancias. Además de han generado todos los algoritmos necesarios para poder implementar este dispositivo dentro de un esquema de teleoperación en el cual se imponga cualquier esquema de control en la interconexión de los sistemas.

El rango de utilización del MAGISTER-P es desde un sistema abierto (display de impedancias) en control de par y cuyas señales están disponibles desde una cómoda interfaz hasta un sistema con un lazo de control en velocidad regulado mediante la técnica del par computado.

La utilización de robots paralelos como dispositivos maestros permite una utilización muy intuitiva incluso para el usuario inexperimentado. El sencillo diseño mecánico de los robots paralelos, y sus especiales características hacen de éste diseño una interfaz robusta y precisa, adecuada para ser utilizada tanto en teleoperación real como en experimentación con tecnología haptics.





# CAPÍTULO 4

## SIMULADOR DINÁMICO DE ARQUITECTURAS DE TELEOPERACIÓN

---

*“Puedes llegar a cualquier parte, siempre que andes lo suficiente”*  
Alicia en el país de las  
Maravillas. Lewis Carroll

### 4.1. Introducción

Tal y como ha quedado patente en los capítulos previos, la experimentación real de plataformas de teleoperación de  $n$  grados de libertad es un problema complejo. Los simuladores de robots dinámicos proporcionan una importante ayuda en dicho reto. Por un lado, permiten simplificar los problemas de comunicaciones entre la zona local y remota, otorgando además la posibilidad de modelar estas comunicaciones y realizar estudios de validez de estos modelos. Por otra parte, su utilización como visualizadores predictivos o la posibilidad de modificar el punto de vista del operador justifica por sí solo su desarrollo.

Existen varias librerías de simulación dinámica que permiten implementar mundos virtuales con capacidades dinámicas; en cambio, pocas de ellas están

pensadas para incluir el modelo dinámico completo de estructuras articuladas. Sin embargo, disponer del modelo dinámico completo del manipulador esclavo, unido a un “aceptable” conocimiento de los parámetros dinámicos reales del entorno, permite generar una plataforma de experimentación virtual con grandes posibilidades, como la de poder insertar campos de fuerza que actúen sobre eslabones intermedios del dispositivo esclavo, o simular eslabones situados sobre bases móviles.

#### **4.1.1. Estructura del capítulo**

El contenido de este capítulo versa sobre las ideas básicas que han permitido el desarrollo de un simulador de tareas de teleoperación que incluya el modelo dinámico completo del entorno.

En primer lugar se realiza un breve resumen del amplísimo mundo de la simulación computacional, y se citan algunas de las múltiples posibles formulaciones dinámicas. Posteriormente se detalla la arquitectura general del software desarrollado, para pasar a detallar algunos de los aspectos de mayor importancia para la realización de experimentos con ayuda de dicho simulador.

Finalmente se desarrollan algoritmos cinemáticos de indexación y mapeado de espacios de trabajo que permiten ampliar las posibilidades de la plataforma experimental.

### **4.2. Dinámica computacional**

La continua evolución de las capacidades gráficas y de cómputo, los desarrollos en métodos numéricos, así como la algorítmica de las aproximaciones a los métodos de análisis clásicos de la mecánica teórica, y los avances de las teorías de control, han permitido superar ampliamente los métodos gráficos tradicionales, aplicados al análisis y diseño de servomecanismos.

En la práctica la aplicación para control en tiempo real de elaborados métodos de simulación dinámica, depende de los avances en la potencia de procesamiento de los ordenadores usados. Durante las décadas pasadas diversos investigadores han desarrollado los formalismos de la dinámica de los sistemas multicuerpos [173],[148],[168]. Las mayores aportaciones de estas publicaciones se enfocaron en los siguientes aspectos: 1) formulación eficiente, 2) precisión, 3) relación física de varias teorías de la mecánica, y 4)

selección de sistemas de coordenadas y métodos de reducción del orden de los modelos. En contraste con la sólida y evolucionada mecánica analítica desarrollada hace unos 160 años por portentos de la mecánica clásica como Euler, Lagrange y Hamilton, se podría inferir que en muchos aspectos la dinámica computacional de los sistemas multicuerpos es una área relativamente joven.

Filosóficamente este estilo de lograr sintetizar las leyes de la mecánica, continua siendo el reto de los investigadores que exploran el mundo de la dinámica de los sistemas multicuerpos. Trabajando desde el principio de D'Alembert, Kane desarrollo las nociones de velocidad generalizada, velocidades parciales, y velocidades angulares parciales, estas ideas reducen la labor envuelta en la formulación de las ecuaciones del movimiento, y hacen posible simplificar las ecuaciones finales. Generando una notable base del conocimiento sobre el modelo del sistema, dado por las velocidades generalizadas.

Es conocido que la eficiencia de un formalismo depende de como las restricciones han sido tratadas, por lo que las coordenadas usadas para describir la configuración de un sistema multicuerpos tiene una significativa influencia en el modelado. Los sistemas comúnmente usados, son los de coordenadas generalizadas [187], coordenadas cartesianas, parámetros de Euler [148], [89], y las coordenadas naturales [110].

Las ecuaciones que gobiernan un sistema dinámico de múltiples cuerpos con restricciones consisten de una serie de ecuaciones diferenciales-algebraicas con restricciones. Las ecuaciones de restricción dependen de la topología del sistema, y debido a la estructura multivariable de estas ecuaciones, el análisis de los sistemas multicuerpos es complejo haciendo imprescindible los métodos computacionales. Un método computacional reduce la posibilidad del error de calculo humano, más aun cuando es necesario incluir en los modelos aspectos de la elasticidad de los sistemas mecánicos. En el caso del problema dinámico directo, las numerosas coordenadas dependientes que se crean en un modelado se deben reducir a coordenadas independientes, mediante la aplicación de métodos matemáticos y algoritmos.

El análisis de los sistemas mecánicos multicuerpos puede ser dividido en tres categorías. En la primera categoría se da lugar a un gran numero de ecuaciones mixtas diferenciales-algebraicas (DAE), que a su vez dan lugar a matrices dispersas. Las coordenadas y multiplicadores de Lagrange son tratados como incógnitas y solucionadas simultáneamente [153]. En la segunda categoría, el numero de ecuaciones diferenciales es minimizado para obtener un numero reducido de las ecuaciones que deben ser integradas [187].

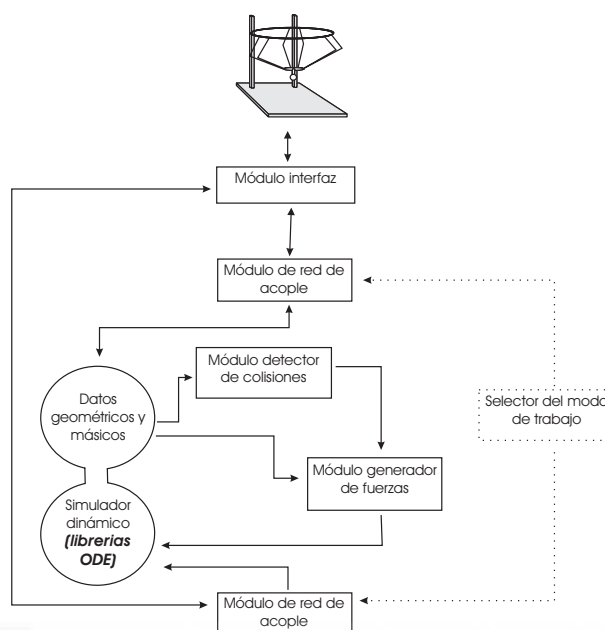


Figura 4.1: Arquitectura del Simulador.

La tercera categoría ha sido propuesta por Serna [175], que introdujo un funcional con una energía ficticia y un factor de penalización como parte de un Lagrangiano. El nuevo funcional es usado para modificar las ecuaciones de Lagrange a través del principio de Hamilton.

### 4.3. Arquitectura general del simulador

En este apartado se muestra una visión general de la arquitectura del simulador desarrollado para investigación en teleoperación. La figura 4.1 muestra el esquema de los diferentes módulos implementados en dicho simulador.

#### *Entorno de desarrollo y librerías gráficas utilizadas*

La herramienta de desarrollo utilizada ha sido Visual C++ y las librerías gráficas están basadas en OpenGL. La interfaz de usuario se ha programado con las librerías Qt, de forma que se asegure la portabilidad entre diferentes sistemas operativos. De hecho, el simulador ha sido probado indistintamente bajo Linux Suse 8 y bajo Windows 2000, si bien la interfaz con el Magister-P



sólo funciona bajo Windows debido a la necesidad de utilizar las librerías proporcionadas por dSpace.

Gracias a la utilización de OpenGL, se dispone de un objeto “camera” que proporciona la posibilidad de modificar el punto de vista del operador utilizando los movimientos del ratón, y lo que es más importante, sirve de sistema de referencia para los algoritmos de mapeado e indexación de los espacios de trabajo entre el dispositivo maestro y el esclavo.

#### *Información geométrica y base de datos másica*

El primer paso para construir un simulador es diseñar las estructuras para almacenar los parámetros geométricos e inerciales de los cuerpos que aparecen en el mundo generado. Un cuerpo rígido tiene varias propiedades que deben ser continuamente contabilizadas. Algunas de ellas varían con el tiempo:

- Vector de posición  $(x, y, z)$  del sistema de referencia ligado al cuerpo
- Velocidad lineal,  $(v_x, v_y, v_z)$
- Orientación del cuerpo, representada mediante cuaternios  $(q_0, q_1, q_2, q_3)$  o mediante matrices de rotación
- Vector de la velocidad angular

Y otras que se consideran invariantes:

- Masa del cuerpo
- Centro de masas expresado en el sistema de referencia ligado al cuerpo
- Matriz de inercias expresada en el sistema de referencia ligado al cuerpo

#### *Módulo Interfaz*

Se ha diseñado una clase genérica (dhInterface) cuya derivación permite conectar cualquier dispositivo mecánico al simulador. Durante el desarrollo de la Tesis se utilizaron tanto el PHANTOM 1.5 Premium Prototype disponible en el departamento de Ingeniería de Sistemas Industriales de la Universidad Miguel Hernández, como el dispositivo Magister-P diseñado en el capítulo 3.

En el caso del PHANTOM se utilizan las librerías GHOST en C++ proporcionadas por Sensable para derivar la clase dhPhantom, que contiene la funcionalidad del dispositivo como display de impedancias.

En el caso del Magister-P se utilizan las librerías MBlib desarrolladas a partir de los algoritmos del capítulo 3 para derivar la clase dhMagister, que contiene la funcionalidad del dispositivo tanto funcionando como display de impedancias o como display de admitancias.

#### *Módulo de red de acople*

El módulo de red de acople es el módulo en el que se implementa el esquema de control bilateral correspondiente, y por tanto es donde se encuentran los reguladores calculados (o sintonizados) que garanticen la estabilidad del sistema.

Además, es el responsable de decidir qué datos son enviados al esclavo real y qué datos son realimentados al maestro. Por tanto tiene una funcionalidad extra cuando se utiliza el simulador como ayuda a un esquema de teleoperación real. En este sentido el módulo de red de acople realiza las mismas funciones que el *Administrator module* en [94].

En este módulo se implementan también los algoritmos de indexación y mapeado de espacios de trabajo [159].

#### *Módulo de colisiones*

El módulo de colisiones tiene como función detectar los pares de cuerpos que posiblemente colisionan. La información de entrada de este módulo es el modelo geométrico y la posición actual de los objetos que forman parte de la escena. El algoritmo de colisiones utilizado está basado en una representación de las AABB (axis aligned bounding boxes) de los 3D-poliedros de la escena [143]. En este algoritmo, los objetos están rodeados de cajas cuyos ejes permanecen alineados a los ejes del sistema de referencia inercial de la escena, y se contabilizan las superposiciones de dichas cajas. La figura 4.2 muestra la construcción de un árbol jerárquico binario de un objeto 2D.

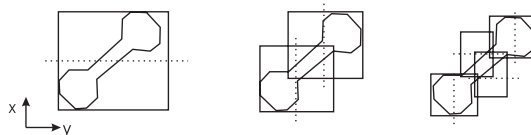


Figura 4.2: Ejemplo de AABB en un objeto 2D.

Este módulo almacena la información en estructuras que contienen la posición de contacto, el vector normal y la profundidad de penetración de cada par de geometrías que colisionan.

#### *Módulo generador de fuerzas de contacto*

A partir de la información de los puntos de contacto proporcionada por el módulo de colisiones y de los parámetros inerciales y másicos de los objetos, el módulo generador de fuerzas calcula las fuerzas de reacción a la interacción de los elementos de la escena. Para realizar el cálculo de estas fuerzas, se soluciona el problema lineal (LCP - linear Complementary Problem) que se obtiene a partir de la configuración de contactos del sistema [143]. La relación lineal entre la aceleración relativa  $\vec{a}_i$  entre dos cuerpos en contacto y la fuerza de reacción  $\vec{F}_i$  en el punto de contacto  $C_i$  se expresa por:

$$\vec{a}_i = \begin{bmatrix} a_{ii_n} & a_{i(i+1)_t} & a_{i(i+2)_k} \\ a_{(i+1)i_n} & a_{(i+1)(i+1)_t} & a_{(i+1)(i+2)_k} \\ a_{(i+2)i_n} & a_{(i+2)(i+1)_t} & a_{(i+2)(i+2)_k} \end{bmatrix} \begin{bmatrix} F_{i_n} \\ F_{i_t} \\ F_{i_k} \end{bmatrix} + \begin{bmatrix} b_{i_n} \\ b_{i_t} \\ b_{i_k} \end{bmatrix} = \mathbf{A}_i \vec{F}_i + \vec{b}_i \quad (4.1)$$

donde el subíndice  $n$  indica la componente a lo largo de la dirección normal al contacto, y los subíndices  $t$  y  $k$  son las componentes contenidas en el plano tangente al contacto. Los coeficientes de la matriz  $\mathbf{A}_i$  y el vector  $\vec{b}_i$  se calculan a partir de los parámetros dinámicos y los desplazamientos relativos de los objetos en el punto de contacto  $C_i$ . Para resolver la ecuación 4.1 se utiliza el algoritmo de Dantzig, según la implementación de [143].

#### *Motor Dinámico*

Una vez calculadas las fuerzas de contacto (y pares) que actúan sobre cada uno de los objetos que conforman la escena, el motor dinámico es el encargado de resolver las ecuaciones de movimiento y de actualizar los datos geométricos y dinámicos de los mismos. En esta tesis se ha utilizado una librería de libre distribución, denominada ODE [150] como motor dinámico. Sobre esta librería se han implementado las clases necesarias para la generación de escenas que contengan eslabones articulados. La figura 4.3 muestra un esquema de la jerarquía de clases desarrollada.

Por tanto, la construcción de robots esclavos se hace de forma sencilla, sin más que añadir los parámetros geométricos y dinámicos, y definir las

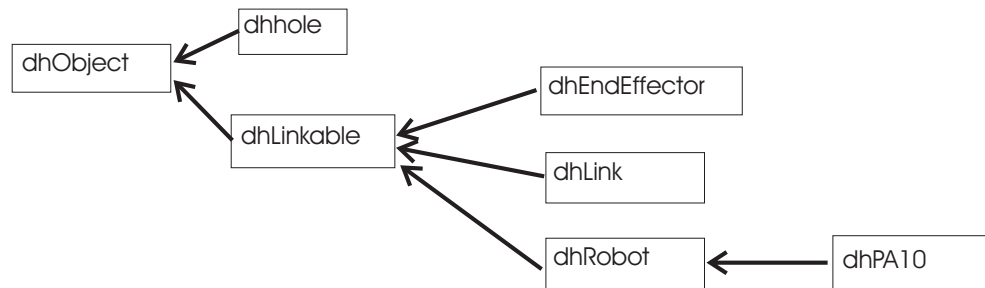


Figura 4.3: Jerarquía de clases desarrolladas sobre la librería ODE.

relaciones entre los diferentes eslabones en el constructor del objeto.

## 4.4. Interfaz del simulador

Se muestra en este apartado los conceptos básicos para la utilización del simulador desarrollado en este tesis. Un aspecto muy importante dentro de la experimentación en teleoperación lo constituyen las interfaces de operador diseñadas [80]. Obviando el diseño de la interfaz mecánica, (ya tratada en el capítulo 3), se debe prestar especial atención al resto de aspectos, como son el problema tecnológico de conectar dicho dispositivo mecánico a la propia plataforma de experimentación y la interfaz de visualización del entorno remoto.

### 4.4.1. Interfaz genérica para conexión del maestro

Por interfaz genérica para la conexión del maestro se entiende el módulo interfaz comentado anteriormente. Los diferentes dispositivos maestros deben poder conectarse a la plataforma de experimentación, y para ello se deben proporcionar los métodos necesarios. No se incluyen en este apartado los problemas de estabilidad del dispositivo háptico, que se tratarán más adelante.

Con objeto de disponer de una plataforma en la que se puedan conectar cualquier dispositivo mecánico, se implementa una interfaz genérica a partir de la cual derivaran las diferentes clases de los interfaces hápticos utilizados (figura 4.4).

La interfaz genérica desarrollada en el simulador se fundamenta en la clase **dhInterface** cuya declaración se muestra a continuación.

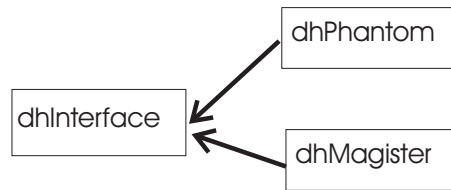


Figura 4.4: Jerarquía de las clases de la interfaz maestra.

```

class dhInterface {

public:
    dhInterface(dhVector<dhRobot*> r) : robots(r){};
    virtual void enable() = 0;
    virtual void disable() = 0;
    virtual void setForce(const dVector3 force, const dVector3 torque) =0;
    void setScalePos(const dReal s){scalePos=s;};
    void setScaleForce(const dReal s){scaleForce=s;};
    void setScaleTorque(const dReal s){scaleTorque=s;};

protected:
    const dhVector<dhRobot*> robots;
    dReal scalePos, scaleForce, scaleTorque;

};
  
```

#### 4.4.1.1. PHANToM

La conexión del PHANToM al simulador se implementa utilizando las librerías GHOST proporcionadas por Sensable. El PHANToM es un dispositivo serie utilizado como maestro de impedancias.

```

class dhPhantom : public dhInterface {
public:
    dhPhantom(dhVector<dhRobot*> r);
    ~dhPhantom();
    int getTransform();
    virtual void setForce(const dVector3 force, const dVector3 torque);
    virtual void enable();
    virtual void disable();
  
```

```
protected:
    gstPHANToM *phantom;
    gstScene *scene;
    gstSeparator *root;
    gstTorqueEffect *effect;
    gstTransformMatrix matrix;

    dVector3      position;
    dMatrix3      rotation;

    gstVector force, torque;

    int started;
};
```

Las funciones miembro `getTransform()` y `setForce()` permiten trabajar con el PHANToM como display de impedancias.

#### 4.4.1.2. MAGISTER-P

El Magister-P diseñado en el capítulo 3 es un dispositivo háptico de arquitectura paralela capaz de funcionar como display de impedancias y como display de admitancias. Las librerías `Clib` proporcionadas por `dSpace` y las librerías `MBlib` desarrolladas permiten la implementación de la clase `dhMagisterP` para la conexión de dicho dispositivo al simulador desarrollado.

```
class dhMagisterP : public dhInterface {
public:
    dhMagisterP(dhVector<dhRobot*> r);
    virtual ~dhMagisterP();
    int getTransform();
    virtual void setTransform(const dVector3 position,
                             const dVector3 rotation);

    int getForces();
    virtual void setForce(const dVector3 force, const dVector3 torque);

    virtual void enable();
    virtual void disable();

protected:
```

```
int ds_error,state;
char* board_name;
unsigned int board_index;

long double qe[91];
long double entrada[97];
long double entrada_fuerza[12];
long double fart[6];
dVector3 position;
dMatrix3 rotation;

int started;
};
```

Debe resaltarse que para poder utilizar el Magister-P como interfaz maestra, la tarea de tiempo real que controla el dispositivo debe estar ejecutándose sobre la tarjeta dSpace, y los nombres y alias de las variables utilizadas sobre la tarjeta dSpace deben coincidir con los utilizados por la clase.

#### 4.4.2. Interfaz de visualización

La interfaz de visualización desarrollada es la mostrada en la figura 4.5, y ha sido implementada haciendo uso del objeto GLDrawer de las librerías Qt, sobre el cual pueden implementarse las librerías OpenGL. Esta decisión ha permitido disponer de objetos “camera” en la escena cuyos sistemas de referencia son de gran ayuda en el desarrollo de algoritmos cinemáticos.

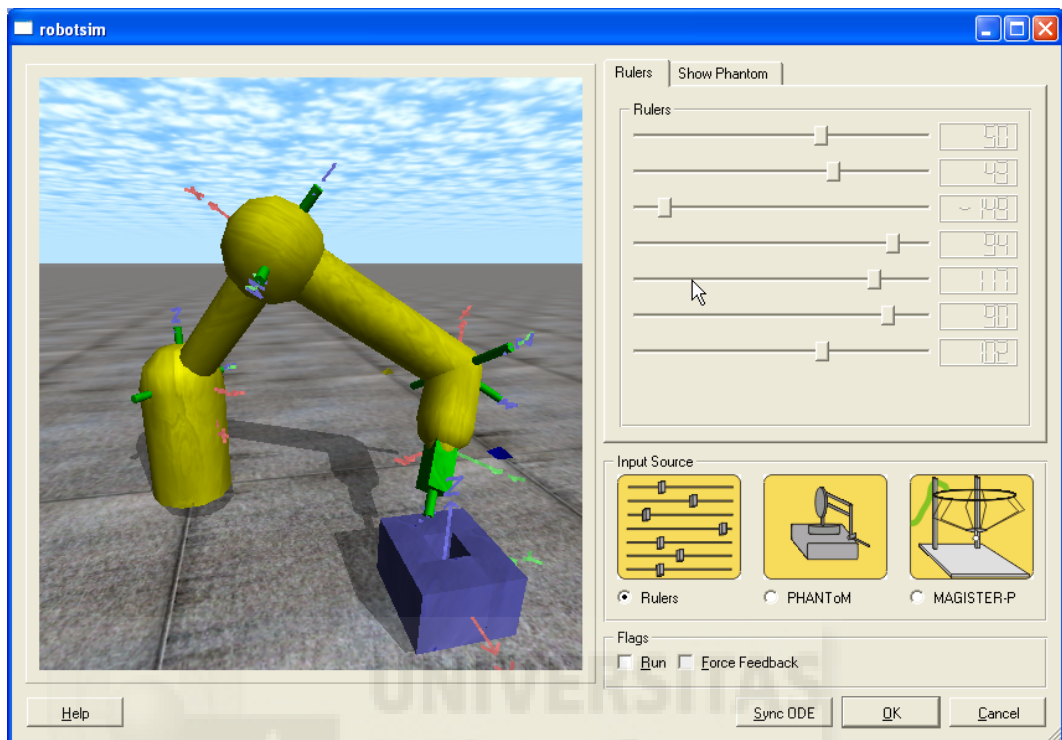


Figura 4.5: Interfaz de visualización del simulador desarrollado.

## 4.5. Red de acople: esquema de control

Se ha desarrollado una interfaz genérica que permita una cómoda implementación de los diferentes esquemas de control bilateral. A partir de la clase `dhController` mostrada se derivan los diferentes esquemas de control y redes de acople que garanticen la estabilidad del sistema.

```
class dhController {
public:
    enum dhControllerType
    {
        dhForces,
        dhVelocities
    };
    dhController(const int size);
};
```



```

    virtual dhControllerType control(const dReal *reference,
        const dReal *pose, const dReal *velocity, dReal *result) = 0;

protected:
    const int size;
    dReal time;
};

```

## 4.6. Simulación de escenas

Con el fin de facilitar la rápida implementación de diferentes mundos virtuales se ha desarrollado una clase `dhScene` que proporciona la funcionalidad básica de los mundos virtuales, de forma que cada experimento se construya en el constructor de la clase derivada de `dhScene`.

```

class dhScene {
public:
    dhScene();
    ~dhScene();
    virtual void step(const dReal dt);
    virtual void setGravity(const dReal x, const dReal y, const dReal z);
    virtual void addObject (dhObject* obj);
    virtual void draw();
    virtual void syncODE();
    static void collisionCallback (void *pData, dGeomID o1, dGeomID o2);
    virtual void collide (dGeomID o1, dGeomID o2);
protected:

    dReal time;
    dhVector<dhObject*> objects;
    dWorldID ODEWorld;
    dSpaceID ODESpace;
    dJointGroupID ODEJointGroup;
    dVector3 gravity;
};

```

La función `step()` se ejecuta en cada paso de integración y se invoca por los Qt-timers definidos en la aplicación.



Figura 4.6: Jerarquía de las clases de la escena.

## 4.7. Algoritmos cinemáticos de mapeado de espacios de trabajo

Hirzinger [159] estudia problemas de escalado entre los espacios de trabajo del maestro y el esclavo. También estudia la problemática de cinemáticas diferentes, solucionándolo con un indexado y mapeado en el espacio cartesiano de los espacios de trabajo de maestro y esclavo.

Sayers [171] establece una correspondencia cinemática basada en el sistema de referencia del punto de vista utilizado en la interfaz visual (el objeto “*camera*”) de forma que diseña un algoritmo de reindexación y mapeado de los espacios de trabajo de los diferentes dispositivos.

Para realizar un mapeado de los espacios de trabajo de diferentes dispositivos, lo lógico es trabajar con coordenadas cartesianas de los efectores finales, independizando a los dispositivos de sus propias configuraciones cinemáticas. La implementación de una correspondencia cinemática implica que los comandos de movimiento del operador (y por supuesto, en el ámbito de un estudio dinámico, las realimentaciones de esfuerzo reflejadas) sean funciones del punto de vista del operador en la interfaz gráfica de usuario. Es decir, si el operador mueve el dispositivo maestro hacia su izquierda, el dispositivo esclavo visualizado en la interfaz de usuario debe moverse hacia la izquierda, independientemente de la posición relativa del esclavo real respecto al operador. Esto hace que la interfaz de usuario aparezca de forma natural para el operador (figura 4.7).

Paralelo al problema de mapeado de espacios, aparece el problema de reindexado y escalado de los espacios de trabajo. El operador debe ser capaz de controlar un robot esclavo, que posiblemente tenga un gran espacio de trabajo, utilizando para ello un dispositivo maestro, que posiblemente tenga un espacio de trabajo relativamente pequeño. Además, en ocasiones se requiere realizar operaciones precisas en el entorno remoto sin que para ello sea necesario disponer de un maestro excesivamente preciso. O en el caso opuesto, se requiere realizar grandes desplazamientos en el robot esclavo con movimientos pequeños del dispositivo maestro. Una correcta combinación de

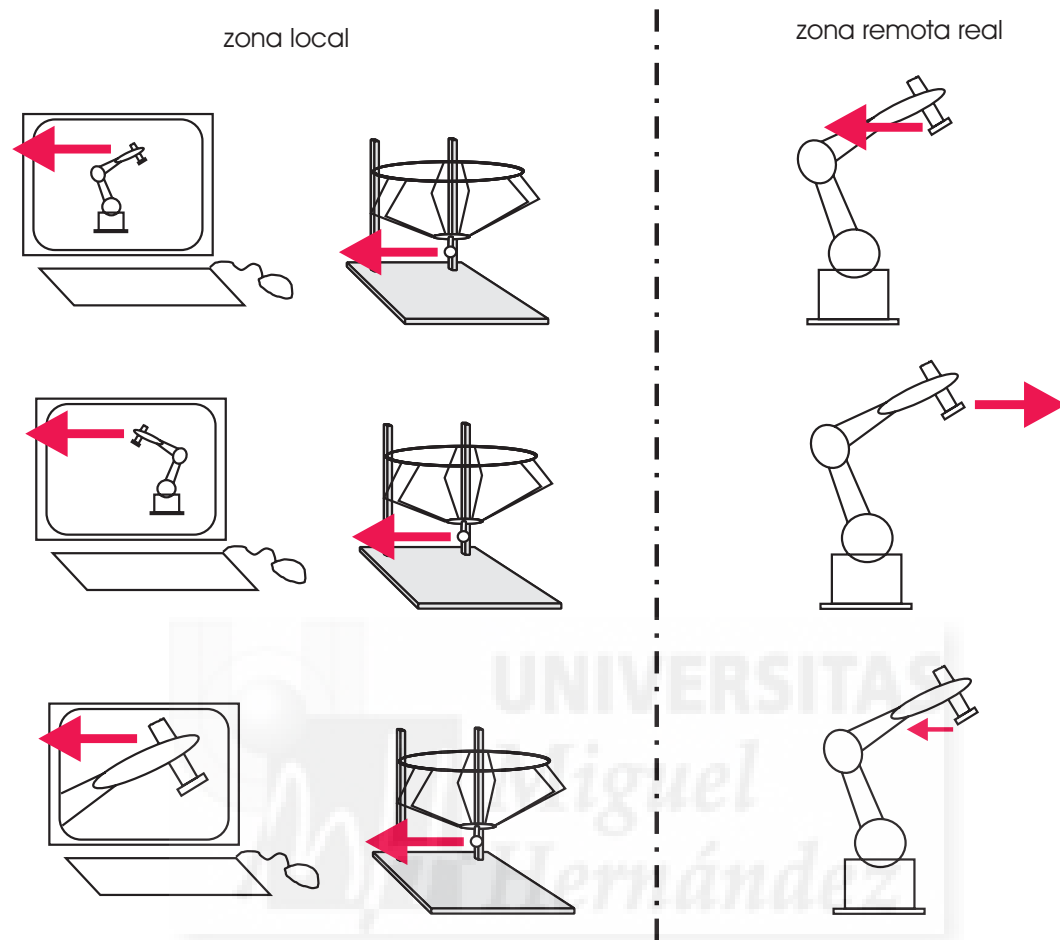


Figura 4.7: Mapeado e indexación de espacios de trabajo.

operaciones de reindexado y escalado de espacios de trabajo permite aportar soluciones a los problemas planteados anteriormente.

Para establecer las diferentes relaciones cinemáticas se establecen las siguientes definiciones:

Tabla 4.1: Variables de los algoritmos de indexación y mapeado

$R^m$	Es la matriz de rotación del efector final del maestro respecto del sistema de referencia establecido en la base del maestro. El subíndice $i$ indica la configuración inicial (tras la reindexación), mientras que el subíndice $n$ representa la configuración actual.
$p^m$	Es el vector de posición del efector final del maestro relativo al sistema de su base. Los subíndices tienen el mismo significado que anteriormente.
$R^s$	Es la matriz de rotación del efector final el esclavo respecto del sistema de referencia establecido en la base del esclavo. Los subíndices tienen el mismo significado que anteriormente.
$p^s$	Es el vector de posición del efector final del esclavo relativo al sistema de su base. Los subíndices tienen el mismo significado que anteriormente.
$s^{ms}$	Es el factor de escala de las coordenadas del maestro al esclavo
$p^{ms}$	Representa el vector de traslación de las coordenadas del maestro a las coordenadas del esclavo.
$R^{ms}$	Representa la matriz de rotación de las coordenadas del maestro a las coordenadas del esclavo.
$d$	Representa la distancia aparente del operador sobre la base del robot en la interfaz de usuario. En una interfaz opengl representa la distancia medida en el mundo simulado entre el sistema de la base del esclavo y el foco de la cámara utilizada para visualizar.
$k$	Representa el factor de escala constante entre el sistema maestro y el sistema esclavo. Este factor se elige de forma que cuando el espacio de trabajo del robot esclavo está completamente visible en la interfaz, el esclavo real puede moverse en todo su espacio sin necesidad de reindexación del espacio del maestro.

De acuerdo a las definiciones de la tabla 4.1, se consideran los siguientes casos:

CASO 1: Se empieza considerando el caso más simple, donde sólo se permiten movimientos de traslación relativa de los efectores finales. En este caso, los ejes de la base del maestro y los ejes

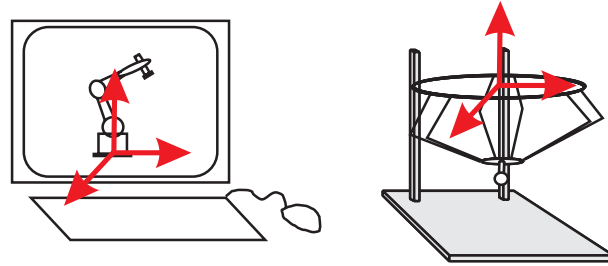


Figura 4.8: Casos 1 y 2 de los algoritmos de mapeado e indexación.

de la base de la representación gráfica del esclavo son paralelos (figura 4.8).

$$\begin{aligned}
 s^{ms} &= d \cdot k \\
 R^{ms} &= I \\
 p^{ms} &= p_i^s - p_i^m \cdot s^{ms} \\
 R_n^s &= R_i^s \\
 p_n^s &= p^{ms} + p_n^m \cdot s^{ms}
 \end{aligned} \tag{4.2}$$

CASO 2: Se permiten tanto los movimientos de traslación como los de rotación. En este caso, los ejes de la base del maestro y los ejes de la base de la representación gráfica del esclavo son paralelos (figura 4.8).

$$\begin{aligned}
 s^{ms} &= d \cdot k \\
 R^{ms} &= (R_i^m)^{-1} R_i^s \\
 p^{ms} &= p_i^s - p_i^m \cdot s^{ms} \\
 R_n^s &= R_n^m R^{ms} \\
 p_n^s &= p^{ms} + p_n^m \cdot s^{ms}
 \end{aligned} \tag{4.3}$$

CASO 3: Se permiten tanto los movimientos de traslación como los de rotación. En este caso, los ejes de la base del maestro y los ejes de la base de la representación gráfica del esclavo tienen una orientación relativa arbitraria. Se denomina C a la matriz de rotación que define esta transformación (figura 4.9).

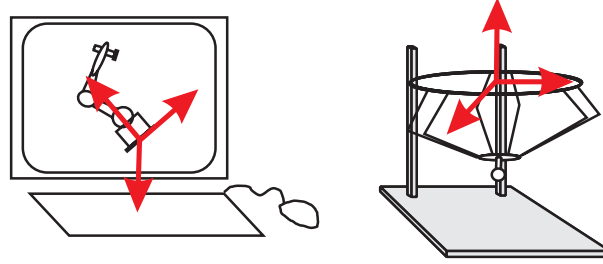


Figura 4.9: Caso 3 de los algoritmos de mapeado e indexación.

$$\begin{aligned}
 s^{ms} &= d \cdot k \\
 R^{ms} &= (R_i^m)^{-1} \cdot (C \cdot R_i^s) \\
 p^{ms} &= (C \cdot p_i^s) - p_i^m \cdot s^{ms} \\
 R_n^{s'} &= R_n^m R^{ms} \\
 p_n^{s'} &= p_n^{ms} + p_n^m \cdot s^{ms} \\
 R_n^s &= C^{-1} \cdot R_n^{s'} \\
 p_n^s &= C^{-1} \cdot p_n^{s'}
 \end{aligned} \tag{4.4}$$

## 4.8. Conclusiones

El desarrollo de un simulador robótico para tareas de teleoperación proporciona importantes ventajas en la experimentación de diferentes esquemas de control bilateral. Se ha desarrollado un esquema general de simulador virtual, sobre el cual se puedan ir ampliando las funcionalidades del mismo gracias a la programación orientada a objetos. La arquitectura implementada permite conectar fácilmente cualquier dispositivo maestro del cual se dispongan de librerías para acceder al mismo, así como modelar de forma rápida diferentes esclavos y escenarios. También cabe destacar que entre las posibilidades del simulador desarrollado se encuentra el implementar varios manipuladores esclavos en la misma escena, e incluso conectar varios dispositivos maestros, para realizar experimentos de control cooperativo.

# CAPÍTULO 5

## EXPERIMENTACIÓN CON SISTEMAS HÁPTICOS DE $N$ GRADOS DE LIBERTAD

---

*“En teoría, no existe diferencia entre teoría y práctica;  
en la práctica si la hay”*

Jan L.A. Van der Snepschent

### 5.1. Introducción

Después de obtener los elementos necesarios para la experimentación de sistemas hápticos y teleoperados, se plantea realizar la interconexión de dichos elementos de forma que se pueda implantar entre ellos cualquier esquema bilateral háptico o de teleoperación.

El primer aspecto importante que debe reseñarse es la propia naturaleza de la interconexión establecida. Los desarrollos teóricos de esquemas de control bilateral [158],[21] se realizan bajo el supuesto de que el entorno local y el entorno remoto son sistemas en lazo abierto y cuyas señales de control son

señales de voltaje, intensidad, pares o momentos. Sin embargo, en los artículos que muestran experimentación con sistemas reales comerciales, el entorno local y el entorno remoto son sistemas en lazo cerrado, estables, dotados de una cierta cantidad de autonomía y cuyas señales de control pueden ser comandos de movimiento de alto nivel. Esta dificultad para la experimentación con sistemas comerciales de los esquemas de control bilateral teóricos radica en la necesidad de conectar dos sistemas en tiempo real (entorno local y entorno remoto) entre sí mediante comunicaciones que se integren en una única tarea de tiempo real.

Este hecho revela otra de las características más importantes del simulador desarrollado en el capítulo 4. La inclusión del simulador en una tarea de tiempo real simplifica el problema tecnológico de las comunicaciones entre el entorno real remoto y el entorno local.

### 5.1.1. Estructura del capítulo

Se presenta en primer lugar la arquitectura hardware empleada en el sistema experimental y los experimentos llevados a cabo para la calibración del dispositivo háptico diseñado en el capítulo 3.

Posteriormente, se muestran los requisitos necesarios para poder conectar un dispositivo kinestésico a la plataforma desarrollada. Para ello se debe asegurar la estabilidad de los mismos. Se presenta un método de diseño de reguladores para sistemas de  $n$  grados de libertad. La aplicación de este método a la plataforma diseñada no es inmediata, debido a que no se dispone de un modelo algebraico de la dinámica del maestro, sino de un modelo numérico. Sin embargo, las conclusiones obtenidas de la aplicación de este método si son válidas como guía para la obtención de unos valores experimentales y su posterior validación. Además, el método es aplicable en configuraciones particulares del sistema, pudiendo obtener valores de los reguladores para ciertas configuraciones del mismo (se eligen las peores posiciones).

Finalmente se muestran algunos resultados obtenidos con el MAGISTER-P durante la realización de experimentos de manipulación háptica de objetos virtuales.



## 5.2. Implementación del dispositivo háptico; modos de trabajo

En este apartado se realiza una descripción de la arquitectura hardware utilizada en el MAGÍSTER-P. Utilizando el software dSpace ControlDesk y las librerías Rtlb, se programan los algoritmos anteriores, permitiendo así la implementación de los diferentes modos de control del dispositivo. La interfaz necesaria para conectar este dispositivo a la plataforma de teleoperación (realizado con las librerías Clib) se estudia más adelante, entendiéndose en este apartado el MAGÍSTER-P como un dispositivo individual desconectado de un esquema de teleoperación.

### 5.2.1. Arquitectura hardware

En la arquitectura hardware del MAGÍSTER-P se ha optado por utilizar una tarjeta multitejes programable que permita una cómoda implementación de los diferentes esquemas de control. La solución adoptada consiste en una tarjeta dSpace DS1103 PPC [72]. Entre las características de esta tarjeta se encuentran su interfaz con Matlab+Simulink a través de las librerías RTlib y Mlib, lo cual facilita la implementación de los algoritmos anteriores dentro de las tareas de tiempo real.

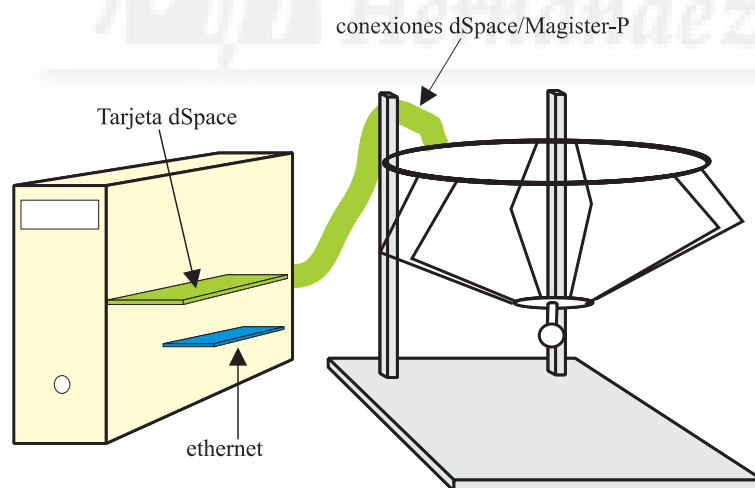


Figura 5.1: Arquitectura Hardware del Magíster-P

Esta arquitectura permite cerrar un lazo de control de tiempo real en la zona local de un esquema de teleoperación, pudiendo así programar esquemas que aseguren la estabilidad del maestro.

#### **5.2.1.1. Tarjeta dSpace DS1103, librerías RTlib y software ControlDesk**

En los diferentes manuales de usuario y programador de la tarjeta dSpace [72], librerías RTLib [73] y software ControlDesk [74] se encuentra información detallada sobre la utilización de estos productos. Aquí se presenta únicamente una breve descripción de estos productos y de las capacidades de los mismos que han sido utilizadas en el desarrollo de esta tesis.

El software ControlDesk permite la modificación de las variables del proceso de tiempo real. Este software se utiliza en los experimentos detallados en este capítulo.

Las librerías RTLib proporcionadas por el fabricante de la tarjeta permiten una programación sencilla de las tareas de tiempo real en la dSpace, haciendo uso de las capacidades de Matlab, Simulink y la Real Time Workshop de Mathworks. Sin embargo, una de las dificultades encontradas hace referencia a la implementación de los algoritmos desarrollados en lenguajes de programación estándar. RTLib soporta la gran mayoría de bloques Simulink, pero los bloques Matlab Fcn no están entre ellos. Por tanto, el código escrito en Matlab no es directamente compilable por el compilador para el procesador Motorola Power PC de la tarjeta dSpace.

La solución a esta dificultad se encuentra en la utilización de S-Functions desarrolladas en ANSI-C. Hay que anotar en este punto que no se puede utilizar el traductor de Matlab de fichero.m a fichero.c (mcc) puesto que éste hace uso de librerías propias de Matlab de las cuales no se dispone del código fuente, y si bien las S-Functions generadas de esta forma funcionan bien en simulaciones de Simulink, el compilador para el procesador PPC no es capaz de generar el fichero objeto.

#### **5.2.1.2. Librerías MBlib de dinámica multicuerpo**

Para solucionar el problema planteado, se han desarrollado en ANSI-C unas librerías que permitan la implementación de los algoritmos presentados en este capítulo en la tarjeta dSpace.

Para el desarrollo de las mismas se implementó en primer lugar unas librerías de tratamiento matricial que permitieran un rápido paso del código Matlab a código C. Se ha mantenido el nombre y sintaxis de las funciones Matlab, de forma que el código sea autoexplicativo. Para el cálculo de matrices inversas se utiliza el método de Crout de descomposición LU [149].

### 5.3. Calibración del dispositivo háptico

En primer lugar se plantean una serie de experimentos destinados a la calibración y comprobación de las características del dispositivo diseñado en el capítulo 3. Se obtienen gráficas experimentales que muestran el ancho de banda y el espacio de trabajo del Magister-P.

#### Experimentos con 1 solo brazo

Los experimentos 1x tenían como propósito comprobar el funcionamiento, calibrar el sistema y obtener valores iniciales de los reguladores en vacío. Las figuras 5.2 y 5.3 muestran los esquemas utilizados en los experimentos 1A y 1F.

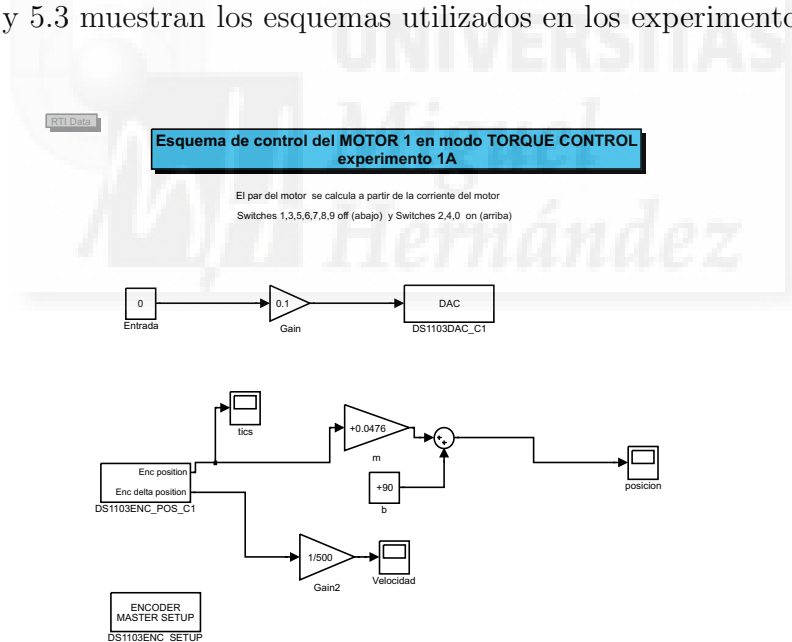


Figura 5.2: Esquema del experimento 1A

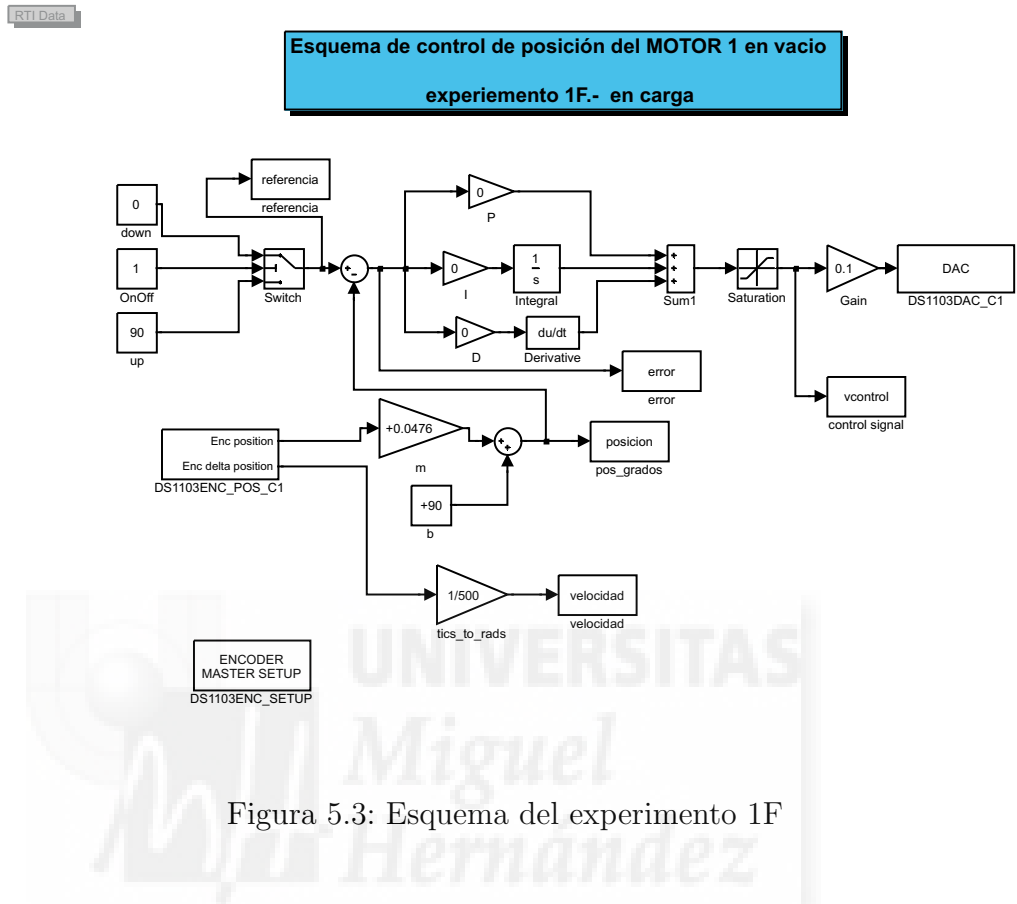


Figura 5.3: Esquema del experimento 1F

### Experimentos con el MAGISTER-P completo

La figura 5.4 muestran el esquema básico del Magíster-P utilizado para implementar los esquemas de display de impedancias y display de admitancias. En base a este esquema se realizan los experimentos de ancho de banda de estímulos, introduciendo en la entrada de fuerzas señales de frecuencia determinada. El tiempo de ciclo de este esquema en la tarjeta dSpace es de 0.000005 segundos. Este dato coincide con el expresado en el manual de la tarjeta.

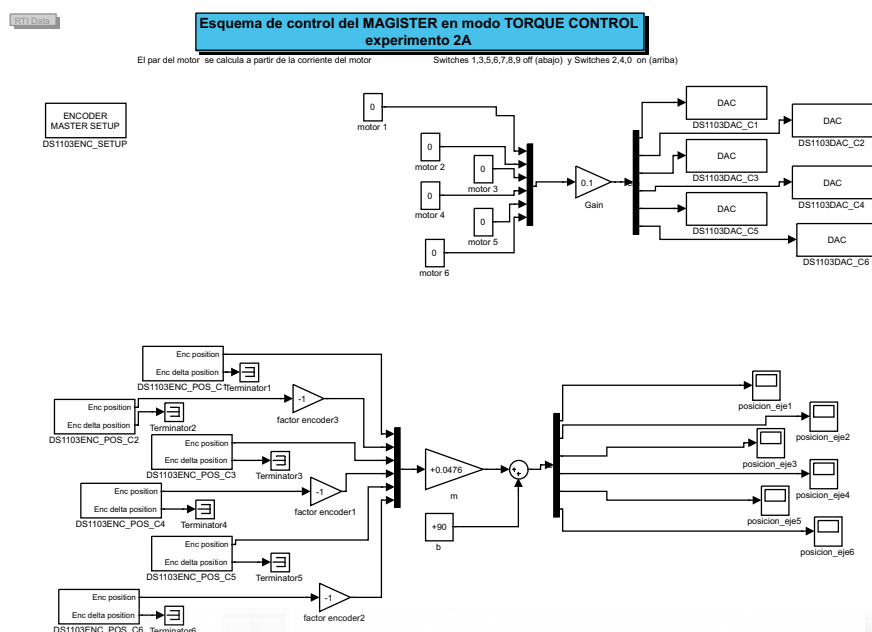


Figura 5.4: Esquema del experimento 2A

### 5.3.1. Como display de impedancias...

Para utilizar un dispositivo mecánico como display de impedancias se deben cumplir los siguientes requisitos: para leer la posición se requiere disponer de sensores de posición y del modelo cinemático directo que proporcione la posición y orientación del efector final en coordenadas del espacio cartesiano. Para reflejar un vector de fuerzas generalizadas se requiere disponer de la matriz jacobiana del mecanismo y de la posibilidad de comandar directamente pares a los actuadores en un esquema en lazo abierto.

El tiempo de ciclo de este esquema en la tarjeta dSpace es obviamente dependiente del cálculo iterativo de la función cinemática directa. En la implementación realizada, se utilizó un error de 0.001 para el algoritmo de Newton-Raphson. Este valor permite que durante una utilización del Magister-P con una velocidad habitual del operador, el tiempo de ciclo no excediera de 0.01 segundos. Uno de los trabajos futuros que se plantean es reescribir los algoritmos con una formulación más eficiente que permita rebajar este límite.

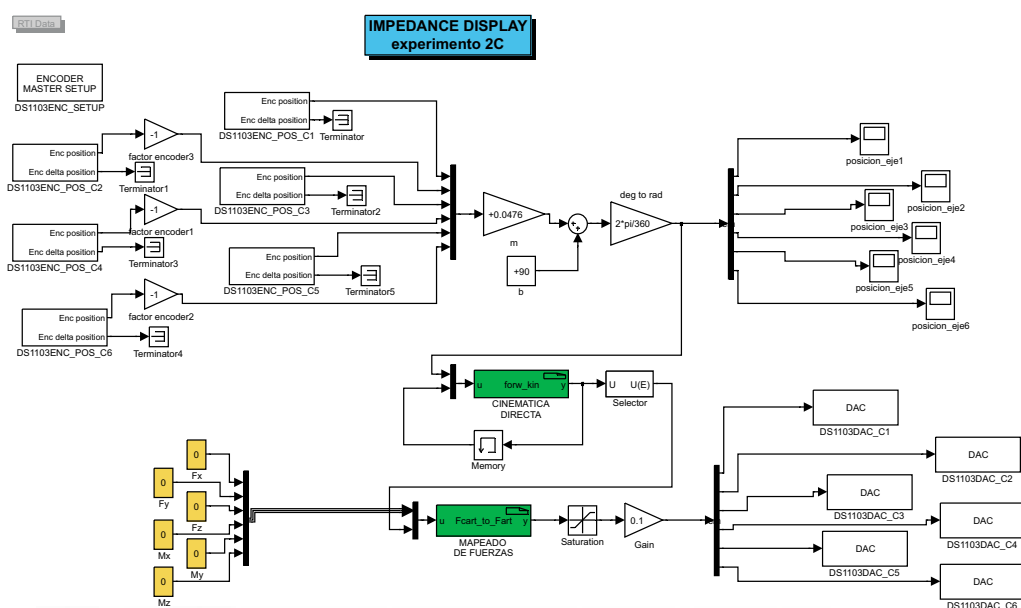


Figura 5.5: Esquema del experimento 2C. Display de Impedancias

### 5.3.2. Como display de admitancias...

Para utilizar un dispositivo mecánico como display de admitancias se deben cumplir los siguientes requisitos: para leer el vector de fuerzas generalizadas se requiere disponer de un sistema capaz de obtener las fuerzas. Típicamente estos sistemas pueden ser un bote sensor de fuerzas o utilizar la intensidad de los motores para obtener los pares de cada actuador y utilizando la jacobiana obtener el vector de fuerzas. Para reflejar la posición del entorno remoto se necesita disponer del modelo dinámico del mecanismo de forma que se puedan hallar los reguladores necesarios para establecer un control en posición. Además es obvio que la estructura mecánica debe ser suficientemente robusta para vencer las fuerzas ejercidas por el operador.

En la experimentación como display de admitancias se ha utilizado un regulador PD en cada eje. Es conocido que en teleoperación, los requerimientos de precisión son diferentes a la operación normal de un robot, debido a la aparición del operador en el lazo de control. Por esta razón, se puede prescindir del efecto integrador y de los efectos perjudiciales que el mismo

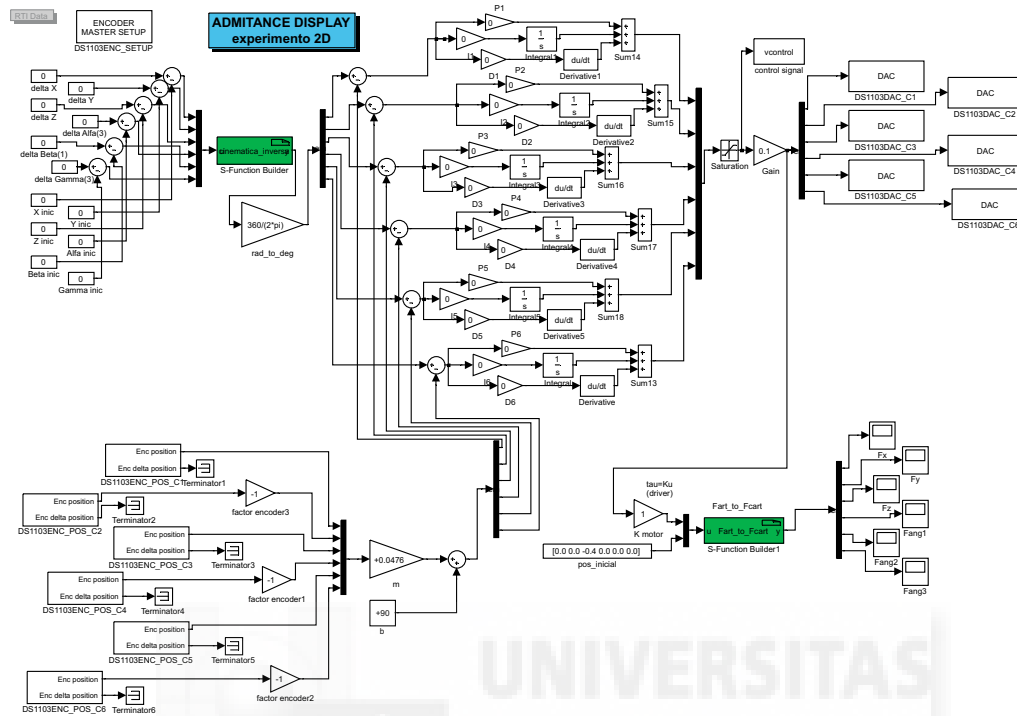


Figura 5.6: Esquema del experimento 2D. Display de Admitancias

tiene en el esquema anterior [30]. Por lo tanto, la implementación de sistemas de control en posición más costosos computacionalmente, como pudiera ser el par calculado [64], [30], no son justificables dentro de un esquema de teleoperación.

Los requerimientos temporales de este esquema son análogos al caso anterior (display de impedancias), debido a que para poder calcular la jacobiana es necesario resolver la cinemática directa. Sin embargo, se ha probado (figura 5.6) un método similar al método de la jacobiana para el cálculo de trayectorias [64] en el que la jacobiana se evalúa en un punto particular y permanece constante para el entorno de este punto. De esta forma, el cálculo iterativo desaparece, quedando el algoritmo geométrico de la cinemática inversa, reduciéndose el tiempo de ciclo a 0.00002 segundos aproximadamente.

### 5.3.3. Resultados experimentales

Los esquemas utilizados en el resto de los experimentos y los ficheros y resultados de los mismos se encuentran en el cd adjunto. Algunos de estos resultados se muestran en la figura 5.7

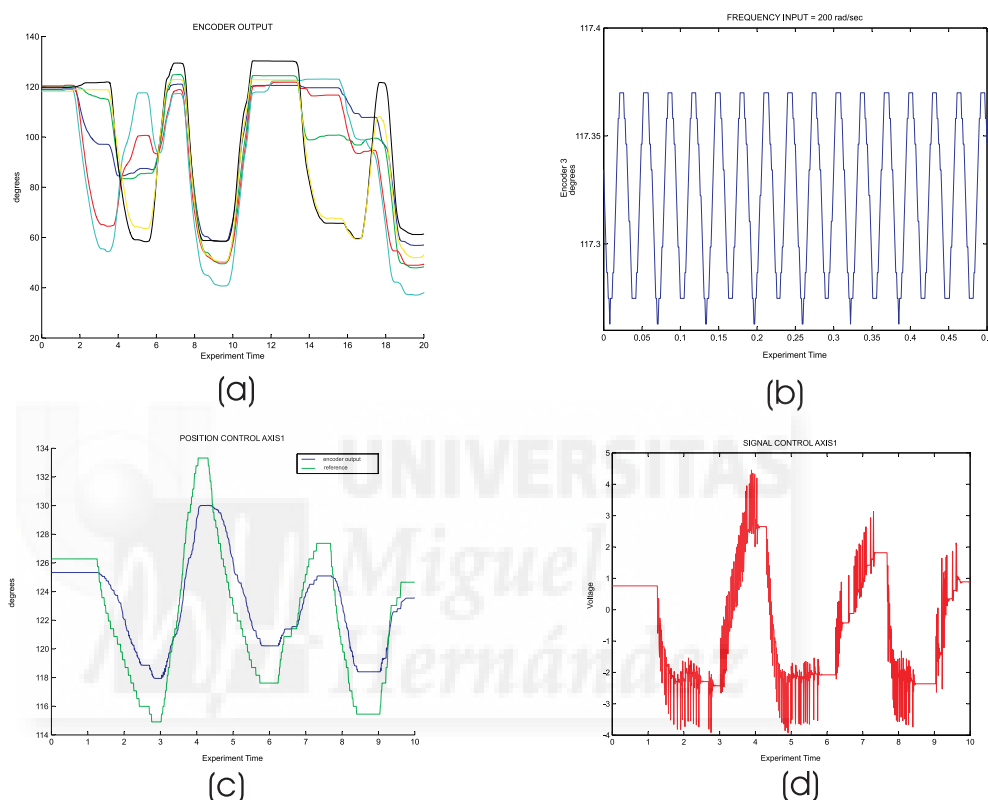


Figura 5.7: Experimentos con el dispositivo maestro.

La figura 5.7-(a) muestra la lectura en grados de los encoders entre los extremos del espacio de trabajo del joystick, siendo la misma  $135^\circ$  cuando el paralelogramo está completamente extendido y de  $40^\circ$  cuando está prácticamente recogido. A partir de este experimento y de otros similares se construye la figura 5.8 que muestra el espacio de trabajo del Magister-P a orientación constante. Como ya se vio en el capítulo 3, para realizar un estudio completo del espacio de trabajo, hay que realizar un análisis de las singularidades de fuerza que pueden aparecer, es decir, configuraciones en las cuales las fuerzas que actúan en alguna dirección en el efector final no pueden ser soportadas



por los pares generados en los actuadores, es decir, se vuelve incontrolable. Sin embargo, en esta plataforma particular, en el caso de trabajar como display de impedancias, las singularidades de fuerza no tienen sentido, y en caso de trabajar como display de impedancias, debido al poco peso del eslabón efector final, un sólo actuador es capaz de mover el extremo del joystick y arrastrar al resto de las cadenas seriales, luego las singularidades de fuerza deben ser forzadas por el operador o bien por un comando erróneo de los diferentes actuadores.

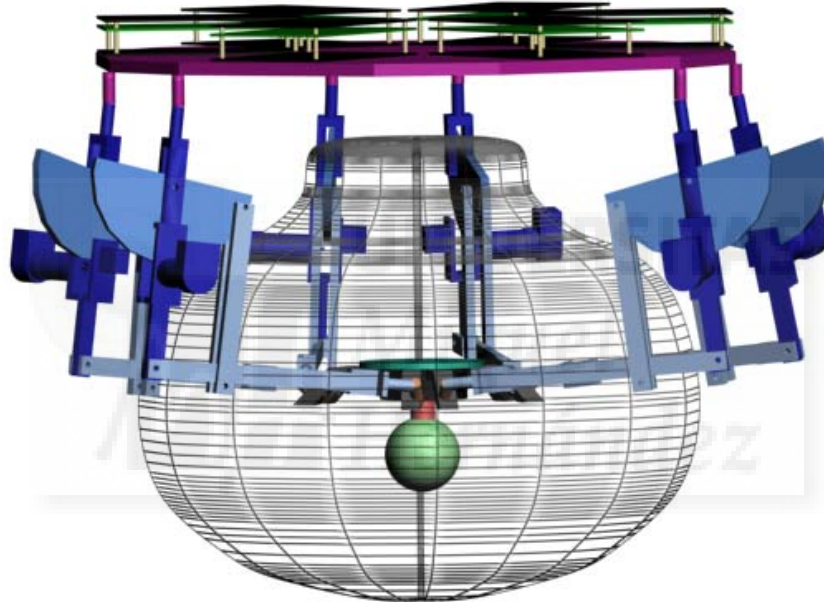


Figura 5.8: Espacio de trabajo con orientación del efector final constante.

La figura 5.7-(b) muestra el ancho de banda de estímulos del prototipo diseñado. Se grafica la salida de los encoders ante una entrada de una señal senoidal de 200 rad/s de frecuencia y 0.5 de amplitud. De hecho, el ancho de banda de los actuadores es mucho mayor, y otros experimentos demuestran que el dispositivo es capaz de reflejar hacia el operador señales de más de 450 rad/s, sin embargo la resolución de los encoders utilizados no permite

graficar estas señales. Estos experimentos revelan que el dispositivo diseñado alcanza actualmente la zona de 100 Hz de la figura 2.6. Sin embargo, y debido a la ausencia de reductores engranados, la mejora mecánica de la articulación esférica permite llegar a valores superiores a 320 Hz, a partir los cuales ya no es posible distinguir entre dos fuerzas consecutivas.

Las figuras 5.7-(c) y (d), relativas al experimento 2D muestran el comportamiento dinámico del sistema con un simple lazo de control en posición. Los valores de los reguladores PD utilizados en este experimento son los mostrados en la tabla 5.1. Como puede observarse en la figura, el dispositivo diseñado puede funcionar como display de admitancias con una velocidad de respuesta aceptable, habiendo sacrificado para ello la precisión en el posicionamiento. La figura (d) muestra la señal de control generada para seguir la trayectoria.

Tabla 5.1: Experimento 2D.- Parámetros PD

P	+0.04
D	+0.003

Un resumen de la experimentación llevada a cabo con el Magíster-P como dispositivo individual puede verse en la tabla 5.2

## 5.4. Conexión del dispositivo háptico

### 5.4.1. Un diseño para la estabilidad del sistema

La aparición del operador y el cruce de informaciones en el esquema bilateral modifica el comportamiento del sistema global, de forma que es necesario encontrar un método de diseño del control que permita garantizar la estabilidad del sistema.

Aunque en el caso de utilizar como maestro un dispositivo cerrado como el PHANToM se pudiera pensar, en principio que la estabilidad del mismo viene garantizada por el correcto uso de las funciones de la librería GHoST (gracias al uso interno de un tipo particular de red de acople, *god-object* [167]), la estabilidad del sistema global se ve afectada por varios factores adicionales; por ejemplo, en el caso de utilizar como entorno remoto una simulación dinámica,

Tabla 5.2: Experimentos con MAGISTER-P como dispositivo individual

	Disposición mecánica	driver	objetivo	Resultados destacados
Exp1A	1 brazo en vacío	torque control	obtener valores de la regresión para lectura en grados del encoder	cuentas inic.=630 m=+0.0476 b=+90 <sup>o</sup>
Exp1B	1 brazo en vacío	torque control	Introducir ruido blando para observar ancho de banda	la transm. por cable no introd. restricciones
Exp1C	1 brazo en vacío	torque control	controlar mediante PID en posición	P=0.03 I=0.002 D=0
Exp1D	1 brazo en vacío	torque control	repetir 1A en carga	Ok
Exp1E	1 brazo en vacío	torque control	repetir 1B en carga	Ok
Exp1F	1 brazo en vacío	torque control	repetir 1C en carga	Ok
Exp2A	Magister completo	torque control	implem. esquema general	Figura 5.7-(a)
Exp2B	Magister completo	torque control	comprobar ancho de banda	Figura 5.7-(b)
Exp2C	Magister completo	torque control	IMPEDANCE DISPLAY	Ok
Exp2D	Magister completo	torque control	ADMITTANCE DISPLAY	Figura 5.7-(c)-(d)

los estudios de Colgate [63] demuestran que la naturaleza discreta de los procesos numéricos implícitos en la simulación de sistemas dinámicos produce entornos remotos no disipativos, y puede producir inestabilidad en el sistema global. Además, es conocido que la aparición de retardos en las comunicaciones provoca del mismo modo inestabilidad en el sistema global. Por tanto, e independientemente de los dispositivos utilizados en el esquema de teleoperación, es necesario garantizar la estabilidad del sistema global (incluyendo la configuración del esquema bilateral utilizado) bajo ciertos condicionantes.

Existen varios estudios de estabilidad de sistemas teleoperados [158], [21], y de sistemas hápticos [2], [63], [92], sin embargo, la mayoría de estos estudios trabajan con *benchmarks* de sistemas lineales de 1 grado de libertad. La generalización a sistemas multivariables de  $n$  grados de libertad de estos trabajos depende de la formulación utilizada, y en ocasiones requiere una reformulación total del trabajo. En concreto, las formulaciones basadas en teoría moderna de espacio de estados y las basadas en la analogía del cuadripolo son las más sencillas de generalizar a sistemas multivariables.

El método presentado aquí es una generalización de los trabajos de Adams y Hannaford [2] para 1 grado de libertad en los maestros HDBF y Escalibur.

Si se considera el modelo dinámico simplificado de un dispositivo mecánico:

$$\tau = A\ddot{\theta} + H(\theta, \dot{\theta}) + G(\theta) \quad (5.1)$$

Se puede reescribir el vector  $H(\theta, \dot{\theta})$  de fuerzas como la matriz  $D(\theta, \dot{\theta})\dot{\theta}$ , siendo  $D$  una matriz  $N \times N$  de fuerzas de coriolis y centrífugas, y el vector  $G(\theta)$  como  $G \cdot \theta$  de forma que la matriz  $G$  contabilice sólo los efectos producidos por la fuerza de la gravedad. De esta forma la ecuación anterior queda como:

$$A\ddot{\theta}(t) + D\dot{\theta}(t) + G \cdot \theta(t) = K \cdot u(t) \quad (5.2)$$

donde ahora el vector  $u(t)$  contabiliza tanto las fuerzas externas ejercidas por el operador como las fuerzas producidas por los actuadores, y  $K$  es una matriz de distribución que aplica cada fuerza del vector  $u(t)$  sobre su correspondiente eslabón.

$$u(t) = [ f_m(t) \quad f_{act}(t) ]^T \quad (5.3)$$

donde  $f_m$  es la fuerza aplicada por el operador y  $f_{act}$  las fuerzas de los actuadores.

Si se denomina  $C_v$  a una matriz de distribución de las velocidades:

$$\begin{bmatrix} \dot{x}_m(t) \\ \dot{x}_{act}(t) \end{bmatrix} = C_v \dot{\theta}(t) \quad (5.4)$$

La ecuación que gobiernan el comportamiento dinámico pueden reescribirse en el dominio de Laplace como:

$$\begin{bmatrix} \dot{x}_m(s) \\ \dot{x}_{act}(s) \end{bmatrix} = C_v \cdot s (As^2 + Ds + G)^{-1} \cdot K \cdot \begin{bmatrix} f_m(s) \\ f_{act}(s) \end{bmatrix} \quad (5.5)$$

o de una forma más simplificada como:

$$\begin{bmatrix} \dot{x}_m \\ \dot{x}_{act} \end{bmatrix} = \begin{bmatrix} Y_{mm}(s) & \vdots & Y_{ma}(s) \\ \cdots & \cdots & \cdots \\ Y_{am}(s) & \vdots & Y_{aa}(s) \end{bmatrix} \begin{bmatrix} f_m(s) \\ f_{act}(s) \end{bmatrix} \quad (5.6)$$

donde los subíndices están referidos a la variable de salida en primer lugar y a la variable de entrada en segundo lugar.

A partir de la ecuación 5.6, particularizando para un esquema bilateral concreto y aplicando el criterio de Llewelyn (ecuación 2.8) se obtiene un método de diseño que asegura la estabilidad bajo las requisitos de pasividad en la zonas remota y local. Se muestra a continuación la aplicación de este método a un esquema fuerza-posición clásico (maestro como display de impedancias) en el que se añade una red de acople (el regulador) que garantice la estabilidad del sistema.

#### *esquema fuerza-posición*

En el caso de que el maestro esté funcionando como dispositivo de impedancias, los encoders ópticos proporcionan una lectura digital (de periodo  $T$ ) de la variable de posición  $x_{act}^*$ , que sirve para generar la variable de salida  $\dot{x}_s$ .

La fuerza digital realimentada desde el entorno remoto  $f_s^*$  se convierte a través de un retenedor de orden cero en la fuerza de los actuadores. Utilizando la nomenclatura de la figura 2.29 se obtiene la siguiente figura:

Incluyendo el integrador de la velocidad y el retenedor, las ecuaciones que

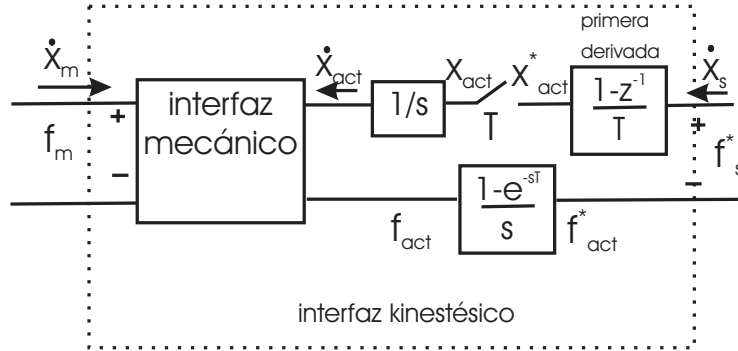


Figura 5.9: Implementación del esquema fuerza-posición

gobiernan el cuadripolo de la figura 5.9 quedan como:

$$\begin{bmatrix} \dot{x}_m \\ x_{act} \end{bmatrix} = \begin{bmatrix} Y_{mm}(s) & \vdots & Y_{ma}(s) \left( \frac{1-e^{sT}}{s} \right) \\ \dots & \dots & \dots \\ Y_{am}(s) \left( \frac{1}{s} \right) & \vdots & Y_{aa}(s) \left( \frac{1-e^{sT}}{s^2} \right) \end{bmatrix} \begin{bmatrix} f_m(s) \\ f_{act}^*(s) \end{bmatrix} \quad (5.7)$$

y si se asume que en la transformada de Laplace de la señal digital  $x_{act}^*$ ,  $x_{act}(j\omega) = 0$  para  $|\omega| \geq \frac{\pi}{T}$ , es decir:

$$x_{act}^*(s) = \frac{1}{T} x_{act}(s) \quad \forall \quad 0 \leq \omega \leq \frac{\pi}{T} \quad (5.8)$$

y se introduce la aproximación de la velocidad mediante la primera derivada, se obtiene:

$$\begin{aligned} \begin{bmatrix} \dot{x}_m \\ \dot{x}_s \end{bmatrix} &= \begin{bmatrix} Y_{mm}(s) & \vdots & Y_{ma}(s) \left( \frac{1-e^{sT}}{s} \right) \\ \dots & \dots & \dots \\ \frac{1}{T} \frac{1}{T} (1 - e^{sT}) Y_{am}(s) \left( \frac{1}{s} \right) & \vdots & \frac{1}{T} \frac{1}{T} (1 - e^{sT}) Y_{aa}(s) \left( \frac{1-e^{sT}}{s^2} \right) \end{bmatrix} \begin{bmatrix} f_m(s) \\ f_s^*(s) \end{bmatrix} \\ &= \begin{bmatrix} Y_{mm}(s) & \vdots & Y_{ms}^*(s) \\ \dots & \dots & \dots \\ Y_{sm}^*(s) & \vdots & Y_{ss}^*(s) \end{bmatrix} \begin{bmatrix} f_m(s) \\ f_s^*(s) \end{bmatrix} \end{aligned} \quad (5.9)$$

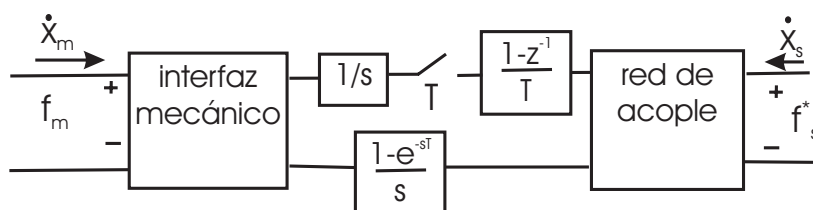


Figura 5.10: Red de acople en esquema fuerza-posición

Si se combina el cuadripolo de la ecuación 5.9 con una red de acople con parámetros ajustables dada por la ecuación 5.10 (figura 5.10).

$$Z_{c_i}(z) = b_{c_i} + k_{c_i} T \frac{z}{(z-1)} \quad (5.10)$$

se obtiene la ecuación

$$\begin{bmatrix} \dot{x}_m \\ \dot{x}_s \end{bmatrix} = \begin{bmatrix} Y_{mm}(s) & \vdots & Y_{ms}^*(s) \\ \dots & \dots & \dots \\ Y_{sm}^*(s) & \vdots & Y_{ss}^*(s) + \frac{1}{Z_{c_i}(e^{sT})} \end{bmatrix} \begin{bmatrix} f_m(s) \\ f_s^*(s) \end{bmatrix} \quad (5.11)$$

donde la aplicación de las condiciones de estabilidad (ecuación 2.8) derivan en las siguientes desigualdades:

$$\operatorname{Re}(Y_{mm}(s)) \geq 0 \quad (5.12)$$

$$2\operatorname{Re}(Y_{mm}(s)) \operatorname{Re}\left(Y_{ss}^*(s) + \frac{1}{Z_{c_i}(e^{sT})}\right) \geq |Y_{ms}^*(s)Y_{sm}^*(s)| + \operatorname{Re}(Y_{ms}^*(s)Y_{sm}^*(s)) \quad (5.13)$$

Cuyo cumplimiento garantiza la estabilidad del sistema global.

Desarrollos similares pueden hacerse con configuraciones diferentes al esquema fuerza-posición presentado en este epígrafe.

#### 5.4.2. Conexión del MAGISTER-P; Estabilidad del sistema implementado

El modelo dinámico simplificado (ecuaciones 5.14 y 5.15) obtenidas en el capítulo 3 permiten aplicar el desarrollo anterior al maestro diseñado en

esta tesis (figura 5.11), sin más que sustituir la ecuación 5.2 por la primera ecuación del sistema 5.14 y la ecuación 5.4 por la ecuación 5.15 y recordar la distribución del vector de coordenadas generalizadas. El vector  $\ddot{q}$  es un vector columna cuyos 6 primeros valores son la posición y orientación del eslabón 1, es decir, la posición del operador ( $x_m$ ), y el vector  $Q^A$  sus primeros elementos son las fuerzas del operador ( $f_m$ ). Por otra parte, el vector  $\nu$  tiene la misma ordenación que el vector  $\dot{q}$ .

$$\begin{bmatrix} M_{(78 \times 78)} & \phi_{q(78 \times 78)}^T \\ \phi_{q(78 \times 78)} & 0 \end{bmatrix} \begin{bmatrix} \ddot{q}_{(78 \times 1)} \\ \lambda_{(78 \times 1)} \end{bmatrix} = \begin{bmatrix} Q_{(78 \times 1)}^A \\ \gamma_{(78 \times 1)} \end{bmatrix} \quad (5.14)$$

$$\phi_q \dot{q} = \nu \quad (5.15)$$



Figura 5.11: Maestro MAGISTER-P

Sin embargo, para obtener un valor numérico en la desigualdad 5.13, se debe simplificar el modelo al movimiento en un sólo eje de forma que las funciones de transferencia ( $Y_{mm}, Y_{ms}^*, Y_{sm}^*, Y_{ss}^*$ ) sean funciones SISO unidimensionales. Esta simplificación no tiene sentido en el maestro diseñado, dado el alto grado de acoplamiento de los actuadores del dispositivo y a la dificultad de obtener un punto de funcionamiento donde evaluar  $\phi_q(q)$  y que los resultados obtenidos sean válidos para todo el espacio de trabajo.



No obstante, el método anterior presenta una serie de conclusiones que permiten partir de unas consideraciones generales a la hora de obtener los valores de los parámetros de los reguladores utilizados.

De la ecuación 5.12  $Y_{mm}(s)$  es por definición la función de transferencia que relaciona las fuerzas aplicadas por el operador con la velocidad del maestro cuando las fuerzas de los actuadores son nulas. Dado que el dispositivo mecánico sin actuadores es pasivo, su función de transferencia será positiva y real y la desigualdad se cumplirá siempre.

Por otra parte, si se reordena la ecuación 5.13 se obtiene la siguiente desigualdad:

$$\operatorname{Re} \left( \frac{1}{Z_{c_i}(e^{sT})} \right) \geq \frac{|Y_{ms}^*(s)Y_{sm}^*(s)| + \operatorname{Re}(Y_{ms}^*(s)Y_{sm}^*(s))}{2\operatorname{Re}(Y_{mm}(s))} - \operatorname{Re}(Y_{ss}^*(s)) \quad (5.16)$$

en la que se observa que es necesario un valor mínimo del valor de amortiguación en  $Z_{c_i}$ , lo que se interpreta como que no se puede reflejar un contacto completamente rígido en el maestro, sino que la rigidez máxima que puede ser reflejada está entre otras cosas, en relación inversa a la impedancia del dispositivo maestro  $Y_{mm}(s)$ , es decir, cuanto mayor sea la impedancia del dispositivo maestro sin actuadores ( $Y_{mm}(s)$ ), mayor será la capacidad de reflejar contactos rígidos. Este resultado está de acuerdo a los estudios de Colgate [61].

La frecuencia de muestreo ( $T$ ) es otro factor determinante en la estabilidad del sistema. En el caso del Magister-P la frecuencia de muestreo está limitada inferiormente por el tiempo de cómputo de los algoritmos cinemáticos del dispositivo, y por tanto su valor ya estaba determinado previamente, de forma que no se puede bajar de cierto límite inferior.

Observando estos requisitos, en los experimentos con el simulador y el maestro Magister-P se han ajustado los valores de los parámetros de la red de acople en función del tiempo de integración de la simulación. Derivando la clase *dhController* presentada en el capítulo 4 se construyen los reguladores que permiten estabilizar el sistema. Los valores presentes en la tabla 5.3 han sido comprobados que permiten una simulación estable de un esquema fuerza-posición siempre que no se produzcan fallos de robustez en la solución numérica de la cinemática del maestro.

Tabla 5.3: Parámetros de simulación estable: fuerza-posición

Tiempo de integración	0.001 seg
Rigidez $K_{c_i}$	200.0
Amortiguamiento $b_{c_i}$	5.0e-4
Factor de escala de reflexión de fuerzas	0.001

### 5.4.3. Conexión del MAGISTER-P; Uso de las librerías CLIB

Las comunicaciones con el hardware de la tarjeta dSpace 1103PPC están controladas completamente por el dispositivo DSP esclavo existente en la propia tarjeta. Las aplicaciones externas como un simulador o un gestor de comunicaciones bilateral no pueden acceder directamente al hardware de la tarjeta, y por tanto deben de hacerlo mediante llamadas a funciones del DSP. Según esta estructura, el esquema bilateral implementado debe comunicarse a través del DSP con la aplicación de tiempo real que realiza el control del dispositivo mecánico.

El procedimiento para establecer dicha aplicación es el siguiente:

1. Registrar la aplicación cliente en el DSP de la tarjeta.
2. Seleccionar la tarjeta que contiene la tarea de tiempo real.
3. Reservar recursos de memoria para intercambio de variables entre la tarjeta dSpace y la aplicación cliente.
4. Acceder a las variables de la tarjeta a través de las funciones existentes para ello. (Los nombres de las variables deben coincidir con aquellos programados en la tarjeta).
5. Liberar los recursos de memoria asignados.
6. Anular el registro de la aplicación.

Para ilustrar el procedimiento de conexión se muestran a continuación el constructor y destructor de la clase `dhMagisterP` desarrollada en el capítulo 4, si bien se ha suprimido parte del código de inicialización de algunas variables.

```
dhMagisterP::dhMagisterP(dhVector<dhRobot*> r) : dhInterface(r) {
    char* board_name;
    int ds_error;
    int state;
    unsigned int    board_index;

    UInt32  address_axis1,address_axis2,address_axis3;
    UInt32  address_axis4,address_axis5,address_axis6;

    board_name="DS1103";
    started=0;

    //registrar la aplicacion. Se anula en el destructor;
    ds_error=DS_register_host_app("MAGISTER");

    //obtener el indice de la tarjeta
    if(ds_error== DS_NO_ERROR)
        ds_error=DS_board_index(board_name,&board_index);

    //obtener el estado de la tarjeta
    if(ds_error== DS_NO_ERROR)
        ds_error=DS_is_reset(board_index,&state);

    //leer valores de las direcciones. NOMBRES VBLES EXP2A-HC
    if(ds_error== DS_NO_ERROR)
        ds_error=DS_get_var_addr(board_index,"pos1",&address_axis1);
    if(ds_error== DS_NO_ERROR)
        ds_error=DS_get_var_addr(board_index,"pos2",&address_axis2);
    if(ds_error== DS_NO_ERROR)
        ds_error=DS_get_var_addr(board_index,"pos3",&address_axis3);
    if(ds_error== DS_NO_ERROR)
        ds_error=DS_get_var_addr(board_index,"pos4",&address_axis4);
    if(ds_error== DS_NO_ERROR)
        ds_error=DS_get_var_addr(board_index,"pos5",&address_axis5);
    if(ds_error== DS_NO_ERROR)
        ds_error=DS_get_var_addr(board_index,"pos6",&address_axis6);

    if(ds_error != DS_NO_ERROR){
        ErrorHandler(ds_error);
        return;
    }
}
```

```

    }
}
//*****
dhMagisterP::~dhMagisterP() {
    disable();
    DS_unregister_host_app();
}

```

#### 5.4.4. Resultados experimentales

Para comprobar el funcionamiento del dispositivo como interfaz háptica en una aplicación de manipulación virtual, se propone realizar varias inserciones de un vástago prismático anclado en el extremo del robot virtual en un orificio que se encuentra sobre un cubo que posee dinámica propia y por tanto responde a las fuerzas generadas en el contacto vástago-orificio (figura 4.5). La gráfica 5.12 muestra los resultados de un experimento en el que se realizaron 4 inserciones. El control utilizado en este experimento fue un esquema fuerza-posición, de forma que se implementó el dispositivo maestro como display de impedancias y se cerró el lazo de posición del esclavo virtual con un control PD.

### 5.5. Conclusiones

Se ha desarrollado una plataforma para la experimentación de sistemas hápticos capaz de albergar gran parte de la casuística de estos sistemas.

Por una parte, la integración de nuevos dispositivos robóticos puede realizarse gracias a los interfaces genéricos diseñados para tal efecto. Se muestra la implementación hardware del dispositivo diseñado en el capítulo 3 y se muestran las características del mismo, en comparación con otros dispositivos del mercado.

Por otra parte, la implementación de un esquema específico de control se realiza via implementación software de una forma cómoda, derivando de las clases genéricas que definen la red de acople.

Finalmente, se muestran los resultados experimentales obtenidos durante la ejecución de una tarea de manipulación de un objeto virtual bajo un esquema clásico fuerza-posición.

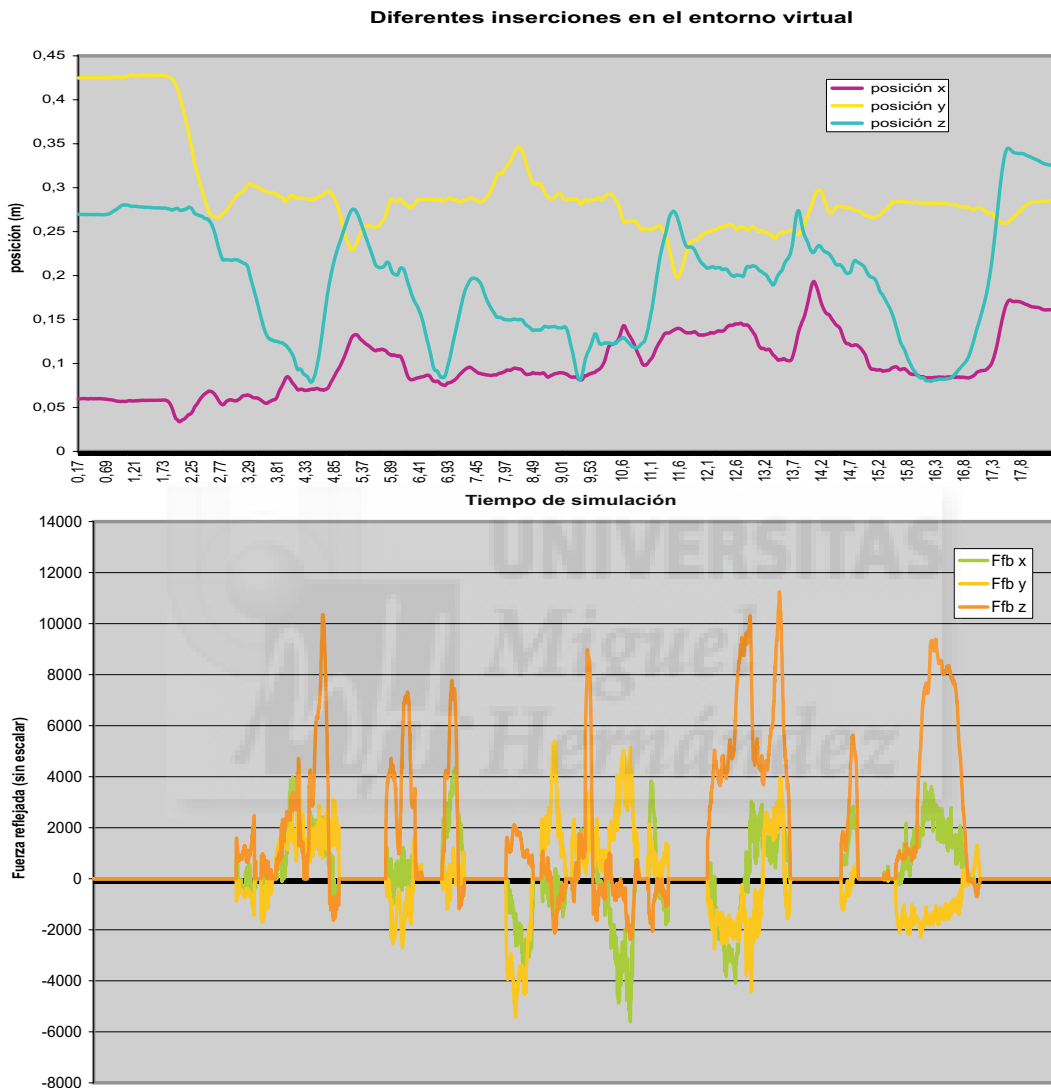


Figura 5.12: Inserciones con esquema fuerza-posición



# CAPÍTULO 6

## APLICACIÓN A SISTEMAS DE TELEOPERACIÓN ABIERTOS DE $N$ GRADOS DE LIBERTAD

---

*“El movimiento se demuestra andando.”*  
Refrán popular.

### 6.1. Introducción

El objetivo original de la teleoperación impone el control y manejo de un dispositivo remoto que interactúa con un entorno. Como se ha visto en el capítulo 2, este manipulador remoto puede ser un dispositivo real o un simulador capaz de realizar cálculos de esfuerzos. El espíritu de generalidad en la experimentación con plataformas teleoperadas que motiva esta tesis nos lleva a tratar de encontrar un dispositivo remoto real que sea capaz de conectarse a la plataforma experimental del mayor número de formas posibles. Es decir, encontrar un manipulador robótico capaz de ser programado de forma independiente y conectarse a la plataforma experimental mediante un protocolo

de comunicaciones, y que a la vez también disponga de la característica de poder comandar directamente los servos del robot de manera que se pueda implementar un esquema de teleoperación en el cual las comunicaciones se han substituido por conexiones eléctricas directas.

### 6.1.1. Estructura del capítulo

En primer lugar se ha utilizado como criterio la naturaleza de las comunicaciones en la plataforma experimental para realizar una clasificación de las posibles plataformas de teleoperación que pueden experimentarse con las herramientas desarrolladas en esta tesis. En esta clasificación se muestra una vez más la necesidad de disponer de un robot esclavo cuyo lazo de control esté abierto.

Seguidamente se justifica la elección del robot Mitsubishi PA-10 en función de los requerimientos deseados para el robot esclavo. Una vez se dispone del robot esclavo, se estudia las diferentes posibilidades del mismo desde el punto de vista de la interconexión con la plataforma de teleoperación, planteando las características de cada una de las mismas.

En el caso de “abrir” completamente el robot, es necesario disponer de un modelo dinámico lo más aproximado posible de forma que sea factible su interconexión con un dispositivo maestro. Para ello se presentan los modelos cinemáticos y dinámicos del robot PA-10, utilizando para ello un modelo simplificado y basándose en los valores de identificación de [111].

Posteriormente, se presentan la interfaz desarrollada para el caso de un robot con arquitectura abierta, mostrando la forma de trabajar con esta interfaz.

Y, finalmente se muestra la implementación de un esquema fuerza-posición entre el Magister-P y el robot esclavo seleccionado.

## 6.2. Comunicaciones en los esquemas de teleoperación

Las diferentes posibilidades de conexión entre la zona local y la zona remota así como la **naturaleza** y **configuración** de dichas conexiones proporciona un criterio de clasificación de las plataformas de teleoperación.

Atendiendo a la naturaleza de las conexiones se debe distinguir entre aquellas implementaciones en las que las comunicaciones no se integran en



el lazo de tiempo real (comunicaciones con retardos) (figuras 6.1, 6.2 y 6.3) y aquellos esquemas en los cuales las comunicaciones se integran dentro de una tarea global de tiempo real que controla tanto la zona remota como la zona local, y por tanto los retardos en las comunicaciones son perfectamente deterministas (figuras 6.4 y 6.5).

Por otra parte, atendiendo a la configuración de las conexiones, se puede encontrar la clasificación (de uso más extendido) de las diferentes arquitecturas bilaterales de control, como son los clásicos esquemas fuerza-posición, posición-posición, presión-posición, controlador IPC, etc.

En esta ocasión, y remarcando el enfoque tecnológico de este capítulo se utiliza una clasificación en función de la naturaleza de las conexiones.

### **6.2.1. Comunicaciones con retardos**

El tercer elemento constituyente, y aquel que le proporciona identidad propia, de un esquema de teleoperación remota son las comunicaciones establecidas entre la zona local y la zona remota. Cuando estas comunicaciones no se integran dentro de un lazo de tiempo controlado, nos encontramos ante la situación de que los retardos en las comunicaciones son indeterminados.

La plataforma diseñada en esta Tesis permite la experimentación de estos esquemas utilizando dispositivos robóticos comerciales o de investigación. La solución tecnológica adoptada consiste en una comunicación basada en RPC (remote procedure calls) utilizando bien los comandos de alto nivel de los dispositivos mecánicos (figuras 6.1, 6.2) o bien estructuras de datos que contengan la información intercambiada (figura 6.3).

#### **6.2.1.1. Comunicaciones a nivel de scripts**

En el caso de utilizar los comandos de alto nivel del robot esclavo, la experimentación queda reducida a aquellas configuraciones de control bilateral cuyos esquemas no necesitan acceder directamente a los servos del esclavo. Utilizando la clasificación clásica de arquitecturas bilaterales usada en [158], las tablas 6.1 y 6.2 muestran posibles configuraciones que se pueden experimentar con las plataformas de las figuras 6.1 y 6.2 respectivamente.

Mientras que si se utiliza el maestro PHANToM, éste sólo puede funcionar reflejando fuerzas e introduciendo posiciones en el esquema bilateral.

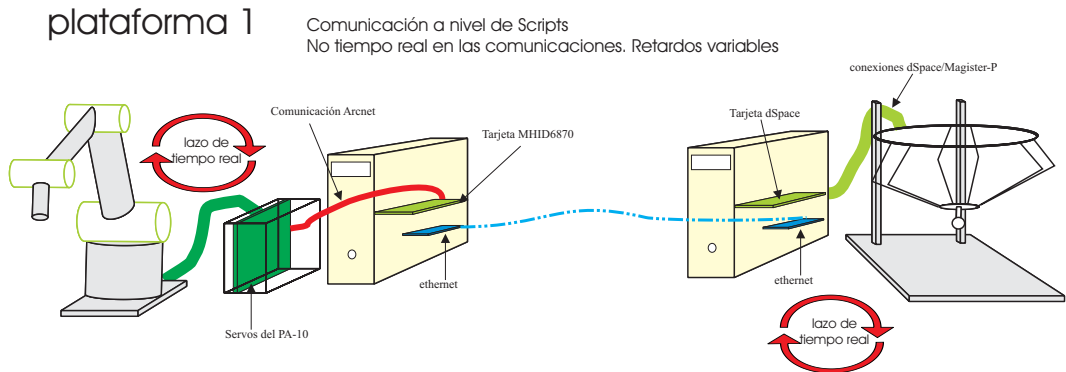


Figura 6.1: Plataforma de experimentación a nivel de scripts

Tabla 6.1: Esquemas en plataforma 1 (MAGISTER-P)

Esquema de control bilateral	Modo de trabajo del esclavo	Modo de trabajo del maestro
Posición-Posición	cerrado (lazo posición)	modo de admitancias
Fuerza-Posición	cerrado (lazo posición)	modo de impedancias

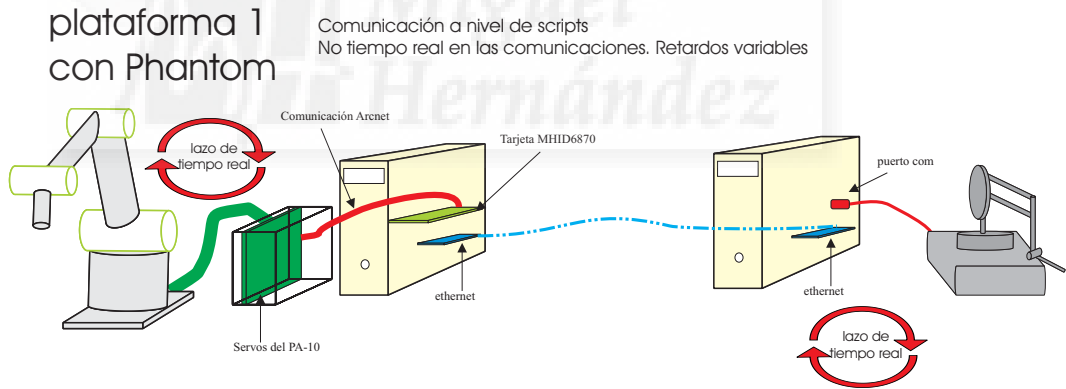


Figura 6.2: Plataforma de experimentación a nivel de scripts con el maestro PHANToM

### 6.2.1.2. Comunicaciones a nivel de señales

La característica de poder acceder directamente a comandar pares a los servos del esclavo e incluso modificar el lazo en velocidad de forma que

Tabla 6.2: Esquemas en plataforma 1 (PHANToM)

Esquema de control bilateral	Modo de trabajo del esclavo	Modo de trabajo del maestro
Fuerza-Posición	cerrado (lazo posición)	modo de impedancias

se puedan comandar señales de velocidad a los servos posibilita la implementación de nuevas configuraciones de esquemas bilaterales con sistemas de  $n$  grados de libertad. La figura 6.3 y la tabla 6.3 muestran posibles implementaciones de este caso.

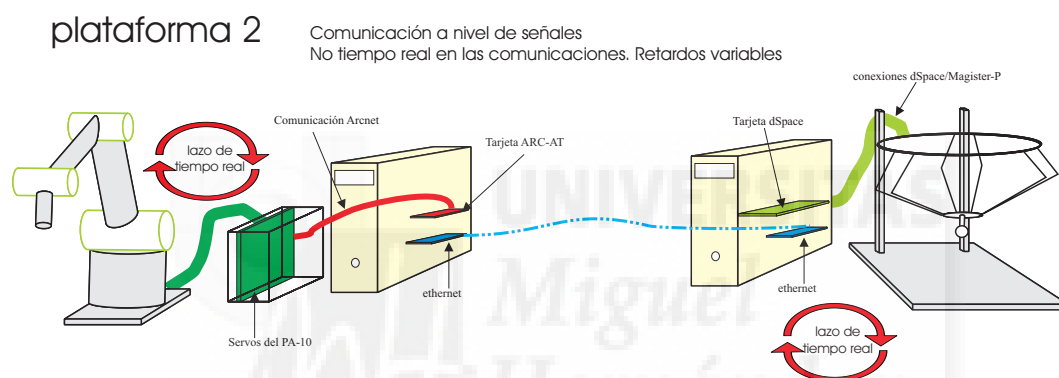


Figura 6.3: Plataforma de experimentación a nivel de señales con retardos variables

Tabla 6.3: Esquemas en plataforma 2 (MAGISTER-P)

Esquema de control bilateral	Modo de trabajo del esclavo	Modo de trabajo del maestro
Posición-Velocidad	Abierto (lazo posición)	modo de admitancias
Fuerza-Velocidad	Abierto (lazo posición)	modo de impedancias
Fuerza-Fuerza	Abierto (lazo pares)	modo de impedancias

### 6.2.2. Comunicaciones en tiempo real

Una primera aproximación en un intento de solucionar el efecto perjudicial de los retardos en las comunicaciones ha sido incluir las mismas en el lazo de control del esquema bilateral. Esta es la solución adoptada por los sistemas de teleoperación comerciales cerrados. Sin embargo, hasta ahora la experimentación de este caso se había visto reducida a simulaciones o dispositivos sencillos de 1 grado de libertad y comandados por una única tarjeta de entradas/salidas sobre la cual se implementaba el esquema de control. La plataforma diseñada permite la experimentación con dispositivos de  $n$  grados de libertad. Para ello se hace uso de la comunicación RS-232 disponible tanto en el computador de la zona remota como en la propia tarjeta dSpace de la zona local (figura 6.4).

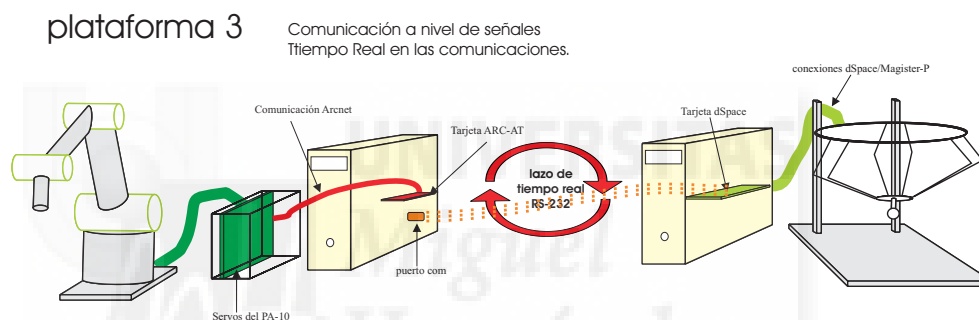


Figura 6.4: Plataforma de experimentación a nivel de señales

Sin embargo, el desarrollo de comunicaciones en tiempo real implica un problema tecnológico de sincronización importante, debido sobre todo a la necesidad de controlar las colisiones de información en el canal de comunicación. En el caso de la plataforma utilizada en esta tesis, el hecho de utilizar procesos numéricos iterativos para calcular la cinemática directa del maestro paralelo añade una dificultad adicional al problema en la sincronización de las tareas. Estos problemas ponen una vez más de manifiesto la utilidad del simulador diseñado en el capítulo 4, pues permite la resolución de todos estos problemas tecnológicos como paso previo a la implementación con sistemas reales. La figura 6.5 muestra un esquema de la plataforma de experimentación diseñada en la cual se ha hecho uso del simulador dinámico de tareas de teleoperación.

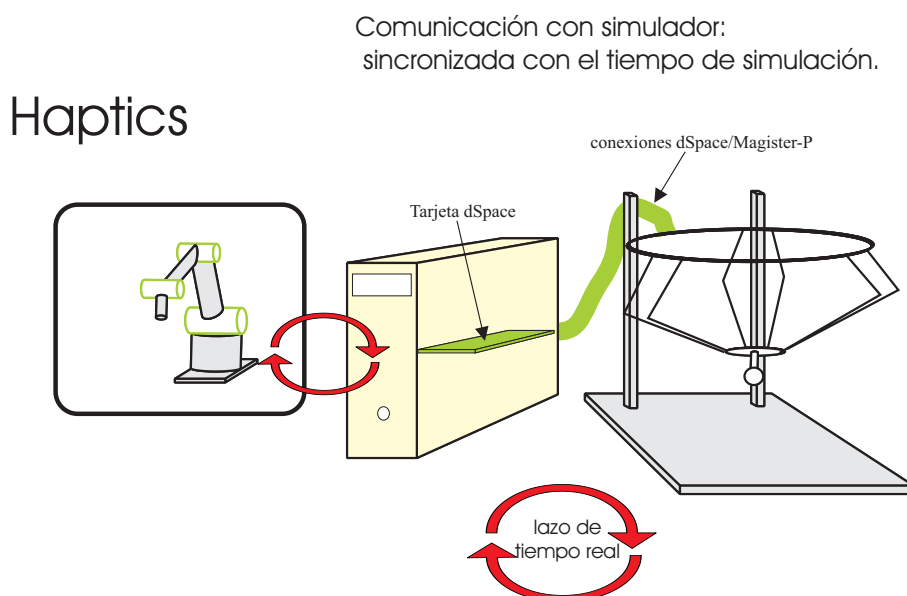


Figura 6.5: Plataforma de experimentación haptics

## 6.3. Elección de un dispositivo esclavo abierto

### 6.3.1. Justificación de la elección

En el mercado se pueden encontrar múltiples manipuladores robóticos que pueden usarse en tareas de teleoperación. Sin embargo, la inmensa mayoría de estos manipuladores se venden como soluciones cerradas desde el punto de vista del control. Muchos de ellos tan sólo ofrecen la posibilidad de programarse mediante lenguajes de programación propietarios [163], [129], [186] limitando las capacidades de comunicación a protocolos basados en RPC [162]. En los últimos controladores que están apareciendo también se oferta la posibilidad de programar el dispositivo mediante un lenguaje de programación genérico, usualmente C++, ampliando así las posibilidades de comunicación con el dispositivo robótico.

Sin embargo, y por razones obvias de protección de la integridad física del manipulador, se pueden encontrar muy pocos ejemplos de manipuladores completamente abiertos desde el punto de vista del control [152]. Para la

experimentación con dispositivos abiertos, la opción más común dentro de los grupos de investigación ha sido “abrir” un robot PUMA560 y tratar de identificar el modelo dinámico del mismo [23].

La aparición del robot Mitsubishi PA-10 (Portable General Purpose Intelligent Arm) en el mercado ofrece una nueva posibilidad a los investigadores dedicados al control dinámico de robots. Este robot dispone de unas librerías en C (Palib) que permiten controlar el robot. El robot tiene además la posibilidad de acceder directamente a los servos de los motores mediante una comunicación arnet (Attached Resource Computer Network). Esta doble posibilidad permite implementar cualquier tipo de esquema de teleoperación que se desee.

### 6.3.1.1. Arquitectura de control abierta

El robot de propósito general MHI PA-10 estructura su controladora según un modelo de capas que intenta simplificar la comunicación entre las propias capas y hacer más fácil el acceso a las mismas.

#### *Arquitectura hardware*

La controladora del robot se estructura en cuatro capas funcionales tal y como se muestra en las figuras 6.6 y 6.7. Estas capas se encuentran físicamente ubicadas en diferentes dispositivos, de forma que existe una separación física entre las mismas.

Nivel 1: Unidad mecánica

Está constituida por el brazo mecánico.

Nivel 2: Unidad ServoDriver

Está constituida por los cuatro servos de los actuadores del PA-10. Se accede a ella mediante comunicación ARCNET (red de paso de token implementada sobre un par trenzado).

Nivel 3: Unidad de control de movimiento

Está constituida por la tarjeta de control de ejes mitsubishi MHID 6780 sobre bus ISA. En ella se encuentran implementados los algoritmos de control dinámico y cinemático del brazo robot. Esta

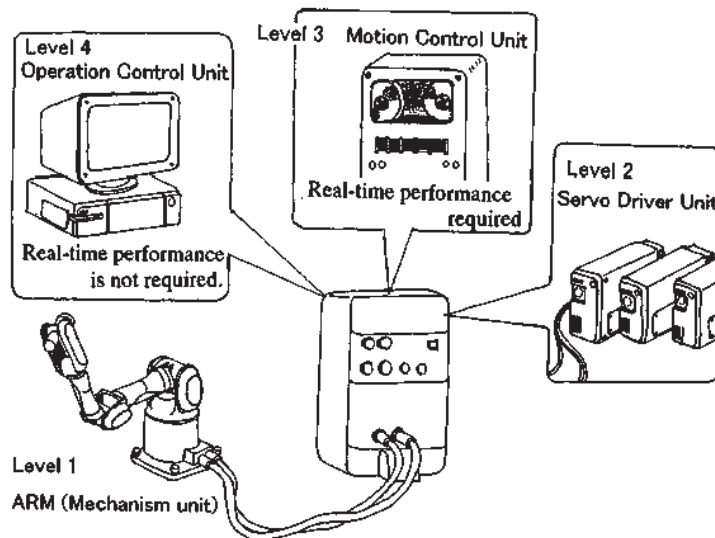


Figura 6.6: Estructura de capas del Mitsubishi PA10

tarjeta establece una comunicación en tiempo real con los servos del nivel 2. Al mismo tiempo, al estar conectada al PC que constituye el nivel 4, se comunica con el mismo mediante las librerías Palib.

Nivel 4: Unidad de control

Está constituida por un PC general y por las librerías en C para describir el movimiento del robot.

*Acceso a los diferentes niveles*

Nivel 1:

Para acceder al Nivel 1 sólo se tiene la posibilidad de hacerlo a través del Nivel 2, de forma que se garantice que la señal PWM necesaria para comandar los actuadores del brazo robot han sido generadas correctamente. Estos dos niveles se comunican entre sí mediante un cable de potencia y otro de señal y deben encontrarse siempre en cualquier configuración del sistema PA10.

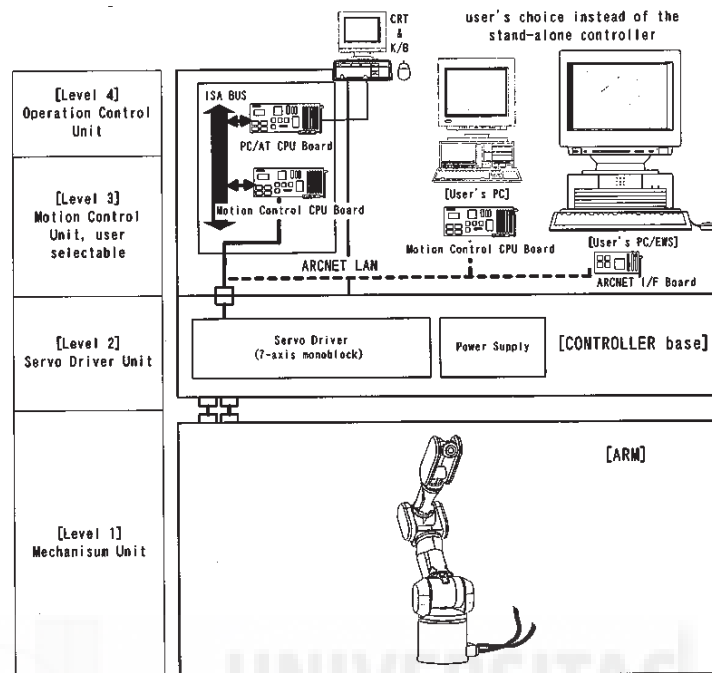


Figura 6.7: Arquitectura abierta del Mitsubishi PA10

#### Nivel 2:

Para acceder al Nivel 2 se puede usar la tarjeta controladora del Nivel 3, o bien sustituir esta tarjeta por una tarjeta de comunicaciones ARCNET que se utilice para enviar las tramas necesarias a los servos del robot. Esta segunda opción requiere el desarrollo de unas librerías de comunicaciones con los servos del robot y de la implementación de las mismas sobre un sistema de tiempo real que garantice dichas comunicaciones.

#### Nivel 3:

Para acceder al Nivel 3 en el caso de utilizar la tarjeta MHID6870 se deben usar las librerías proporcionadas por Mitsubishi® para ello. Sin embargo, si se usa una tarjeta de comunicaciones general, se deben desarrollar las librerías correspondientes que aseguren un comando correcto a los servos del robot.



Nivel 4:

El acceso al Nivel 4 es la programación sobre el propio PC que tiene las librerías Palib o las desarrolladas para las comunicaciones en general.

En el caso particular de esta tesis, se han realizado desarrollos sobre el sistema PA-10 completo, utilizando la Palib, de forma que se podía establecer una comunicación mediante ethernet con el sistema maestro y también sobre el sistema PA-10 abierto a partir del Nivel 2, de forma que se disponía del robot abierto y comunicación directa sobre los servos, pudiendo implementar los reguladores de acuerdo a las características del propio esquema de teleoperación elegido.

#### 6.3.1.2. Interfaz para la programación del robot

En el caso de utilizar la tarjeta controladora MHID6870 se ha implementado una arquitectura cliente-servidor basada en llamadas RPC [147] que permita la utilización de las propias librerías (palib.c) del robot en el software cliente.

En el caso de acceder al Nivel 2 de los servos del robot a través de una comunicación arcnet, ha sido necesaria la implementación de un software de comunicaciones que permite enviar y recibir tramas arcnet. El apartado 6.5.1 muestra algunos detalles de este software.

#### 6.3.2. Descripción del brazo esclavo

Se trata de un brazo redundante de 7 grados de libertad con las siguientes características:

- El robot posee 7 articulaciones. La configuración de las articulaciones desde la base del robot es: R - P - R - P - R - P - R (R: Rotación, P: Pivote).
- Peso del robot: 35 kg.
- Capacidad de carga: 10 kg.
- Máxima velocidad integrada: 1550 mm/s.
- Accionamientos: Motores AC.

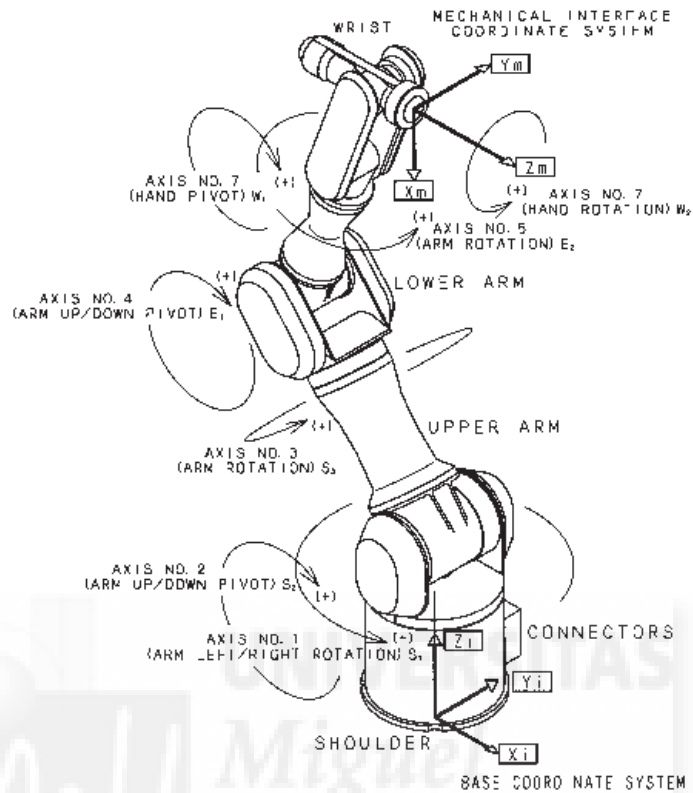


Figura 6.8: Brazo mecánico del Mitsubishi PA10

## 6.4. Modelado de un dispositivo esclavo abierto

### 6.4.1. Modelado cinemático del brazo serie

Al contrario que ocurría con los robots paralelos, en los robots seriales la solución de la cinemática es muy sencilla, por lo que se presentan aquí tan solo referencias a los algoritmos utilizados en el simulador del capítulo 4 [169].

En esta tesis, y a pesar de que el robot elegido como esclavo para la experimentación tiene 7 grados de libertad, se ha trabajado con la tercera articulación bloqueada, de forma que se tiene un robot antropomórfico de 6 grados de libertad.

*cinemática directa*

El problema cinemático directo se plantea en términos de encontrar una matriz de transformación que relaciona el sistema de coordenadas ligado al cuerpo en movimiento respecto a un sistema de coordenadas que se toma como referencia. Para lograr esta representación se usan las matrices de transformación homogénea 4x4, la cual incluye las operaciones de traslación y la orientación. Se muestra a continuación un ejemplo en Matlab de la función desarrollada para solucionar dicho problema.

```
>>q=zeros(6,1);
>>T=directkinematic6(q)
```

T =

```
1.0000    0    0    0.4500
    0    1.0000    0    0
    0    0    1.0000    0.8950
    0    0    0    1.0000
```

*cinemática inversa*

La cinemática inversa consiste en hallar los valores de las coordenadas articulares del robot  $q = [q_1, q_2, \dots, q_n]^T$  conocida la posición y orientación del extremo del robot.

A pesar de que en la literatura [30], [84], [64] se pueden encontrar diversos métodos genéricos para la resolución de la cinemática inversa que pueden ser implementados en computadora, suele ser habitual la resolución por medio de métodos geométricos. La mayor parte de los robots suelen tener cadenas cinemáticas relativamente sencillas, que facilitan la utilización de los métodos geométricos. Para muchos robots, si se consideran sólo los tres primeros grados de libertad, se tiene una estructura planar. Este hecho facilita la resolución del problema. Asimismo los últimos tres grados de libertad suelen usarse para la orientación de la herramienta, lo cual permite una resolución geométrica desacoplada de la posición de la muñeca del robot y de la orientación de la herramienta. Se muestra la ayuda de la función desarrollada en Matlab para la solución geométrica de un robot serial de 6 grados de libertad.

```

% INVERSEKINEMATIC6   Inverse Kinematic
% Q=INVERSEKINEMATIC6(T,CODO, MUNECA)
% devuelve el vector de coordenadas articulares correspondiente
% a la solución cinemática inversa de la mano del manipulador
% en la posición y orientación expresadas en la matriz T.
% CODO = 1 indica codo del robot arriba.
% CODO = -1 indica codo abajo.
% MUNECA = 1 indica que la muñeca del robot se sitúa
% por debajo de la coordenada expresada en T.
% MUNECA = -1 significa que la muñeca se sitúa por arriba.

```

## 6.4.2. Modelado dinámico del brazo serie

El principal escollo que hay que salvar en el modelado dinámico de un robot serial es el conocimiento de sus parámetros dinámicos. En esta tesis se han seguido los desarrollos de [111] que han permitido la obtención de un modelo dinámico de robot que ha sido validado con experimentación.

### 6.4.2.1. Modelo dinámico simplificado

Implementar un control dinámico de robots es todavía hoy un reto importante para los investigadores en robótica. La complejidad del problema, y el desconocimiento de los parámetros dinámicos hace que en la mayoría de las ocasiones se utilicen controles PD o PID donde el control se realiza independiente para cada articulación, confiando en el desacople de los eslabones debido al efecto de las reductoras de los actuadores y al alto valor de la constante proporcional [30].

Sin embargo, obtener del modelo dinámico del robot, junto con la mencionada capacidad de “abrir” el robot permite disponer de las herramientas necesarias para la experimentación en teleoperación.

Si consideramos un modelo simplificado del robot, en el cual se desprecian los efectos de rozamiento seco y viscoso en las articulaciones:

$$\tau = A\ddot{\theta} + H(\theta, \dot{\theta}) + G(\theta) \quad (6.1)$$

Donde  $\tau$  es el vector de fuerzas generalizadas,  $\theta$  es el vector de coordenadas generalizadas,  $A$  es la matriz de inercias,  $H$  el vector de fuerzas de coriolis y

centrífugas, y  $G$  el vector de gravedad.  $\theta, \dot{\theta}$  y  $\ddot{\theta}$  son las posiciones articulares, las velocidades articulares y las aceleraciones articulares respectivamente.

Para poder trabajar con este modelo se necesitan los parámetros dinámicos del robot.

#### 6.4.2.2. Datos del fabricante

Los parámetros geométricos y másicos el robot han sido proporcionados por el fabricante del mismo, y se encuentran resumidos en las tablas 6.4 y 6.5.

Tabla 6.4: Parámetros Denavit-Hartenberg Mitsubishi PA-10 robot

Eslabón	$a$ (m)	$\alpha$ (rad)	$d$ (m)	$\theta$ (rad)
1	0.0	0.0	0.315	0.0
2	0.0	-1.5708	0.0	0.0
3	0.0	1.5708	0.450	0.0
4	0.0	-1.5708	0.0	0.0
5	0.0	1.5708	0.500	0.0
6	0.0	-1.5708	0.0	0.0
7	0.0	1.5708	0.08	0.0

Tabla 6.5: Parámetros másicos del robot Mitsubishi PA-10

Eslabón	Masa (kg)	c.d.g. (m)		
1	9.78	0,0	0,0	-0,166
2	8.41	0,0	-0,0632	0,0
3	3.51	0,0	0,0	-0,112
4	4.31	0,0	-0,046	0,0
5	3.45	0,0	-0,0632	0,0
6	1.46	0,0	0,003	0,0
7	0.24	0,0	0,0	0,0

### 6.4.2.3. Identificación de los parámetros dinámicos

Se define la inercia aparente o dominante de un eslabón del robot como la inercia total medida en la articulación en cuestión. Para ello el resto de los eslabones están frenados y constituyen un cuerpo rígido (composite body method). El modelo dinámico de un solo eslabón rígido es:

$$I\ddot{\theta} + k_{vn}\dot{\theta} + g(\theta) = u \quad (6.2)$$

Donde  $I$  es la inercia dominante,  $k_{vn}$  es el coeficiente de rozamiento viscoso,  $g(\theta)$  es el término de la gravedad y  $u$  la entrada al actuador.

El control PD de dicha articulación se expresa en [64] como

$$u = k_p(\theta - \theta_d) - k_v(\dot{\theta} - \dot{\theta}_d) + \hat{g}(\theta) \quad (6.3)$$

Donde el subíndice  $d$  indica el valor deseado,  $k_v$  y  $k_p$  son las ganancias derivativa y proporcional y  $\hat{g}(\theta)$  la estimación de la compensación de la gravedad. Si se asume que  $g(\theta) \approx \hat{g}(\theta)$ , el sistema en lazo cerrado queda

$$I\ddot{\theta} + k_{vn}\dot{\theta} = -k_p(\theta - \theta_d) - k_v(\dot{\theta} - \dot{\theta}_d) \quad (6.4)$$

Que estableciendo a cero la posición y velocidad angular deseadas, la ecuación 6.4 queda como:

$$\ddot{\theta} + \frac{k_{vn} + k_v}{I}\dot{\theta} + \frac{k_p}{I}\theta = 0 \quad (6.5)$$

Ecuación de un segundo orden [151] dado por:

$$(s^2 + 2\zeta\omega_n s + \omega_n^2)\theta(s) = 0 \quad (6.6)$$

Donde

$$\frac{k_{vn} + k_v}{I} = 2\zeta\omega_n \quad \frac{k_p}{I} = \omega_n^2 \quad (6.7)$$

Por lo tanto, la inercia dominante de cada eslabón puede ser medida directamente de la respuesta del robot. La manera más sencilla de hacer esto es cancelar el efecto del rozamiento viscoso, obteniendo una oscilación no amortiguada, y midiendo la frecuencia natural, obtener la inercia dominante. Para conseguir una oscilación no amortiguada, se establece  $\zeta = 0$  en la ecuación 6.6, quedando  $(s^2 + \omega_n^2)\theta(s) = 0$ , lo que significa:

$$\frac{k_{vn} + k_v}{I} = 0 \quad k_{vn} + k_n = 0 \quad (6.8)$$

Es decir, el procedimiento experimental para medir las inercias dominantes de cada eslabón es el siguiente:

1. frenar todos los eslabones menos aquel que quiera medirse
2. aplicar un par igual al termino gravitacional, es decir hacer  $k_p=k_v=0$ . Esto hace que al mover el eslabón a mano, el termino gravitacional se compense y el robot se mueve libremente con un ligero amortiguamiento debido a la fricción de los motores y a la transmisión mecánica.
3. decrementar muy lentamente y a intervalos muy pequeños el valor de  $k_v$  hasta que no aparezca el amortiguamiento anterior, es decir, que  $k_v$  cancele el efecto del rozamiento viscoso  $k_{vn}$ .
4. aumentar el valor de  $k_p$  hasta un valor suficiente para medir con comodidad el periodo T de las oscilaciones constantes del eslabón.

$$\hat{I} = \frac{k_p T^2}{4\pi^2} \quad (6.9)$$

Obviamente, los valores medidos según este procedimiento son dependientes de la configuración, y por tanto solo válidos en las configuraciones donde se realizan los experimentos.

El siguiente paso es obtener una relación entre las inercias aparente de los eslabones y los parámetros del modelo de la ecuación 6.1. Para ello se utilizan los desarrollos de Khatib [23]. Estas relaciones se han obtenido de forma simbólica (utilizando Maple) a partir del modelo de las ecuaciones de Lagrange-Euler [84]. El desarrollo realizado puede encontrarse en el cd adjunto. De esta forma, se obtienen expresiones simbólicas que permiten calcular, a partir de los valores experimentales obtenidos por Jamisola, los valores de los parámetros de la ecuación 6.1.

Como ejemplo se muestra la ecuación obtenida para el valor del coeficiente  $a_{17}$  de la matriz de inercias:

$$\begin{aligned} a_{17} := & .125*\cos(q2)*( \cos(q4)*\cos(q6)-\cos(q5)*\sin(q4)*\sin(q6) ) \\ & +.125*\sin(q2)*( \sin(q3)*\sin(q5)*\sin(q6)+\cos(q3)*(-\cos(q6) \\ & *\sin(q4)-\cos(q4)*\cos(q5)*\sin(q6) ) ) \end{aligned}$$

Estos desarrollos permiten obtener un modelo dinámico en modo simbólico válido y fácilmente implementable en el simulador del capítulo 4.

## 6.5. Conexión del dispositivo esclavo

### 6.5.1. Apertura del lazo de control del brazo robot

Para poder desarrollar una interfaz de comunicación con los servos del PA-10 es necesario conocer las características de dicha comunicación. Una información más detallada sobre la misma se puede encontrar en [155].

En primer lugar se debe conocer que el Nivel 2 está constituido por 4 servos que comandan los 7 ejes del brazo mecánico según la arquitectura mostrada en la figura 6.9. Para establecer una comunicación con estos servos debe conocerse la dirección arcnets de los mismos, establecida mediante unos switches en la tarjeta (por defecto: FEH), y además establecer una dirección a la tarjeta de comunicaciones (por defecto: FFH).

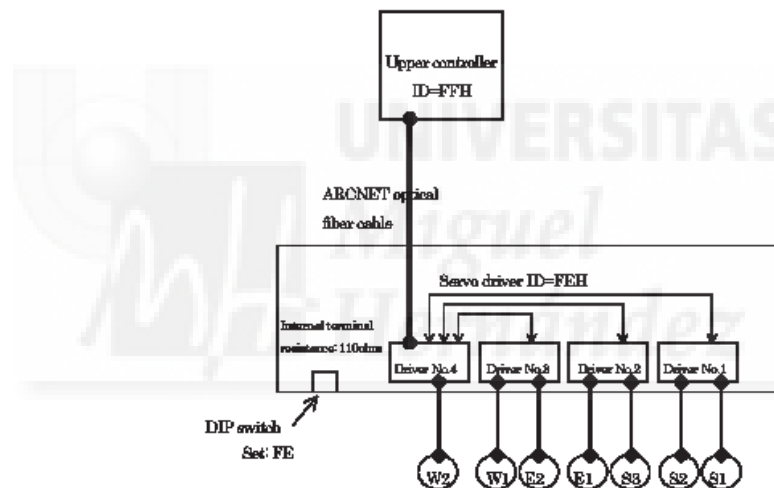


Figura 6.9: Arquitectura de los servos del Mitsubishi PA10

La comunicación con los servos del PA-10 se establece mediante tramas enviadas por el par trenzado. La figura 6.10 muestra los diferentes comandos que permiten entrar y salir de los diferentes modos de control de los servos. Nos centramos en esta breve explicación en el modo de control. Como se observa en la figura, para entrar al control mode primero de debe enviar el comando 'S', de forma que los servos se encuentran a partir de ese momento esperando tramas 'C' con los comandos de control con una frecuencia de 10



ms. Para finalizar este modo se realiza mediante el envío de una trama 'E'. De esta forma la comunicación sigue la estructura:

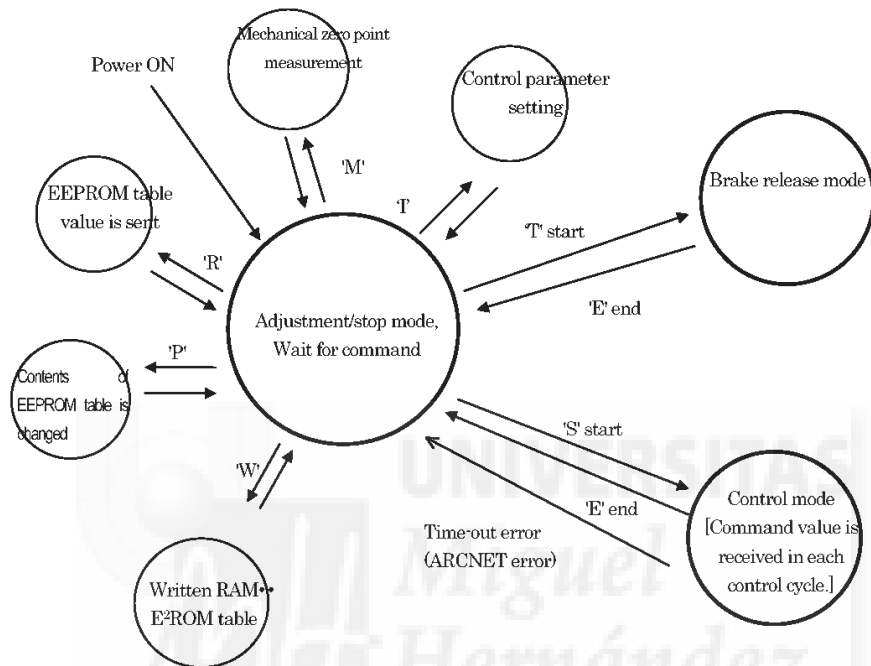
$$'S' \rightarrow 'C' \rightarrow 'C' \rightarrow 'C' \rightarrow 'C' \rightarrow 'E'$$


Figura 6.10: Modos de comunicación de los servos del PA10

Se muestra a continuación las estructuras de las tramas arcnet enviadas y recibidas. Información similar sobre el resto de las tramas se puede encontrar en [155].

```
//ESTRUCTURAS TRAMA PARA MANDAR
```

```
//COMANDO DE 1 EJE
```

```
typedef struct {
    unsigned char    Status;    // Status del eje (1 byte)
    unsigned short  Tolck;     // Par motor comandado (2 bytes)
    unsigned short  Speed;     // Velocidad comandada (2 bytes)
```

```

}C_COM;

//TRAMA COMPLETA
typedef struct {
    unsigned char    ChkNo;    // N° de check
    unsigned char    Code;     // 'S', 'R', 'C' o 'E'
    C_COM            Drive[7]; // 7 Estruct. C_com de los 7 ejes
} SEND_SD;

//ESTRUCTURAS TRAMA PARA RECIBIR

//DATOS DE 1 EJE
typedef struct {
    unsigned short   Status; //Status del eje (2 bytes)
    long             Rez;    //Angulo del resolver (4 Bytes)
    unsigned short   TolK;  //Par del eje (2 bytes)
} RC_COM;

//TRAMA COMPLETA
typedef struct {
    unsigned char    ChkNo;    // N° de check devuelto
    unsigned char    Code;     // 'S' , 'C' o 'E' devuelto
    RC_COM            Sts[7];   // 7 RC_COM de los 7 ejes
    unsigned short   MStatus;  // Master servo status (2 bytes)
} RECV_SD;

```

#### 6.5.1.1. Interfaz de comunicación en arquitectura abierta

La tarjeta arcnet utilizada es la ARC-AT de la casa Toyomicrosystems Corporation (TMC), basada en el controlador COM20020 capaz de establecer comunicaciones entre 156 Kbps y 10 Mbps entre un máximo de 255 nodos.

Siguiendo las instrucciones del manual de los servos del PA-10, para poder establecer la comunicación, se debe trabajar a 5 Mbps, y utilizar el transmisor HYC2458S. Según esta configuración, los jumpers de la tarjeta ARC-AT deben configurarse de la forma que indica la tabla 6.6:

En la interfaz desarrollada se han utilizado las siguientes estructuras para completar las tramas a mandar mediante arcnet:

##### **Estructuras para mandar datos**

Estructura c\_com: contiene 1 byte de status, 2 bytes del comando de

Tabla 6.6: Configuración de los jumpers de la tarjeta ARC-AT.

Posición	Comentario
JP1: IRQ2	Según software desarrollado
JP2: 40Mhz	Comunicación 5 Mbps
JP3: INV	polaridad led
JP4: Z	HYC2485S
JP5: Y	HYC2485S
JP6: Z	HYC2485S
JP7: Z	HYC2485S
JP8: Z	HYC2485S
JP9: Z/Y	HYC2485S
JP10: Z/Y	HYC2485S
JP11: ON	Terminación
JP12: ISO	HYC2485S
JP13: ISO	HYC2485S
JP14: ISO	HYC2485S
JP15: ISO	HYC2485S
JP16: ISO	HYC2485S

par enviado y 2 bytes del comando de velocidad enviado. Esta estructura se utiliza para almacenar los datos relativos a un eje y construir la estructura `send_sd`.

Estructura `send_sd`: contiene 1 byte de `chkno`, 1 byte de código enviado ('S', 'C' o 'E') y 7 estructuras `c_com` con los datos de cada eje.

#### **Estructuras para recibir datos**

Estructura `rc_com`: contiene 2 bytes de `status`, 4 bytes del ángulo del resolver y 2 bytes del par leído en el eje mediante la intensidad. Esta estructura se utiliza para almacenar los datos relativos a un eje y construir la estructura `recv_sd`.

Estructura `recv_sd`: contiene 1 byte de `chkno`, 1 byte de código recibido ('S', 'C' o 'E'), 7 estructuras `rc_com` con los datos de cada eje y 2 bytes de `mstatus` con el estado general de los servos.

#### **Estructuras para almacenar datos**

Estructura `axis`: contiene 2 bytes de `status` de un eje, 4 bytes del ángulo

del resolver y 2 bytes del par leído en el eje mediante la intensidad.

### Funciones miembro desarrolladas

Se muestran en la tabla 6.7 una relación de las funciones miembro desarrolladas para esta interfaz con objeto de facilitar la comprensión del código del cd adjunto.

Tabla 6.7: Funciones miembro de la clase CServo.

Función	Descripción
CServo::Inicializar	inicializa la tarjeta arcnet
CServo::mandar_trama	manda la estructura _SendPacket[256]
CServo::recibir_trama	recibe la trama _RecvPacket
CServo::mandar	Efectúa 3 intentos de mandar la trama llamando a mandar_trama
CServo::recibir	Efectúa 3 intentos de recibir la trama llamando a recibir_trama
CServo::set_Ccom	Actualiza la estructura_Ccom
CServo::set_eje	
CServo::vSetData	Construye la estructura_SendPacket
CServo::vSendInitC	inicializa la tarjeta arcnet
CServo::iRecvC	Recibe la respuesta al comando 'C'
CServo::iRecvR	Recibe la respuesta al comando 'R'
CServo::iSendC	Manda el comando 'C'
CServo::iSendR	Manda el comando 'R'
CServo::iSendSE	iSendSE

Finalmente, la figura 6.11 muestra una imagen de la interfaz desarrollada para el comando de pares a los servos del Mitsubishi Pa-10.

## 6.6. Implementación de un esquema de teleoperación

Finalmente se muestra el esquema de la implementación de un esquema fuerza-posición entre el dispositivo maestro Magister-P diseñado y el robot esclavo PA-10 según la figura 6.3.

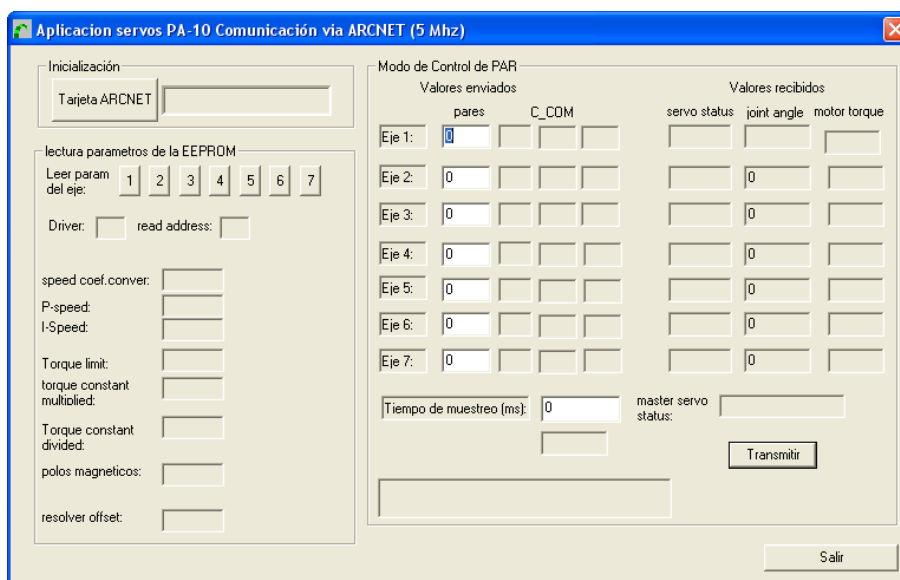


Figura 6.11: Graphics User Interface de comunicaciones

Para realizar la comunicación entre los distintos computadores que intervienen en el sistema se ha utilizado una arquitectura cliente - servidor mediante el protocolo RPC (Remote Procedure Call, llamada a procedimiento remoto). En concreto se ha utilizado el protocolo RPC/XDR (Remote Procedure Calls / External Data Representation, llamadas a procedimientos remotos / representación externa de datos) basado en la especificación de Sun. Este protocolo permite desarrollar aplicaciones que ejecutan procedimientos (funciones) de otros computadores conectados en red, sin tener en cuenta que los procedimientos no son ejecutados localmente. Es decir, el protocolo RPC proporciona un mecanismo mediante el cual un proceso, el proceso cliente, puede comunicarse con otro proceso, el proceso servidor, simplemente ejecutando una llamada a un procedimiento como si fuera una subrutina en el sistema local. En la figura 6.12 se muestra la arquitectura del sistema junto a los flujos de datos.

En la implementación del sistema de comunicaciones se ha utilizado la herramienta Distinct ONC RPC/XDR Toolkit - 32. Esta herramienta ha permitido el desarrollo de aplicaciones cliente - servidor en C y C++ para Windows 2000 utilizando la especificación estándar de ONC RPC/XDR (Remote

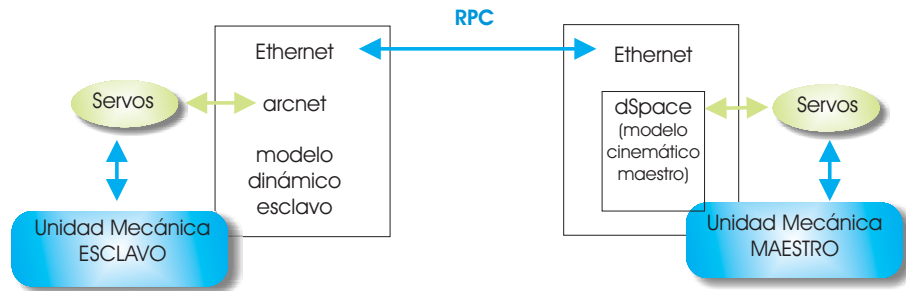


Figura 6.12: Implementación de un esquema de teleoperación

Procedure Calls / External Data Representation) de Sun. Esta herramienta dispone de una biblioteca y de un compilador, llamado RPCGEN para generar todo el código según el protocolo RPC.

## 6.7. Conclusiones

Se ha desarrollado una plataforma para la experimentación de sistemas teleoperados capaz de albergar gran parte de la casuística de estos sistemas.

El acceso a bajo nivel a los servos de un robot comercial ha estado impedido por los fabricantes de robots, principalmente por el peligro y la dificultad de implantar posteriormente un control adecuado para el movimiento del robot. Sin embargo, para realizar una experimentación de los esquemas de control bilateral, es necesario disponer de dispositivos mecánicos “abiertos” a los cuales se les pueda interconectar otro dispositivo que actúe como maestro, e implementar entre ambos un esquema de control.

Este capítulo presenta las ideas básicas necesarias para la obtención de un modelo identificado del robot seleccionado, a la vez que muestra el procedimiento de acceso directo a los servos del robot, obteniendo así las herramientas necesarias para trabajar con un robot “abierto” y ser capaz de controlarlo.

# CAPÍTULO 7

## CONCLUSIONES Y TRABAJO FUTURO

---

*“He revisado mis notas y no me gustan. He pasado tres días en la U.S. Robots y lo mismo habría podido pasarlos en casa con la enciclopedia Telúrica.”*  
*Susan Calvin. Robotpsicóloga*  
*I. Asimov. Yo, Robot*

### 7.1. Conclusiones generales

La experimentación con sistemas teleoperados de  $n$  grados de libertad encierra una serie de dificultades teóricas y tecnológicas que hacen de la misma un campo apasionante y claramente multidisciplinar. Cuando se pretende abordar un tema como el citado, es obligado un análisis previo de los elementos que componen el problema, así como de la casuística del mismo. A lo largo de este proceso han ido constantemente apareciendo diferentes aspectos implícitos en la teleoperación que abrían nuevas vías de investigación. El objetivo inicial de la tesis, disponer de sistemas abiertos con los que se pudiera experimentar, ha ido derivando, entre otras metas parciales, en el diseño, modelado y análisis de un dispositivo kinestésico novedoso basado en

la modificación de una plataforma paralela. Desarrollado este dispositivo, y obtenidos el resto de los objetivos de la tesis, como un esclavo abierto y un simulador dinámico de robots, se abre ahora un campo verdaderamente amplio de posibilidades en la experimentación de sistemas bilaterales. Algunos de los caminos de este campo, como el desarrollo de algoritmos de mapeado e indexación de espacios de trabajo entre el dispositivo maestro y el dispositivo esclavo han empezado a ser investigados en este mismo trabajo. Otros caminos que se han ido abriendo a lo largo del trabajo podrán igualmente ser probados experimentalmente con la plataforma diseñada.

### 7.1.1. Contribuciones aportadas

Recordando el apartado de contribuciones originales presentado en el capítulo 1, se repasa a continuación el citado esquema en el que se citan los logros conseguidos y la utilidad de los mismos:

- Se ha desarrollado un estudio desde el punto de vista fisiológico y ergonómico del diseño de interfaces mecánicas de operador. Dicho estudio debe considerarse como punto de partida para el diseño de interfaces kinestésicas. Los aspectos tratados en el mismo deben estar, y han estado, presentes en el proceso de concepción de una interfaz destinada a trabajar como maestro en un esquema de teleoperación.
- Los diferentes estados de arte de sistemas teleoperados a menudo son presentados bajo el punto de vista de la configuración de la información intercambiada. Se ha tratado en esta tesis de adicionar a este enfoque un punto de vista tecnológico de forma que se disponga de un criterio para la clasificación de los posibles esquemas de control bilateral que se pueden implementar en función de los dispositivos mecánicos de los que se disponga.
- Se han desarrollado los algoritmos de análisis de un robot paralelo de cadena cinemática tipo URS, y la implementación del control de un dispositivo de este tipo. El resultado de estos trabajos ha sido por una parte una herramienta vrml desarrollada sobre Matlab+Simulink<sup>®</sup> que permite la visualización del funcionamiento de dichos algoritmos, y por otra parte el desarrollo de la librería de dinámica multicuerpo **mblib** desarrollada en ansi-c que puede utilizarse en los programas de control que se quieran implementar en la tarjeta dSpace.



- Se dispone de un método de obtención de los parámetros dinámicos válidos para el modelo dinámico de un robot serial, y del procedimiento de “apertura” de la controladora del robot seleccionado. El resultado ha sido un software de comunicaciones arcnet capaz de acceder a los servos del robot Mitsubishi® PA-10 y comandar pares a los mismos. Dicho software puede ser implementado dentro de cualquier otro proyecto más general.
- Se ha desarrollado un simulador dinámico de esquemas teleoperados que permite la inclusión del modelo dinámico de uno o varios esclavos, y que además es capaz de conectar varios maestros que controlan a uno o varios esclavos. El resultado ha sido un software genérico y abierto que incluye herramientas de visualización OpenGL como cámaras que permiten un fácil desarrollo de nuevos algoritmos que hagan uso de las capacidades dichos objetos.
- Se muestra la implementación hardware y software de una plataforma para la experimentación de arquitecturas de control bilateral. El resultado de este último punto ha sido la consecución del objetivo inicial de la tesis de disponer de una plataforma de experimentación de tareas teleoperadas y hapticas sobre la cual se pudieran implementar diferentes arquitecturas tanto físicas como lógicas.

## 7.2. **Objetivos pendientes**

La presente tesis tiene el espíritu de aportar una herramienta lo más genérica posible que ayude en la experimentación en control bilateral. Consecuentemente, cuanto mayor sea el grado de satisfacción en los objetivos marcados, mayor será el campo abierto a continuación. Por tanto, los diferentes caminos que han ido apareciendo a lo largo de la tesis presentan múltiples aspectos (cada cual más interesante) dignos de ser continuados con el mismo interés que se ha puesto en el desarrollo del trabajo previo. Con objeto de hacer una clasificación de estos caminos, se ha seguido un criterio de agrupación por capítulos de la tesis.

### 7.2.1. En modelado y análisis de robots paralelos...

El ingreso en el mundo de los robots paralelos ha supuesto una aventura apasionante que merece ser comentada. Las ventajas esta tipología de robots han sido explotadas para su utilización como dispositivos hápticos de forma exitosa. Sin embargo, varios aspectos han quedado abiertos, los desarrollos numéricos utilizados en esta tesis proporcionan una solución iterativa a los problemas cinemático y dinámico. Por tanto, la utilización de estos modelos en esquemas de control en tiempo real es por naturaleza incompatible. La solución aportada ha sido la medición del tiempo de cómputo máximo para un error máximo admisible determinado con objeto de sincronizar la tarea de tiempo real. Esta solución funciona en la mayoría de los casos, pero no es robusta, en el sentido de qué pasa cuando el proceso iterativo no converge. Aparte de este problema, el tiempo de cómputo de un proceso iterativo es mucho más elevado que el de una solución cerrada. En otro ámbito, el modelo dinámico numérico permite simular cualquier punto de funcionamiento del dispositivo, pero no permite un diseño algebraico de los reguladores que estabilicen el sistema.

Por tanto, la obtención de una solución cerrada para el dispositivo diseñado, además de aparecer como un problema excitante, permitiría la optimización de los desarrollos de la tesis, aportando robustez a la plataforma experimental.

### 7.2.2. En identificación y modelado de robots seriales...

Ya se han comentado las ventajas de la apertura del lazo de control del robot utilizado como esclavo. Sin embargo varios aspectos relativos al modelado han quedado inacabados.

En primer lugar, el software desarrollado permite el envío de pares a los servos del robot. Sin embargo, los servos de dicho robot pueden comandarse con señales de velocidad, y los valores de dicho lazo pueden modificarse.

Por otra parte, se ha supuesto una configuración antropomórfica para el brazo robot, bloqueando la tercera articulación del mismo. En realidad el robot es redundante con 7 grados de libertad. La disponibilidad del robot abierto junto con el desarrollo de una pseudojacobiana del mismo permitiría experimentar en control de brazos robots basados en modelos del mismo.

### **7.2.3. En robótica virtual y simulación dinámica...**

El campo de la simulación computacional se amplía maravillosamente a medida que se avanza en él. El simulador desarrollado sobre una librería de libre distribución hace uso intensivo de los algoritmos implementados en dicha librería. Sin embargo, son muchas las mejoras que pueden ir adicionándose a dicho código, otros detectores de colisiones, otros generadores de fuerzas, otros integradores numéricos..., de forma que cada de estos problemas uno puede ser tratado como un módulo independiente.

Por otra parte, la naturaleza discreta de los procesos numéricos impone nuevos retos teóricos en el aspecto de estabilidad.

### **7.2.4. En comunicaciones en tiempo real...**

Los problemas tecnológicos de comunicaciones han sido solucionados para parte de los posibles esquemas de control bilateral estudiados. Sin embargo, la implementación de nuevos esquemas puede requerir nuevas modificaciones. La aconsejable migración de todo el sistema desarrollado a un sistema de tiempo real estricto permitiría la ampliación de las capacidades del sistema.





## Parte II

### ANEXOS





# Apéndice A

## Planos mecánicos del MAGISTER-P

---

En el presente anexo se muestran algunos planos de fabricación utilizados en la fabricación de uno de los brazos del MAGISTER-P.

Los ficheros .dwg (versión de Mechanical Desktop 6.0) y los ficheros .mpf (ascii) de control numérico (sinumerik 810D) se encuentran en el cd adjunto.

### A.1. Planos generales

### A.2. Planos de fabricación

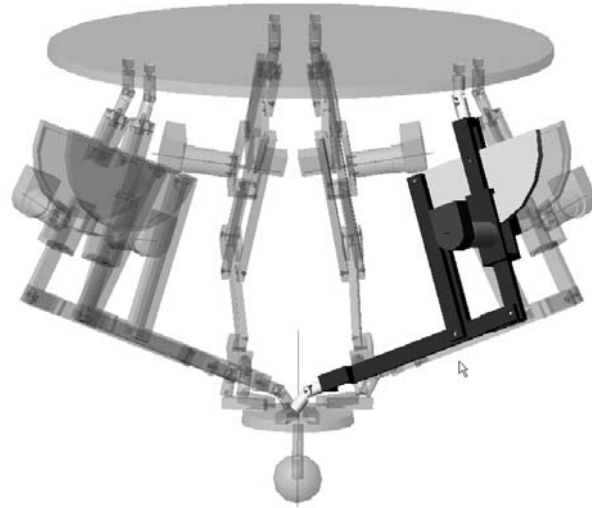


Figura A.1: Modelo Mechanical Desktop

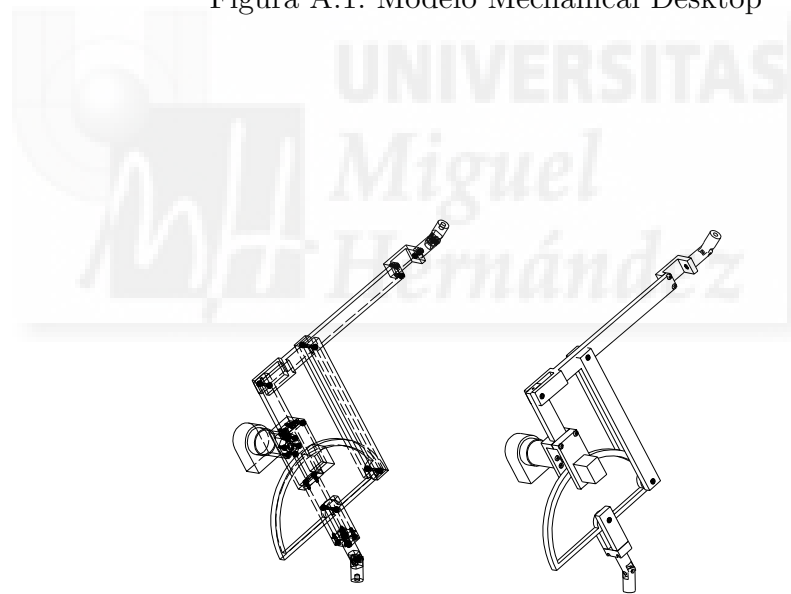


Figura A.2: Brazo mecánico



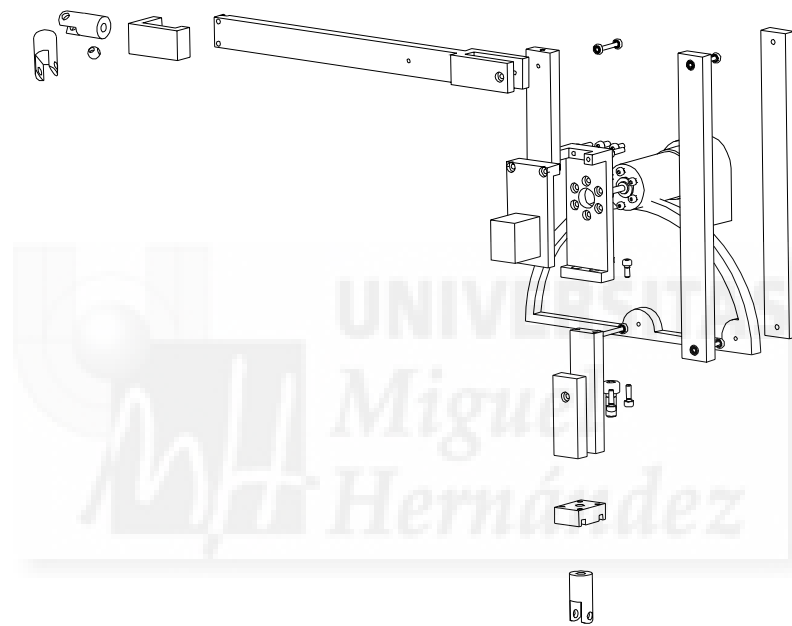
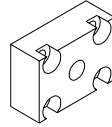
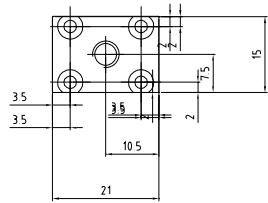
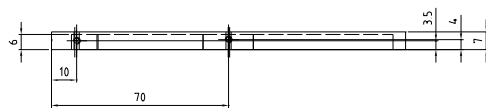
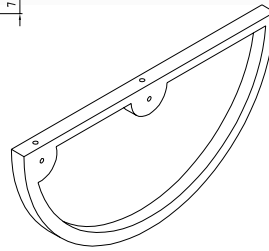
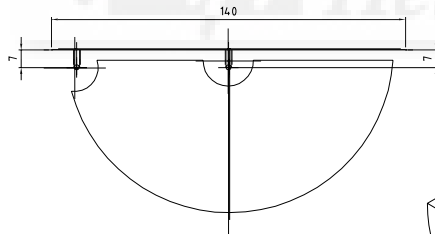


Figura A.3: Despiece de un brazo

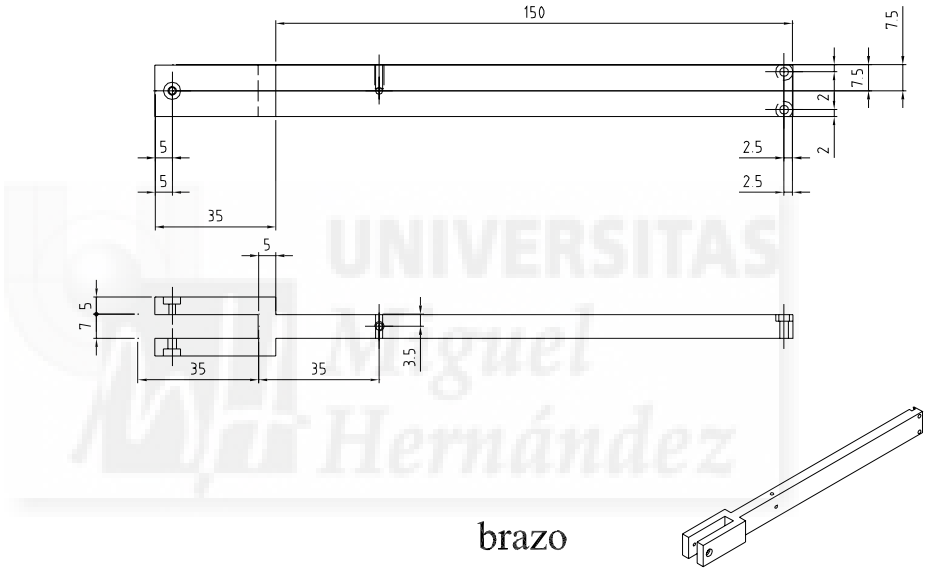




Base\_brazo



Disco



brazo

# Apéndice B

## Conexiones Eléctricas del MAGISTER-P

---

El presente anexo muestra las conexiones eléctricas realizadas en el interfaz háptico MAGISTER-P.

### B.1. Conexiones eléctricas motor EM2342-012CR/LC3002

Tabla B.1: Características motor-driver

Tensión nominal	12 volt
Resistencia	1.90 $\Omega$
Potencia de salida	17.00 W
Eficiencia	80 %
Velocidad en vacio	8100 rpm
Intensidad de vacio	0.075A
Par estático	80.0 mNm
Par de fricción	1.00 mNm

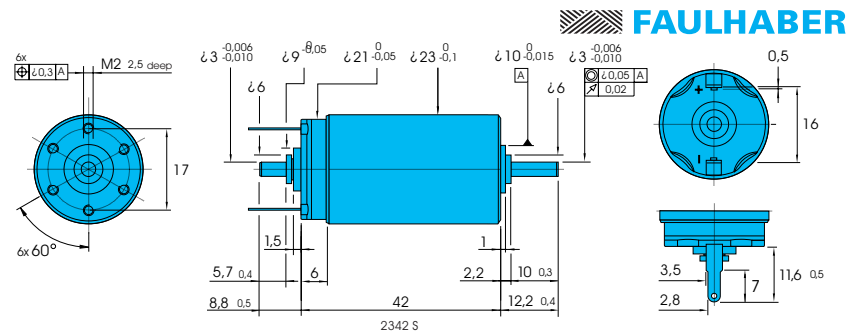


Figura B.1: Dimensiones de los motores EM2342-012CR

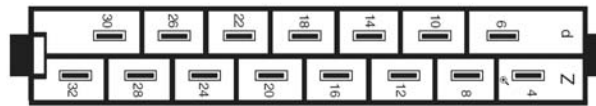


Figura B.2: Conector DIN41612

## B.2. Conexiones LC3002+encoders HP / dSpace ds1103

La información para realizar las conexiones con la tarjeta dspace se encuentra en los manuales de instalación proporcionados por dSpace GMBH.

### *Pines de los encoders HP 5500*

Para una mejor lectura de la señal proporcionada por los encoders, se recomienda colocar una resistencia de pull-up de valor 2.7 k entre los pines Vcc-channel B, Vcc-channel A, y Vcc-channel I.

### *Conexiones de la tarjeta DS1103 PPC*

En las tablas siguientes se muestran solo las señales utilizadas de la tarjeta DS1103. Las tablas están ordenadas de la siguiente manera:

- En la primera columna figura el conector Sub-D y el pin de la tarjeta DSPACE.

- La segunda columna (Señal), muestra el nombre que se le da a la señal en la documentación de la tarjeta DSPACE1103.
- La tercera columna relaciona el pin del conector Sub-D de la tarjeta DSPACE con el pin del conector DB50 colocado en la base del Magíster-P.
- En la cuarta columna figura la señal. a la que se conectó el pin.
- En la quinta columna aparece un breve comentario de la señal conectada.

***P1 Conector analógico***

Tabla B.2: Conexiones analógicas

conector Sub-D	Señal	Conector DB50	Conectado a:	Descripción:
P1B 41	GND	P1B 23	Pin 14 del driver 1	Comando motor 1
P1A 41	GND	P1A 23	Pin 14 del driver 2	
P1B 25	DACH1	P1B 24	Pin 16 driver 1	Comando motor 1
P1A 25	DACH2	P1A 24	Pin 16 driver 2	Comando motor 2
P1B 9	GND	P1B 25	Pin 14 del driver 3	Comando motor 3
P1A 9	GND	P1A 25	Pin 14 del driver 4	
P1B 42	DACH3	P1B 26	Pin 16 driver 3	Comando motor 3
P1A 42	DACH4	P1A 26	Pin 16 driver 4	Comando motor 4
P1B 26	GND	P1B 27	Pin 14 del driver 5	Comando motor 5
P1A 26	GND	P1A 27	Pin 14 del driver 6	
P1B 10	DACH5	P1B 28	Pin 16 driver 5	Comando motor 5
P1A 10	DACH6	P1A 28	Pin 16 driver 6	Comando motor 6

***P2 Conector digital***

No hay conexiones en el conector digital

***P3 Conector digital/Encoders***

Tabla B.3: Conexiones encoders

conector Sub-D	Señal	Conector DB50	Conectado a:	Descripción:
P3B 23	GND	P3B 18	Pin 1 encoder 6	GND.
P3B 7	PHI0(6)	P3B 19	Pin 3 encoder 6	Canal A encoder 6
P3B 40	PHI90(6)	P3B 20	Pin 5 encoder 6	Canal B encoder 6
P3B 24	IDX(6)	P3B 21	Pin 2 encoder 6	Canal I encoder 6
P3B 8	GND	P3B 22	Pin 1 encoder 1	GND.
P3B 41	PHI0(1)	P3B 23	Pin 3 encoder 1	Canal A encoder 1
P3B 25	PHI90(1)	P3B 24	Pin 5 encoder 1	Canal B encoder 1
P3B 9	IDX(1)	P3B 25	Pin 2 encoder 1	Canal I encoder 1
P3B 42	GND	P3B 26	Pin 1 encoder 2	GND
P3B 26	PHI0(2)	P3B 27	Pin 3 encoder 2	Canal A encoder 2
P3B 10	PHI90(2)	P3B 28	Pin 5 encoder 2	Canal B encoder 2
P3B 43	IDX(2)	P3B 29	Pin 2 encoder 2	Canal I encoder 2
P3B 27	GND	P3B 30	Pin 1 encoder 3	GND
P3B 11	PHI0(3)	P3B 31	Pin 3 encoder 3	Canal A encoder 3
P3B 44	PHI90(3)	P3B 32	Pin 5 encoder 3	Canal B encoder 3
P3B 28	IDX(3)	P3B 33	Pin 2 encoder 3	Canal I encoder 3
P3B 12	GND	P3B 34	Pin 1 encoder 4	GND
P3B 45	PHI0(4)	P3B 35	Pin 3 encoder 4	Canal A encoder 4
P3B 29	PHI90(4)	P3B 36	Pin 5 encoder 4	Canal B encoder 4
P3B 13	IDX(4)	P3B 37	Pin 2 encoder 4	Canal I encoder 4
P3B 46	GND	P3B 38	Pin 1 encoder 5	GND
P3B 30	PHI0(5)	P3B 39	Pin 3 encoder 5	Canal A encoder 5
P3B 14	PHI90(5)	P3B 40	Pin 5 encoder 5	Canal B encoder 5
P3B 47	IDX(5)	P3B 41	Pin 2 encoder 5	Canal I encoder 5
P3B 16	VCC2	P3B 46	Pin 4 encoder 1	Vcc encoder 1
P3A 16	VCC2	P3A 46	Pin 4 encoder 3	Vcc encoder 3
P3B 49	VCC2	P3B 47	Pin 4 encoder 5	Vcc encoder 5
P3B 33	VCC3	P3B 48	Pin 4 encoder 2	Vcc encoder 2
P3A 33	VCC3	P3A 48	Pin 4 encoder 4	Vcc encoder 4
P3B 17	VCC3	P3B 49	Pin 4 encoder 6	Vcc encoder 6



# BIBLIOGRAFÍA

---





# Bibliografía

- [1] [Adachi 1994] Adachi, Y. 1994 “ Development of a Haptic Interface for Virtual Reality” 2<sup>nd</sup> Japan-France Congress on Mechatronics, Takamatsu, Japan, 4pp, November
- [2] [Adams 1999] Adams, R.J., “Stable Haptic Interaction with Virtual Environment” PhD. Thesis. University of Washinton 1999
- [3] [Adel-Malek] Karim Abdel-Malek, Wei Yu and Jerry Duncan. “Human Placement for Maximum Dexterity”. Submitted to SAE Digital Human Modeling and Simulation.
- [4] [Adelstein 1987] Adelstein, B. D.,Rosen, M. J. (1987). “A Two-Degree-of-Freedom Loading Manipulandum for the Study of Human Arm Dynamics.” In ASME Winter Annual Meeting 1987 , pp. 111-112 , Boston, Massachusetts
- [5] [Adelstein 1990] Adelstein, B. D., Rosen, M. J. (1990). “A High Performance Two Degree-of-Freedom Kinesthetic Interface.” In N. Durlach (Ed.) In Proceedings of the Engineering Foundation Conference on Human Machine Interfaces for Teleoperators and Virtual Environments , Santa Barbara, CA
- [6] [Adelstein 1992] Adelstein, B. D.,Rosen, M. J. (1992). “Design and Implementation of a Force Reflecting Manipulandum for Manual Control Research.” In ASME Winter Annual Meeting 1992 , Anaheim, CA
- [7] [Albus 1989] Albus James S., Lumia Ronald, y McCain H. G.. “NASA/NBS Standard Reference Model for Telerobot Control System Architecture (NASREM)“. Departamento de Comercio de EE.UU1989

- [8] [Ali Navhi 1998] Ali Navhi, D. Nelson, J.Hollerbach D. Jonhson. "Haptic manipulation of Virtual Mechanism form Mechanical CAD designs" Proceedings of the 1998 IEEE International Conference of Robotics and Automation.
- [9] [Almonacid 2002] Almonacid, M. "Modelado, simulación y control de movimiento de robots paralelos trepadores". PhD. Thesis. UMH 2002
- [10] [Ait-Ahmed 1993] Ait-Ahmed M. "Contribution à la modélisation géométrique et dynamique des robots parallèles". Ph.D. Thesis, Université Paul Sabatier, Toulouse, 2 Febrero 1993
- [11] [Anderson 1989] R.J. Anderson y M.W. Spong, "Bilateral Control of Teleoperators with Time Delay", IEEE Transactions on Automatic Control, Vol. 34, No. 5, Mayo 1989
- [12] [Anderson 1995] R.J. Anderson, "SMART: A Modular Control Architecture for Telerobotics", IEEE Robotics and Automation Magazine, Vol. 2, No. 3, Septiembre, 1995
- [13] [Ang 1998] Lim Ser Yong, Wong Hong Yang and Marcelo Ang. "Robot task execution with telepresence using virtual reality technology". 1998 International Conference on Mechatronic Technology.
- [14] [Angeles 1992] Angeles J. and Zanganeh K.E. "The semi-graphical solution of the direct kinematics of general platform manipulators". In ISRAM, pages 45-52, Santa-Fe, 11-13 Noviembre 1992
- [15] [Aracil 1995] Aracil R., Ferre M., Peñín L. F., y Barrientos A. "Robtet: a new Teleoperated System for Live-Line Maintenance". The 7th International Conference on Transmission and Distribution Construction and Live Line Maintenance. ESMO 95. 1995
- [16] [Aracil 1996] Aracil R., Peñín L. F., Ferre M., Barrientos A., Tuduri A., y Val F. "ROBTET: Robot on Live-Line Maintenance". International Conference on Live Maintenance. 1996
- [17] [Aracil 1999] Aracil,R. Saltaren,R. Sabater,JM. "TREPA, Parallel climbing robot for maintenance of palm trees and large structures" 2nd International Workshop and Conference on Climbing and Walking Robots, CLAWAR'99 ISBN 186058-207-9

- [18] [Aracil 1999b] Barrientos,A;Luengo,O;Mora,A; Aracil,R. “Teleoperated Backhoe excavator with haptic control” 16th. IAARC/IFAC/IEEE International Symposium on Automation and Robotics in Construction IS-ARC’99. Madrid September 22-24, 1999
- [19] [Arai 1990] Arai T., Cleary K., and others . “Design, analysis and construction of a prototype parallel link manipulator”. In IEEE Int. Workshop on Intelligent Robots and Systems (IROS), volume 1, pages 205-212, Ibaraki, Japan, 3-6 Julio 1990
- [20] [Arai 1993] Arai T., Stoughton R., and Jaya Y.M. “Micro hand module using parallel link mechanism”. In Japan/USA Symp. on Flexible Automation, pages 163-168, San Francisco, 13-1 Julio 1993
- [21] [Arcara 2001] Arcara, P., “Control of haptic and robotic telemanipulation systems”. PhD. Thesis. Università degli studi di bologna. 2001
- [22] [Aristoteles] Aristoteles: “De Anima in Focus”. Ed. Michael Durrant, New York, Routledge, 1993.
- [23] [Armstrong 1986] Armstrong,b. Khatib,O. Burdick,J. “The explicit dynamic model and inertial parameters of the PUMA 560 Arm”. Proceedings IEEE International Conference on Robotics and Automation, 1986, pp. 510-518.
- [24] [Argyris 1982] Argyris, J. H. and Mlejnek, H. P. Dynamics of Structures. North-Holland, Amsterdam, 1991 ISBN 0-444-89045-9
- [25] [Atkinson 1977] Atkinson, W. D., Bond, K. E., Tribble, G. L., III, Wilson, K. R. (1977). “Computing with Feeling.” Comput. Graphics 2 , pp. 97-103
- [26] [ATR 1989] ATR (1989). “ATR Communication Systems Artificial Intelligence Department 1989 Annual Report”. (Brochure/Research Summary) , Advanced Telecommunications Research Institute International , Japan
- [27] [Azorin 2001] J.M. Azorin, R. Aracil, J.M. Sabater, C. Fernandez “Bilateral control of teleoperators with time-delay: stability analysis” IFAC Workshop on Internet Based Control Education IBCE 2001

- [28] [Bardofer 2000] Bardofer, A., Munih, M., “Connecting a Haptic Interface with a Robot”. Proceedings of the 10th mediteranean electrotechnical conference. MELECON 2000.
- [29] [Barrientos 1995] A. Barrientos, L.F. Peñín y J. Carrera, “Automatización de la fabricación. Autómatas programables, actuadores y transductores”. Sección de publicaciones de la ETSIIM, Madrid, 1995.
- [30] [Barrientos 1997] A. Barrientos, L.F. Peñín, C. Balaguer y R. Aracil, “Fundamentos de robótica”, McGraw-Hill, Madrid, 1997
- [31] [Basdogan 1997] Basdogan, C., Ho, C., Srinivasan, M.A. (1997) “A for Displaying Shape and Texture of 3D Objects in Virtual Environments”, Proceedings of the ASME Dynamic Systems and Control Division, Vol. 61, pp. 77-84.
- [32] [Basdogan 1998] Basdogan, C. “The role of Haptic Communication in Shared Virtual Environments”. PUG 98 Phantom Users Group.
- [33] [Baumgarte 1972] Baumgarte, J. “Stabilization of Constrains and Integrals of Motion”, Computer Methods in Applied Mechanics and Engineering, Vol. 1 pp. 1-16, 1972
- [34] [Bejczy 1990] A. Bejczy, W. Kim, and S. Venema, “The phantom robot: Predictive display for teleoperation with time delay” in Proc IEEE Int. Conf. Robot Automat. 1990 pp.546-551
- [35] [Bergamasco 1992] Bergamasco, M. (1992). “Design of Hand Force Feedback Systems for Glove-like Advanced Interfaces.” In Proceedings of the 1992 IEEE Workshop on Robot and Human Communication , pp. 286-293 , Tokyo, Japan
- [36] [Blackmon 1996] Blackmon and Stark, “Model based supervisory control in telerobotics” Presence-Teleop. Virtual Environment, vol 5 n<sup>o</sup>2 pp. 205-223
- [37] [Blattner 1992] Blattner, M. (Ed.) (1992). “Computer Mediated Interaction”. , ACM Press
- [38] [Bolanowski 1988] Bolanowski, S. J. J., Gescheider, G. A., Verrillo, R. T., Checkosky, C. M. (1988). “Four channels mediate themechanical aspects

- of touch.” *The Journal of the Acoustical Society of America* , pp. 1680-1694
- [39] [Bolt 1982] Bolt, R. A. (1982). “I. Virtual manual-graphical space; II: Tactual Simulation. (Concept Paper)” , MIT Architecture Machine Group
- [40] [Bouzit 93] Bouzit, M., Richard and P. Coiffet, 1993 “LPR Dextrous Hand Master Control System” Technical Report, Laboratoire de Robotique de Paris, 21 pp., January
- [41] [Brooks 1977] Brooks, F. P., Jr. (1977). “The Computer “Scientist” as Toolsmith - Studies in Interactive Computer Graphics.” In B.Gilchrist (Ed.) *Information Processing, Proceedings of the IFIP Congress 1977* , pp. 625-634
- [42] [Brooks 1988] Brooks, F. P., Jr. (1988). “Grasping Reality Through Illusion- Interactive Graphics Serving Science.” In *Fifth Conference on Computers and Human Interaction, CHI '88 Proceedings* , pp. 1-11 , Washington, DC
- [43] [Brooks 1990] Brooks, F. P., Jr., Ouh-Young, M., Batter, J. J., Kilpatrick, P. J. (1990). “Project GROPE - Haptic Displays for Scientific Visualization.” *Computer Graphics, Proceedings of SIGGRAPH'90*, Boston, MA
- [44] [Brooks 1986] Brooks, T. L., Bejczy, A. K. (1986). “Survey of Hand Controllers for Teleoperation”. (NASA TECH BRIEF Vol 10, No 1,Item 88) , NASA JPL
- [45] [Brown 1994] Brown, J. M., Colgate, J. E. (1994). “Physics-Based Approach to Haptic Display.” In *Proceedings of ISMCR 94, Topical Workshop on Virtual Reality* , Houston, TX
- [46] [Burdea 1991] Burdea, G., Zhuang, J. (1991). “Dextrous telerobotics with force feedback - an overview. Part 1: Human factors.” *Robotica* 9 , pp. 171-178
- [47] [Burdea 1994] Burdea, G. and Coiffet, P. 1994. “Virtual Reality Technology” . New York: John Wiley and Sons, Inc

- [48] [Burdea 1996] Burdea, G. "Force and Touch Feedback for Virtual Reality". Ed John Wiley and Sons, inc.
- [49] [Burdea 1999] Burdea, G. "Invited Review: The synergy between Virtual Reality and Robotics". (invited paper). IEEE Transactions on Robotics and Automation. Vol 15, nº3, June 1999
- [50] [Buttolo 1994] Buttolo, P., Braathen, P., Hannaford, B. (1994). "Sliding Control of Force Reflecting Teleoperation: Preliminary Studies." PRESENCE
- [51] [Buttolo 1996] Pietro Buttolo, D. Kung, B. Hannaford "Manipulation in Real, Virtual and Remote Environments" Proceedings IEEE Conference on System, Man and Cybernetics, Vol 5 pp. 4656-61.
- [52] [Caldwell 1995] Caldwell, D. G. Medrano-Cerda and C. Bowler, 1995 "Pneumatic Muscle Actuators al Low Weight Drive Units in a Robotic Exoskeleton" Technical Report, University of Salford, Manchester
- [53] [Cauchy 1813] Cauchy A. "Deuxième mémoire sur les polygones et les polyèdres". Journal de l'Ecole Polytechnique, pages 87-98, Mayo 1813
- [54] [Charles 1985] Charles J., y Vertut J. "Cable Controlled Deep Submergence Teleoperator System. Teleoperated Robotics in Hostile Environments". Society of Manufacturing Engineers. 1985
- [55] [Cholewiak 1991] Cholewiak, R and A. Collins "Sensory and Physiological Bases of Touch" in M.Heller and W.Schiff Eds., The Psychology of Touch, Lawrence Erlbaum Associates.
- [56] [Chua 1987] L.O. Chua, C.A. Desoer y E.S Kuh, "Linear and Nonlinear Circuits", McGraw-Hill, New York, 1987.
- [57] [Cohen 1995] J.Cohen, M.Lin, D.Manocha and M.Ponamgi. "I-collide: An interactive and exact collision detection system for large-scale environments" In Proc. of ACM Interactive 3D Graphics Conf. Pages 189-196, 1995
- [58] [Coiffet 1993] Coiffet, P., M. Bouzit and G.Burdea, 1993. "The LPR Dextrous Hand Master" VR Systems Fall 93 Conference, Sig Advanced Applications, New York City, October



- [59] [Colgate 1988] Colgate, J. E., Hogan, N.(1988). "Robust control of dynamically interacting systems." *Int. Journal Control* , pp. 65-88
- [60] [Colgate 1993] Colgate, J. E., Grafing, P. E., Stanley, M. C., Schenkel, G. (1993). "Implementation of Stiff Virtual Walls in Force-Reflecting Interfaces." In *VRAIS '93* , pp. 202-208 , Seattle, WA
- [61] [Colgate 1994a] Colgate, J. E., Brown, J. M. (1994). "Factors Affecting the Z-Width of a Haptic Interface." In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation* , pp. 3205-3210
- [62] [Colgate 1994b] Colgate, J. E., Schenkel, G. G. (1994). "Passivity of a Class of Sampled-Data Systems: Application to Haptic Interfaces." In *Proceedings of the American Control Conference*
- [63] [Colgate 1997] Colgater, J.E., Brown,J.M. "Passive implementation of multibody simulations for haptic display". In *proceedings of the 1997 ASME International Mechanical Engineering Congress and Exhibition*. Vol. DSC-61, pp.85-92
- [64] [Craig 1989] Craig, J.J., "Introduction to Robotics: mechanics and Control" 2nd Edition Ed. Addison-Wesley 1989.
- [65] [Crouch 1977] Crouch Donald S. "Mars Viking Surface Sampler Subsystem". American Nuclear Society 25th Conference on Remote System Technology. 1977
- [66] [Cutt 1991] Cutt, P. S. (1991). "Tactile Feedback Technology". (Report on company's memory metal tactile array; brochure) , TiNi Alloy Company, Oakland, CA
- [67] [Danescu 1994] G. Danescu, M. Dahan, "Prolègomènes n<sup>o</sup> 6" (INRIA), Janvier 1994.
- [68] [Dario 1996] P. Dario, M. Bergamasco, A. M. Sabatini, "Advanced Interfaces for Teleoperated Robotic Surgery", in *ISCAMI 1: Integrated System for the Management and Manipulation of Medical Images*, J. Demongeot A. Sousa Pereira (eds.) Springer-Verlag France, Paris 1991
- [69] [Dasgupta 1994] B. Dasgupta, T.S. Mruthyunjaya, "A canonical formulation of the direct position kinematics for a general 6-6 Stewart platform", *Mechanism and Machine Theory*, 29(6):819-827, April 1994.

- [70] [Denne 1994] Denne Debelopments Ltd. 1994, "PemRAM Type 3300/150" Company brochure, Wimborne, UK.
- [71] [Douady 1991] Douady D. "Contribution à la modélisation des robots parallèles: conception d'un nouveau robot à 3 liaisons et six degrés de liberté". Ph.D. Thesis, Université Paris VI, Paris, 9 Décembre 1991
- [72] [Dspace1103] DS1103 PPC Controller Board. Installation and Configuration Guide. dSpace GmbH 2002
- [73] [DspaceRTlib] Real-Time Interface Implementation Guide. dSpace GmbH 2002
- [74] [DspaceCD] Control Desk Experiment Guide. dSpace GmbH 2002
- [75] [Edin 1993] Edin, B. B., Howe, R., Westling, G., and Cutkowsky, M. 1993. "A physiological method for relaying frictional information to a human teleoperator". Transactions on Systems, Man, and Cybernetics 23 (2), 427-432
- [76] [Ellis 1993] Ellis, R., O. Ismaeil and M. Pipsett, 1993, "Design and Evaluation of a High Performance Prototype Planar Haptic Interface". Proceeding of ASME WAM, DSC-Vol 49, ASME, New York, pp.55-64.
- [77] [Esposito 1997] C. Esposito, D. Sullivan, U. Frank y R. Cibulskis, "Field Applications of Robotics Systems in Hazardous Waste, Site Operations", en: Robotics and Remote Systems for Hazardous Environments, Ed. m. Jamshidi y P.J. Eicker, Prentice-Hall, 1993.
- [78] [Exos 1995] EXOS, I. (1995). The Force. Brochure/Press Kit , Exos, Inc. , Woburn, Massachusetts
- [79] [Ferre 1996] Ferre M., Aracil R., Peñín L. F., y Barrientos "A Multimedia Interface for Teleoperated Robots". Proc. of the SPIE. Telemanipulator and Telepresence Technologies III. Vol. 2901. . 1996.
- [80] [Ferre 1997] Ferre, M.: Diseño de interfaces avanzadas para robots teleoperados. Desarrollo de un entorno de teleoperación con características multimedia. UPM-DISAM. 1997.
- [81] [Ferrel 1967] Ferrel W. R. y Sheridan T. B. "Supervisory control of remote manipulation". IEEE Spectrum 4, N<sup>o</sup> 10. 1967

- [82] [Fink 1997] B. Fink y otros, "Cartesian Controlled Heavy Machines Supported by Advanced Human-Machine-Interfaces", 14th International Symposium on Automation and Robotics in Construction, Pittsburgh, 1997.
- [83] [Flatau 1977] Flatau Carl R. SM-229. "A New Compact Servo Master-Slave Manipulator". American Nuclear Society 26th Conference on Remote System Technology. 1977
- [84] [Fu-Gonzalez-Lee 1987] Fu, K.S, Gonzales, R.C. Lee C.S.G. "Robótica: Control, Visión e inteligencia" McGraw-hill, 1987
- [85] [Gillespie 1992] Gillespie, B. (1992). "Dynamical Modeling of the Grand Piano Action." In International Computer Music Conference (ICMC) Proceedings , pp. 77-80 , San Jose, CA
- [86] [Gillespie 1993] Gillespie, B. (1993). "The Virtual Piano Action: Design and Implementation." In International Computer Music Conference(ICMC) Proceedings , Aarhus, Denmark
- [87] [Goertz 1964] Goertz Raymond C. "Manipulators Systems Development at ANL". Proceedings of the 12th Conference on Remote Systems Technology, ANS. 1964.
- [88] [Greaves 1994] Greaves Anna E. R. "State of the Art in Nuclear Telerobotics : Focus on the Man/Machine.Connection". Proc. of the SPIE, Vol. 2351 : Telemanipulator and Telepresence Technologies. 1994
- [89] [Haug 1989] E.J. Haug, Computer Aided Kinematics and Dynamics Of Mechanical Systems, Allyn and Bacon, 1989.
- [90] [Hannaford 1989] Hannaford, B. (1989). "A Design Framework for Teleoperators with Kinesthetic Feedback." IEEE Transactions on Robotics and Automation
- [91] [Hannaford 1989b] Hannaford, B. (1989). "Stability and Performance Tradeoffs in Bi-Lateral Telemanipulation." International Conference on Robotics and Automation, pp. 1764-1767 , Scottsdale, Arizona
- [92] [Hannaford 1993] Hannaford, B., Venema, S. (1993). "Kinesthetic Feedback in Virtual Environments and Teleoperation." In W.Barfield (Ed.) Handbook of Human Factors in Virtual Environments

- [93] [Hayward 1998] Hayward V. and Kurtz R. "Preliminary study of serial-parallel redundant manipulator". In NASA Conference on Space Telerobotics, pages 39-48, Pasadena, 31 Enero 1989
- [94] [Heguy 2001] O.Heguy, N.Rodriguez,H.Luga, J.P.Jessel, and Y.Duthen; Virtual environment for cooperative assistance in teleoperation, in Proc. 9th Int. Conf. in Central Europe on Computer Graphics, Visualization and Computer Vision.
- [95] [Herndon 1984] Herndon J. N., et al "The State of the Art Model M-2 Maintenance System". Proc. of the 1984 National Topical Meeting on Robotics and Remote Systems, of the AMS. . 1984
- [96] [Henri 1994] Henri P. And Hollerbach 1994 "An analytical and Experimental Investigation of a Jet Pipe Controlled Electropneumatic Actuator" IEEE Inter. Conf. On Robotics and Automat. NY pp.300-306
- [97] [Herve 1992] Hervé J.M. "Group mathematics and parallel link mechanisms". In IMACS/SICE Int. Symp. on Robotics, Mechatronics, and Manufacturing Systems, pages 459-464, Kobe, 16-20 Septiembre 1992
- [98] [Hirzinger 1993] Hirzinger G., Brunner B., Dietrich J., y Heindl J. 1993. "Sensor-Based Space Robotics- ROTEX and Its Telerobotic Features". IEEE Transactions on Robotics and Automation. Vol. 9, N<sup>o</sup> 5.
- [99] [Hogan 1985] N. Hogan, "Impedance Control: An approach to manipulation: Part I-Theory", ASME Journal of Dynamic Systems, Measurement and Control, Vol. 107, Marzo 1985
- [100] [Hollerbach 1997] J.Hollerbach ,Ali Navhi, D. Nelson, D. Jonhson. "Haptic interfacing for Virtual prototyping of Mechanical CAD designs" Proceedings of the 1997 ASME design Engineering Technical Conferences.
- [101] [Husty 1994] M.L. Husty, "An algorithm for solving the direct kinematic of Stewart-Gough type platforms",in: Rapport de Recherche (research report) TR-CIM-94-7, Universite McGill, Montreal, June 1994.
- [102] [Inoue 1985] Inoue H., Tsusaka Y., and Fukuizumi T. "Parallel manipulator". In Proc. 3rd ISRR, pages 321-327, Gouvieux, France, 7-11 Octubre 1985

- [103] [IRB 340] IRB, “IRB 340 Picker Reference Manual”, ABB Industries
- [104] [Ishii 1993] Ishii, M., Sato, M. (1993). “A 3D Interface Device with Force Feedback: A Virtual Work Space for Pick-and-Place Tasks.” In IEEE (Ed.) VRAIS '93 , pp. 331-335 , IEEE , Seattle
- [105] [Iwata 1990] Iwata, H. (1990). “Artificial Reality with Force-Feedback: Development of Desktop Virtual Space with Compact Master Manipulator.” Computer Graphics, Proceedings of Siggraph '90
- [106] [Iwata 1992] Iwata, H. (1992). “Force Display for Virtual Worlds.” In Proc.of International Symposium of Artificial Reality and Tele-existence 1992
- [107] [Iwata 1993] Iwata, H. (1993). “Pen-based Haptic Virtual Environment.” In VRAIS'93 , pp. 287-292 , Seattle, WA
- [108] [Iwata 1994] Iwata, H., Hiroaki, Y. (1994). “Desktop Force Display. Invited Interactive Demonstration”, in The Edge, at Siggraph'94 , Orlando, FL
- [109] [Iwata 1991] Iwata, H., Nakagawa, T., Nakashima, T. (1991). “Force Display for Presentation of Rigidity of Virtual Objects.” In 7th Symposium on Human Interface October , pp. 55-58 , Kyoto, Japan
- [110] [Jalon 1986] Garcia de Jalon, J., Unda, J., and Avello, A., 1986, Natural Coordinates for the Computer Analysis of Multibody Systems, Computer Methods in Applied Mechanics and Engineering, Vol. 56, pp. 309-327.
- [111] [Jamisola 1999] Jamisola,R. Ang, M.J., Ming, T.,Khatib, O. Yong,S. “Dynamics Identification and Control of an Industrial Robot”. In Proceedings of the Ninth International Conference on Advanced Robotics (ICAR99) 25-27 Oct 1999.
- [112] [Jandura 1994] Jandura, L. And M. Srinivasan “Esperiments on Human Performance in Torque Discrimination and Control” Procc. Of 1994 ASME Winter Annual Meeting.
- [113] [Kraft] Kraft TeleRobotics. <http://www.krafttelerobotics.com/>

- [114] [Kohli 1988] Kohli D., Lee S-H, Tsai K-Y, and Sandor G.N. "Manipulator configurations based on Rotary-Linear (R-L) actuators and their direct and inverse kinematics." *J. of Mechanisms, Transmissions and Automation in Design*, 110:397-404, Décembre 1988
- [115] [Lawrence 1992] D.A. Lawrence, "Designing Teleoperator Architectures for Transparency", *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, Nice, Francia, Mayo 1992.
- [116] [Lazard 1992] D. Lazard, "Stewart platform and Gröbner basis", in: *ARK*, pp. 136–142, Ferrae, 7-9 September 1992.
- [117] [Lazard 1994] D. Lazard, J.P. Merlet, "Stewart platform has 12 configurations", in: *Proceedings of the IEEE Int. Conf. On Robotics and Automation*, pp. 2160–2165, San Diego, May 8-13, 1994.
- [118] [Lee 1993] Lee H-Y. and Roth B. "A closed-form solution of the forward displacement analysis of a class of in-parallel mechanisms". In *IEEE Int. Conf. on Robotics and Automation*, pages 720-724, Atlanta, 2-6 Mayo 1993
- [119] [Lin 1993] M.C. Lin "Efficient Collision Detection for Animation and Robotics" PhD thesis, Department of electrical Engineering and Computer Science, University of California
- [120] [Lorena 1997] S.J. Lorenc, B.E. Handlon y L.E. Bernold, "Development of a Robotic Bridge Maintenance System", *14th International Symposium on Automation and Robotics in Construction*, Pittsburgh, 1997.
- [121] [Llewelyn 1952] Llewelyn, F.B. "Some fundamental properties of transmission systems". *Proc. IRE*, vol 40, pp 271-283, 1952
- [122] [Marconi 1985] Marconi . "The Gadfly manipulator. Research Report 732", Marconi Research Centre, Junio 1985
- [123] [Marconi 1986] Marconi . "Development of the Tetrabot robotic manipulator. Research report", Marconi Research Centre, Junio 1986
- [124] [Martin 1985] Martin H. Lee, y Kuban Daniel P. "Teleoperated Robotics in Hostile Environments". *Robotics International of SME*. 1985

- [125] [Massie 1993] Massie, T. H. (1993). "Design of a Three Degree of Freedom Force-Reflecting Haptic Interface. SB" , Electrical Engineering and Computer Science, MIT
- [126] [Massie 1993b] Massie, T. H., Salisbury, J. K. J. (1993). "Force Reflecting Haptic Interface". Filed Oct 1, 1993. U.S Patentpending
- [127] [Massie 1994] Massie, T. H., Salisbury, K. (1994). "The PHANToM Haptic Interface:A Device for Probing Virtual Objects." Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems , Chicago, IL
- [128] [McCallion 1979] McCallion H. and Pham D.T. "The analysis of a six degrees of freedom work station for mechanized assembly". In Proc. 5th World Congress on Theory of Machines and Mechanisms, pages 611-616, Montréal, Julio 1979
- [129] [Melfa] MELFA. Mitsubishi Melfa Reference Guide. Mitsubishi
- [130] [Merlet 1988] Merlet J-P. "Force-feedback control of parallel manipulators". In IEEE Int. Conf. on Robotics and Automation, Philadelphie, 24-29 Abril 1988.
- [131] [Merlet 1989] Merlet J-P. "Manipulateurs parallèles, 4eme partie : mode d'assemblage et cinématique directe sous forme polynomiale". Research Report 1135, INRIA, Décembre 1989
- [132] [Merlet 1990] Merlet J-P. "An algorithm for the forward kinematics of general 6 d.o.f. parallel manipulators". Research Report 1331, INRIA, Noviembre 1990.
- [133] [Merlet 1991] Merlet J-P. and Gosselin C. "Nouvelle architecture pour un manipulateur parallèle à 6 degrés de liberté". Mechanism and Machine Theory, 26(1):77-90, 1991.
- [134] [Merlet 1992] J.P. Merlet, "Direct kinematics and assembly modes of parallel manipulators", in: International Journal of Robotics Research, 11(2), 150-162, April 1992.
- [135] [Merlet 1993] Merlet J-P. "Closed-form resolution of the direct kinematics of parallel manipulators using extra sensors data". In IEEE Int.

- Conf. on Robotics and Automation, pages 200-204, Atlanta, 2-7 Mayo 1993
- [136] [Merlet 1994] Merlet J-P. and Mouly N. "Espaces de travail et planification de trajectoire des robots parallèles plans". Research Report 2291, INRIA, Abril 1994 .
- [137] [Merlet 1995] Merlet J-P. "Designing a parallel robot for a specific workspace". Research Report 2527, INRIA, Abril 1995 .
- [138] [Milgram 1997] Milgram, P., Ballantyne, J.. "Real World Teleoperation via Virtual Environment Modelling", in Proc. International Conference on Artificial Reality & Tele-existence ICAT 97
- [139] [Millman 1992] Millman, P., Stanley, M., Grafing, P., Colgate, J. E (1992). "A system for the implementation and kinesthetic display of virtual environments." In Proceedings of 1992 SPIE 1833
- [140] [Millman 1994] Millman, P. A., Colgate, J. E. (1994). "Experimental Study of Haptic Information Pickup in Manipulation." In Telemanipulator Technology, Proceedings of 1994 SPIE
- [141] [Morikawa 1992] Morikawa Hajime, Takanashi Nobuaki, y Tagawa Norio. "A Telerobot System for Space Antenna Assembly Experiments". Proc. IEEE/RSJ. International Conference on Intelligent Robots and Systems 1992
- [142] [Mosher 1960] Mosher, R., y Wendel B. "Force reflecting electro-hydraulic servo-manipulator". Electro-Technology 66. 1960
- [143] [Murilo 2001] Murilo G.Coutinho. Dynamic simulations of multibody systems. ISBN 0-387-95192-X. Ed. Springer-Verlag.
- [144] [Nakamura 1989] Nakamura Y. and Ghodoussi M. "Dynamics computation of closed-link robot mechanisms with non redundant and redundant actuators". IEEE Trans. on Robotics and Automation, 5(3):294-302, Junio 1989
- [145] [Nakashima 1992] Nakashima K. and others. "Development of the parallel manipulator". In IMACS/SICE Int. Symp. on Robotics, Mechatronics, and Manufacturing Systems, pages 419-424, Kobe, 16-20 Septiembre 1992



- [146] [Nair 1994] P. Nair, "On the forward kinematics of parallel manipulators", in: *The Int. J. Of Robotics Research*, 13(2):171-188, April 1994.
- [147] [Ñeco 2001] Ñeco, R.P. Técnicas de procesamiento del lenguaje natural para la teleoperación de robots. PhD. Thesis. 2001 UMH
- [148] [Nikravesh 1988] P.E. Nikravesh, *Computer-Aided Analysis of Mechanical Systems*, Prentice Hall, 1988
- [149] [Numerical Recipes 2002] *Numerical Recipes in C*. 2002
- [150] [ODE 2001] Russel Smith; *Open Dynamics Engine v0.03 User Guide*. <http://opende.sf.net> 2001.
- [151] [Ogata] Ogata, K. *Ingeniería de control moderna*. Ed. Prentice-Hall
- [152] [Oonishi] Oonishi, Ken. "The open manipulator system of the MHI PA-10 Robot". Mitsubishi Heavy Industries, Ltd. Japan.
- [153] [Orlandea 1977] Orlandea, N., Chace, M. A., and Calahan, D.A., 1977, "A Sparsity-Oriented Approach to the Dynamic Analysis and Design of Mechanical Systems", Part 1-2, *ASME Journal of Engineering for Industry*, pp. 773-784.
- [154] [Oyama 1993] Oyama, Tsunemoto Tachi and Inoue, "Experimental study on remote manipulation using virtual reality" *Presence-Teleop. Virtual Environ.* vol 2 n<sup>o</sup>2
- [155] [pa10-servos] *Portable General Purpose Intelligent Arm. Servo Driver. Operating Manual- 91-00353 Rev.0*. Mitsubishi Heavy Industries, Ltd. Japan.
- [156] [Parish 1997] D.W. Parish, "Introduction to the standardized teleoperation system (STS) for converting any vehicle to teleoperated control", *ANS 7th Topical Meeting on Robotics and Remote Systems*, Augusta, 1997.
- [157] [Park 1986] Park T., "A Hybrid Constraint Stabilization-Generalized Coordinate Partitioning Method for Machine Dynamics", *Journal of Mechanisms, Transmission, and Automation in Design*, Vol. 108, No. 2, 1986, pp 211-216.

- [158] [Peñin 1998] Peñin, L.F., “Control Bilateral de robots teleoperados. Contribuciones en reflexión de fuerzss”. Tesis Doctoral. DISAM.UPM 1998.
- [159] [Preusche 2000] C. Preusche and G. Hirzinger; Scaling issues for teleoperation, in Proc. PUG 2000 (Phantom Users Group), 2000
- [160] [Raimondi 1992] Raimondi T. y L. Galbiati. “Manipulators Mascot IV used in JET and prospects of enhancement“. Teleoperation : Numerical Simulation and Experimental Validation. 1992
- [161] [Raju 1989] Raju, G.J., Verghese G.C. y Sheridan T.B. “Design issues in 2-port network models of bilateral remote teleoperation” Proceedings of IEEE Conference of Robotics and Automation, 1989
- [162] [RAP] ABB Flexible Automation AB. Factory Ware Interface. S4C Plus.
- [163] [Rapid] ABB Flexible Automation AB. Rapid Reference On-line Manual
- [164] [Reboulet 1991] Reboulet C. and others . “Rapport d’avancement projet VAP, thème 7, phase 3”. Research Report 7743, CNES/DERA, Enero 1991
- [165] [Salisbury 1992] Salisbury, J. K., Srinivasan, M. A. (1992). “Haptic Interfaces.” In V. MIT (Ed.) BBN Report No.7661 Virtual Environment Technology for Training (VETT) , BBN
- [166] [Salisbury 1994] Salisbury, K., Massie, T. (1994). “The PHANToM Haptic Interface.” In Proc. AAAI Spring Symposium Series, Toward Physical Interaction and Manipulation , Stanford,CA
- [167] [Salisbury 1995] Salisbury, K., Brock, D., Massie, T., Swarup, N., Zilles,C. (1995). “Haptic Rendering: Programming Touch Interaction with Virtual Objects.” In Proceedings of 1995 ACM Symposium on Interactive 3D Graphics , Monterey,CA
- [168] [Saltaren 1996] Saltarén, R.J. “Herramientas para el modelado y diseño paramétrico de robots, basadas en la dinámica de sistemas multicuerpos”. Tesis Doctoral. DISAM-UPM 1996.

- [169] [Saltaren 2000] Saltarén,R.J., Sabater,J.M., Azorín, J.M., Almonacid,M. Prácticas de robótica utilizando Matlab Ed. UMH 2000
- [170] [Sarcos] Sarcos Co. <http://www.sarcos.com>
- [171] [Sayers 1998] Craig Sayers, Remote Control Robotics. Springer-Verlag 1998
- [172] [Schroer 1988] Schroer Bernard J. "Telerobotics Issues in Space Application". Robotics and Autonomous Systems, Vol. 4. N<sup>o</sup> 4. 1988.
- [173] [Shabana 1999] Shabana, Dynamics of Multibody systems, second edition, Cambridge University Press, 1999.
- [174] [Sheridan 1987] T.B. Sheridan et. al, "MIT research in telerobotics", en Proceedings Workshop on Space Telerobotics, Vol. 2, JPL Publications, Julio 1987
- [175] [Serna 1989] Bayo E., and Serna, M.A., 1989. Penalty Formulations for the Dynamics Analysis of Elastic Mechanisms. ASME Journal Mechanisms , Transmissions, and Automation in Design, Vol. 111, pp. 312-326
- [176] [Seow 1988] Seow. "Physiology of Touch, Grip and Gait" in J.Webster Ed., Tactile Sensing for Robotics and Medicine, John Wiley and Sons
- [177] [Seguchi 1990] Seguchi Y., Tanaka M., and others . "Dynamic analysis of a truss-type flexible robot arm." JSME Int. J., 33(2):183-190, Octubre 1990
- [178] [Sugimoto 1987] Sugimoto K. "Kinematic and dynamic analysis of parallel manipulators by means of motor algebra". J. of Mechanisms, Transmissions and Automation in Design, 109:3-7, Marzo 1987
- [179] [Stansfield 1992] Stansfield. "Haptic perception with an articulate, sense robot hand". Robotica, 10,(1992) pp. 497-508
- [180] [Srinivasan 1989] Srinivasan, M. A. (1989). "Surface Deflection of Primate Fingertip Under Line Load." Journal of Biomechanics, pp. 343-349
- [181] [Srinivasan 1990] Srinivasan, M. A. (1990). "Encoding of Shape in the Responses of Cutaneous Mechanoreceptors." , Massachusetts Institute of Technology

- [182] [Shimizu 1993] Shimizu, S. S.Ino, M.Sato, T.Odagawa, T.Ixumi, M.Takahashi and T.Ifukube “Development of Actuator Using Metal Hydride for Force Display to Elbow Joint” *Journal of Robotics and Mechatronics*, Vol 5 n<sup>o</sup> 3 pp.220-225
- [183] [Stewart 1965] Stewart D. “A platform with 6 degrees of freedom”. *Proc. of the Institution of mechanical engineers*, 180 (Part 1, 15):371-386, 1965
- [184] [Vertut 1976] Vertut Jean et al. “The MA 23 Bilateral Servomanipulator System”. *American Nuclear Society 24th Conference on Remote System Technology*. . 1976.
- [185] [Vertut 1985] Vertut Jean, y Coiffet Philippe. *Teleoperation and Robotics, Evolution and Development*. Editorial Kogan Page. 1985
- [186] [V+] V+ Languaje Reference Guide. Staubli.
- [187] [Wehage 1982] Wehage, R. A., Haug, E. J. “Generalized Coordinate Partitioning for Dimension Reduction in Analysis of Constrained Dynamic System”, *Journal of Mechanical Design*, Vol. 104, No. 1 pp 247-255, 1982
- [188] [Wernli 1982] Wernli Robert L. *Robotics Undersea. Teleoperated Robotics in Hostile Environments*. 1982
- [189] [Wiker 1989] Wiker, S., Hershkowitz and J.Zik, 1989 “teleoperator Comfort and Psychometric Stability: Criteria for Limiting Master Controller Forces of Operation and Feedback During Telemanipulation” *Procc. Of NASA Conf. on Space Telerobotics*. Vol.1 NASA
- [190] [Yokokohji 1992] Yokokohji, Y. y T. Yoshikawa, “Bilateral Control of Master-Slave Manipulators for Ideal Kinesthetic Coupling”, *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, Nice, Francia 1992
- [191] [Zamanov 1991] Zamanov V.B and Sotirov Z.M. A contribution to the serial and parallel manipulator duality. In *8th World Congress on the Theory of Machine and Mechanisms*, pages 517-520, Prague, 26-31 Agosto 1991