



Universidad de Valladolid



ESCUELA DE INGENIERÍAS
INDUSTRIALES

Máster en Ingeniería Industrial

MÁSTER EN INGENIERÍA INDUSTRIAL

ESCUELA DE INGENIERÍAS INDUSTRIALES
UNIVERSIDAD DE VALLADOLID

TRABAJO FIN DE MÁSTER

**Estrategia de control del robot UR3
para entornos de cirugía robotizados**

Autor: Álvaro Muñoz García
Tutor: Juan Carlos Fraile Marinero

Valladolid, septiembre, 2020

Resumen

La cirugía endoscópica endonasal es una técnica mínimamente invasiva para la extirpación de tumores en la base del cráneo. La presencia de estructuras anatómicas críticas, como la arteria carótida y el nervio óptico, hace que la precisión de los movimientos sea sumamente importante.

La cirugía robótica es una excelente herramienta para aquellas intervenciones en las que la precisión es crítica. En este Trabajo Fin de Máster se propone un sistema robótico basado en el brazo robótico *UR3* de *Universal Robots* utilizando el entorno de trabajo *ROS*.

Se ha desarrollado un sistema de control basado en fuerza que garantice minimizar las fuerzas de contacto que puedan aparecer durante la intervención, validado a través de la simulación de diferentes escenarios de colisión y contacto.

Abstract

Endoscopic endonasal surgery is a minimally invasive technique to remove skull base tumors. Precision is extremely important due to the presence of critical anatomical structures, such as the carotid artery and the optic nerve.

Robotic surgery is an excellent tool for those procedures in which precision is key. In this thesis a robotic system based on the *Universal Robots UR3* robotic arm using the *ROS* framework is proposed.

A force-based control system that ensures the minimization of the contact forces that may appear during the intervention has been developed and validated through the simulation of different collision and contact scenarios.

Palabras clave

Cirugía mínimamente invasiva; Cirugía endoscópica endonasal; Cirugía robótica; ROS; Control de fuerza

Keywords

Minimally invasive surgery; Endoscopic endonasal surgery; Robotic surgery; ROS; Force control

Agradecimientos

Con la redacción de este Trabajo Fin de Máster pongo fin a una etapa de mi vida que empezó desde que tengo recuerdos. Han sido muchos años detrás de una mesa aprendiendo, estudiando, sufriendo y también disfrutando, y parece que de momento, esta etapa ha llegado a su fin.

Querría agradecer a mi tutor Juan Carlos la implicación y preocupación que ha mostrado durante la elaboración de este TFM y durante mi estancia como trabajador en ITAP.

También querría hacer mención especial a mi padre, a mi madre y a mi hermana, con los que he podido compartir los buenos y malos momentos de esta etapa. Gracias también a todos los compañeros y amigos de dentro y fuera de la universidad, que han hecho mucho más fácil este viaje.

Hasta la próxima.

Índice general

Índice general	VII
Índice de figuras	XI
Índice de tablas	XV
1. Introducción y objetivos	1
1.1. Introducción	1
1.2. Objetivos	3
2. Estado del arte	5
2.1. Evolución de técnicas quirúrgicas	5
2.1.1. Cirugía mínimamente invasiva	5
2.1.2. Cirugía endoscópica endonasal	7
2.1.3. Cirugía robotizada	9
2.1.4. Cirugía robotizada aplicada al campo endonasal	15
2.1.5. Conclusiones	21
2.2. Sistemas de control	22
2.2.1. Métodos de control de <i>compliance</i>	22
2.2.1.1. Métodos de control activos directos	24
2.2.1.2. Métodos de control activos indirectos	25
2.2.2. Conclusiones	27
2.3. Sistemas de control. Aplicaciones	29
2.3.1. Campo endonasal	29
2.3.2. Otros campos	31

3. Descripción <i>hardware</i> y <i>software</i>	35
3.1. Hardware	35
3.1.1. Brazo robótico	35
3.1.2. Pinza	37
3.1.3. Sensor de fuerza	37
3.1.4. Intercambiador de herramientas	38
3.1.5. Dispositivos médicos	40
3.1.6. Cráneo y soporte del cráneo	42
3.1.7. Portaherramientas para el endoscopio	43
3.2. Software	46
3.2.1. <i>MoveIt</i> y <i>tf2</i>	46
3.2.2. Paquetes conexión <i>UR3</i>	47
3.2.3. <i>URScript</i>	49
3.2.4. <i>Rviz</i> y <i>URSim</i>	50
4. Sistemas de control para abordaje robótico en el campo endonasal	53
4.1. Introducción	53
4.1.1. Descripción escenario completo	53
4.1.2. Sistemas de control	54
4.2. Sistema de control sin realimentación de fuerza	55
4.2.1. Biblioteca de movimientos	55
4.2.2. Flujo de trabajo y limitaciones	58
4.3. Sistema de control con realimentación de fuerza	61
4.3.1. Biblioteca <i>URScript</i>	61
4.3.2. Descripción sistema de control	63
4.3.3. Sintonización controlador	67
4.3.4. Filtrado señal de fuerza	69
5. Validación sistema de control con realimentación de fuerza	75
5.1. Introducción	75
5.2. Escenarios	77

5.2.1. Escenario 1	77
5.2.2. Escenario 2	81
5.2.3. Escenario 3	84
5.2.4. Escenario 4	87
5.2.5. Escenario 5	90
5.3. Mejora del eje Z	91
5.4. Conclusiones	94
6. Conclusiones y líneas futuras	95
6.1. Conclusiones	95
6.2. Líneas futuras	96
Bibliografía	99
A. Robot UR3	107
A.1. Especificaciones técnicas	108
A.2. Brida de salida de la herramienta	109
B. Sensor de fuerza HEX-E	111
B.1. Especificaciones técnicas	112
B.2. Planos	112

Índice de figuras

1.1. Esquema preliminar del proyecto <i>Craneal</i> [1]	2
2.1. Comparativa entre colecistectomía abierta y colecistectomía laparoscópica [2]	6
2.2. Visión general de las partes implicadas en una EETS. Traducción de [3] . .	7
2.3. Sellado del injerto con pegamento biológico. Traducción de [3]	9
2.4. Diagrama de las diferentes generaciones de sistemas quirúrgicos robotizados. Traducción de [4]	11
2.5. Sistema quirúrgico Da Vinci [5]	13
2.6. Sistemas robóticos de tercera generación	14
2.7. Sistema robótico de cuarta generación <i>PillCam</i> [6]	14
2.8. Ejemplo de nube de puntos y espacio de trabajo. Adaptación de [7]	16
2.9. Espacio de trabajo aproximado a un tronco de cono. Adaptación de [8] . .	16
2.10. Soporte de endoscopio Endocrane de Karl Storz [9]	17
2.11. Endoscope Robot en una sala de operaciones de la Universidad de Brescia. (A) Brazo robótico. (B) Pedal. (C) Soporte del endoscopio anclado a la camilla. El cirujano comprueba su posicionamiento. (D) Vista general del sistema [10]	20
2.12. Métodos de control de <i>compliance</i> . Adaptación de [11]	22
2.13. Dispositivo RCC del fabricante ATI Industrial Automation [12]	23
2.14. Esquema de control fuerza-posición híbrido. Traducción de [11]	25
2.15. Esquema de control fuerza-posición paralelo. Traducción de [11]	26
2.16. Esquema control de impedancia. Traducción de [11]	27
2.17. Esquema control de admitancia. Traducción de [11]	28
2.18. Setup experimental del sistema [13]	30

2.19. Setup experimental del sistema SmartArm [14]	30
2.20. Esquema de control del sistema SmartArm [14]	31
2.21. Esquema conceptual [15]	32
2.22. Esquema de control híbrido posición-fuerza [15]	32
2.23. Esquema de control para lijado de aspas [16]	33
3.1. Brazo robótico UR3 [17]	36
3.2. Pinza 2F-85 de Robotiq [18]	37
3.3. Sensor de fuerza HEX-E del fabricante OnRobot [19]	38
3.4. Sistema Tool Changer de Smartshift Robotics [20]	39
3.5. Intercambiador de herramientas desarrollado en ITAP	40
3.6. Óptica HOPKINS modelo 28132 BA de Karl Storz	40
3.7. Aspirador 649179 B de Karl Storz [21]	41
3.8. Electrodo de coagulación 28164 ED de Karl Storz [21]	41
3.9. Taladro 252682 de Karl Storz [21]	41
3.10. Cureta 28164 KB de Karl Storz [21]	41
3.11. Modelo del cráneo listo para anclaje	42
3.12. Escenario completo utilizando la primera versión del agarre del endoscopio	43
3.13. Prototipo de agarre con orientación cambiada	44
3.14. Singularidad por alineación de muñeca [22]	44
3.15. Segunda revisión del agarre del endoscopio	45
3.16. Escenario completo	45
3.17. Sistemas de referencia del robot UR3	48
3.18. Comparación entre el modelo de simulación y el entorno real	49
3.19. Plugin de MoveIt para Rviz	50
3.20. Capturas del simulador URSim ejecutándose en una máquina virtual . . .	51
4.1. Escenario completo con sistemas de referencia	54
4.2. Flujo de trabajo de alto nivel de MoveIt [23]	58
4.3. Comparación entre el modelo de colisión detallado y reducido	60
4.4. Primera versión del esquema de control con realimentación de fuerza	64

4.5.	Reacciones antes variaciones de fuerza de gran valor	65
4.6.	Aplicación de un umbra de velocidad de 30 mm/s	65
4.7.	Segunda versión del esquema de control con realimentación de fuerza	66
4.8.	Respuesta del controlador P antes una entrada escalón	67
4.9.	Respuesta del controlador PD antes una entrada escalón	68
4.10.	Respuesta ante filtro paso bajo con frecuencia de corte 20 Hz	69
4.11.	Respuesta ante filtro paso bajo con frecuencia de corte 10 Hz	70
4.12.	Respuesta ante filtro paso bajo con frecuencia de corte 5 Hz	70
4.13.	Respuesta ante filtro paso bajo con frecuencia de corte 3 Hz	70
4.14.	Respuesta ante filtro de la media móvil con tamaño de ventana 16	71
4.15.	Respuesta ante filtro de la media móvil con tamaño de ventana 32	71
4.16.	Respuesta ante filtro de la media móvil con tamaño de ventana 64	72
4.17.	Respuesta ante filtro de la media móvil con tamaño de ventana 128	72
4.18.	Versión final del esquema de control con realimentación de fuerza	73
5.1.	Esqueleto de escenarios de colisión	75
5.2.	Escenario montado sobre el modelo del cráneo	76
5.3.	Sección del escenario 1	77
5.4.	Representación de la fuerza	78
5.5.	Escalonamiento de la fuerza y la posición	79
5.6.	Relación entre fuerzas en el eje Y y el eje Z	79
5.7.	Representación de la posición en 3D en el escenario 1	80
5.8.	Sección del escenario 2	81
5.9.	Representación de la fuerza y la posición en el escenario 2	82
5.10.	Representación de la posición en 3D en el escenario 2	83
5.11.	Sección del escenario 3	84
5.12.	Representación de la fuerza y la posición en el escenario 3	85
5.13.	Representación de la posición en 3D en el escenario 3	86
5.14.	Sección del escenario 4	87
5.15.	Representación de la fuerza y la posición en el escenario 4	88
5.16.	Representación de la posición en 3D en el escenario 4	89

5.17. Esquema del escenario 5	90
5.18. Representación de la fuerza y la posición en el escenario 5	90
5.19. Escenario 5 con mejoras en el eje Z y distintos valores K_{FUERZA}	92
5.20. Superposición de la fuerza sin filtrar en la nueva configuración del eje Z	93
A.1. Especificaciones técnicas del robot UR3 [17]	108
A.2. Brida de salida de la herramienta. Todas las unidades están en mm [17]	109
B.1. Especificaciones técnicas del sensor de fuerza HEX-E [24]	112
B.2. Planos del sensor de fuerza HEX-E [24]	112

Índice de tablas

2.1. Resultados del espacio de trabajo a partir de la nube de puntos [7]	15
2.2. Tabla resumen de los prototipos de sistemas robóticos para acceso endonasal	19
3.1. Resolución libre de ruido del sensor de fuerza HEX-E en función del eje de aplicación [24]	38
4.1. Número de triángulos de los modelos del robot	60
4.2. Descripción de comandos de movimiento de URScript	62
4.3. Nivel de ruido y retraso al utilizar filtro paso bajo	71
4.4. Nivel de ruido y retraso al utilizar filtro de la media móvil	72
5.1. Parámetros de control utilizados en los escenarios	76

Capítulo 1

Introducción y objetivos

En este primer capítulo introductorio se presenta la problemática que ha dado pie a la realización del Trabajo Fin de Máster (TFM), el proyecto en el cual se encuentra englobado y los objetivos que se persiguen.

1.1. Introducción

El presente TFM se ha realizado como parte del plan de estudios del Máster en Ingeniería Industrial de la Universidad de Valladolid, en el instituto universitario de investigación ITAP (Instituto de las Tecnologías Avanzadas de la Producción).

El trabajo realizado se encuentra enmarcado dentro del proyecto *Craneal*, financiado por el Ministerio de Economía, Industria y Competitividad con referencia DPI2016-80391-C3-1-R. Se trata de un proyecto en colaboración con la Universidad de Málaga y la Universidad Miguel Hernández de Elche. Una parte del trabajo desarrollado en este TFM se publicó en el congreso "Robot 2019: Fourth Iberian Robotics Conference", en concreto, el *paper* se tituló: "Integration of a Surgical Robotic Co-worker in an Endoscopic Neurosurgical Assistance Platform" [25].

La cirugía endoscópica endonasal es una técnica mínimamente invasiva para intervenciones en la base del cráneo. El material quirúrgico específico, así como el endoscopio, son introducidos a través de las fosas nasales del paciente. Pese a las ventajas que supone este enfoque frente a uno tradicional, como la reducción del tiempo de recuperación y evitar cicatrices en la cara, supone otros retos. La presencia de estructuras anatómicas críticas como la arteria carótida y el nervio óptico hace que la precisión de los movimientos sea sumamente importante.

El proyecto *Craneal* presenta un asistente robótico diseñado especialmente para la neurocirugía de abordaje endonasal. Consta de un brazo robótico teleoperado por el cirujano con realimentación háptica y un nivel de asistencia variable, encargado de portar las he-

herramientas quirúrgicas; otro brazo robótico autónomo guiará el endoscopio por la cavidad nasal. Además se utilizará un intercambiador de herramientas automático que permita el cambio de herramientas de forma rápida y segura. El sistema aportará información acerca de los desplazamiento de la materia cerebral, debido al cambio en la presión interior al practicar el orificio en la base del cráneo. Este fenómeno se conoce como *brain shift*. De esta forma, se libera al cirujano de un número de tareas que le permitirá concentrarse en las maniobras críticas, aportando seguridad al procedimiento. En la Figura 1.1 se muestra un esquema preliminar del sistema robótico.

El proyecto global plantea los siguientes objetivos globales:

- Diseño y desarrollo de un sistema robótico especializado en el abordaje endonasal para neurocirugía
- Diseño y desarrollo de un sistema de navegación neuroquirúrgica para abordaje endonasal
- Diseño y desarrollo de técnicas de cooperación entre el cirujano y el sistema robótico durante el abordaje endonasal
- Integración y evaluación de la eficiencia del sistema robótico completo

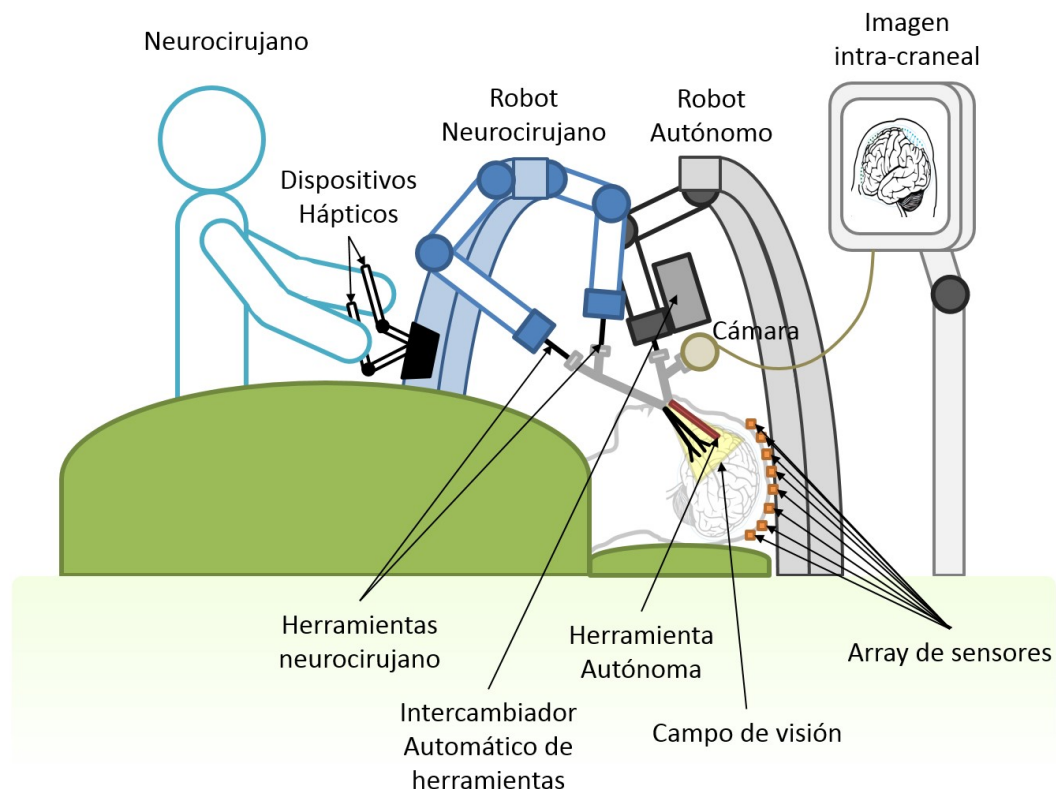


Figura 1.1: Esquema preliminar del proyecto Craneal [1]

El proyecto *Craneal* tiene un alcance muy amplio del cual únicamente se cubrirán las siguientes partes: programación y comunicación con el robot autónomo, diseño de sistemas de control para el robot autónomo en función de la fase de la intervención, desarrollo de un intercambiador de herramientas, así como su interacción con la plataforma robótica y creación de soportes para anclar las herramientas quirúrgicas al brazo robótico.

1.2. Objetivos

Una vez planteados los objetivos generales del proyecto *Craneal*, y habiendo acotado el alcance del TFM, se pueden fijar los siguientes objetivos para el desarrollo del mismo:

- Revistar la literatura para conocer el estado actual de desarrollo de la cirugía robótica en el campo endonasal y los diferentes enfoques existentes para abordar un control de fuerza, tanto en entornos quirúrgicos como industriales
- Desarrollo de primitivas de movimiento ejecutadas en un ordenador remoto bajo el entorno de trabajo *ROS* que permita controlar el brazo robótico *UR3*
- Creación de un entorno de simulación que represente el estado del robot en el escenario de trabajo de forma gráfica
- Diseño de un sistema de control con realimentación de fuerza que se adapte a las condiciones dinámicas de la intervención, minimizando los potenciales daños y aumentando la seguridad
- Validación del sistema de control con realimentación de fuerza mediante la simulación de posibles escenarios de contacto y colisión

El TFM está estructurado de la siguiente manera: en el capítulo 2 se describe el estado del arte de la evolución de las técnicas quirúrgicas y los sistemas de control de *compliance*. En el tercer capítulo se hace una descripción del *hardware* y *software* utilizado para el desarrollo del proyecto. En el capítulo 4 se describen los dos sistemas de control desarrollados, con y sin realimentación de fuerza. En el quinto capítulo se lleva a cabo la validación del sistema de control con realimentación de fuerza y finalmente en el capítulo sexto se resumen las conclusiones del TFM y las líneas futuras de trabajo.

Capítulo 2

Estado del arte

Una vez introducido el tema del TFM, en este segundo capítulo se desarrolla el estado del arte de la evolución de las técnicas quirúrgicas, desde la cirugía mínimamente invasiva hasta la cirugía robótica, centrándose finalmente en el campo endonasal; también se cubrirán los llamados métodos de control de compliance, que se adaptan al entorno en el que se mueve el robot.

2.1. Evolución de técnicas quirúrgicas

Este apartado recoge la aparición de la cirugía mínimamente invasiva desde una perspectiva general, para más adelante centrarse en cirugía endoscópica endonasal. Finalmente se revisará la bibliografía para hablar sobre las plataformas robotizadas que existen en el mercado o bien están en desarrollo, tanto de carácter generalista, como aquellas específicamente diseñadas para el acceso endonasal.

2.1.1. Cirugía mínimamente invasiva

La cirugía mínimamente invasiva (CMI) describe una parte de la cirugía que abarca todas las disciplinas tradicionales. Puede considerarse una filosofía, un modo de pensar. Consiste en la utilización de pequeñas incisiones utilizando sistemas de imagen miniaturizados para reducir el traumatismo de la exposición quirúrgica. El término “cirugía de acceso mínimo” describe mejor la filosofía, utilizar incisiones mínimas para acceder a los lugares dónde se va a operar [26].

Apareció por primera vez en 1987 con un procedimiento llamado colecistectomía laparoscópica, cuyo objetivo es la extirpación de la vesícula biliar [27]. A partir de ese momento, el número de procedimientos fue aumentando gracias a los avances en la tecnología y mejora de la destreza por parte de los cirujanos. En la Figura 2.1 se muestra una comparativa entre un enfoque tradicional de cirugía abierta, frente a un procedimiento de CMI.

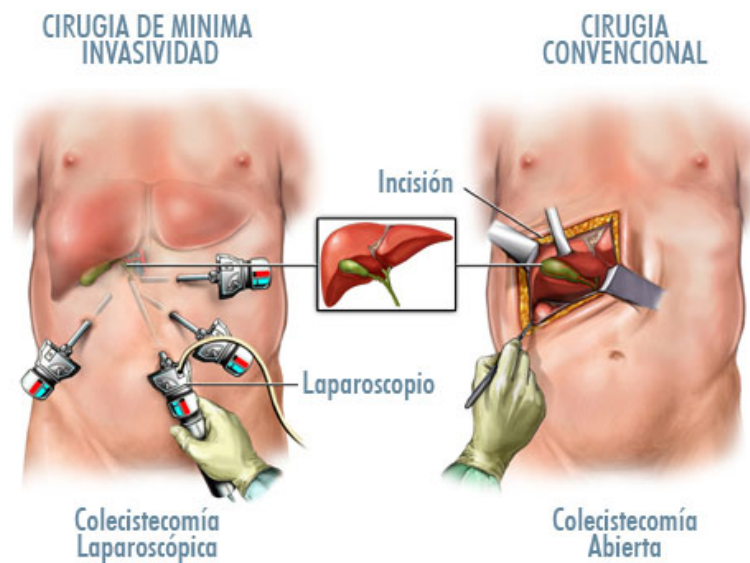


Figura 2.1: Comparativa entre colecistectomía abierta y colecistectomía laparoscópica [2]

Ventajas y limitaciones

A continuación se resumen las ventajas más importantes de la CMI en relación a cirugía abierta:

- El tamaño de las incisiones es menor, con el consecuente beneficio en términos estéticos y trauma en los tejidos sanos
- Posibilidad de evitar anestesia general
- El riesgo de infección es menor
- Disminución del dolor postoperatorio y mejora de la respuesta inmunológica
- Las estancias hospitalarias son menores, habilitando una reincorporación laboral más rápida y morbilidad asociada a la estancia postoperatoria reducida

Pese a las numerosas ventajas de la CMI, también hay que tener en cuenta las siguientes limitaciones, algunas inherentes a la naturaleza del equipamiento utilizado:

- Pérdida del tacto, haciendo que la manipulación de tejidos pase a depender más de la visualización, y no tanto de la realimentación de fuerza
- Pérdida de coordinación ojo-mano
- La visualización de las imágenes proporcionadas por el endoscopio en un monitor en 2 dimensiones supone la pérdida de visión espacial, haciendo que los movimientos sean menos intuitivos, aumentando la dificultad del procedimiento

- Capacidad de movimiento restringida
- Los temblores del cirujano no son amortiguados, sino que se transmiten a lo largo de los instrumentos rígidos [28]

Algunas limitaciones pueden superarse con formación específica, adaptándose a la palpación con instrumentos. Otras limitaciones, como la pérdida de profundidad, se van paliando con avances tecnológicos, como la introducción de endoscopios que proporcionan vídeo en tres dimensiones [29].

2.1.2. Cirugía endoscópica endonasal

En el apartado anterior se han descrito cuáles son las características de la CMI con carácter general. Este apartado se centrará en el abordaje de tumores en la base del cráneo. Los posibles tratamientos de estos tumores pueden consistir en cirugía, radioterapia y/o quimioterapia [30]. En los casos en los que cirugía es necesaria, el enfoque tradicional ha sido la microcirugía, en la cual se practica una incisión bajo el labio superior y se retira una gran parte del tabique nasal, de forma que el cirujano pueda ver directamente el área de acceso al tumor.

La cirugía endoscópica transesfenoidal transnasal es la técnica de CMI cuyo objetivo es la extirpación de tumores en la base del cráneo, de hipófisis o glándula pituitaria. También se conoce por sus siglas en inglés EETS (*Endoscopic endonasal trans-sphenoid surgery*).

Como su nombre indica, transesfenoidal significa a través del seno esfenoidal, que se accede desde los orificios nasales. Para ello, hay que hacer una incisión mínima en la parte posterior de la cavidad nasal. En la Figura 2.2 se muestra una representación del acceso a zona tumoral y las cavidades que atraviesa [3].

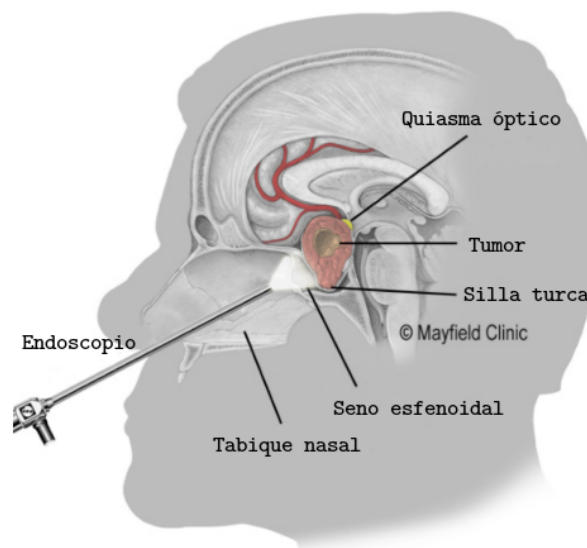


Figura 2.2: Visión general de las partes implicadas en una EETS. Traducción de [3]

Descripción del procedimiento quirúrgico

La técnica que se va a describir es conocida como 2 cirujanos 4 manos, o técnica de las 4 manos. Se introduce el endoscopio por el orificio nasal izquierdo y avanza hasta la parte posterior de la cavidad nasal. Una pequeña parte del tabique nasal es retirada utilizando un taladro a través del orificio derecho, y se abre el acceso al seno esfenoidal. Un cirujano es el encargado del lado izquierdo, por dónde además del endoscopio también se introduce un aspirador quirúrgico, mientras que el otro cirujano utiliza el lado derecho para introducir las diferentes herramientas.

En la parte posterior del seno esfenoidal se encuentra un hueso que recubre la glándula pituitaria, llamado silla turca. Este delgado hueso se retira para exponer el revestimiento del cráneo llamado duramadre. Ésta es abierta, dando acceso al tumor y a la glándula pituitaria.

A través del agujero practicado en la silla turca, el cirujano extirpa el tumor utilizando instrumentos de agarre largos. Una vez se ha extirpado toda la parte visible del tumor, se inspecciona si este ha crecido lateralmente dentro del seno cavernoso. Debido a la presencia de nervios y vasos sanguíneos, la extirpación completa del mismo puede ser complicada sin causar daños, por lo que la masa tumoral restante podrá ser eliminada posteriormente con tratamientos de radiación.

El último paso es cerrar la apertura practicada en la silla turca, utilizando, si fuera posible, un injerto de hueso del tabique nasal. De no serlo, puede sustituirse por un injerto de material sintético. En cualquiera de los casos, se aplica pegamento biológico sobre el injerto en el seno esfenoidal. Este pegamento permite la curación y evita la fuga de líquido cefalorraquídeo del cerebro al seno esfenoidal y a la cavidad nasal [3, 31]. En la Figura 2.3 se muestra una representación de la zona de acceso y zona tumoral una vez ha finalizado la intervención.

Ventajas y limitaciones

La EETS está ganando aceptación como tratamiento de primera línea para la mayoría de los tumores pituitarios. En [32] se realizó una revisión bibliográfica de la que se pueden extraer las siguientes ventajas respecto a un enfoque de microcirugía:

- Menor morbilidad al evitar la craneotomía, la retracción del cerebro y la reducción de la manipulación neurovascular
- Mejora de la calidad de vida en términos de incomodidad y percepción del dolor del paciente
- Tiempo de operación y duración de la estancia en la hospital reducida, con las consecuentes ventajas que esto supone

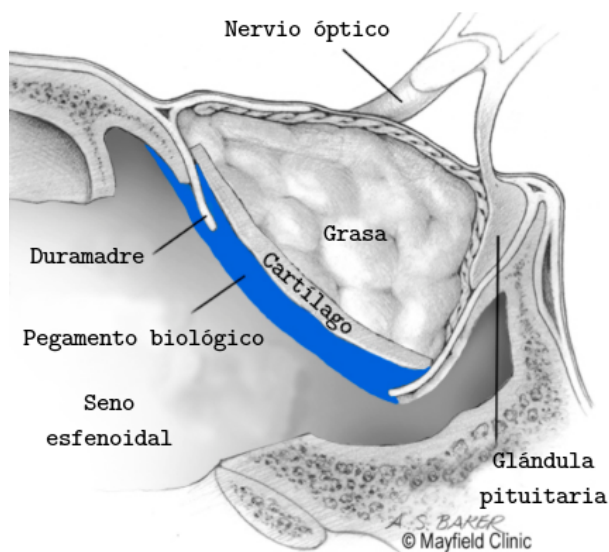


Figura 2.3: Sellado del injerto con pegamento biológico. Traducción de [3]

- Menor pérdida de sangre
- Mejor respuesta visual y endocrinológica
- Mayor resección¹ de tumores

Respecto a las limitaciones encontramos las siguientes:

- La sinusitis se considera una contraindicación para el enfoque transfenoidal debido a la posible propagación transcraneal de la infección
- La ausencia de neumaticidad² del seno esfenoidal puede plantear complicaciones debido a la posición cercana de la arteria carótida
- La transición desde microcirugía a EETS supone una acentuada curva de aprendizaje
- Dependiendo de la posición, consistencia o tamaño del tumor, un enfoque EETS puede no ser practicable

2.1.3. Cirugía robotizada

El nacimiento de la cirugía robótica tuvo lugar en un momento en el que las técnicas de cirugía mínimamente invasiva iban siendo adoptadas por los cirujanos, y se buscaba paliar las limitaciones inherentes a dichas técnicas. Algunos sugieren que la cirugía robótica es la evolución natural de la CMI.

¹Resección se refiere a extirpación quirúrgica

²La neumaticidad es la presencia de cavidades de aire dentro de huesos

En 1985 se introdujo el primer robot aplicado al campo médico, el Puma 200, utilizado para la realización de biopsias cerebrales selectivas. Fue diseñado para mejorar las biopsias tradicionales en términos de precisión. Desde ese momento, el desarrollo de la cirugía robótica no ha dejado de crecer gracias a los avances tecnológicos y sus ventajas técnicas [28].

Ventajas técnicas y obstáculos para su implantación

La cirugía robotizada ofrece las siguientes ventajas técnicas en comparación con la CMI.

- Mejor visualización, con visión estereoscópica con capacidad de magnificación
- Eliminación del temblor de las manos, lo que mejora la precisión
- Mejor maniobrabilidad, que en algunos sistemas permite hasta 7 grados de libertad
- Los movimientos pueden ser escalados, de forma que un movimiento grande del cirujano se transforme en uno pequeño del brazo robótico, mejorando la precisión
- Mejora de la ergonomía del cirujano, reduciendo el estrés y logrando mayores niveles de concentración
- Integración de datos en tiempo real y/o previamente registrados, de modo que se pueda llevar a cabo una compensación de factores intraoperatorios complejos, como el movimiento debido a los latidos del corazón

Naturalmente, también existen desventajas asociadas a estos sistemas, o más bien, obstáculos que dificultan o justifican su implantación:

- Coste. El sistema quirúrgico moderno *Da Vinci* tiene un coste aproximado de 1 800 000 €, al que hay que añadir una cuota anual de 145 000 € y un coste de instrumental por operación de 2 300 €, lo cual es significativamente superior a operaciones que utilicen cirugía abierta o CMI
- Curva de aprendizaje. Aunque los sistemas quirúrgicos robóticos a priori requieren una curva de aprendizaje menor a la CMI, dado que es más intuitiva, y así lo demuestran algunos estudios, no existe suficiente evidencia que lo soporte. Esto es debido a que los cirujanos normalmente ya se encuentran familiarizados con técnicas de CMI antes de utilizar sistemas robóticos, por lo que los resultados pueden estar sesgados
- Limitaciones de espacio y operativas. El espacio del quirófano debe ser suficiente como para dar cabida a las grandes dimensiones de los sistemas actuales; además

del personal médico necesario para una operación habitual, deberá haber personal especializado en la plataforma robótica; tiempo necesario de preparación tanto preoperatoria como postoperatoria del sistema robótico.

- Equipamiento auxiliar y propiedad intelectual. La CMI tiene una gran dependencia de equipo auxiliar que muy probablemente no pueda ser utilizado directamente por el sistema robótico. Por ello, el fabricante deberá desarrollar sus propias herramientas que muy probablemente sólo sean compatibles con sus sistemas robóticos.
- Resultados clínicos y evidencia. La comparación entre CMI y cirugía robótica es ambigua. Aunque algunos estudios muestran ventajas en la duración de la estancia hospitalaria total y pérdida de sangre (además de los beneficios asociados a la mayor precisión), el tiempo de operación, el coste y los tiempo de preparación del quirófano son superiores.

Evolución en el campo de la cirugía robotizada

Desde el primer prototipo hasta los prototipos y desarrollos actuales, ha habido un gran desarrollo tecnológico. En la revisión bibliográfica realizada por [4] distingue 5 generaciones, tal y como se muestra en la Figura 2.4.

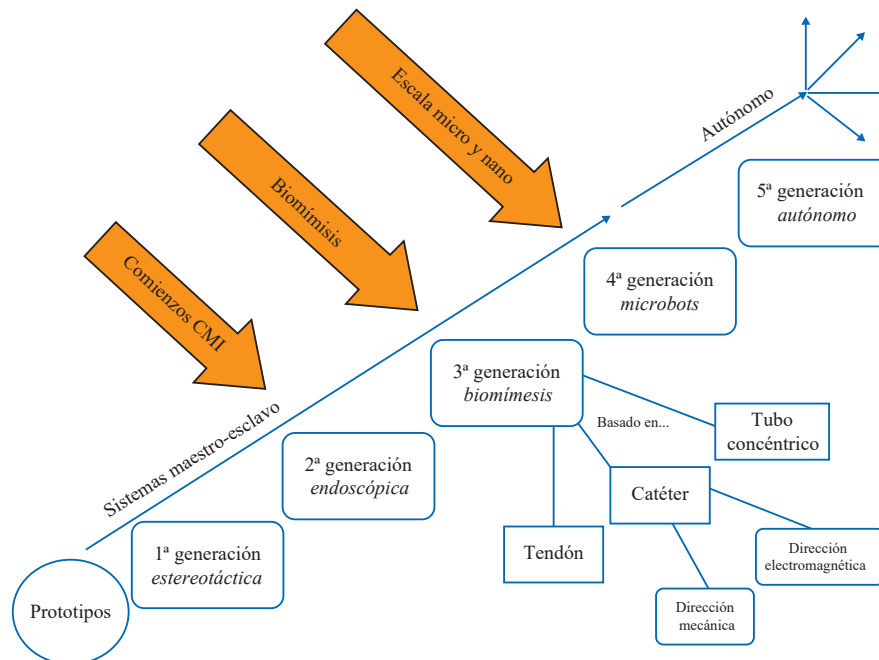


Figura 2.4: Diagrama de las diferentes generaciones de sistemas quirúrgicos robotizados. Traducción de [4]

1ª generación - Estereotáctica. El primer robot quirúrgico utilizado fue el PUMA 200 en 1985 para realizar una biopsia cerebral estereotáctica, en la cual el cirujano situó

los brazos del robot en la posición adecuada para realizar la tarea [33]. En otros campos como la ortopedia, se utilizaron robots para realizar procedimientos que tenían una estrategia claramente matemática y mecánica bien definida, descartando procedimientos que dependieran del tacto o en la que la manipulación de tejidos fuera crítica. Ejemplos de estos robots son el SCARA, ROBODOC y ArcoBot [34].

Las conclusiones de esta primera generación fueron que la supervisión humana era necesaria para cada paso que realizaba el robot, no siendo posible que este realizara tareas secuenciales sin supervisión, utilizando un paradigma maestro-escalvo. La segunda conclusión fue que era necesario aumentar la destreza y la flexibilidad en la manipulación de tejidos para poder extender las aplicaciones de la cirugía robótica.

2ª generación - Endoscópica. La segunda generación ha supuesto la mayor evolución y expansión de los robots quirúrgicos. Esta generación fue el resultado de la introducción de PROBOT de *Imperial College* [35], que tenía la capacidad de manipular tejidos blandos. En esta época ya se había establecido la CMI, y estos sistemas daban solución a los siguientes problemas: dificultad de acceso a zonas con restricciones anatómicas (como la pelvis o la cavidad torácica) en las cuales la fuerza a ejercer era muy elevada, falta de precisión en algunas tareas, dificultades debido a la visualización en dos dimensiones y falta de tacto al manipular tejidos.

Durante esta generación se desarrollaron los dos sistemas robóticos más conocidos de forma simultánea. El sistema robótico *Zeus (Computer Motion)*, disponible comercialmente a partir de 1998, seguido por el sistema robótico *Da Vinci (Intuitive Surgical Inc)* en el año 2000. Ambos sistemas fueron capaces de realizar una cirugía de *bypass* de la arteria coronaria como prueba de concepto.

En 2001 se realizó con el sistema *Zeus* la primera teleoperación trasatlántica, dónde el cirujano se encontraba en Nueva York, EEUU y el paciente en Estrasburgo, Francia [36].

En 2003 *Intuitive Surgical* adquirió *Computer Motion*. A partir de entonces el único sistema disponible comercialmente fue el *Da Vinci*, que en la actualidad sigue en utilización y continuo desarrollo. En la Figura 2.5 se muestra el sistema completo.

Aunque existen otras plataformas robóticas, este último ha permanecido como líder del mercado, ofreciendo las siguientes características:

- Entorno cómodo y puesto de operación ergonómico
- Imágenes en 3D que ofrecen la percepción de profundidad
- Controles intuitivos (coordinación ojo-mano)
- Anulación de temblores fisiológicos



(a) Consola del cirujano



(b) Carro del paciente



(c) Carro de visión

Figura 2.5: Sistema quirúrgico Da Vinci [5]

- Movimiento de las muñecas del robot con 7 grados de libertad, habilitando la realización de microcirugías complejas
- Acceso a zonas del cuerpo complicadas, como la pelvis, en la que cirugía abierta o MIS dan lugar a fuerzas de torsión excesivas o a la necesidad de realizar grandes incisiones

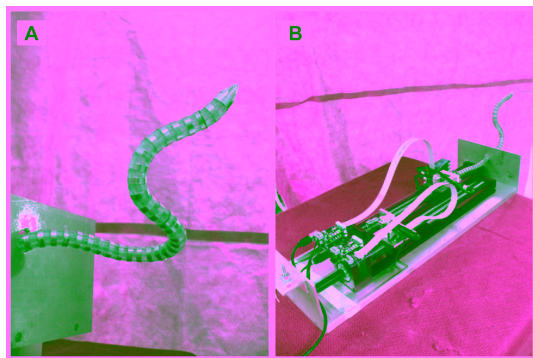
Los sistemas robóticos de segunda generación son los más extendidos y asentados en el uso clínico actual.

3ª generación - Biomímesis. Durante el desarrollo de la segunda generación de robots, la CMI comenzó a adoptar tecnologías basadas en la naturaleza, como endoscopios con forma de serpiente o instrumentos con los extremos articulados, para permitir el acceso a zonas complicadas. La tercera generación de robots adoptó estos principios de biomímesis y tecnología con múltiples articulaciones. La mayor parte de estos robots fueron diseñados en la época de los 2000.

Los desarrollos actuales pueden dividirse en dos categorías:

- Sistemas flexibles impulsados por tendones, como el *CardioArm* de *Imperial College* [37]
- Sistemas de navegación por catéteres, que incluyen tanto sistemas de dirección mecánica (como el sistema *Amigo* de *Catheter Robotics* [38]) como dirección electromagnética. Un subgrupo de estos sistemas son dispositivos que utilizan tubos concéntricos para aumentar la capacidad de ejercer fuerza a distancia

En la Figura 2.6 se muestran ejemplos de ambos sistemas.



(a) Sistema *CardioArm* de *Imperial College* [37]



(b) Sistema *Amigo* de *Catheter Robotics* [38]

Figura 2.6: *Sistemas robóticos de tercera generación*

4ª generación - Microbots. El concepto de robot microscópicos ha estado presente desde el comienzo de la cirugía robótica. Estos robots podrían entrar en el cuerpo dejando una huella quirúrgica mínima y trabajar tanto en solitario como en colaboración con un grupo de robots para visualizar y tratar no sólo las enfermedades quirúrgicas convencionales, sino también las no quirúrgicas, como los procesos infecciosos e inmunológicos que podrían ser manejados a nivel celular. Sin embargo, los avances actuales no han conseguido alcanzar esos niveles de funcionalidad, aunque se sigue trabajando en ellos. En 2008, [39] presentó un sistema en el cual se guía una cápsula endoscópica a través del tracto gastrointestinal con la ayuda de un brazo robótico industrial externo utilizando imanes permanentes, tanto en el extremo del robot como en la cápsula. Se consiguió un gran nivel de precisión en pruebas *ex vivo*.

En la actualidad las cápsulas endoscópicas que se encuentran en uso clínico funcionan únicamente como dispositivos de captura de imágenes pasivo. Un ejemplo de este sistema es la *PillCam*, la cual se introduce por vía oral y recorre el intestino delgado transmitiendo las imágenes en tiempo real, careciendo de un sistema de guiado adicional [6], mostrado en la Figura 2.7.



Figura 2.7: *Sistema robótico de cuarta generación PillCam* [6]

5ª generación - Autónomo. Aunque desde la primera generación de robots quirúrgicos se han diseñado con el objetivo de dotarles de un cierto nivel de autonomía, realizando tareas pre-programadas, la idea de robots plenamente autónomos con capacidad crítica y

de tomar decisiones sigue siendo meramente conceptual.

Los avances en *machine learning* podrán ayudar a que estos dispositivos autónomos se conviertan en una realidad en un futuro.

2.1.4. Cirugía robotizada aplicada al campo endonasal

Pese al auge y el desarrollo de la cirugía robótica en las últimas décadas, su avance en el campo de la otorrinolaringología ha sido mucho más lento. Prueba de ello es que a fecha de 2013 el único sistema aprobado para por la FDA³ para operaciones de cuello y cabeza fue el *Da Vinci* [41], y en la actualidad se ha sumado el sistema *Flex Robotic System* (*Medrobotics*) [42].

Este desarrollo más lento se ha debido a las restricciones anatómicas que limitan la capacidad de estos sistemas robóticos. El tamaño de las herramientas, la falta de diseño para el trabajo con tejidos blandos y el gran volumen de trabajo que ocupa, limita la utilización del *Da Vinci* en este campo de aplicación. El sistema *Flex Robotic System* fue diseñado para cirugía robótica transoral, lo cual también presenta limitaciones para la aplicación que nos concierne [43].

Antes de comenzar el diseño de un sistema robótico aplicado al campo endonasal, es interesante conocer cuales son las restricciones geométricas asociadas. En [7] se lleva a cabo un experimento para determinar cual es el espacio de trabajo en una intervención EETS. Con ayuda de un sistema de seguimiento óptico y de adquisición de datos, estudiaron el espacio de trabajo de 70 cadáveres, accediendo tanto por el orificio nasal izquierdo como derecho. La captura de datos comienza en el orificio nasal y terminar en el seno esfenoidal (Figura 2.8). La nube de puntos obtenida fue aproximada al volumen de un cilindro cuyas dimensiones se muestran en la Tabla 2.1.

	Orificio	Mínimo	Máximo	Media	Desviación típica
Diámetro [mm]	Izdo	14.28	25.89	19.32	3.23
	Dcho	13.28	27.26	18.84	3.46
	Ambos	13.28	27.26	19.08	3.32
Longitud [mm]	Izdo	36.01	70.46	53.55	8.39
	Dcho	40.50	70.00	54.24	5.91
	Ambos	36.01	70.46	53.90	7.20

Tabla 2.1: Resultados del espacio de trabajo a partir de la nube de puntos [7]

³La FDA (*Food and Drug Administration*) es la agencia reguladora de alimentos y fármacos en Estados Unidos [40], equivalente a la AEMPS (Agencia Española de Medicamentos y Productos Sanitarios) en España

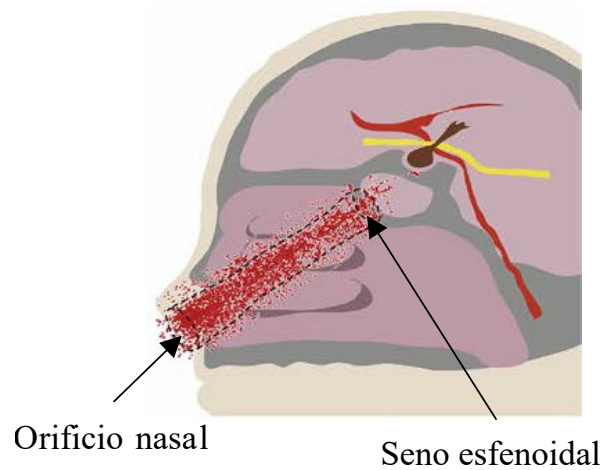


Figura 2.8: Ejemplo de nube de puntos y espacio de trabajo. Adaptación de [7]

La primera limitación de los resultados obtenidos es debida a la utilización de cadáveres conservados a 0 °C sin el uso de formol, por esta razón, el espacio de trabajo es un poco más amplio y los tejidos blandos se encogen, lo cual puede distorsionar los resultados. La segunda limitación se debe a que el origen de la totalidad de los cadáveres era del sureste asiático, pudiendo haber variaciones respecto a otros continentes.

En [8] se realiza un estudio de las necesidades cinemáticas y dinámicas de una interfaz háptica para la simulación de intervenciones EETS. Para la estimación del espacio de trabajo, se apoyan en un estudio de la geometría de la cavidad nasal realizado por [44], y determinan que se puede reducir a un tronco de cono cuyas dimensiones se muestran en la Figura 2.9. En este caso, también comienza en los orificios nasales, aunque finaliza en la glándula pituitaria.

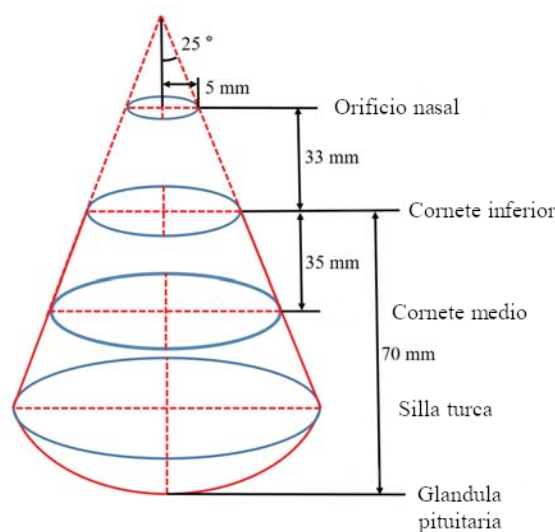


Figura 2.9: Espacio de trabajo aproximado a un tronco de cono. Adaptación de [8]

Soportes endoscópicos mecánicos

En el apartado 2.1.2 se describía la técnica de 4 manos para el acceso endonasal, la cual supone un trabajo conjunto de al menos dos cirujanos, uno de los cuales sostiene el endoscopio y el otro se encarga de la disección quirúrgica, requiriendo de alta coordinación entre ambos. Un primer avance fue la utilización de soportes para el endoscopio no robotizados, mejorando la precisión y permitiendo eliminar los temblores, y consecuentemente, disminuyendo la fatiga. En la Figura 2.10 se muestra el soporte *Endocrane* del fabricante *Karl Storz* [9].



Figura 2.10: Soporte de endoscopio *Endocrane* de *Karl Storz* [9]

En [45] se realizó una encuesta a neurocirujanos en relación al uso de soportes endoscópicos. Los resultados mostraron que la mayoría de los neurocirujanos preferían no utilizar los soportes endoscópicos en intervenciones en la base del cráneo. Los encuestados destacaron las siguientes características positivas y negativas:

- Características positivas: estabilidad, número de articulaciones (grados de libertad), maniobrabilidad y seguridad
- Características negativas: movimientos poco suaves, grados de libertad limitados en comparación a una mano, deriva hacia abajo (debido al propio peso del endoscopio), falta de flexibilidad, caro y voluminoso

Prototipos de sistemas robóticos en el campo endonasal

Para superar las limitaciones de los soportes mecánicos, se desarrollaron los sistemas robóticos para su utilización en la campo endonasal. A continuación, se revisa la biblio-

grafía referente a sistemas robóticos para acceso endonasal que se encuentran en fase de desarrollo, analizando sus principales características.

Se pueden encontrar dos interfaces de control diferentes: telemanipulación (TM) y modo cooperativo (MC). La TM se utiliza para controlar la posición del endoscopio mediante voz, movimientos de cabeza, pedal o *joystick*. El MC es una alternativa en la que el cirujano y el brazo robótico cooperan para sostener el endoscopio. El cirujano puede mover manualmente el instrumento, como si se tratara de una cirugía convencional, y el robot mantiene la posición del endoscopio una vez el cirujano detiene el movimiento.

Los diferentes prototipos se presentan en la Tabla 2.2.

Todos los prototipos resumidos en la Tabla 2.2, tienen un principio de funcionamiento similar basado en un sistema robótico actuado por motores, en cambio, iArmS [58] es la excepción. Se trata de un brazo mecánico articulado, con un encoder, freno y sensor de fuerza alojado en cada articulación, y su objetivo es que el brazo del cirujano pueda reposar sobre éste, disminuyendo el temblor y la fatiga. Las articulaciones son capaces de mantener su posición, así como de liberarse para poder alcanzar una nueva posición. La principal limitación de este sistema es su coste (98.000 \$), lo cual lo hace inaccesible para muchos centros.

En términos generales, los autores coinciden en las limitaciones que tiene un sistema de telemanipulación basado en *joystick*. No sólo hace necesario que tenga que ser controlado por otro cirujano, sino que ambos deben estar muy bien compenetrados. Esto ha dado lugar a la utilización de otros sistemas de telemanipulación, como aquellos basados en pedales, seguimiento de cabeza o control por voz. Otra de las principales limitaciones que comparten son las dimensiones del sistema robótico, lo cual limita los movimientos del cirujano.

A continuación se va a hablar del primer sistema comercial en este campo, el *Endoscope Robot* [59].

Sistema comercial *Endoscope Robot*

El sistema *Endoscope Robot* desarrollado por *Medineering Surgical Robots* [59] se ha puesto recientemente a la disposición para la realización de pruebas clínicas. Se trata de un robot compacto de 7 grados de libertad desarrollado específicamente para trabajar como un soporte endoscópico en operaciones en la base del cráneo. El interfaz de control es un pedal accionado por el cirujano que tiene tres modos de funcionamiento: avance y retroceso en planos coronales, rotación y traslación.

En la Figura 2.11 se muestra el sistema completo.

En diciembre de 2018 se realizó un estudio [60] contando con la participación de 30 cirujanos especializados en la base del cráneo. Se plantearon 2 tareas a realizar sobre un

Autor (Año)	Nombre	Interfaz control	Tarea
Nimsky [46] 2004	EVO1	Joystick (TM)	Soporte endoscopio
Wurm [47] 2005	A73	Autónomo + Joystick (TM)	Soporte endoscopio
Nathan [48] 2006	AESOP	Control por voz (TM)	Soporte endoscopio
Strauss [49] 2007	-	Joystick (TM)	Soporte endoscopio
Xia [13] 2008	-	MC	Soporte endoscopio o taladro
Yoon [50] 2011	-	Doble joystick (TM)	Soporte endoscopio con sistema de flexión activa
Eichhorb [51] 2011	Tx40	Joystick (TM) y seguimiento autónomo	Soporte endoscopio
Swaney [52] 2012	-	Consola cirujano (TM)	Sistema robótico con 2 pinzas, aspirador y endoscopio flexible con fuente de luz
Trevillot [53] 2013	HYBRID	MC	Soporte endoscopio
Schneider [54] 2013	TENTACLE LIKE	Joystick (TM)	Manipulador instrumentos
Cabuk [55] 2015	SP ROBOTIC SYSTEM	Joystick (TM)	Soporte endoscopio
Swaney [56] 2015	-	Interfaz de usuario (TM)	2 brazos robóticos. Pinza y cureta
Chan [57] 2016	FREE	Control por voz (TM)	Soporte endoscopio
Bolzoni [43] 2017	BEAR	Seguimiento cabeza (TM)	Soporte endoscopio
Ogiwara [58] 2017	iArmS	Operado sin motores (MC)	Soporte brazo cirujano
Marinho [14] 2020	SmartArm	Consola cirujano (TM)	2 brazos robóticos. Herramientas y/o endoscopio

Abreviaciones: *MC*, modo cooperativo; *TM*, telemanipulación.

Tabla 2.2: Tabla resumen de los prototipos de sistemas robóticos para acceso endonasal

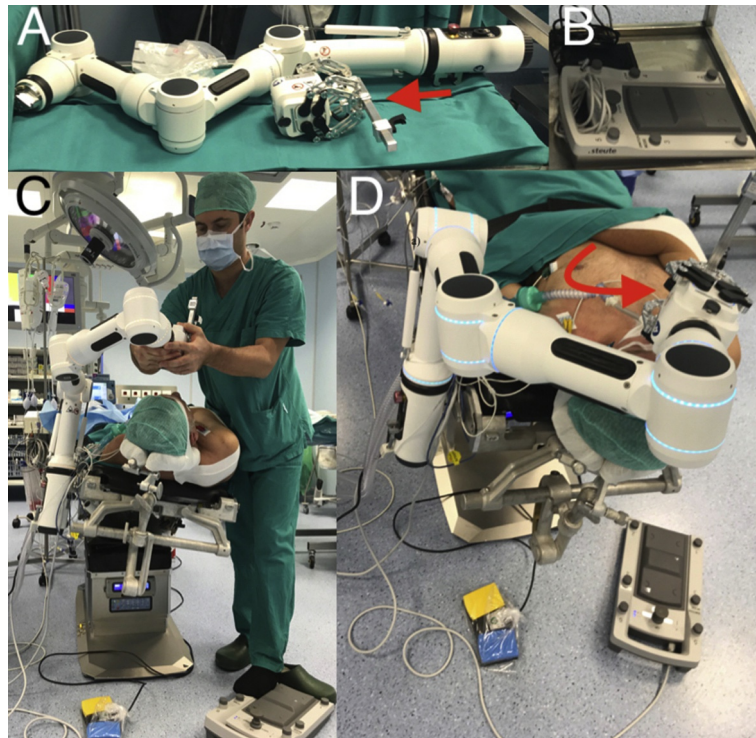


Figura 2.11: *Endoscope Robot en una sala de operaciones de la Universidad de Brescia. (A) Brazo robótico. (B) Pedal. (C) Soporte del endoscopio anclado a la camilla. El cirujano comprueba su posicionamiento. (D) Vista general del sistema [10]*

NeuroEndoTrainer [61] (simulador de la cavidad nasal y base del cráneo de código abierto), utilizando el sistema robótico y sin utilizarlo, para poder comparar los resultados. La primera tarea consistió en una maniobra de agarre, introduciendo el endoscopio por un orificio, y la herramienta de agarre por el otro. La segunda tarea consistió en el posicionamiento de un injerto, para ello, en la fase robótica podían utilizar las dos manos, guiando el endoscopio con el pedal, mientras que en la fase no robótica únicamente podían dedicar una mano a la manipulación del injerto, controlando la otra el endoscopio.

El análisis de los resultados mostraron que la utilización del sistema *Endoscope Robot* en la primera tarea no supuso una ventaja, sino más bien una carga adicional. En cambio, en la tarea segunda permitía liberar una de las manos para poder utilizar una segunda herramienta, mejorando el tiempo de intervención, así como su percepción del estrés y exigencia de la tarea.

Aunque se trate de un estudio preliminar, y sea necesario la realización de más pruebas, muestra que la mayoría de los cirujanos creen que la aplicación de la robótica podría ser beneficioso en la cirugía endoscópica en la base del cráneo, dando solución a las limitaciones actuales de maniobrabilidad del endoscopio [10].

2.1.5. Conclusiones

En la actualidad existen un gran número de desarrollos activos en fase de prototipado. La mayor parte de ellos se centran únicamente en la función de portar y guiar el endoscopio, desarrollando sistemas de telemanipulación lo más intuitivos y menos intrusivos posible. También existen otros desarrollos que incorporan diversos elementos, como el aspirador y la cureta, en busca de un sistema robótico más completo.

Únicamente el sistema *Endoscope Robot* se encuentra en fase de pruebas clínicas a la espera de analizar los resultados.

Se concluye que el campo de la cirugía endonasal robótica no se encuentra en un estado de madurez tan avanzado como por ejemplo la laparoscopia, pero que progresivamente va cobrando más importancia.

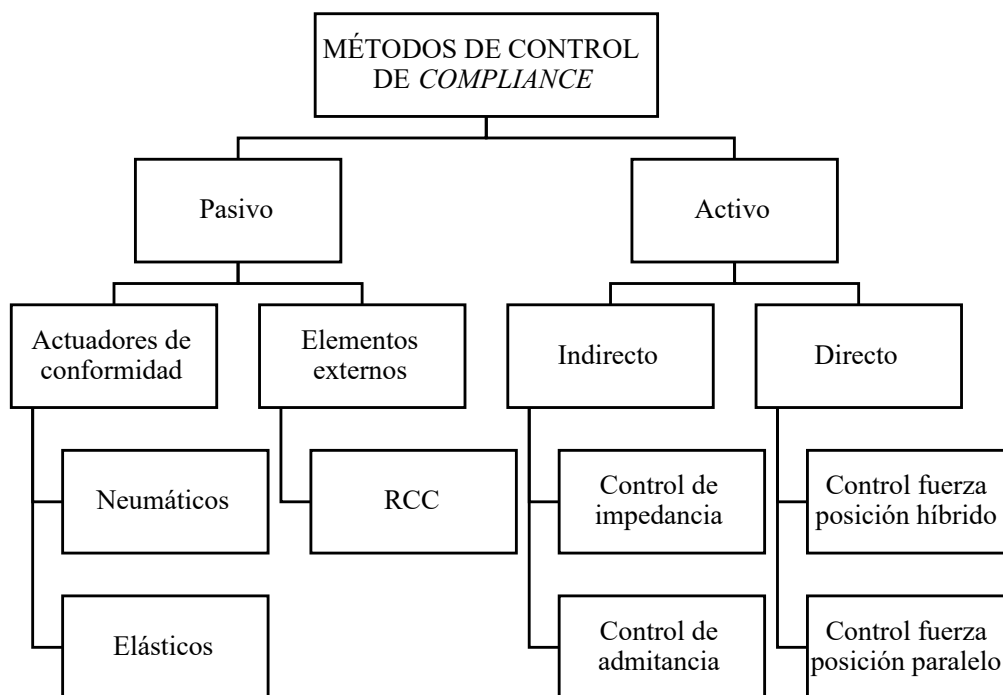
2.2. Sistemas de control

2.2.1. Métodos de control de *compliance*

En un aplicación industrial convencional, los robots ejecutan tareas repetitivas de forma precisa, en las cuales el entorno está estrictamente bajo control. En estos escenarios, el control por posición es la mejor solución, siempre y cuando apenas haya contacto con el entorno.

En cambio, entornos no estructurados, como el que se plantea en una intervención quirúrgica, donde el conocimiento del entorno es impreciso, restringido o cambiante, se requiere algún tipo de control de fuerza para que el robot logre su objetivo de manera segura.

Uno de los características necesarias para un robot robusto es su capacidad de adaptar las fuerzas de interacción cuando manipula un objeto o contacta con una superficie. Este problema es conocido como *compliance control* (podría traducirse al español como control de conformidad o de cumplimiento, aunque ninguna de las traducciones se ha encontrado en la literatura, por lo tanto se designará como control de *compliance*) y trata de garantizar que el robot se adapte a las fuerzas de interacción en lugar de resistirse a ellas. En la Figura 2.12 se muestra una visión general de los métodos clásicos de control de *compliance*.



Abreviaciones: *RCC*: *Remote Center of Compliance*, en español, Centro Remoto de *compliance*

Figura 2.12: Métodos de control de *compliance*. Adaptación de [11]

Pueden dividirse en métodos pasivos o activos:

- Métodos pasivos: incorporan elementos físicos situados entre el robot y el entorno,

encargados de controlar y limitar las fuerzas. En la Figura 2.13 se muestra un ejemplo de dispositivo RCC diseñado para corregir errores de alineación en aplicaciones de ensamblaje automatizadas

- Métodos activos: utilizan técnicas de control de realimentación para controlar la fuerza y el movimiento al entrar en contacto con el entorno



Figura 2.13: Dispositivo RCC del fabricante ATI Industrial Automation [12]

Los métodos activos pueden abordar un rango más amplio de aplicaciones, y su adaptación a diferentes escenarios es más simple y rápida, pese a que su diseño inicial es más complejo. Aunque los métodos activos son ampliamente utilizados en investigación, los entornos industriales suelen optar por métodos pasivos. Pese a su falta de flexibilidad, suelen ofrecer un control más preciso, además, se tiene gran control sobre el entorno para poder adaptarlo a posibles cambios.

El mejor actuador para llevar a cabo un control de fuerza sería una fuente de fuerza ideal, es decir, un actuador que ejerciera exactamente la fuerza comandada independientemente de la carga. Los actuadores reales, naturalmente, no cumplen esta característica. El movimiento de una carga generará fuerzas adicionales en la salida del actuador, llamado impedancia del actuador. Cuanto mayor sea la frecuencia de movimiento de la carga, mayor será esa impedancia. Otro factor importante es el ancho de banda, que especifica la frecuencia máxima hasta la que se pueden medir fuerzas de forma precisa. Un actuador ideal tendría una impedancia nula y un ancho de banda infinito.

Por lo tanto existen dos opciones:

- Utilizar un actuador lo más parecido posible a uno ideal, con baja impedancia y alto ancho de banda
- Utilizar un método de control activo, que permita controlar un motor eléctrico (siendo de por sí una mala fuente de fuerza) de forma que se comporte como un actuador de baja impedancia

En la aplicación que cubre en este TFM se utilizará un método de control activo, debido al bajo o nulo control sobre el entorno y la alta flexibilidad que proporciona. Por ello, a continuación se detallan los diferentes métodos de control activos.

Los métodos de control activos utilizan las medidas de fuerza y posición del robot como realimentación del lazo de control para generar una salida que garantice el comportamiento deseado. La manera en la que los errores de posicionamiento o de fuerza son tratados permiten dividir los métodos de control activo en directos o indirectos:

- Métodos de control directos: el sistema controla directamente la fuerza de contacto en base a una consigna. Por ejemplo, en un sistema de control fuerza-posición híbrido, el usuario define qué direcciones son controladas en posición, y cuáles en fuerza
- Métodos de control indirectos: el sistema controla la fuerza indirectamente a través del posicionamiento. Por ejemplo, en un control de impedancia, el usuario define una relación dinámica entre la fuerza y el movimiento

La implementación práctica de ambos métodos es similar, aunque su principio de funcionamiento es diferente. Los métodos de control activos también se pueden clasificar en función de su comportamiento dinámico o estático:

- Métodos de control basados en un modelo dinámico: son aquellos en los que la respuesta dinámica del sistema tiene importancia. Para ello, es necesario un modelo dinámico completo del robot, por lo que es complejo tanto de diseñar como de implementar
- Métodos de control basados en un modelo estático: la respuesta dinámica del sistema no tiene tanta importancia como su valor estacionario. Estos métodos son más sencillos de implementar, y no requieren la descripción del modelo dinámico del robot

2.2.1.1. Métodos de control activos directos

Control fuerza-posición híbrido

Este método trata el contacto como un problema geométrico, en el que hay un conjunto de restricciones geométricas que se deben tener en cuenta. Se definen algunos grados de libertad como controlados por fuerza, mientras que el resto serán controlados por posición. Este enfoque se basa en el hecho de que la mayoría de las tareas pueden dividirse en dos subespacios independientes, uno en el que se controlan las fuerzas de contacto y otro que controla el movimiento del robot, de forma que el control de fuerza y posición es independiente. Por ejemplo, en una aplicación industrial de lijado se debería controlar en fuerza la dirección perpendicular a la superficie a lijar, y en posición las direcciones paralelas a dicha superficie.

La Figura 2.14 muestra un esquema de control del sistema, en el cual:

- La posición y orientación del efector final se denota por X
- La fuerza f y el momento m en las tres dimensiones ejercido por el efector final, se designan conjuntamente por h .
- Los torques comandados se designan por τ
- El subíndice d indica que se trata de una referencia, tanto de posición como de fuerza

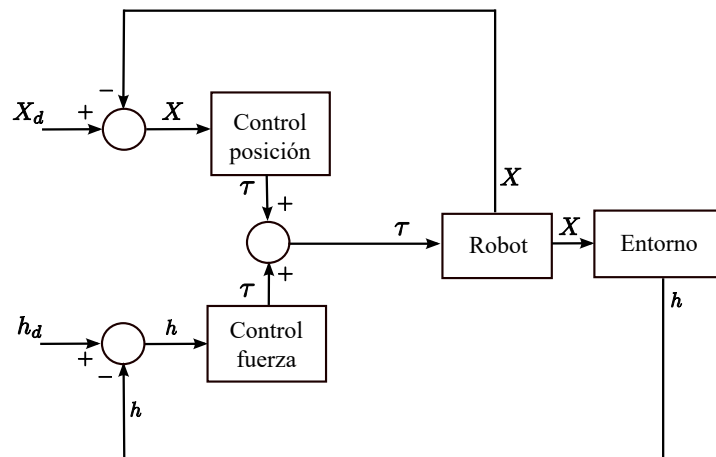


Figura 2.14: Esquema de control fuerza-posición híbrido. Traducción de [11]

Control fuerza-posición paralelo

La diferencia entre el control fuerza-posición híbrido y paralelo, es que éste último no utiliza diferentes subespacios de control para la fuerza y la posición, sino que combina y pondera las contribuciones de cada controlador en uno único. En este caso, el controlador da prioridad a errores de fuerza, dominando la respuesta del controlador. Por lo tanto, se tolera un error de posicionamiento para asegurar un seguimiento correcto de la fuerza.

En la Figura 2.15 se muestra un esquema de control.

2.2.1.2. Métodos de control activos indirectos

Dentro de los métodos de control indirectos se diferencia entre control de impedancia y control de admitancia, aunque a veces se utiliza el término control de impedancia para referirse a cualquiera de ellos indistintamente. Ambos persiguen el mismo objetivo: modificar la impedancia del robot, pero desde diferentes perspectivas. El control de impedancia tiene como entrada la posición y como salida la fuerza, en cambio, el control de admitancia tiene como entrada la fuerza y como salida la posición.

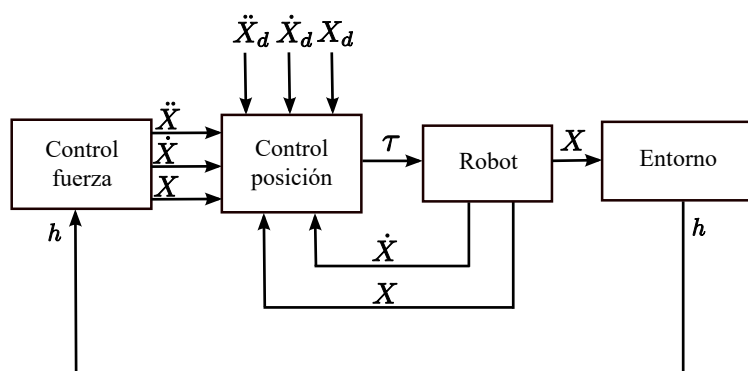


Figura 2.15: Esquema de control fuerza-posición paralelo. Traducción de [11]

Control de impedancia

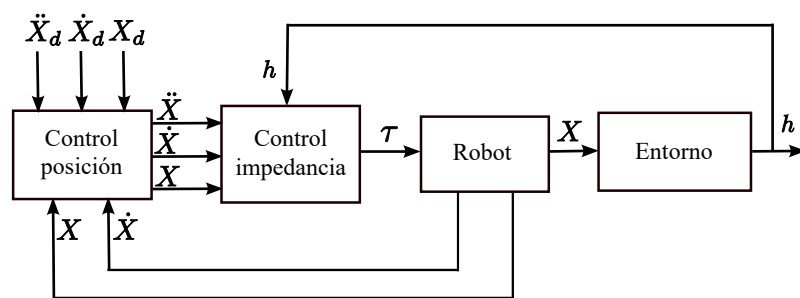
El control de impedancia regula de forma indirecta las fuerzas de contacto generando un movimiento que establece una relación dinámica deseada entre el robot y su entorno.

En contraposición a los métodos directos cubiertos en el apartado 2.2.1.1, el control de impedancia utiliza una única ley de control para regular de forma simultánea tanto la posición como la fuerza. Las matrices utilizadas en el control de impedancia tienen dimensiones físicas: rigidez, amortiguamiento e inercia. Estos parámetros van a definir el comportamiento dinámico del robot, simulando un efector final que contara con muelles, amortiguación e inercia. En la Figura 2.16a se muestra un esquema del control de impedancia.

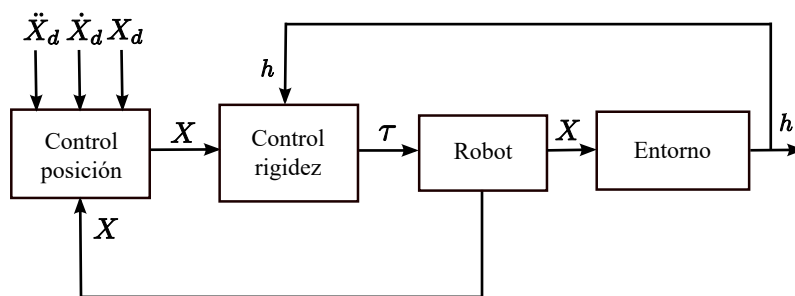
Control de rigidez El control de rigidez es una simplificación del control de impedancia. Únicamente es necesario establecer una acción proporcional que controlará el robot de manera que se comporte como un muelle de 6 grados de libertad respecto a las fuerzas y momentos aplicadas al efector final. Este método se centra únicamente en la respuesta estacionaria, por lo que no requiere un conocimiento completo de la dinámica del robot. En resumen, el control de rigidez establece una relación estática entre las fuerzas ejercidas sobre el entorno y la variación en la posición y orientación generada. En la Figura 2.16b se muestra un esquema del control de rigidez, se puede ver que la única entrada es la fuerza y la posición, ya que no necesita conocer valores de velocidad o aceleración.

Control de admitancia

El control de admitancia tiene el mismo principio de funcionamiento que el control de impedancia, sin embargo, separa explícitamente el control de posición del control de admitancia. El controlador de posición es un lazo interno diseñado para ofrecer un posicionamiento preciso y robusto, la diferencia radica en que el controlador de posición no recibe directamente el movimiento deseado, sino la salida del control de admitancia. La



(a) Control de impedancia completo. Rigidez, amortiguamiento e inercia



(b) Simplificación. Control de rigidez

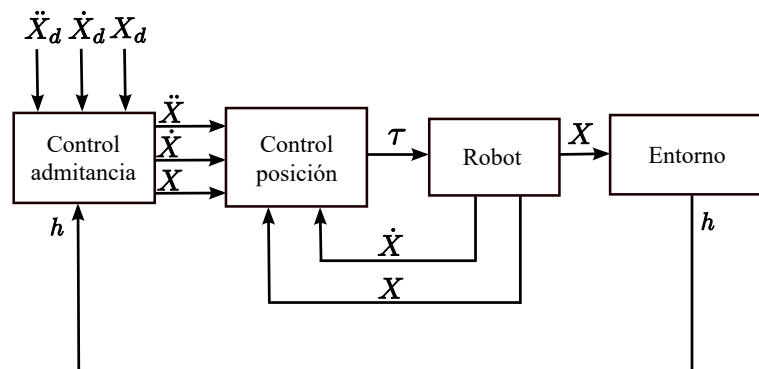
Figura 2.16: Esquema control de impedancia. Traducción de [11]

salida del controlador de posición garantizará una posición y orientación que cumpla con las relaciones dinámicas entre fuerza y movimiento. En la Figura 2.17a se muestra un esquema que ilustra este comportamiento.

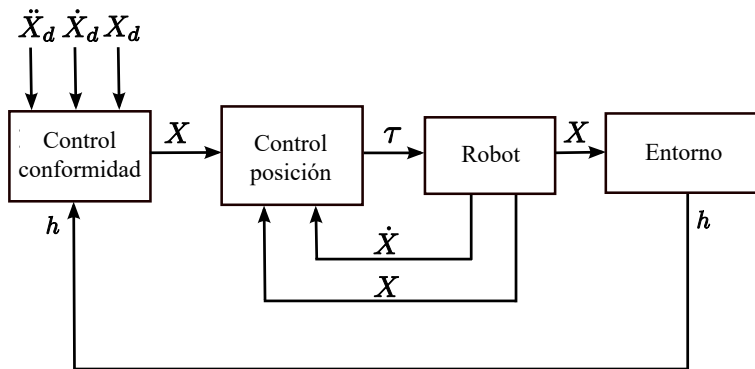
Control de *compliance* Al igual que el control de impedancia contaba con una simplificación llamada control de rigidez, al control de admitancia se le puede aplicar la misma simplificación para obtener el control de *compliance*. Con este control, únicamente es importante el valor estacionario de la relación entre fuerza y movimiento. En la Figura 2.17b se muestra su esquema [11].

2.2.2. Conclusiones

Se han presentado diferentes enfoques para abordar un control de *compliance* que se adapte al entorno y a posibles contactos con éste. Los métodos pasivos, pese a ofrecer un control más preciso, carecen de flexibilidad y adaptabilidad a posibles cambios en el entorno, por lo que limita su uso. Los métodos activos son más flexibles y pueden a su vez dividirse en directos o indirectos, que pese a tener implementaciones similares, difieren en principio de funcionamiento. Para estos métodos se proponen esquemas de control que únicamente son esqueletos a partir de los cuales desarrollar un sistema de control completo adaptado a cada aplicación. Se puede concluir que no existe un método superior a otro, sino que la idoneidad del mismo dependerá de la aplicación.



(a) Control de admittancia completo. Rigidez, amortiguamiento e inercia



(b) Simplificación. Control de *compliance*

Figura 2.17: Esquema control de admittancia. Traducción de [11]

2.3. Sistemas de control. Aplicaciones

2.3.1. Campo endonasal

En el apartado 2.1.4 se presentan varios prototipos de sistemas robóticos aplicados al campo endonasal. La función principal de la mayoría de los mismos es hacer de soporte al endoscopio, y por lo tanto no utilizan lazos de control basados en realimentación de fuerza, sino uno basado únicamente en posición. A continuación se presentan los detalles de los sistemas de control de los 3 prototipos que utilizan algún tipo de control de fuerza.

A73 - Wurm 2005 [47]

El sistema *A73* además de actuar como soporte endoscópico, cuenta con un sistema de navegación parcialmente autónomo. Los autores reflejan que el sistema utiliza un sensor de fuerza-par que puede ser activado o desactivado por el programa de control, de modo que el robot reaccione a las fuerzas encontradas con un grado de sensibilidad ajustable. En cambio, no aporta más información en términos de control.

Xia 2008 [13]

Se desarrolla un sistema robótico guiado por imágenes que proporciona asistencia mecánica a la hora de realizar perforaciones de la base del cráneo. Utiliza un sistema robótico *NeuroMate* [62] al cual han acoplado un sensor de fuerza-par en el efector final. El interfaz de control es cooperativo (tal y como se muestra en la Figura 2.18), es decir, las fuerzas y momentos ejercidos por el cirujano sobre dicho instrumento son convertidas en un movimiento del robot en dicha dirección. Utilizan *virtual fixtures*⁴ para restringir el espacio de trabajo bajo la siguiente ley de control de admitancia

$$\dot{q} = J^{-1}(q) \cdot K(d) \cdot G(f) \cdot \begin{bmatrix} F_w \\ T_w \end{bmatrix} \quad (2.1)$$

en la que \dot{q} es la velocidad angular, J es el Jacobiano en el extremo de la herramienta, cuya inversa J^{-1} permite convertir la velocidad cartesiana en velocidad angular, $K(d)$ es una matriz de escalado en función de la distancia d a la zona prohibida, $G(f)$ es una matriz de ganancia de admitancia en función de la fuerza medida f y F_w, T_w son las fuerzas y pares medidas con respecto al sistema de referencia *world* del robot.

⁴Las llamadas *virtual fixtures* pueden traducirse al español como elementos fijos virtuales y son una superposición de información sobre la percepción que tiene un usuario de un entorno real con el objetivo de mejorar las tareas de telemanipulación. Estas pueden ser fuerzas de repulsión u obstáculos que reaccionan al contacto de forma rígida, elástica o con otras propiedades dinámicas.



Figura 2.18: *Setup experimental del sistema [13]*

SmartArm - Marinho 2020 [14]

El sistema *SmartArm* está compuesto por dos brazos robóticos industriales con herramientas flexibles ancladas a sus extremos distales, utilizando como interfaz de control dispositivos hápticos Phantom Premium [63]. Opera bajo un sistema de control centralizado, debido a la facilidad de integración y mantenimiento, utilizando *Robot Operating System (ROS)* para la comunicación entre procesos. En la Figura 2.19 se muestra una imagen del *setup* experimental.

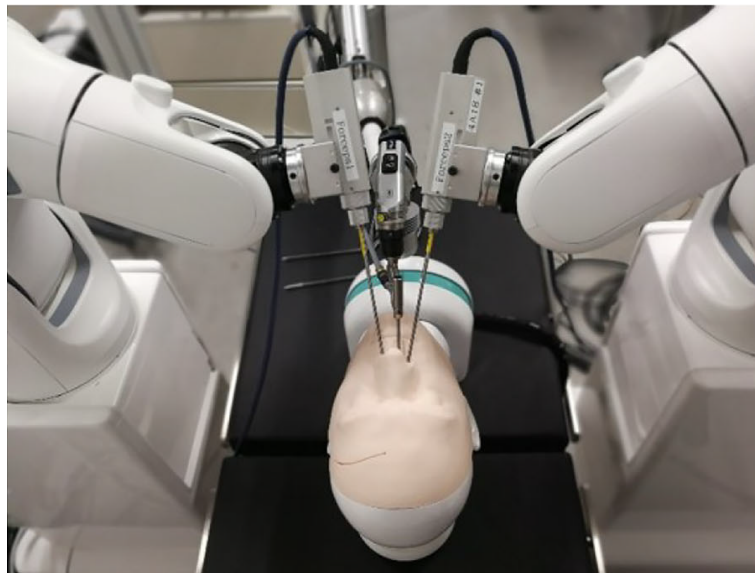


Figura 2.19: *Setup experimental del sistema SmartArm [14]*

La estrategia de control está basada en un marco de trabajo [64] desarrollado en que utiliza el método de las desigualdades de campos vectoriales [65] para la generación de *virtual fixtures*. La estrategia de control puede dividirse en dos bloques, el lado maestro y el lado esclavo.

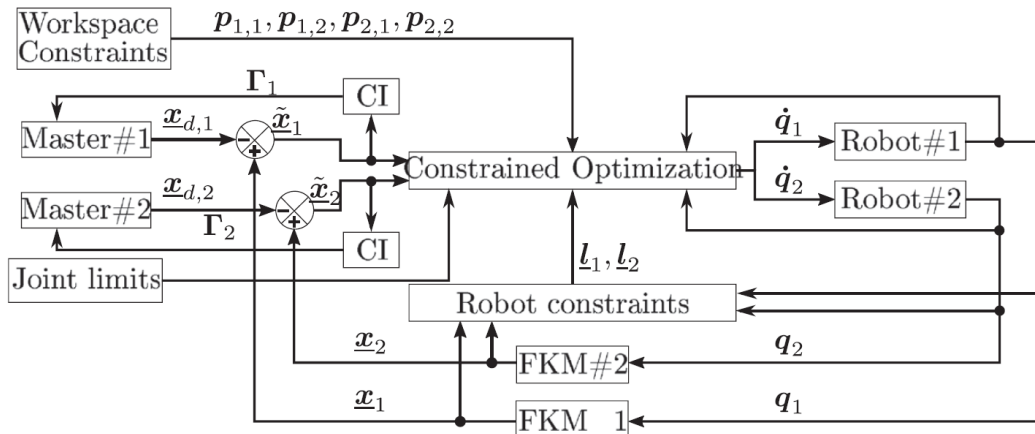
Lado maestro (dispositivo háptico) En el lado maestro se utiliza un sistema de control de impedancia de la forma

$$\gamma = -\eta_f(\tilde{t}^{master}) - \eta_v \cdot \dot{t}_{master} \quad (2.2)$$

en la que γ es la fuerza en el lado maestro, η_f, η_v son coeficientes de rigidez y viscosidad respectivamente, \tilde{t}^{master} es el error de posicionamiento del lado esclavo, visto desde el punto de vista del maestro, y \dot{t}_{master} es la velocidad lineal del interfaz maestro. En lugar de que las *virtual fixtures* sean generadas en el lado maestro (interfaz háptica) y estas afecten el comportamiento del esclavo (robot), se propone la generación de *virtual fixtures* en el lado esclavo en tiempo real, de forma que si hay una discrepancia entre la posición de la herramienta deseada y la posición real, el usuario pueda “sentir” las direcciones en las que el robot tiene dificultad para moverse.

Lado esclavo (brazo robótico) En el lado esclavo, los brazos robóticos son controlados a través de problema optimización de restricciones, utilizando funciones de coste que tienen en consideración la traslación y rotación del efector final, así como las velocidades de los ejes del robot.

En la Figura 2.20 se muestra el esquema de control.



FKM: Forward Kinematic Model, en español, cinemática directa

CI: Cartesian Impedance, se refiere al control de impedancia cartesiano

Figura 2.20: Esquema de control del sistema SmartArm [14]

2.3.2. Otros campos

En la literatura no se encuentran más desarrollos que utilicen sistemas de control con realimentación de fuerza en el campo endonasal, por ello, a continuación se presentan dos trabajos que utilizan un esquema de control muy similar, aún siendo aplicaciones

diferentes, y que podrían ser aplicables a cualquier otro campo que requiera un control basado en fuerza.

Control fuerza-posición en laparoscopia [15]

Se lleva a cabo el diseño e implementación de un sistema robótico de visión para *Single-Port Laparoscopy*⁵ compuesto por una cámara miniaturizada guiada magnéticamente mediante un brazo robótico a lo largo de la pared abdominal. En la Figura 2.21 se muestra un esquema conceptual.

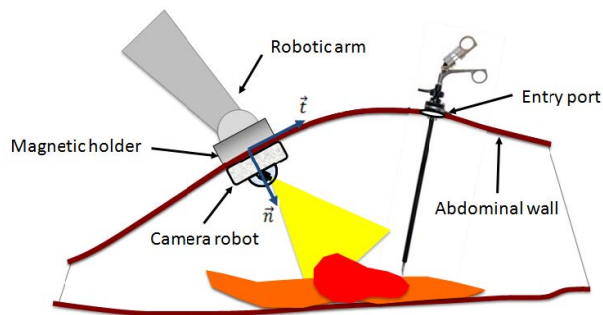


Figura 2.21: Esquema conceptual [15]

Para mover el brazo robótico sobre la pared abdominal se utiliza un sistema híbrido fuerza-posición en el cual la superficie de contacto se ha asemejado a un modelo elástico. En la Figura 2.22 se muestra el esquema de control.

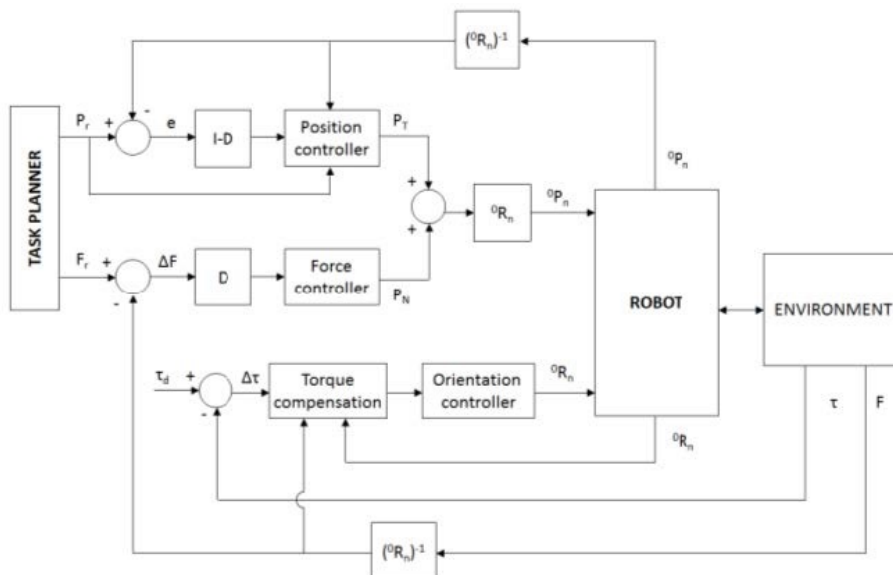


Figura 2.22: Esquema de control híbrido posición-fuerza [15]

⁵Single-Port Laparoscopy (Laparoscopia de puerto único) es una técnica MIS reciente en la cual el procedimiento se lleva a cabo a través de un único punto de entrada, que suele ser ombligo.

La primera del sistema es el planificador de tareas que fija la referencia de posición y fuerza; la fuerza es controlada en la dirección normal a la pared abdominal y la posición en las direcciones tangenciales. Ambas acciones son desacopladas a través de las matrices D y $I-D$ para evitar problemas de interferencias entre el controlador de fuerza y posición. La salida de ambos controladores es combinada y conformará la consigna de posición. La segunda entrada del sistema es la consigna de par, que con el sistema de compensación de par garantiza la perpendicularidad entre la dirección longitudinal del robot y la pared abdominal. La salida del controlador de orientación fija la consigna de orientación, y ambas con enviadas al controlador del robot. El efector final del robot tiene acoplado un sensor de fuerza-par que permite cerrar el lazo de control junto con los valores de posición y orientación del robot.

El controlador de posición consiste en un controlador PI y el controlador de fuerza asume una interacción elástica entre el robot y el entorno, relacionada a través de una constante de rigidez del entorno.

Control fuerza en lijado de aspas para uso aeronáutico [16]

El lijado con asistencia robótica ha surgido como un proceso de acabado en los últimos años para mecanizar componentes con altos requisitos de acabado superficial y flexibilidad. Sin embargo, la consistencia del mecanizado de la superficie es difícil de garantizar en un proceso de este tipo. Por ello, se propone aplicar un método de control híbrido de fuerza-posición combinado con un control PI/PD que se muestra en la Figura 2.23.

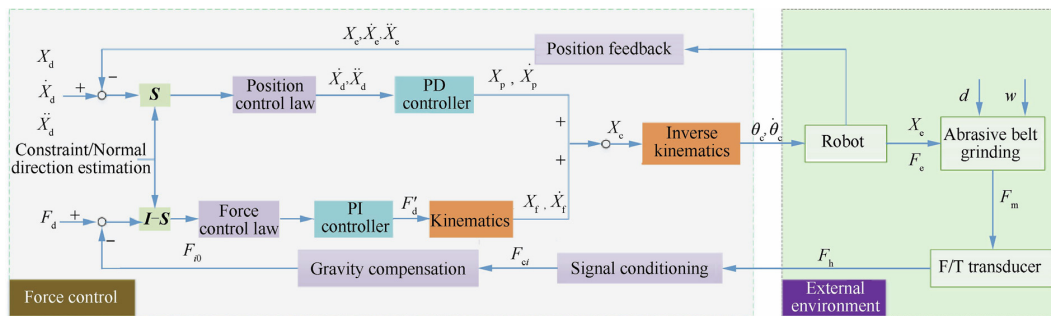


Figura 2.23: Esquema de control para lijado de aspas [16]

Existen bastantes similitudes entre este esquema de control y el mostrado en la Figura 2.22. Esto es debido a que aunque se trate de aplicaciones diferentes, ambos desean controlar la fuerza en la dirección normal a la superficie y la posición en las direcciones tangenciales.

Aún así, existen diferencias. El controlador de posición es un PI en vez de un PD, y el controlado de fuerza, en vez de establecer una relación elástica, utiliza un controlador PI. Debido al mayor peso del sistema, y a las diferencias en las lecturas de fuerza según la orientación del robot, se lleva a cabo una compensación gravitatoria.

Capítulo 3

Descripción *hardware* y *software*

Este tercer capítulo hace de puente entre el estado del arte y el desarrollo de los sistemas de control de capítulos posteriores, creando una visión general del flujo de trabajo y del entorno en el que se encuentra el robot. Se van a presentar las herramientas que se han utilizado para el desarrollo del trabajo a nivel de software y hardware. Esto incluye tanto soluciones comerciales como elementos de diseño y fabricación propia.

3.1. Hardware

Comenzamos por la descripción del *hardware* utilizado, ya que será el que condicione en cierta medida el *software* a utilizar. Los elementos principales de *hardware* son el brazo robótico, la pinza, el sensor de fuerza, el intercambiador de herramientas, los dispositivos médicos, el cráneo y el portaherramientas al que estará anclado el endoscopio.

3.1.1. Brazo robótico

El brazo robótico utilizado es un robot colaborativo modelo *UR3* del fabricante *Universal Robots* (Figura 3.1) . Sus especificaciones técnicas completas se presentan en el Anexo A aunque a continuación se resumen las más importantes [17].

- Carga máxima 3 *kg*
- Alcance 500 *mm*
- Rango de giro de las juntas $\pm 360^\circ$ (rotación infinita en la última junta)
- Velocidad lineal de la herramienta máxima de 1000 *mm/s*
- Repetibilidad ± 0.1 *mm*

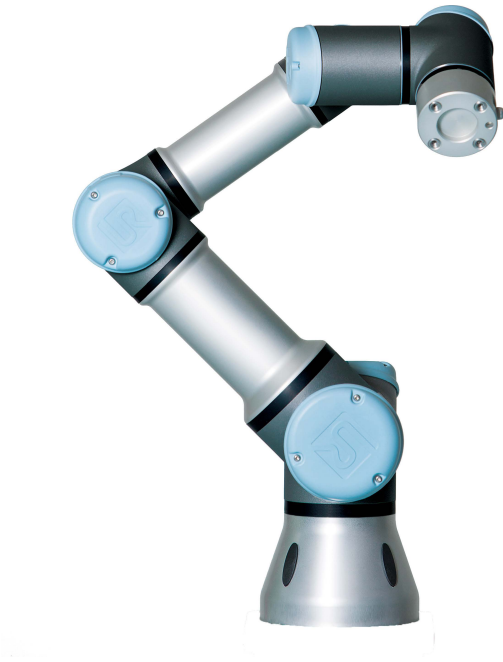


Figura 3.1: Brazo robótico UR3 [17]

- Funcionamiento colaborativo
- Lazo de control a 125 Hz

Sus características técnicas resultan muy adecuadas para la aplicación que nos concierne. La carga máxima es más que suficiente, ya que las herramientas quirúrgicas son ligeras; el alcance pese a no ser muy grande, es suficiente, dando el pequeño espacio de trabajo en el que se moverá; el rango de giro de las juntas es muy elevado y aportará mucha flexibilidad a la hora de posicionar el robot en la mesa de operaciones; la velocidad lineal de la herramienta no es relevante, ya que los movimientos que se realizarán serán a muy baja velocidad; la repetibilidad tiene gran importancia para garantizar la seguridad del procedimiento, y tiene un valor suficiente; el funcionamiento colaborativo aporta un nivel de seguridad adicional, además de facilitar la interacción entre el personal sanitario y el robot; por último, el lazo de control interno funciona a 125 Hz , lo cual limitará la frecuencia máxima del sistema de control a desarrollar.

En cambio, su capacidad colaborativa no es suficiente para el desarrollo de un sistema de control basado en la fuerza. La caja de control del robot obtiene internamente la fuerza ejercida a partir de la corriente suministrada a los motores, teniendo en cuenta la posición y masa de la carga que está moviendo, así como la configuración de ejes del propio robot. Eso implica que la sensibilidad en la medida de la fuerza, en el mejor de los casos, ronda los 5 N [66]. Una sensibilidad de 5 N es claramente insuficiente para esta aplicación, por lo que será necesaria la utilización de un sensor de fuerza externo.

Para aumentar el alcance del robot, sobretodo a la hora de manipular instrumentos con una longitud considerable (como el endoscopio y otros dispositivos médicos), se ha utili-

zando un calzo entre la mesa y la base del robot que lo eleva 304 *mm*.

3.1.2. Pinza



Figura 3.2: Pinza 2F-85 de Robotiq [18]

Durante las primeras etapas de desarrollo se utilizó la pinza 2F-85 del fabricante *Robotiq* [18] (Figura 3.2), ya que era una forma muy ágil y sencilla de coger diferentes herramientas. Más adelante, se descartó su uso por el volumen que ocupaba, las dificultades que tenía el robot para alcanzar ciertas zonas debido a la longitud adicional de la pinza y principalmente porque el desarrollo del intercambiador de herramientas eliminaba la necesidad de utilizarla.

Además, esta tiene un peso de 900 *g*, al que habría que añadir el dispositivo médico que está portando y su soporte. Uno de los objetivos es limitar el peso al mínimo, mejorando el funcionamiento del sistema, ya que cuanto menor sea la carga del robot, mejor será la precisión.

3.1.3. Sensor de fuerza

El sensor de fuerza utilizado es el modelo *HEX-E* del fabricante *OnRobot* (Figura 3.3). Este modelo es compatible con el robot *UR3*, anclándolo directamente a su efector final. Las características completas, así como sus dimensiones, se disponen en el Anexo B, aunque a continuación se muestran sus características más relevantes [24].

- Sensor de 6 ejes, es decir, mide tanto la fuerza como el momento en los ejes XYZ
- Tiene una capacidad nominal de 200 *N* y 10 *Nm*



Figura 3.3: Sensor de fuerza HEX-E del fabricante OnRobot [19]

- Frecuencia de muestreo máxima 500 Hz.

La resolución libre de ruido se resume en la Tabla 3.1.

F_{xy}	F_z	T_{xy}	T_z
0.2 N	0.8 N	0.01 Nm	0.002 Nm

Tabla 3.1: Resolución libre de ruido del sensor de fuerza HEX-E en función del eje de aplicación [24]

La resolución en la medida de fuerza es muy superior a la ofrecida por el brazo robótico y permitirá el desarrollo de un sistema de control preciso. En cambio, hay que tener en cuenta que la resolución en los ejes X e Y es diferente al eje Z, lo cual tendrá unos resultados negativos en su desempeño y habrá que sintonizar el controlador de forma que compense este desequilibrio.

El módulo DAQ (*Data Acquisition*, en español, adquisición de datos) se puede conectar a la caja de control del robot directamente o a un ordenador externo vía *Ethernet* o USB.

3.1.4. Intercambiador de herramientas

En el proyecto *Craneal* se propone un sistema de intercambio de herramientas automático sin necesidad de intervención humana. Se realizó un estudio de mercado de los sistemas comerciales que permitieran llevar a cabo dicho intercambio, encontrando 3 tipos de productos:

- Accionados neumáticamente: el modelo QC-5 de *ATI Industrial Automation* [67] soporta una carga de 8.2 kg. La principal desventaja es que depende de un suministro constante de aire comprimido, lo cual añade complejidad al sistema global y obliga al quirófano a tener una instalación de aire comprimido

- Accionados manualmente: el modelo MC-10 de *ATI Industrial Automation* [68] es un cambiador de herramientas manual que soporta hasta 10 *kg*. Aunque es una opción muy válida, no cumple con el objetivo de permitir un intercambio de herramientas sin intervención humana
- Accionados mecánicamente: el modelo *Tool Changer* de *Smartshift Robotics* [20] es un sistema puramente mecánico, sin alimentación eléctrica o neumática y soporta hasta 25 *kg*, lo cual lo hace una opción muy buena (Figura 3.4)



Figura 3.4: Sistema *Tool Changer* de *Smartshift Robotics* [20]

Como el proyecto *Craneal* se encuentra en una etapa de desarrollo muy temprana, se decidió desarrollar un sistema basado en dicho modelo para verificar el funcionamiento del intercambiador de herramientas en el sistema global. Víctor Moreno San Juan (colaborador del Instituto de Tecnologías Avanzadas de la Producción de la UVa) llevó a cabo dicho desarrollo y se fabricó un prototipo del intercambiador de herramientas con una impresora 3D utilizando material PLA. En la Figura 3.5 se muestran los modelos 3D del intercambiador fabricado.

Se trata de un anclaje puramente mecánico, en la que el macho gira 90° sobre la hembra y finaliza el anclaje. Las rampas en la pieza macho ayudan a hacer el anclaje más fuerte. Con ayuda del sensor de fuerza se puede garantizar un par de apriete para una operación segura y consistente a lo largo de los ciclos de uso.

El macho se encuentra fijado al efector final del robot y las hembras se encuentran en la estación de intercambio. Cabe mencionar que las piezas hembra tienen los mismos puntos de anclaje que el robot original (4 roscas M6x8, según los planos facilitados en el Anexo A.2) para facilitar la compatibilidad con desarrollos anteriores que se anclaran directamente al extremo del robot. La dimensiones del módulo macho son aproximadamente un diámetro de 65 *mm* y una altura de 10 *mm*, y las dimensiones del módulo hembra tiene un mismo diámetro de 65 *mm* y una altura de 20 *mm*.

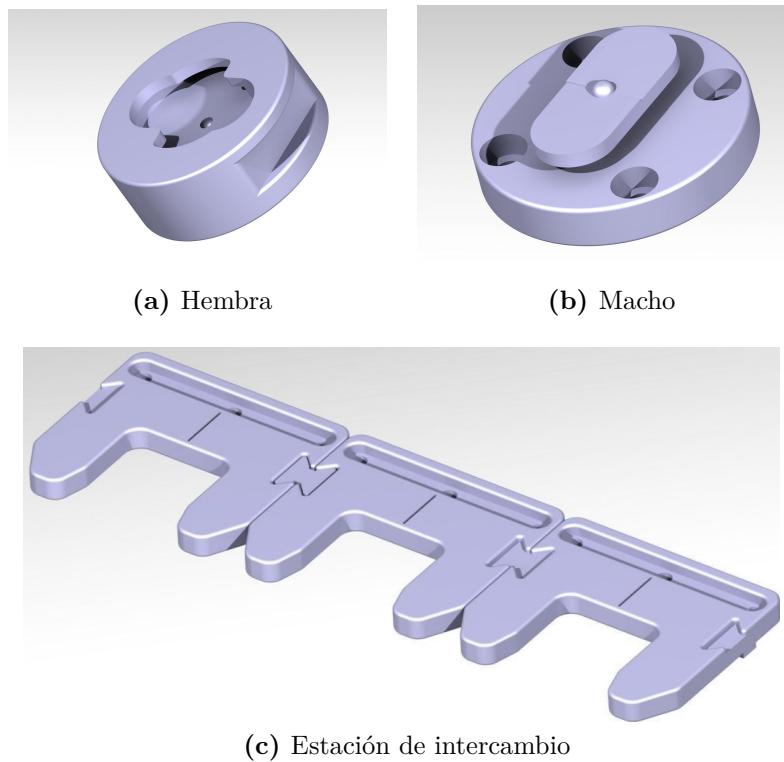


Figura 3.5: Intercambiador de herramientas desarrollado en ITAP

3.1.5. Dispositivos médicos

A continuación se describen los dispositivos médicos utilizados. A pesar de no disponer de la totalidad del material, todos los desarrollos se han realizado pensando en la compatibilidad con los mismos. La selección de material se ha basado en un informe [21] facilitado por *Karl Storz* en la que realizan una recomendación de instrumental para EETS.

Endoscopio Se ha utilizado un endoscopio con óptica *HOPKINS* modelo 28132 BA de *Karl Storz*, con una dirección visual de 30° , un diámetro exterior de 4 mm y una longitud de 18 cm (Figura 3.6).



Figura 3.6: Óptica *HOPKINS* modelo 28132 BA de *Karl Storz*

Aspirador Se ha optado por el modelo 649179 B de *Karl Storz* con tubo flexible, punta cónica, longitud de 18 cm y diámetro exterior de 1.35 mm (Figura 3.7)

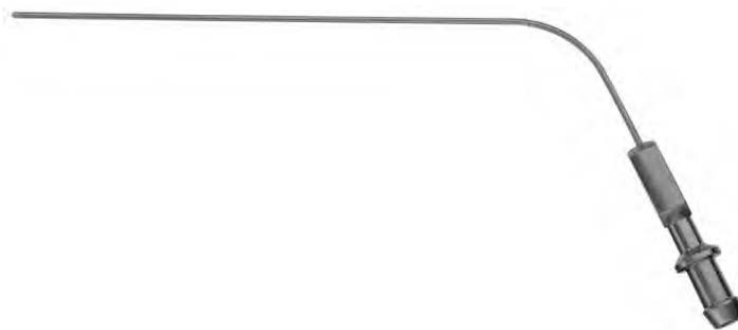


Figura 3.7: Aspirador 649179 B de Karl Storz [21]

Electrodo de coagulación Se trata de una electrodo de coagulación con extremo esférico, modelo 28164 ED de *Karl Storz* con un diámetro de 2 mm y una longitud total de 13 cm (Figura 3.8).



Figura 3.8: Electrodo de coagulación 28164 ED de Karl Storz [21]

Taladro En intervenciones EETS es importante la utilización de taladros diseñados específicamente para esa zona, ya que deben funcionar a altas revoluciones y tener un tamaño reducido. El modelo de taladro elegido ha sido el 252682 de *Karl Storz*, tiene una velocidad de giro de 100 000 rpm, una longitud útil de 93 mm, un diámetro exterior de 7.5 mm y admite fresas con un diámetro de eje de 3.17 mm (Figura 3.9).



Figura 3.9: Taladro 252682 de Karl Storz [21]

Cureta Se ha seleccionado una cureta modelo 28164 KB de *Karl Storz* con forma de cuchara, el extremo en ángulo, un tamaño de 2 mm, mango redondo y longitud de 25 cm (Figura 3.10).



Figura 3.10: Cureta 28164 KB de Karl Storz [21]

3.1.6. Cráneo y soporte del cráneo

Para la realización de pruebas es indispensable contar con un modelo de un cráneo realista. Las tres universidades hemos utilizado el mismo modelo de cráneo impreso con una impresora 3D en material PLA, para facilitar la colaboración y que los resultados sean extrapolables.

La Universidad de Málaga desarrolló como parte del proyecto *Craneal* el sistema de soporte del cráneo, junto con modificaciones en el modelo original del cráneo para facilitar el anclaje. En la Figura 3.12 del siguiente apartado se presenta el escenario completo, con el soporte del cráneo anclado a extrusiones de aluminio de 45x45 mm.

En la Figura 3.11 se muestra el modelo del cráneo, que añade los puntos de anclaje al soporte, el taladro para acceder a la base del cráneo y un punto de calibración, dónde se creará el sistema de referencia común entre ambos brazos robóticos. A modo de referencia, el cubo que abra todo el volumen del cráneo tiene unas dimensiones de 142x198x91 mm.

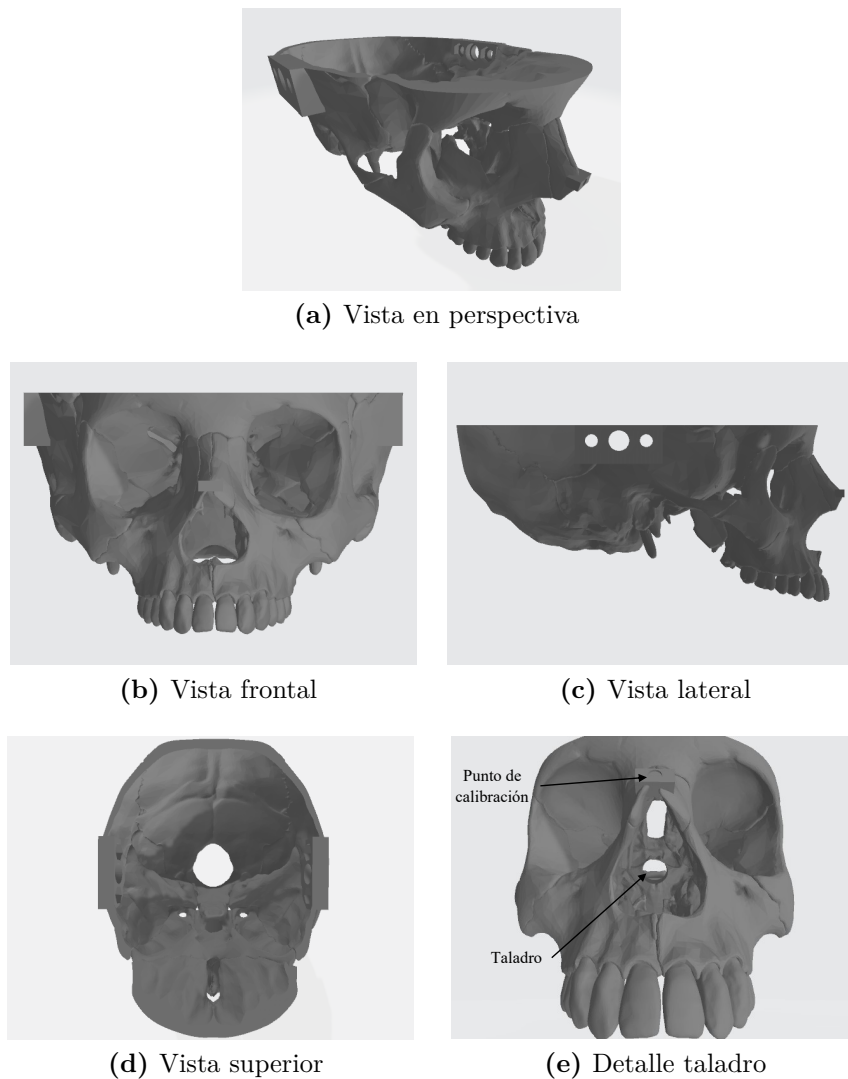


Figura 3.11: Modelo del cráneo listo para anclaje

3.1.7. Portaherramientas para el endoscopio

Se han utilizado dos versiones del portaherramientas para el endoscopio durante el desarrollo del TFM. Ambos han sido fabricados en la impresora 3D y han sido diseñados por Víctor Moreno San Juan (ITAP).

La primera versión fue diseñada para realizar el anclaje con la pinza. Estaba formado por dos cuerpos: en el primero estaba alojado el endoscopio y era dónde la pinza realizaba el contacto; el segundo cuerpo es independiente del primero y es dónde está alojada la óptica, de forma que se pueden intercambiar las ópticas de una forma cómoda. En términos generales, era un agarre muy robusto en el que los dispositivos médicos estaban muy protegidos, en cambio, era muy voluminoso y añadía bastante peso (aproximadamente 1 kg) al conjunto. En la Figura 3.12 se muestra un foto del mismo.

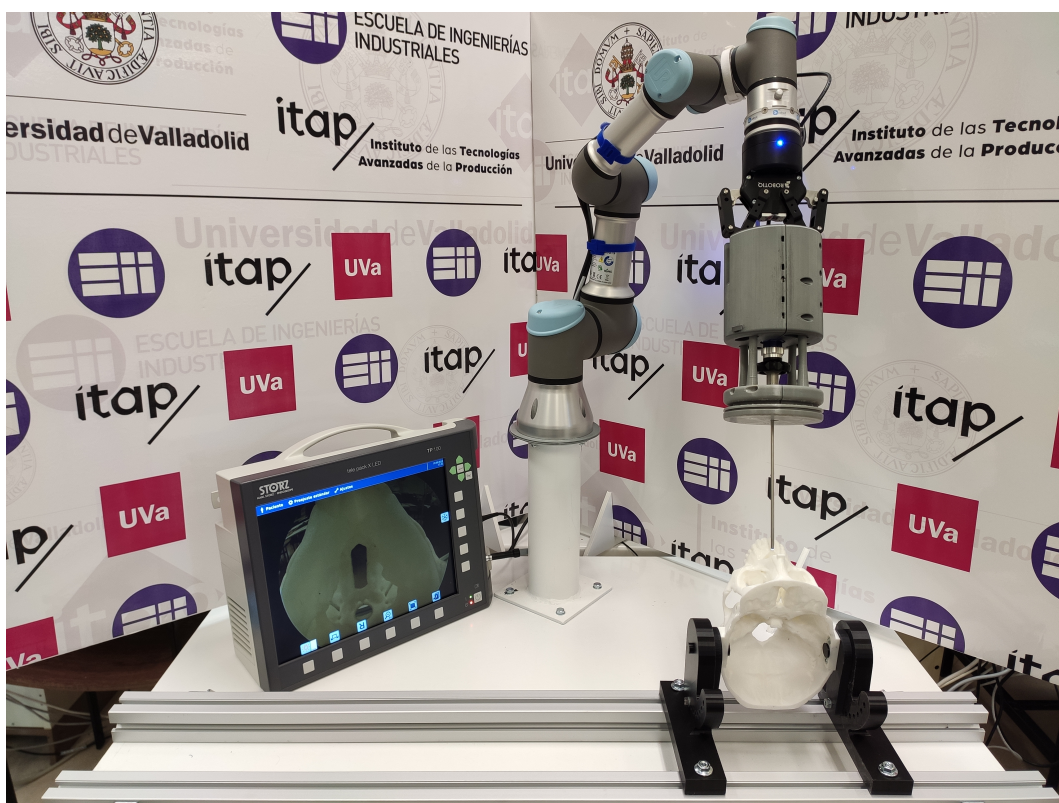


Figura 3.12: Escenario completo utilizando la primera versión del agarre del endoscopio

La segunda versión es un rediseño completo con los siguientes objetivos:

- Disminuir el volumen y peso del agarre
- Eliminar puntos de contacto para la pinza y crear un sistema de anclaje directo al robot (o al intercambiador de herramientas, que comparte el mismo interfaz)
- Cambiar orientación para facilitar la cooperación con el segundo brazo robótico que se utilizará en el proyecto *Craneal*

Definimos el plano XY como el coincidente con la brida de salida del robot, y el eje Z la dirección longitudinal con sentido saliente. En la primera versión, el endoscopio y el eje Z son colineales, lo cual limita los movimientos de ambos brazos robóticos, en cambio, si se orienta el endoscopio de forma que sea paralelo al plano XY y perpendicular al eje Z, mejora la capacidad de movimiento. La forma trivial de implementar esta idea se refleja en el prototipo de la Figura 3.13.



Figura 3.13: Prototipo de agarre con orientación cambiada

En cambio, esta implementación da lugar a una singularidad denominada por *Universal Robots* como *Wrist Alignment Singularity* [22], que se traduciría al español como singularidad por alineación de muñeca. Las articulaciones del hombro, codo y muñeca 1 rotan todas en el mismo plano en el robot *UR3*, como muestran las flechas numeradas 1, 2 y 3 en la Figura 3.14. Sin embargo, cuando también se alinea la articulación de la muñeca 2 (marcada con el número 4) en ese mismo plano, el rango de movimiento se ve muy reducido debido a las singularidades que aparecen. Por ello, la solución más sencilla es modificar el diseño del agarre, haciendo que sostenga el endoscopio girado 90°, y así evitar el giro de la muñeca 2.

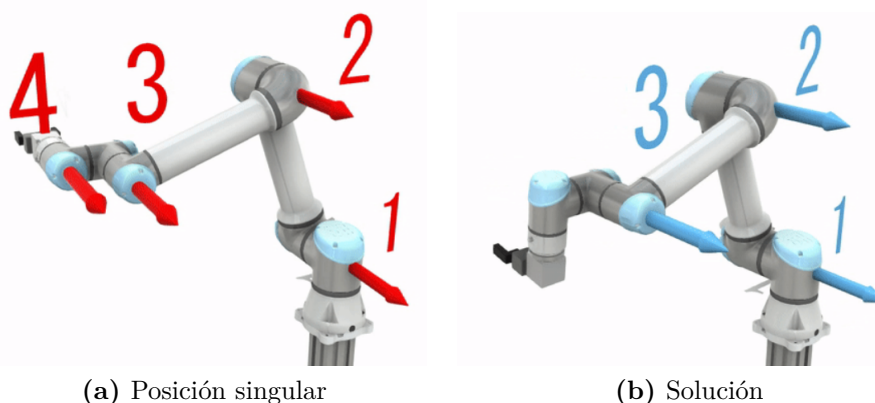


Figura 3.14: Singularidad por alineación de muñeca [22]

Teniendo en cuenta los 3 objetivos enumerados y la necesidad de evitar la singularidad, se presenta la versión final en la Figura 3.15. En dichas imágenes se utiliza una replica del endoscopio y la óptica impresos en 3D, que ha sido de gran utilidad para realizar pruebas sin poner en riesgo el material real.

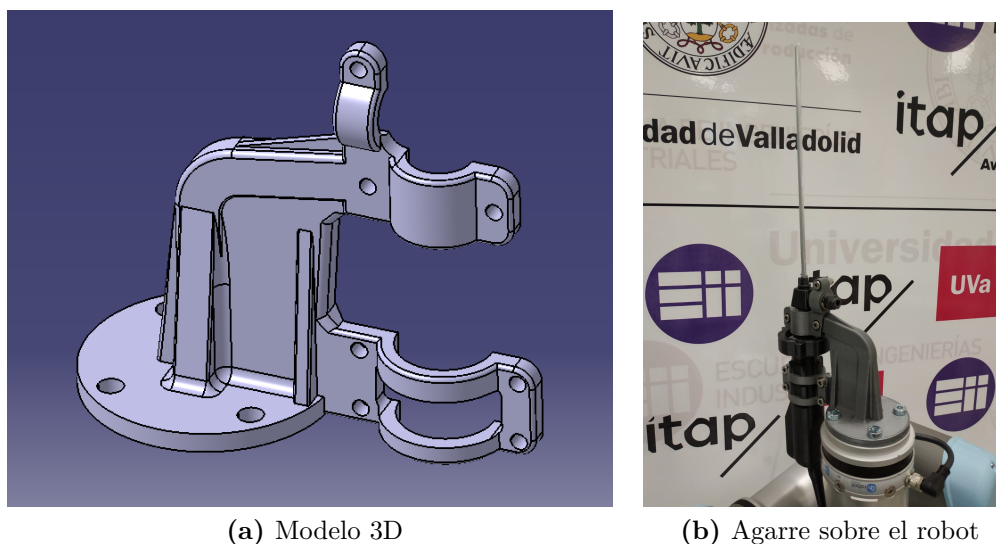


Figura 3.15: Segunda revisión del agarre del endoscopio

Finalmente en la Figura 3.16 se muestra una imagen con el escenario completo.

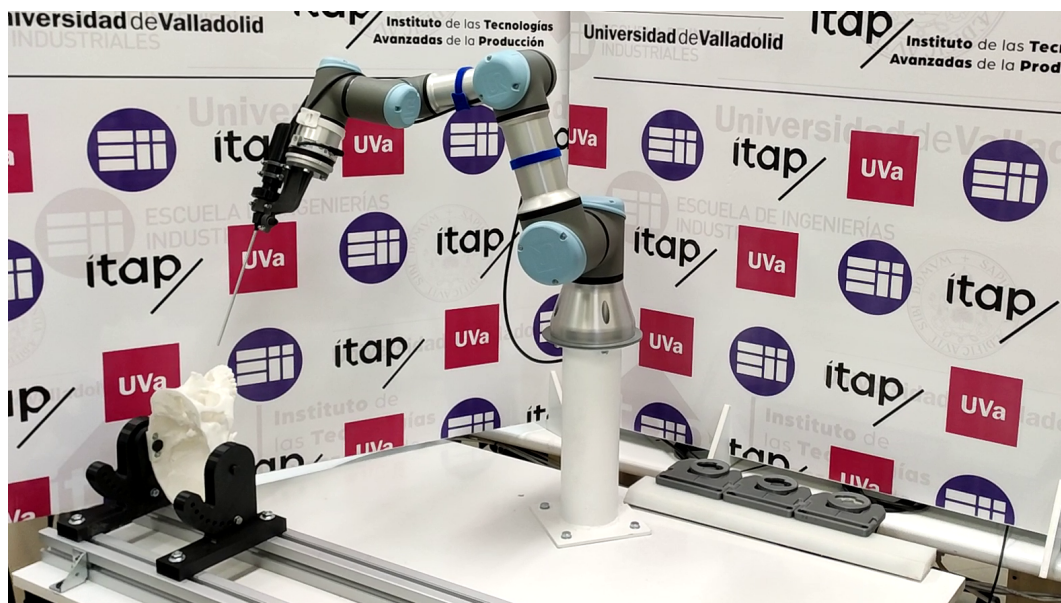


Figura 3.16: Escenario completo

3.2. Software

Se ha diseñado un sistema de control centralizado basado en una distribución Ubuntu 16.04.6 LTS en la cual se ha instalado la versión *Kinetic Kame* de *ROS* (*Robot Operating System*, en español, sistema operativo para robots). *ROS* es un entorno de trabajo pensado para el desarrollo *software* de sistemas robóticos de código abierto. Es una colección de herramientas, bibliotecas y convenciones cuyo objetivo es facilitar la tarea de crear un sistema robótico complejo [69]. Ofrece interfaces tanto para *Python* como *C++*, habiendo escogido este último por contar con funciones más avanzadas.

En *ROS* los procesos individuales se denominan nodos, y se comunican entre ellos a través de *topics*¹ (comunicación unidireccional) y servicios (comunicación bidireccional). De esta manera, en lugar de tener que desarrollar un único proceso con varios hilos de alta complejidad, los programas pueden desarrollarse de forma individual y conectarse posteriormente a través de la infraestructura de *ROS*. Dada la naturaleza colaborativa del proyecto *Craneecal*, este enfoque resulta muy conveniente.

En torno a *ROS* existe un elevado número de bibliotecas y herramientas pensadas para todo tipo de sistemas robóticos que facilitan en gran medida su desarrollo, entre las que se incluyen comunicación simple entre procesos (*topics* y servicios), gestión de sistemas de coordenadas (*tf2*), entornos de trabajo para movimiento (*MoveIt*) programas de simulación (*Gazebo*) y de visualización (*RViz*), entorno de trabajo para la creación de interfaces gráficas (*rqt*), adquisición de datos sencilla (*ros bag*), etc. En los siguientes apartados se resumen aquellas utilizadas de mayor importancia.

3.2.1. *MoveIt* y *tf2*

MoveIt Motion Planning Framework [23] es el entorno de trabajo más utilizado dedicado a la planificación de movimientos y manipulación robótica en *ROS*, siendo compatible con un elevado número de modelos de robots, entre los que se incluye el *UR3* de *Universal Robots*. *MoveIt* está escrito en su totalidad en *C++*, aunque al igual que *ROS*, ofrece un interfaz en *Python* para acceder a las funcionalidades básicas, y utiliza por defecto los sistemas de comunicación de *ROS*.

Para facilitar la labor de desarrollar nuevas funcionalidades para *MoveIt*, así como para activarlas o desactivarlas en función de las necesidades, se utilizan *plugins* independientes al núcleo. Los *plugins* más utilizados son el planificador de movimientos *OMPL* (*Open Motion Planning Library*), detección de colisiones con *FCL* (*Fast Collision Library*), *plugin* para trabajar con cadenas cinemáticas *KDL* (*Kinematics and Dynamics Library*), entre

¹Los *topics* (se podría traducir al español como temas o asuntos) son buses a través de los cuales los nodos intercambian información. Existen nodos suscriptores, que reciben los mensajes de un determinado *topic*, y nodos publicadores, que envían mensajes a dicho *topic*. El tipo de mensaje intercambiado es definido a la hora de crear el *topic*, por lo que ambos nodos saben lo que esperar y lo que enviar [70]

otras [71].

La API para acceder a las funciones de *MoveIt* se puede dividir en 4 bloques, en función de los avanzadas que sean las mismas:

- *Move Group Interface*: se trata del interfaz principal para acceder a la funcionalidad de *MoveIt*, abarcando movimientos en coordenadas articulares y cartesianas, establecer velocidades y aceleraciones, manipulación de objetivos (*pick and place*), etc.
- *MoveIt ROS*: ofrece funcionalidad adicional en términos de la cadena cinemática del robot, añade capacidad de percepción por parte del robot, interacción con otros componentes de *ROS* y en general un control mayor sobre la planificación
- *MoveIt Core*: se trata del control de menor nivel al que se puede acceder, teniendo control sobre las funcionalidades internas de *MoveIt*
- *MoveIt OMPL Interface*: es un interfaz que permite ajustar la planificación de trayectorias cuando se utiliza el *plugin* OMPL

Durante el desarrollo realizado en este TFM se ha utilizado *Move Group Interface* para funcionalidad básica y *MoveIt ROS* para movimientos más avanzados.

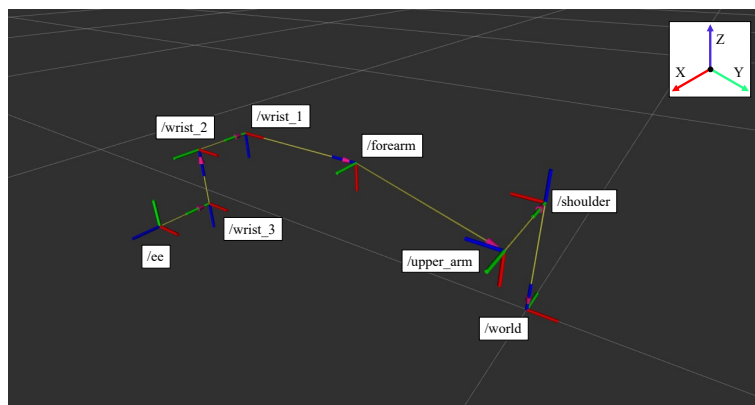
tf2 [72] es un paquete² de *ROS* desarrollado para la gestión de sistemas de coordenadas y permite al usuario llevar un seguimiento de múltiples sistemas de coordenadas a lo largo del tiempo. Mantiene la relación entre estos en una estructura de árbol, y permite transformar puntos o vectores entre dos sistemas de coordenadas cualesquiera en cualquier momento deseado, pasado o actual. Se trata de un paquete muy utilizado en el entorno *ROS* y también es utilizado por *MoveIt* para la gestión de sistemas de coordenadas.

En la Figura 3.17 se muestran los sistemas de coordenadas asociados al robot *UR3* y como están asociados los unos a los otros.

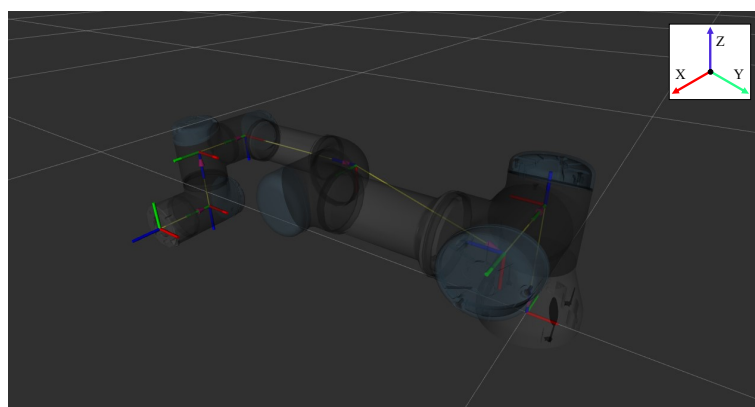
3.2.2. Paquetes conexión *UR3*

Para poder utilizar una plataforma robótica en *ROS* y *MoveIt* se necesitan dos elementos esenciales. El primero es un *driver* que haga de puente entre la caja de control del robot y el ordenador sobre el que está ejecutándose *ROS*. El segundo es un paquete de configuración *MoveIt*, que contiene toda la información referente al mismo: modelo cinemático del robot, articulaciones que pueden colisionar entre ellas, grupos de planificación (por

²En *ROS* el *software* está organizado en paquetes. Un paquete puede contener nodos, una biblioteca, un conjunto de datos, archivos de configuración o cualquier otra información que constituya un módulo útil en sí mismo[73]



(a) Sin vista del robot



(b) Con vista del robot

Figura 3.17: Sistemas de referencia del robot UR3

ejemplo, únicamente la muñeca o las tres últimas articulaciones), posiciones predeterminadas, efectores finales (por ejemplo, una pinza, un taladro, un endoscopio...) y sensores de percepción (si los hubiera).

El modelo cinemático del robot se define a través de un fichero *URDF* (*Universal Robot Description Format*), que especifica articulaciones, dimensiones, geometría, peso, velocidades, aceleraciones, límites, etc. También existe un fichero *SRDF* (*Semantic Robot Description Format*) que es un complemento del fichero *URDF* donde se especifican grupos de articulaciones, transformaciones e información adicional sobre colisiones, entre otros. Ambos ficheros siguen un formato *XML*.

Existen dos paquetes que ofrecen tanto un *driver* de comunicación como un paquete de configuración *MoveIt* alojados en la plataforma *Github* y de código abierto.

- *universal_robot* [74]: contiene el paquete de configuración *MoveIt*, descripción *URDF* y *SRDF* del robot, paquete para cálculos de cinemática inversa y directa, definición de mensajes para intercambio con la caja de control del robot, *driver* básico, paquete de configuración para el *software* de simulación *Gazebo* y modelo 3D para visualización y cálculo de colisiones

- *ur_modern_driver* [75]: es la evolución del *driver* básico presente en el paquete *universal_robot*, que soluciona problemas de estabilidad y añade nueva funcionalidad. A fecha de redacción de este TFM acaba de lanzarse una nueva revisión del mismo, llamado *ur_robot_driver* [76] mejorando estabilidad y funcionalidad, además de haberse realizado en colaboración con *Universal Robots*, con lo que es esperable un mejor desempeño en términos generales.

En la Figura 3.18 se muestra una comparación entre el modelo de simulación y el entorno real. Se han incorporados elementos como la mesa, el calzo y la pinza.

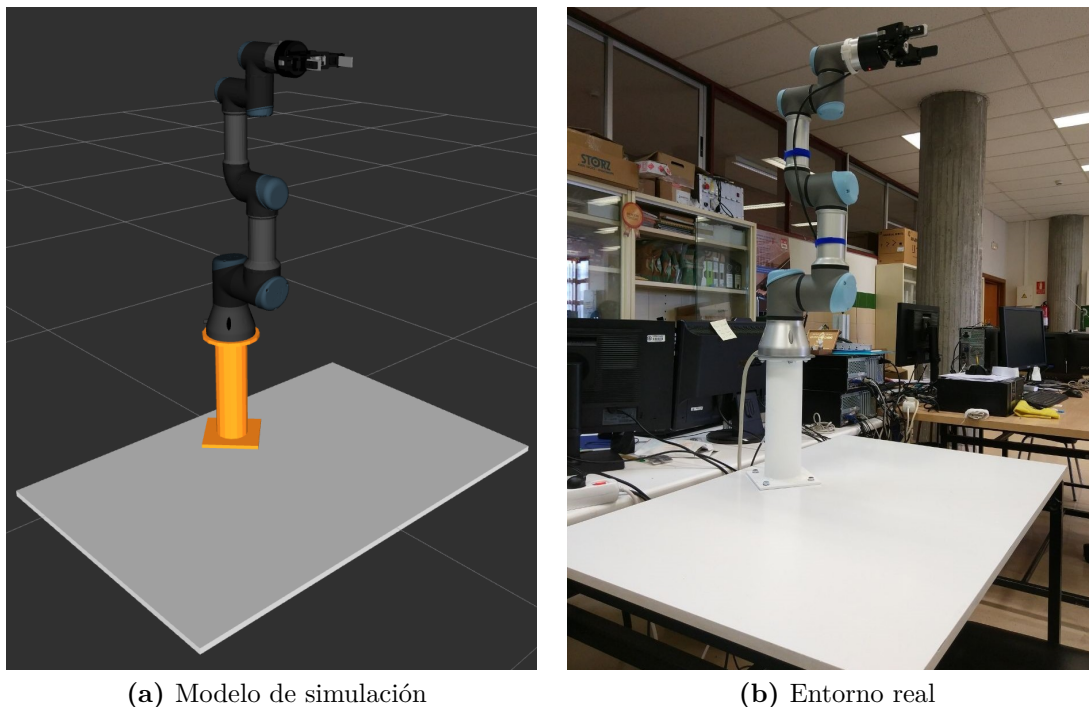


Figura 3.18: Comparación entre el modelo de simulación y el entorno real

3.2.3. *URScript*

El robot *UR3* puede controlarse de forma nativa a dos niveles:

- Utilizando la pantalla *PolyScope* y las herramientas disponibles en el interfaz de usuario
- A nivel de *script*

El lenguaje de programación desarrollado por *Universal Robots* se llama *URScript* [77], que incluye variables, definiciones, comandos de control de flujo y funciones que controlan y monitorean los movimientos del robot, así como sus puertos de entrada/salida.

Al controlador de bajo nivel se le llama *URControl* y se ejecuta como un servicio al que puede acceder tanto la interfaz de usuario como un cliente a través de una conexión TCP/IP. La programación a nivel de *script* consiste en la ejecución de código en ordenador que actúa como cliente y se comunica con el controlador a través de *sockets*.

La comunicación con el controlador de bajo nivel se realiza a través del *driver* descrito en el apartado anterior, y será la forma más directa y rápida de comunicarse con el robot. Es la fuente de información principal: posiciones angulares de los ejes, estado del robot, temperaturas, fuerzas, etc.

3.2.4. *Rviz* y *URSim*

A continuación se presentan dos herramientas que han sido de gran utilidad para el desarrollo de *software* y realización de pruebas: *Rviz* y *URSim*.

Rviz [78] es el visualizador 3D nativo de *ROS*. Permite cargar el modelo tridimensional del robot y ver los movimientos del mismo en tiempo real, visualizar los sistemas de referencia y las relaciones entre ellos, muestra valores angulares de ejes, así como la posición y orientación respecto al sistema de referencia que se elija. *MoveIt* cuenta con un *plugin* para *Rviz* que ofrece una funcionalidad básica (Figura 3.19).

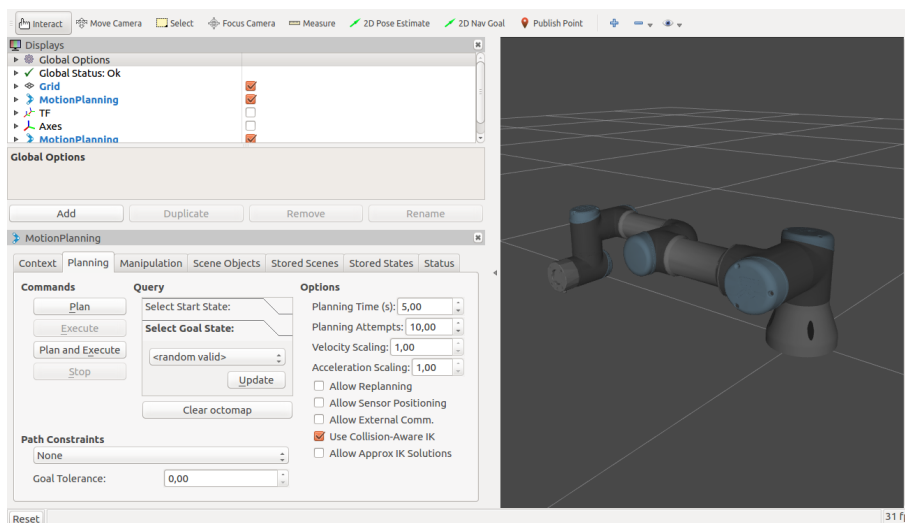


Figura 3.19: Plugin de *MoveIt* para *Rviz*

En general, es una herramienta que muestra información útil de una forma muy ordenada y fácil de acceder. En cambio, no se puede confundir con un simulador, ya que únicamente muestra la información que se le suministra sin hacer ningún tipo de cálculo. A partir del paquete de configuración *MoveIt* se puede ejecutar un controlador que simula en cierta medida el comportamiento del robot, pero se trata de un controlador genérico que no tiene en cuenta inercias, singularidades, y sobretodo, no es el mismo controlador que estará ejecutándose en el robot real, por lo que los resultados no son 100 % comparables.

Sin embargo, es de gran utilidad para verificar ciertas acciones antes de ejecutarlo sobre el robot real, ya que se pueden detectar otros problemas en la programación.

El *software* de simulación *Gazebo* es muy utilizado en entornos *ROS*, y de hecho el paquete *universal_robot* cuenta con la configuración para poderlo ejecutarlo. Aunque sea un simulador real, sigue teniendo la limitación de que no es exactamente el mismo controlador. Por esta razón, para realizar simulaciones se ha utilizado *URSim*.

URSim [79] es proporcionado directamente por *Universal Robots* y realmente se trata del propio sistema operativo del robot. Como la caja de control está basada en una distribución Linux, puede instalarse directamente en una distribución Linux, o la opción elegida, instalarlo en una máquina virtual independiente. Una vez instalado, únicamente hay que conectarse a la misma como si se tratara del robot real, a través del *driver* y especificando la dirección IP.

Las ventajas de utilizar *URSim* frente a otro simulador, es que la respuesta siempre será la más cercana a la realidad. Además, es posible descargarse la misma versión de *firmware* (en el caso del robot utilizado, ésta es la 3.5.4) que se tenga instalada en el robot real, para que la simulación sea lo más realista posible. Como desventaja se puede destacar que la interfaz de usuario es pobre en relación a otros simuladores, pero se puede subsanar ejecutando *RViz* para la visualización de movimientos u otra información que se considere relevante. En la Figura 3.20 se muestran dos capturas de la máquina virtual.

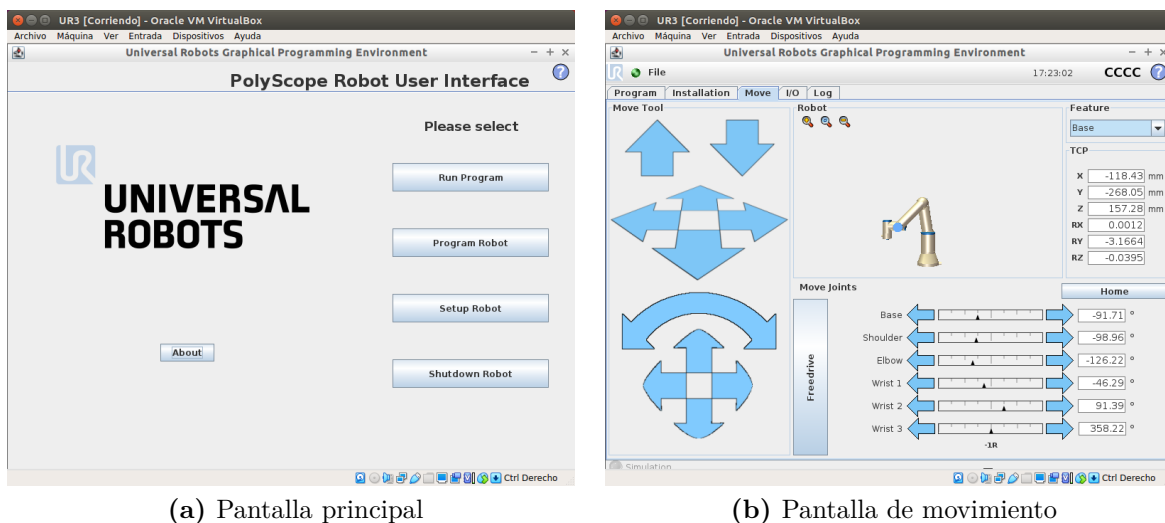


Figura 3.20: Capturas del simulador *URSim* ejecutándose en una máquina virtual

Capítulo 4

Sistemas de control para abordaje robótico en el campo endonasal

El capítulo cuarto cubre la descripción e implementación de los sistemas de control para abordaje robótico en el campo endonasal. Comenzará con una descripción del escenario de trabajo completo y presentación de los sistemas de referencia utilizados. A continuación se describirán los dos sistemas de control desarrollados: el primero de ellos está basado en MoveIt y será utilizado para movimientos grandes sin realimentación de fuerza y el segundo estará basado en URScript y sí contara con realimentación de fuerza.

4.1. Introducción

4.1.1. Descripción escenario completo

Antes de comenzar la descripción de los sistemas de control se va a dar una visión global del escenario de trabajo y los sistemas de referencia principales.

En la Figura 4.1 se presenta el escenario completo. El robot está realizando una maniobra de aproximación a la zona de intervención, alzado sobre la mesa gracias a la plataforma elevadora. El cráneo se encuentra anclado al soporte de manera sólida, y éste a los perfiles de aluminio fijados a la mesa. Se utiliza la segunda revisión del portaherramientas, que en este caso porta un prototipo impreso en 3D de la óptica y el endoscopio. Siguiendo el procedimiento quirúrgico, el endoscopio accedería por el orificio nasal izquierdo, razón por la cual el escenario se encuentra posicionado de tal manera. Finalmente, la estación de intercambio de herramientas se encuentra en la parte posterior.

Se pueden diferenciar 6 sistemas de referencia:

- */root*: tiene su origen en el punto anclaje entre la mesa y la plataforma elevadora del robot

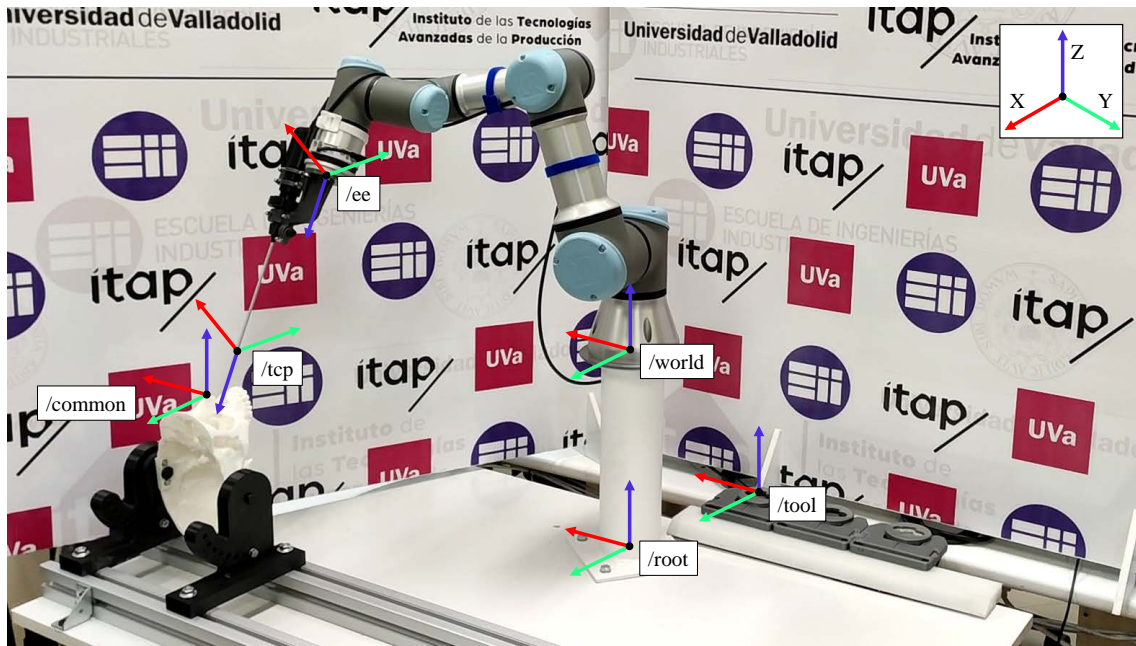


Figura 4.1: Escenario completo con sistemas de referencia

- */world*: es el sistema de referencia de la base del robot, siendo uno de los sistemas de referencia principales al que estarán referenciados muchos otros, por ejemplo, */root* comparte orientación con */world*
- */ee*: es el sistema de referencia del efector final del brazo robótico, coincidente con la brida de salida de la herramienta
- */tcp*: coincidirá con el extremo distal de la herramienta que se esté utilizando en un determinado momento, en este caso el endoscopio. Tiene la misma orientación que */ee*
- */common*: se encuentra en una muesca del modelo tridimensional de cráneo. Será el sistema de referencia común en el escenario conjunto del proyecto *Craneecal* cuando se utilicen dos brazos robóticos
- */tool*: sistema de referencia que coincide con el punto de anclaje de la primera estación del intercambiador de herramientas

4.1.2. Sistemas de control

La utilización de sistemas robóticos en entornos quirúrgicos requiere la adaptación de la estrategia de control a la intervención a realizar. En el caso que nos concierne, la extirpación de tumores en la base del cráneo, las diferentes etapas de la intervención requerirán un comportamiento del robot diferente. Por ejemplo, durante las fases de taladrado se debe tener un control de la fuerza que se está realizando, cuando se mueva por zonas

cercanas a la arteria carótida o el nervio óptico deberá evitarse o minimizar el contacto, cuando se retire la duramadre los movimientos deberán ser muy precisos y delicados, durante la extirpación del tumor se deberá permitir realizar más fuerza aunque por debajo de un umbral, etc. Además de la estrategia de control en la zona de operación, también se deberán desarrollar estrategias mientras no haya herramientas dentro del paciente, es decir, intercambio de herramientas, maniobras de aproximación, etc.

A lo largo de este cuarto capítulo se describirán las 2 estrategias de control implementadas y el trabajo desarrollado en torno a ellas:

1. Sistema de control sin realimentación de fuerza: movimientos libres, grandes o pequeños, basado en la biblioteca de movimiento de *MoveIt*
2. Sistema de control con realimentación de fuerza: sistema de posicionamiento para movimientos dentro del entorno quirúrgico que minimice los contactos y se oponga a ellos. Utiliza la biblioteca de movimiento *URScript* y el sensor de fuerza

4.2. Sistema de control sin realimentación de fuerza

El objetivo de este sistema de control es dotar al robot de una funcionalidad básica controlada a través del entorno *ROS*, implementando los comandos de movimiento que se encuentran tradicionalmente disponibles en los entornos de programación de brazos robóticos (movimientos en coordenadas articulares o cartesianas, movimientos lineales u optimizados, cambios de sistemas de referencia, control de velocidad y aceleración, etc.). También se desea aprovechar el potencial del entorno *MoveIt* para incorporar en la planificación de trayectorias la habilidad de evitar obstáculos, como pueden ser la mesa, la plataforma elevadora, el paciente y el propio robot.

4.2.1. Biblioteca de movimientos

La biblioteca de movimiento de *MoveIt* desarrollada consiste en una clase programada en *C++* que engloba todas las funciones y variables necesarias para su correcto funcionamiento, de forma que su mantenimiento y futuras ampliaciones sean sencillas.

ROS ofrece un número de variables o estructuras que facilitan el intercambio de información, como *geometry_msgs::Pose*, que sirve para representar una *pose*, es decir, una posición y orientación. Se trata de una estructura que tiene las siguientes subdivisiones:

<i>geometry_msgs::Pose pose</i>	<i>geometry_msgs::Point position</i>	<i>float64 x</i> <i>float64 y</i> <i>float64 z</i>
	<i>geometry_msgs::Quaternion orientation</i>	<i>float64 x</i> <i>float64 y</i> <i>float64 z</i> <i>float64 w</i>

La variable del tipo *float64* es otra convención de *ROS* para designar una variable de punto flotante con doble precisión *double*. Como se observa, la nomenclatura utilizada para designar una orientación en *ROS* es el cuaternio. El paquete *tf2* ofrece funciones que permiten realizar conversiones entre ángulos de Euler y cuaternios.

Las funciones pueden agruparse en los siguientes grupos: movimiento, transformaciones, información y otras. La descripción de todas las funciones y explicación de los detalles de implementación se escapan del alcance de este TFM, por lo tanto, únicamente se describirán aquellas funciones más importantes sin hacer hincapié en su implementación.

Funciones de movimiento

- *goto_joints(float64 { $\vartheta_1, \vartheta_2, \vartheta_3, \vartheta_4, \vartheta_5, \vartheta_6$ })*: mueve el robot a una configuración de ejes recibida como argumento, en forma de vector. Cada valor ϑ corresponde a un grado de libertad del robot. Las unidades esperadas son radianes
- *goto_position(geometry_msgs::Point position)*: mueve el *TCP* del robot a unas coordenadas XYZ recibidas como argumento sin restricciones de orientación, es decir, el robot puede tomar cualquier orientación, llevando a cabo una trayectoria óptima. Las unidades esperadas son metros, y estarán referidas al sistema de referencia de la base del robot, */world*
- *goto_position_straight(geometry_msgs::Point position)*: mueve el *TCP* del robot a unas coordenadas XYZ recibidas como argumento sin restricciones de orientación llevando a cabo una trayectoria en línea recta. Las unidades esperadas son metros, y estarán referidas al sistema de referencia de la base del robot, */world*
- *goto_pose(geometry_msgs::Pose pose)*: mueve el *TCP* del robot a una posición y orientación recibidas como argumento siguiendo una trayectoria óptima. Las unidades esperadas son metros y radianes, y estarán referidas al sistema de referencia de la base del robot, */world*
- *goto_pose_straight(geometry_msgs::Pose pose)*: mueve el *TCP* del robot a una posición y orientación recibidas como argumento siguiendo una trayectoria recta. Las

unidades esperadas son metros y radianes, y estarán referidas al sistema de referencia de la base del robot, */world*

- *offset(geometry_msgs::Pose pose)*: devuelve la posición y orientación del *TCP* del robot desplazado una posición y orientación recibidas como parámetro respecto al sistema de referencia */world*
- *offset_tcp(geometry_msgs::Pose pose)*: devuelve la posición y orientación del *TCP* del robot desplazado una posición y orientación recibidas como parámetro respecto al sistema de referencia */tcp*, transformado a coordenadas */world* para poder enviar la consigna al robot correctamente
- *set_speed_scaler(float64 scaler)*: fija un factor de escalado de velocidad. En *MoveIt* no existe la posibilidad de tener un control preciso sobre la velocidad de movimiento, sino que se aplica un factor de escala sobre la velocidad máxima del brazo robótico. Este factor de escalado toma un valor entre 0 y 1
- *set_acceleration_scaler(float64 scaler)*: fija un factor de escalado de aceleración. Al igual que con la velocidad, es la única forma de tener control sobre la aceleración. Este factor de escalado toma un valor entre 0 y 1

Funciones para transformaciones

- *from_tcp_to_world(geometry_msgs::Pose pose)*: convierte una posición y orientación recibido como argumento en coordenadas de */tcp* y lo convierte a */world*
- *from_world_to_common(geometry_msgs::Pose pose)*: convierte una posición y orientación recibido como argumento en coordenadas de */world* y lo convierte a */common*. Esta función únicamente es necesaria cuando se utilizan los dos brazos robóticos
- *frame_tf(string source, string target, geometry_msgs::Pose pose)*: lleva a cabo una transformación de una *pose* entre 2 sistemas de referencia arbitrarios
- *forward_kinematics(float64 { $\vartheta_1, \vartheta_2, \vartheta_3, \vartheta_4, \vartheta_5, \vartheta_6$ })*: lleva a cabo la cinemática directa y devuelve la posición y orientación del *TCP* respecto al sistema de referencia */world* a partir de la configuración de ejes
- *inverse_kinematics(geometry_msgs::Pose pose)*: lleva a cabo la cinemática inversa y devuelve la configuración de ejes a partir de la posición y orientación del *TCP* respecto al sistema de referencia */world*

Funciones de información

- *get_position()*: devuelve la posición del *TCP* respecto al sistema de referencia */world*

- *get_orientation()*: devuelve la orientación del TCP respecto al sistema de referencia */world*
- *get_pose()*: devuelve la posición y orientación del TCP respecto al sistema de referencia */world*
- *get_joints()*: devuelve la configuración de ejes actual

Otras funciones

Dentro de esta categoría se engloban funciones para depuración, sacar información en pantalla, añadir marcadores en *RViz*, comunicación a través de *topics* y otras funciones de control internas.

4.2.2. Flujo de trabajo y limitaciones

En la Figura 4.2 se muestra un esquema conceptual del flujo de información de *MoveIt*.

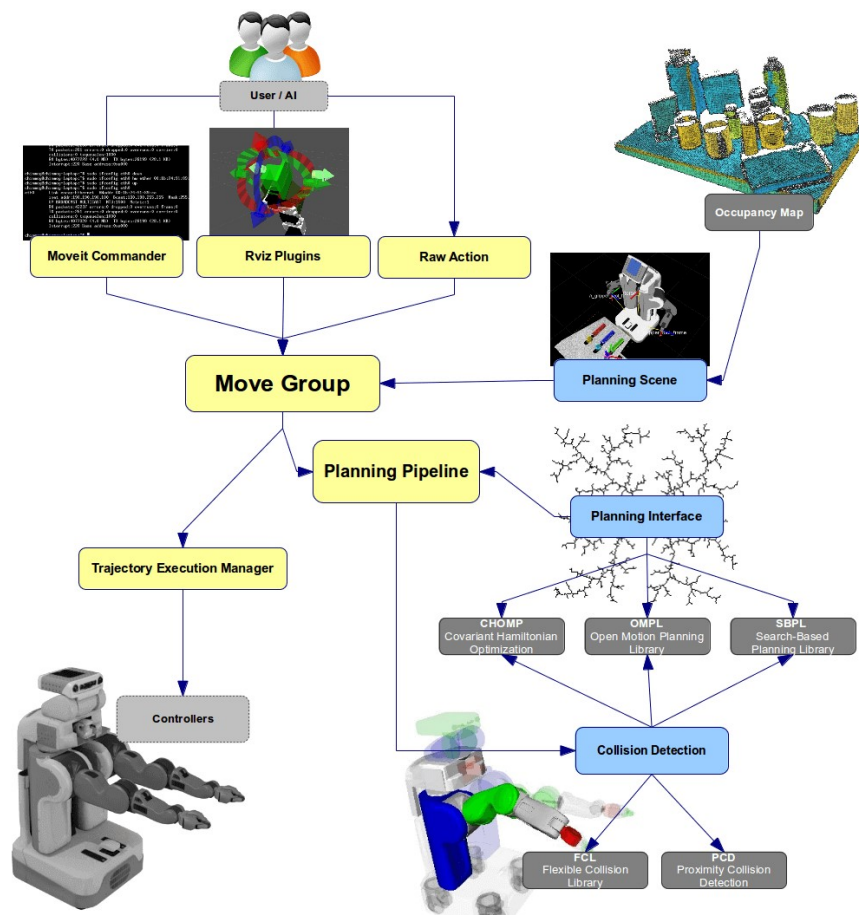


Figura 4.2: Flujo de trabajo de alto nivel de *MoveIt* [23]

La entrada al sistema es una petición por parte del usuario (*User / AI*), que a través de las diferentes *APIs* disponibles envía una petición al nodo *Move Group*. Este nodo tiene una función integradora, recibe peticiones y restricciones geométricas o de trayectoria, y genera las salidas adecuadas. *Planning Scene* se encarga de aportar información sobre los obstáculos que se hayan definido y una representación del estado actual del robot en su entorno. Una vez conocidas la consigna y las restricciones del entorno, se manda una petición a *Planning Pipeline*. El primer paso es garantizar que la trayectoria seguida no colisione con ningún elemento del entorno ni consigo mismo, esto es realizado por el bloque *Collision Detection*, que realiza los cálculos a través de un *plugin* seleccionado por el usuario, en este caso se utiliza la biblioteca *FCL (Flexible Collision Library)*. Conjuntamente se lleva a cabo la planificación de la trayectoria a través del bloque *Planning Interface*, que también se vale de un *plugin* externo para su ejecución. En esta caso se ha utilizado *OMPL (Open Motion Planning Library)*. Una vez se ha planificado la trayectoria, es enviada de vuelta a los bloques *Planning Pipeline* y *Move Group*, dónde es direccionada a *Trajectory Execution Manager* y finalmente enviado al controlador del robot.

El funcionamiento del sistema es muy bueno, y es capaz de planificar trayectorias que eviten los obstáculos de su entorno. En cambio, debido a la arquitectura del sistema, la comunicación entre nodos, y todos los cálculos que debe realizar, el tiempo de planificación es bastante elevado. Cuando en el escenario está presente únicamente el brazo robótico, el tiempo de planificación es bajo, no superando los 50 *ms*, en cambio, cuando se añaden el resto de elementos, los tiempos de planificación oscilan en torno a valores cercanos a 4 segundos.

El cálculo de colisiones está basado en modelos *STL (Stereolithography)* que define un modelo 3D a partir de la unión de triángulos. A mayor número de triángulos, mayor es el detalle, pero el cálculo de colisiones tiene un coste computacional mucho mayor, ya que se deben realizar más comprobaciones. En un esfuerzo para reducir el tiempo de planificación, se identificó el modelo más complejo del escenario, el brazo robótico. En *MoveIt* existen dos modelos diferentes, el modelo visual y el modelo de colisión. El objetivo es reducir la complejidad del modelo de colisión únicamente, reduciendo el tiempo de computación sin comprometer la apariencia visual.

El modelo del robot está constituido por 7 ficheros. En la Tabla 4.1a se muestran los triángulos asociados a cada uno. Observamos que el número total de triángulos son 38 925, mientras que *MoveIt* recomienda en su documentación que el computo total de triángulos no se superior a unos pocos miles.

Para reducir el detalle de estos modelos de colisión se ha utilizado el software *MeshLab*, concretamente la utilidad *Filters / Remeshing, Simplification and reconstruction / Quadratic Edge Collapse Decimation*. El resultado final fue fruto de prueba y error, buscando un equilibrio entre el número total de triángulos, y conservar su geometría. En la Tabla 4.1b se resumen los triángulos una vez aplicada la simplificación, y en la Figura 4.3 la comparación de ambos modelos de colisión.

Fichero	Número de triángulos	Fichero	Número de triángulos
<i>base.stl</i>	7 992	<i>base.stl</i>	600
<i>forearm.stl</i>	4 934	<i>forearm.stl</i>	1 000
<i>shoulder.stl</i>	7 865	<i>shoulder.stl</i>	1 499
<i>upperarm.stl</i>	10 263	<i>upperarm.stl</i>	1 549
<i>wrist1.stl</i>	3 672	<i>wrist1.stl</i>	1 000
<i>wrist2.stl</i>	3 671	<i>wrist2.stl</i>	999
<i>wrist3.stl</i>	528	<i>wrist3.stl</i>	528
TOTAL = 38 925		TOTAL = 7 175	
(a) Modelo original		(b) Modelo reducido	

Tabla 4.1: Número de triángulos de los modelos del robot

Se puede observar que las diferencias entre nivel de detalle son importantes, pero reproduce fielmente el volumen que ocupa el robot, que es, al fin y al cabo, lo que se necesita para el cálculo de colisiones. Esta reducción ha permitido reducir el tiempo de planificación a la ventana de 1 segundo, lo cual es aceptable.

Por esta razón, este sistema de control únicamente se utilizará para realizar movimientos sin restricciones temporales importantes y sobretodo, aquellas movimientos que requieran cálculo de colisiones con el entorno.

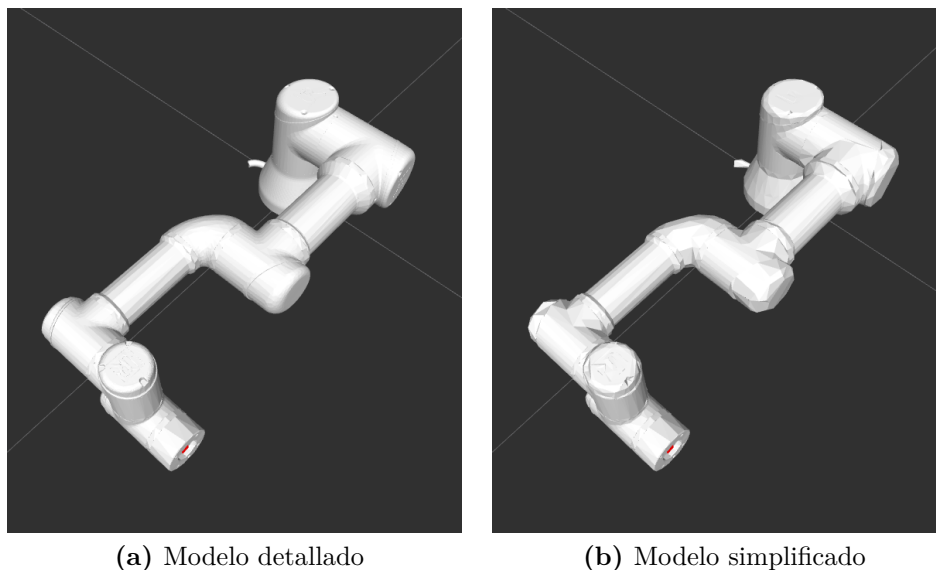


Figura 4.3: Comparación entre el modelo de colisión detallado y reducido

4.3. Sistema de control con realimentación de fuerza

En el apartado 4.2 se describía la utilización de *MoveIt* para la planificación de movimientos. Un enfoque utilizando *MoveIt* junto con *OMPL* como planificador de trayectorias es inviable para un sistema de control con realimentación de fuerza. El tiempo que transcurre desde el momento en el que se conoce el punto objetivo y el instante en el que el robot comienza a moverse es del orden de 1 segundo, dependiendo de la complejidad del movimiento. Un lazo de control con una frecuencia de actualización tan baja daría lugar a un comportamiento inestable por parte del robot, ya que no le daría tiempo a reaccionar a las fuerzas de interacción cuando están apareciendo, sino aquella que aparecieron hace un segundo. Además, el espacio de trabajo es tan pequeño (aproximadamente un cilindro de diámetro 27 mm y longitud 70 mm, tal y como se indicaba en el apartado 2.1.4), que el cálculo de colisiones no es necesario y únicamente incrementa el tiempo de computación. Por esta razón, es necesario explorar otras opciones de control.

A lo largo de este apartado se va a cubrir la creación de una biblioteca basada en los comandos *URScript*, la justificación de los comandos de movimientos utilizados, la descripción del sistema de control adoptado y finalmente la sintonización del controlador para un comportamiento estable.

4.3.1. Biblioteca *URScript*

La alternativa a *MoveIt* consiste en enviar comandos *URScript* a través de *sockets* al controlador del robot. Es la forma más directa de controlar el robot, y por lo tanto, la que permitirá la actualización de consignas a una mayor velocidad, idealmente, al máximo que permite el controlador del robot, 125 Hz.

En la documentación que facilita *Universal Robots* sobre *URScript* [77] se pueden encontrar varios comandos asociados al movimiento, siendo los más relevantes *movej*, *movel*, *servoj*, *speedj* y *speedl*. En la Tabla 4.2 se describe la funcionalidad de cada comando y los argumentos que recibe.

La diferencia entre los comandos *movej* y *servoj* puede no ser evidente a primera vista. El comando *movej* realiza un movimiento desde el punto en el que se encuentre el robot hasta la posición especificada por el usuario, creando una sucesión de puntos de paso y una curva de aceleración y deceleración de forma automática. En cambio, *servoj* es un control de menor nivel en el que únicamente se indica cuál es el siguiente punto al que debe moverse el robot, y es el usuario el responsable de crear una trayectoria acorde a la velocidad deseada, y de comenzar y terminar los movimientos de forma suave.

Para determinar qué comando de movimiento es el más óptimo, se realizó el siguiente experimento: se movió el robot a una posición inicial que le permitiera moverse libremente 100 mm en la dirección positiva del eje X. Se mandaron consignas utilizando los

Comando	Descripción	Argumentos
<i>movej</i>	Se mueve a una posición linealmente en coordenadas articulares	Se especifica la configuración de los ejes, la aceleración y velocidad del eje principal*, el tiempo (prioridad frente a aceleración y velocidad) y radio de giro (en el caso de encadenar varios comandos <i>movej</i>)
<i>movel</i>	Se mueve a una posición linealmente en coordenadas cartesianas	Se especifica la posición y orientación del <i>TCP</i> , la aceleración y velocidad lineal, el tiempo (prioridad frente a aceleración y velocidad) y radio de giro (en el caso de encadenar varios comandos <i>movel</i>)
<i>servoj</i>	Comando para control <i>online</i> del robot especificando una configuración de ejes	Se especifica la configuración de los ejes, el tiempo durante el cual el comando está controlando el robot, un valor de anticipación que suaviza la trayectoria y la ganancia con la cual se mueve a la configuración de ejes objetivo
<i>speedj</i>	Acelera linealmente en coordenadas articulares a una velocidad establecida	Se especifica la velocidad objetivo de cada uno de los ejes, la aceleración del eje principal* y el tiempo que está activo el comando de movimiento
<i>speedl</i>	Acelera linealmente en coordenadas cartesianas a una velocidad establecida	Se especifica la velocidad lineal del <i>TCP</i> , tanto posición como orientación, la aceleración lineal y el tiempo que está activo el comando de movimiento

*El eje principal dependerá del movimiento, y será aquel que tenga que recorrer el mayor espacio, y por lo tanto, tendrá la mayor velocidad y aceleración en comparación al resto de ejes

Tabla 4.2: Descripción de comandos de movimiento de *URScript*

diferentes comandos disponibles de forma que alcanzara la posición final de 100 *mm* a una velocidad constante de 10 *mm/s*. La consigna se actualizaba cada 8 *ms*, coincidiendo con una frecuencia de 125 *Hz*. Los resultados para cada comando fueron los siguientes:

- *movej* y *movel*: aunque los movimientos eran suaves, el robot no alcanzaba las consignas en el tiempo marcado, sino que tardaba más. Se podía observar una ralentización en la ejecución de las consignas a mediada que pasaba el tiempo. Cuando se detuvo la actualización de consignas, el robot aceleró hasta alcanzar la posición final. Por lo tanto, se puede concluir que la utilización de estos comandos no son aptos para un control en tiempo real. La justificación es que el sistema de control del robot planifica una trayectoria con curvas de aceleración y deceleración, y cuando se envía la siguiente consigna no le ha dado tiempo al robot a finalizar la anterior,

por lo tanto el sistema se satura. Además, se desea que el movimiento sea lineal y continuo, no con constantes cambios en la velocidad

- *servoj*: a la hora de ejecutar este comando, además de la configuración de ejes objetivo, se debe fijar 3 parámetros; *time* (tiempo), tiempo durante el cual el comando controla el robot; *lookahead time* (anticipación), permite suavizar o hacer más agresiva la trayectoria; *gain* (ganancia), valor proporcional con el cual intenta alcanzar la consigna. A pesar de haber fijado el tiempo a 8ms (correspondiente a una frecuencia de 125Hz) y modificar los valores de ganancia y anticipación, no se ha conseguido realizar movimientos fluidos. Se observaba una vibración constante que no es admisible para esta aplicación
- *speedj* y *speedl*: han sido los comandos que mejor han funcionado para un control en tiempo real, los movimientos eran suaves y predecibles. La principal desventaja es que el posicionamiento del robot es más complicado, ya que no se comanda directamente una posición, sino una velocidad. Eso condicionará el diseño del sistema de control tal y como se cubrirá en el siguiente a continuación

Por lo tanto, tras probar los diferentes comandos, finalmente se utilizará *speedl*, ya que ofrece los movimientos más suaves, y el hecho de trabajar en coordenadas cartesianas permite ahorrarse el paso de convertirlo a coordenadas articulares (aunque lo tenga que realizar el controlador del robot internamente).

4.3.2. Descripción sistema de control

El sistema de control con realimentación de fuerza puede englobarse dentro de un sistema de control posición-fuerza paralelo. Durante el preoperatorio se establecerá la trayectoria ideal que deberá seguir el endoscopio a lo largo de las diferentes etapas de la intervención, en cambio, puede que existan pequeñas discrepancias a la hora de realizar la intervención. Por este motivo, se desarrolla este sistema de control que estará activo mientras el robot porte el endoscopio, haciendo que se minimice el contacto entre las estructuras internas y éste.

Se pretende que tenga un comportamiento repulsivo, de forma que cuando detecte fuerza en una determinada dirección, se mueva en la dirección opuesta, disminuyendo el valor de esa fuerza. También puede entenderse como un sistema que desea mantener una consigna de fuerza de valor nulo. Es importante mencionar que, debido al pequeño espacio de trabajo, únicamente actuará sobre la posición y no la orientación, manteniéndose constante mientras esté activo este sistema de control. En la Figura 4.4 se muestra el esquema de la primera aproximación al sistema de control.

Los subíndices de todos los parámetros indican el sistema de referencia al que están referidos. El subíndice *W* se refiere a *world*, y corresponde a la base del robot; el subíndice

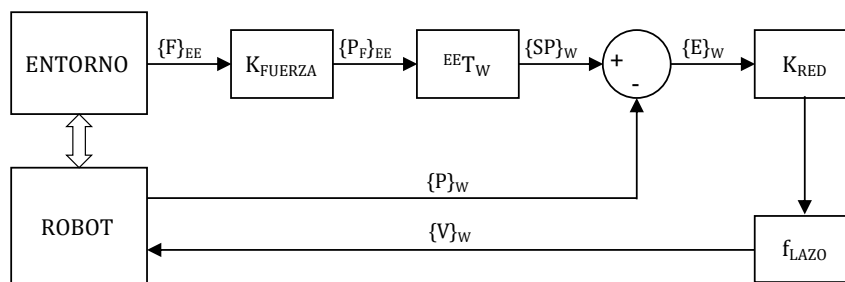


Figura 4.4: Primera versión del esquema de control con realimentación de fuerza

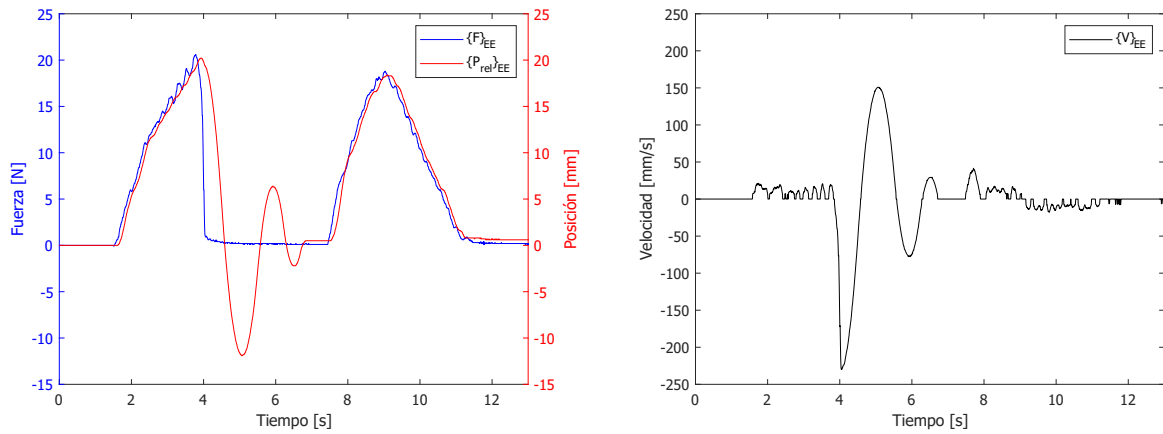
EE atiende a *end effector* y corresponde al sistema de referencia asociado a la brida de salida de la herramienta. Los sistemas de referencia $/ee$ y $/tcp$ tienen la misma orientación, y se asume que el conjunto porta-herramientas y endoscopio es rígido, por lo que las lecturas de fuerza leídas en $/ee$ corresponden a las fuerzas ejercidas en $/tcp$, es decir, el extremo del endoscopio.

Fruto de la interacción entre el robot y el entorno, aparecen unas fuerzas de contacto que se reflejan en la lectura del sensor de fuerza $\{F\}_{EE}$. Se realiza una transformación fuerza-posición a partir de la constante K_{FUERZA} , que tendrá unidades de mm/N . Cuanto mayor sea el valor de esta constante, menor será la fuerza necesaria para producir un mismo desplazamiento. Este desplazamiento $\{P_F\}_{EE}$ es convertido a coordenadas en el sistema de referencia $/world$, constituyendo el *setpoint* o consigna del controlador, designado por $\{SP\}_W$.

Si se resta el *setpoint* y la posición actual del robot respecto a $/world$ $\{P\}_W$, se obtiene el error de posicionamiento del sistema, $\{E\}_W$. Este error es multiplicado por la constante K_{RED} . El valor de esta constante será menor que la unidad, y su objetivo es minorar el valor de error que posteriormente es convertido a velocidad multiplicando por la frecuencia de muestreo del lazo ($125 s^{-1}$). Es necesario minorar este valor de error porque, si no, trataría de alcanzar la consigna con mucha velocidad, además de convertir el sistema en inestable. De esta forma, se asegura que la velocidad con la que se alcanza el valor de consigna es menor y con una menor diferencia entre el valor máximo y el mínimo. Finalmente, la velocidad $\{V\}_W$ es comandada al robot y se cierra el lazo de control.

Para la primera prueba se fijó una constante de reducción K_{RED} de valor 0,1, una constante K_{FUERZA} de valor $1 mm/N$ y una aceleración de $100 mm/s^2$. También se estableció un umbral de $\pm 0,5 mm$, de forma que no hubiera oscilaciones en torno a la posición de equilibrio, debido al ruido de la señal de fuerza. En las gráficas se va a utilizar la nomenclatura $\{P_{rel}\}_{EE}$ para facilitar la interpretación de las mismas. Indica la posición relativa del TCP respecto a la posición inicial en el sistema de referencia $/ee$. En la primera prueba se ha aplicado una fuerza de aproximadamente $20 N$ de forma progresiva en el eje Y (elegido de forma arbitraria), para posteriormente hacerla desaparecer de forma súbita, posteriormente, se ha aplicado y retirado la misma fuerza de forma progresiva en ambos movimientos. En la Figura 4.5a se pueden observar los resultados, representando en azul

la fuerza leída por el sensor de fuerza, en rojo la posición relativa del *TCP* respecto a la posición inicial en coordenadas $/ee$ y finalmente en la Figura 4.5b se representa la velocidad comandada en coordenadas $/ee$.

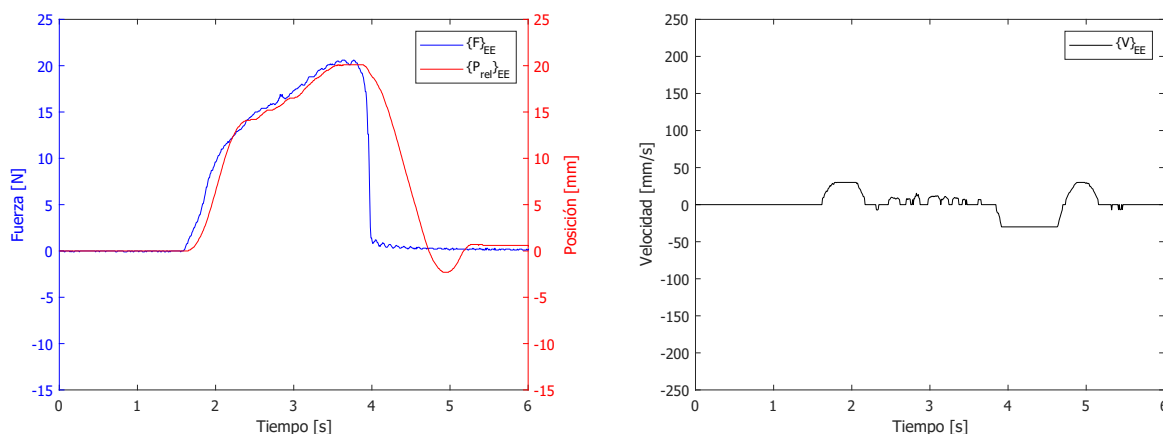


(a) Representación de la fuerza frente a la posición

(b) Representación de la velocidad

Figura 4.5: Reacciones antes variaciones de fuerza de gran valor

Los resultados muestran que si la fuerza cesa de manera repentina, se produce una oscilación cuando el robot vuelve a la posición de equilibrio. En cambio, si la fuerza decremента su valor de forma paulatina, no aparece ese efecto, y el robot sigue la fuerza de forma precisa. Otro efecto que se observa cuando se produce un gran cambio en la fuerza, son unos elevados valores de velocidad al intentar alcanzar la posición de equilibrio. En la siguiente prueba se limitó esta velocidad a ± 30 mm/s, tal y como se muestra en la Figura 4.6.



(a) Representación de la fuerza frente a la posición

(b) Representación de la velocidad

Figura 4.6: Aplicación de un umbral de velocidad de 30 mm/s

Se puede observar que para un mismo cambio en la consigna de 20 N, el sobrepaso tiene una magnitud significativamente menor (de 11.9 mm a 2.3 mm). Naturalmente, se observa un

retardo a la hora de alcanzar la posición objetivo debido a la limitación en la velocidad, en cambio el tiempo de establecimiento ha mejorado, pasando de aproximadamente 3 segundos a 1 segundo.

Estos dos experimentos no tienen mucho valor en términos de un ajuste fino del controlador, ya que se aplica una fuerza de forma manual, y nunca tendrá ni el mismo valor máximo, ni se alcanzará en el mismo instante, pero si que permite obtener las siguientes conclusiones:

- Un controlador proporcional (la constante K_{RED}) no es suficiente para controlar el sistema, por lo tanto, en las siguientes iteraciones se cambiará el bloque por un controlador PID
- Saturar la salida de velocidad ha resultado ser efectivo para controlar el sistema cuando la variación en el valor de consigna es muy grande
- El comportamiento repulsivo es el deseado

En la siguiente iteración se ha sustituido el bloque K_{RED} por un controlador PID, además de añadir la posibilidad de fijar una referencia de posición. El esquema se muestra en la Figura 4.7.

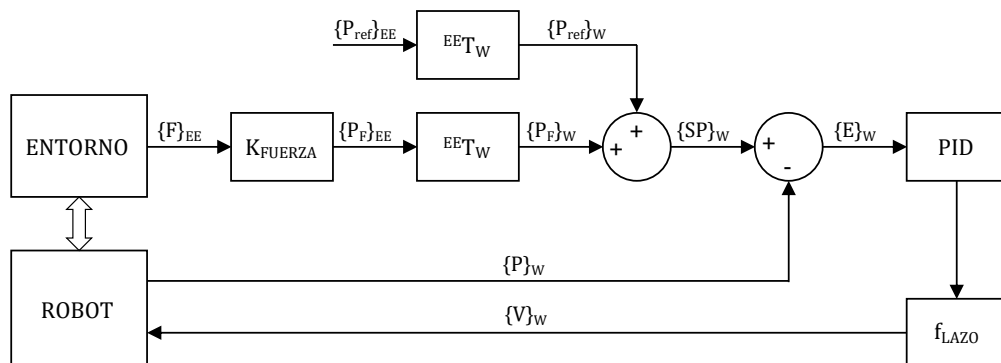


Figura 4.7: Segunda versión del esquema de control con realimentación de fuerza

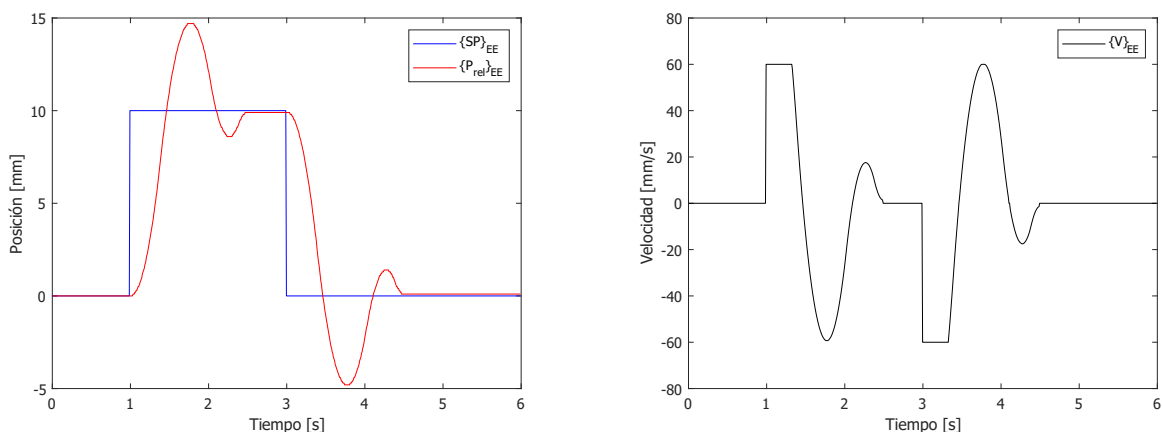
La referencia de posición $\{P_{ref}\}_{EE}$ es especificada bajo el sistema de referencia $/ee$, posteriormente es convertido a coordenadas en el sistema de referencia $/world$, $\{P_{ref}\}_W$. Esta posición es sumada a la obtenida tras la conversión fuerza-posición, designada por $\{P_F\}_W$, obteniendo la consigna final. Inicialmente, tanto la fuerza como la posición tienen el mismo peso en la consigna final, aunque esto podría ajustarse fácilmente aplicando factores que ponderar la acción de cada una. Una vez obtenido el *setpoint*, obtenemos el error que constituye la entrada del controlador PID.

La sintonización del controlador PID se cubre en el siguiente apartado.

4.3.3. Sintonización controlador

El método utilizado para la sintonización del controlador ha sido el método de prueba y error. Para ello, se va a someter al sistema a una entrada escalón de 10 mm , después se esperará dos segundos, y se volverá a una consigna de posición nula. El sistema de control únicamente está pensado para realizar pequeños movimientos incrementales, y por lo tanto únicamente se buscará garantizar su estabilidad en saltos en la consigna de valor $\pm 10\text{ mm}$.

El punto de partida es un valor proporcional igual a $0,1$ sin acción integral ni derivativa. Se ha decidido reducir el umbral a $0,1\text{ mm}$, que coincide con el valor de precisión del brazo robótico para que el posicionamiento sea más preciso; la aceleración se ha mantenido en 100 mm/s^2 y el valor de saturación se incrementado a 60 mm/s para mejorar la capacidad de reacción del sistema. En la Figura 4.8 se muestra la respuesta a la entrada escalón de valor 10 mm .



(a) Representación de la fuerza frente a la posición

(b) Representación de la velocidad

Parámetros

P	I	D	Umbral [mm]	Aceleración [mm/s^2]	Saturación [mm/s]
0,1	0	0	0,1	100	60

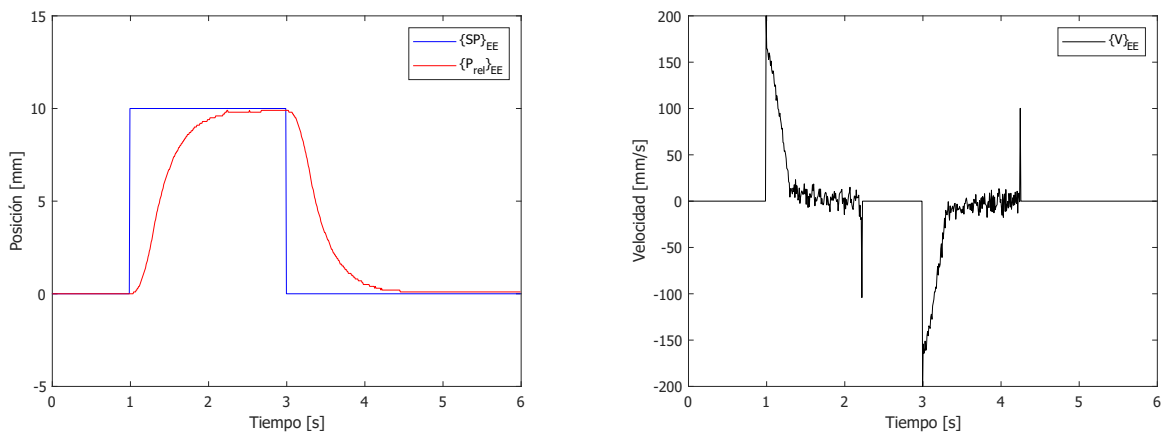
Figura 4.8: Respuesta del controlador P antes una entrada escalón

En la respuesta obtenida ante la entrada escalón de 10 mm se observan dos oscilaciones antes de alcanzar la posición objetivo. El sobrepaso tiene un valor de $4,7\text{ mm}$ y el tiempo de establecimiento es aproximadamente $1,5$ segundos. La velocidad satura a 60 mm/s en la primera oscilación, lo cual contribuye a limitar la amplitud de la segunda.

Desde ese punto de partida, se mantuvo el valor proporcional y se aumento la acción integral en incrementos muy pequeños. No se consiguió obtener una respuesta estable cuando se utilizaba el término integral, por muy pequeño que fuera. Una posible explicación es la señal de ruido del sensor de fuerza, ya que independientemente de si ha llegado o no a la posición objetivo, siempre va a haber oscilaciones debido al ruido, y el sumatorio de error

toma un valor muy alto rápidamente, desestabilizando el sistema.

Por esta razón, se utilizará un controlador PD en vez de un PID. En cambio, como el término derivativo es muy sensible al ruido, este no podrá tomar un valor demasiado elevado. Siguiendo con el método de prueba y error, se llegó a una solución que garantiza la estabilidad del sistema sin sobrepico, aunque con un tiempo de establecimiento de aproximadamente 1 segundo. En la aplicación que nos concierne es más importante que no se produzca el sobrepico pese a demorarse más en alcanzar la consigna. En la Figura 4.9 se muestran los resultados.



(a) Representación de la fuerza frente a la posición

(b) Representación de la velocidad

Parámetros

P	I	D	Umbral [mm]	Aceleración [mm/s ²]	Saturación [mm/s]
0,13	0	0,032	0,1	80	200

Figura 4.9: Respuesta del controlador PD antes una entrada escalón

El valor proporcional toma un valor de 0,13 y el valor derivativo 0,032. Se han realizado dos cambios adicionales: se ha aumentado la velocidad de saturación hasta 200 mm/s con el objetivo de que el primer tramo de la respuesta sea más rápida, aún así, se ha limitado a 80 mm/s² la aceleración, de forma que los movimientos sean más suaves. Como se aprecia en la Figura 4.9a se elimina por completo el sobrepaso y las oscilaciones. En cambio, la Gráfica 4.9b representa perfectamente uno de los efectos negativos de utilizar el término derivativo, la señal de control es mucho mas ruidosa, y por lo tanto, eso se traslada en pequeñas vibraciones. Un valor derivativo de 0,032 ha sido una solución de compromiso entre el nivel de ruido de la señal de velocidad y la eliminación de la oscilación y limitación del tiempo de establecimiento.

Los resultados obtenidos en estos experimentos llevaron a la última iteración a la hora de sintonizar el sistema, que fue añadir una etapa de filtrado de la señal de fuerza.

4.3.4. Filtrado señal de fuerza

En el diseño de la etapa de filtrado de la señal de fuerza se ha utilizado el software *Matlab* junto con *Simulink* para realizar una simulación de la acción de filtrado. Respecto a la señal de entrada, se han buscado representar 3 escenarios: señal sin aplicar ninguna fuerza externa, aplicación de fuerza paulatina y aplicación de fuerza repentina, de esta forma, se obtiene más información de cómo afecta la etapa de filtrado al sistema. Se van a hacer pruebas con los siguientes filtros:

- Filtro de la media móvil: se trata de un filtro digital muy utilizado en el procesamiento de señales debido a la facilidad de implementación y la baja complejidad computacional. La salida de un filtro de la media móvil es la media de los últimos N valores almacenados en una ventana de tamaño N , fijada por el usuario. Es bastante eficaz a la hora de eliminar ruido de alta frecuencia. También existe el filtro de la mediana móvil que es más inmune a valor atípicos muy alejados del valor real, aunque en este caso no se da ese escenario
- Filtro paso bajo: es un filtro que deja pasar las componentes de baja frecuencia de una señal, y atenúa las de alta frecuencia. El usuario fija la frecuencia de corte a partir de la cual comienza la atenuación

Cuanto menor sea la frecuencia de corte del filtro paso bajo, o cuanto mayor sea el tamaño de la ventana del filtro de la media móvil, mejor será el filtrado, pero mayor será el retraso que inducirá, por lo tanto se va a tener que tomar una solución de compromiso entre el nivel de filtrado y el retraso inducido.

Para el filtro paso bajo se muestran los resultados para las siguientes frecuencias de corte: 20 *Hz* (Figura 4.10), 10 *Hz* (Figura 4.11), 5 *Hz* (Figura 4.12) y 3 *Hz* (Figura 4.13).

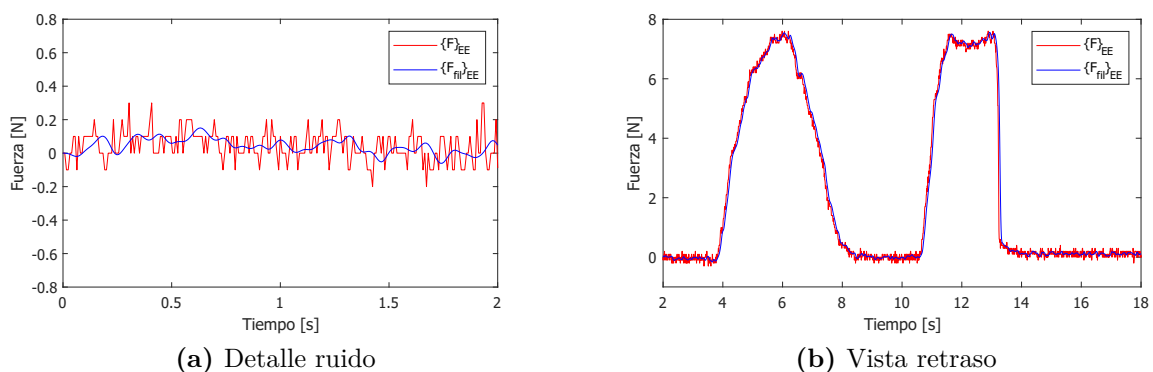


Figura 4.10: Respuesta ante filtro paso bajo con frecuencia de corte 20 *Hz*

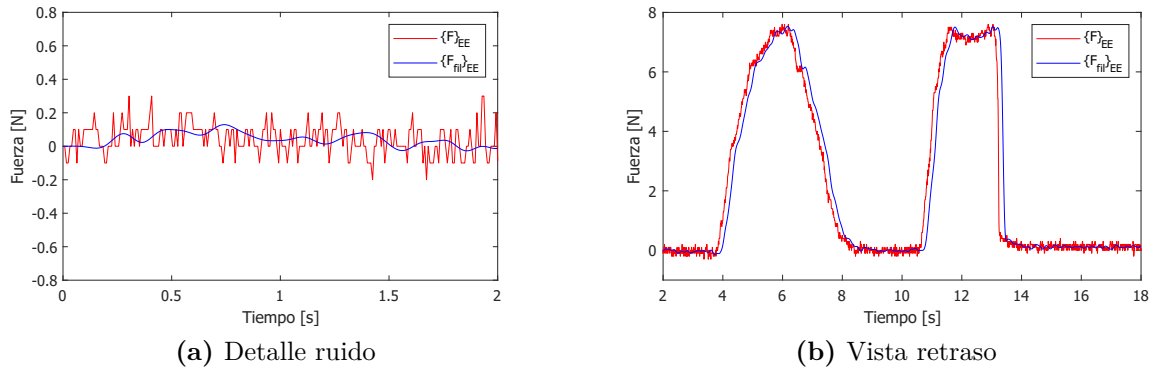


Figura 4.11: Respuesta ante filtro paso bajo con frecuencia de corte 10 Hz

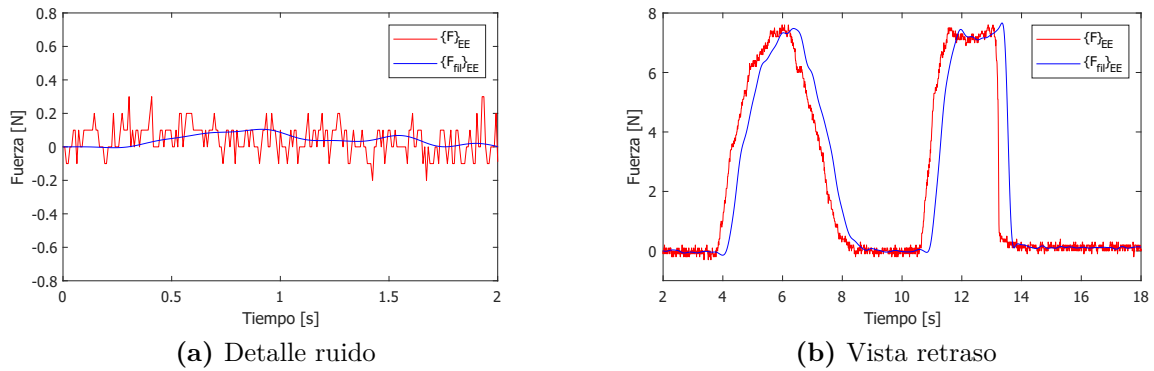


Figura 4.12: Respuesta ante filtro paso bajo con frecuencia de corte 5 Hz

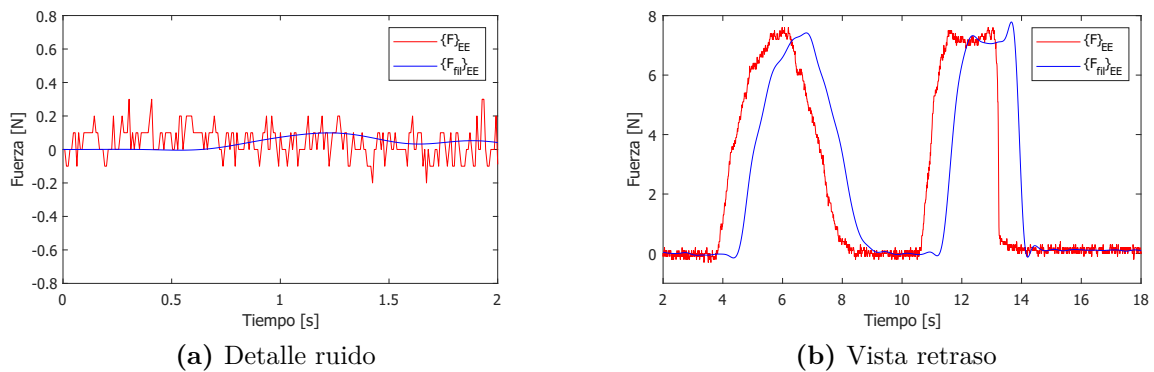


Figura 4.13: Respuesta ante filtro paso bajo con frecuencia de corte 3 Hz

Los resultados muestran lo que se mencionó anteriormente. El filtrado es más efectivo a medida que disminuye la frecuencia de corte, aunque a costa de aumentar el retraso. Este retraso desplaza hacia la derecha la curva de fuerza, sin que haya un retraso adicional cuando los saltos en la señal de fuerza es de mayor magnitud.

Se ha cuantificado el retraso en el instante $t = 13,24$ s, que toma un valor de $0,9$ N en la señal sin filtrar. Se ha elegido ese punto porque es el más crítico en el filtro de la media móvil, y permitirá realizar una comparación más justa entre ambos. En el mismo se produce un salto de 8 N en un intervalo de tiempo muy pequeño, por lo que se puede considerar como el retraso máximo que puede aparecer. También se ha cuantificado el nivel de ruido a través del valor pico a pico de la señal filtrada en el intervalo de tiempo de 0 a 2 segundos. En la Tabla 4.3 se muestran los resultados.

f_{corte} [Hz]	20	10	5	3
$t_{retraso}$ [ms]	80	200	400	810
F_{pk-pk} [N]	0,21	0,16	0,11	0,10

F_{pk-pk} [N] de la señal de fuerza sin filtrar = $0,50$ N

Tabla 4.3: Nivel de ruido y retraso al utilizar filtro paso bajo

A continuación se muestran las gráficas utilizando el filtro de la media móvil con la misma entrada de datos. Se han utilizado tamaños de ventana 16 (Figura 4.14), 32 (Figura 4.15), 64 (Figura 4.16) y 128 (Figura 4.17).

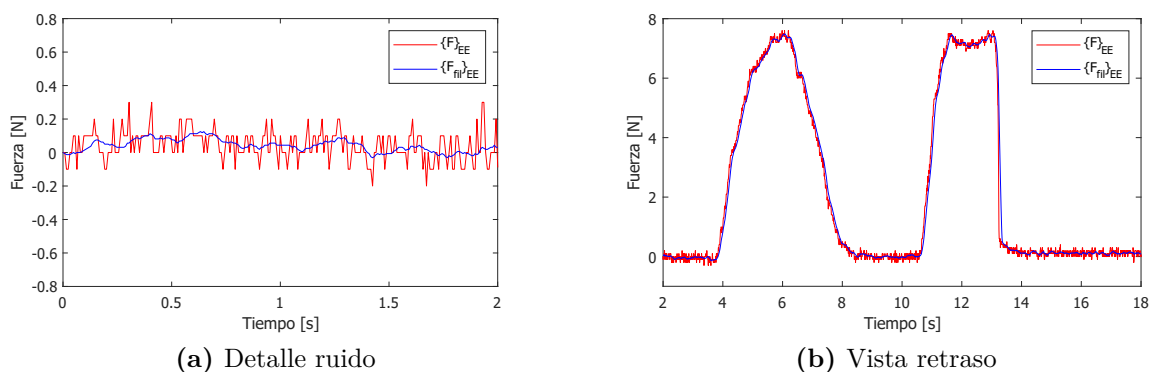


Figura 4.14: Respuesta ante filtro de la media móvil con tamaño de ventana 16

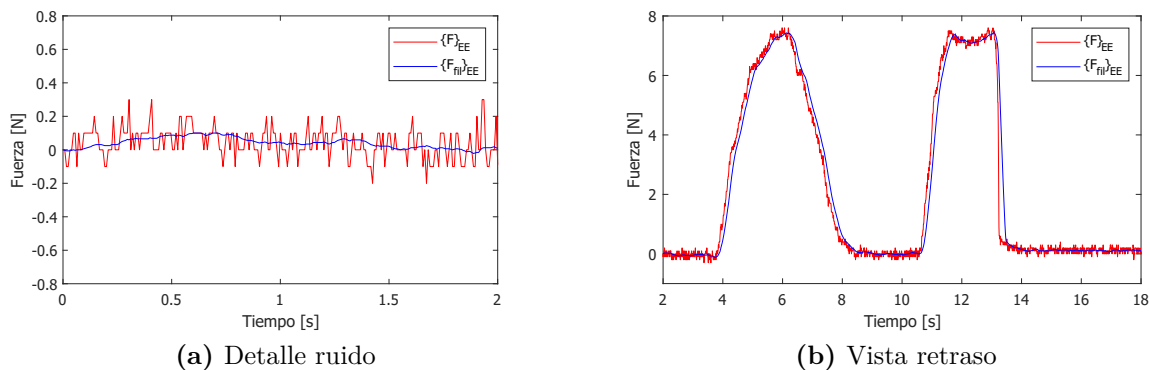


Figura 4.15: Respuesta ante filtro de la media móvil con tamaño de ventana 32

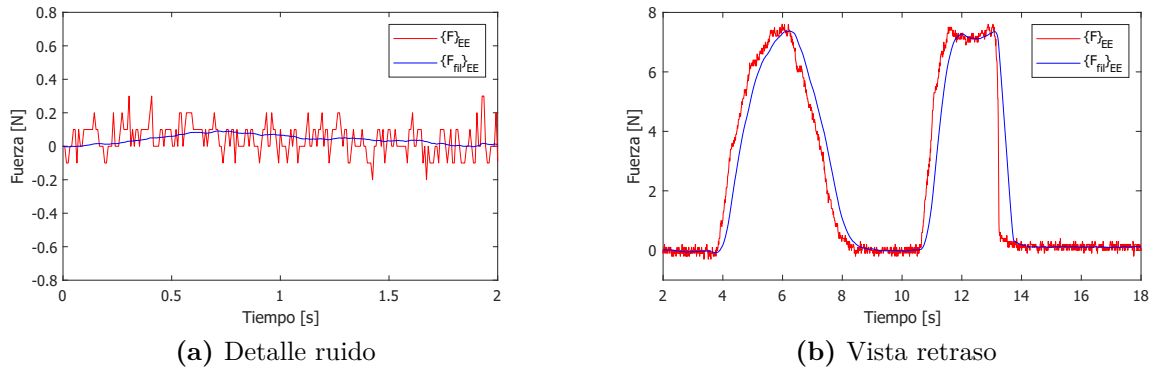


Figura 4.16: Respuesta ante filtro de la media móvil con tamaño de ventana 64

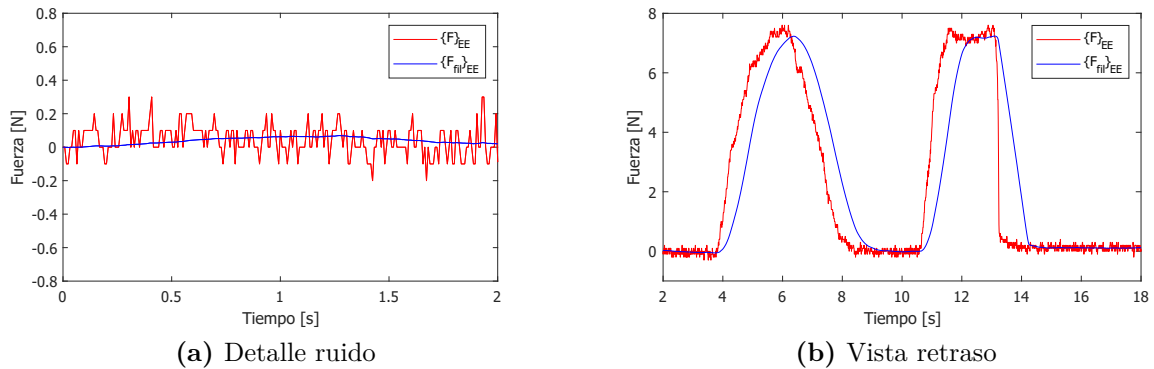


Figura 4.17: Respuesta ante filtro de la media móvil con tamaño de ventana 128

Encontramos un comportamiento diferente respecto al filtro paso bajo. Mientras que este último desplazaba la curva de fuerza hacia la derecha sin importar la velocidad con la que ésta cambiaba de valor, el filtro de la media móvil sí es sensible a estos cambios de velocidad, sobretodo cuando el tamaño de la ventana de filtrado es alto. Este comportamiento es esperado, ya que al realizar la media de la ventana, cuando se produce un cambio repentino, pasará un tiempo hasta que el valor de la media converja al valor real. En la Tabla 4.4 se cuantifican los niveles de ruido y retraso para poderlo compararlo con el filtro paso bajo.

$tam_{ventana}$	16	32	64	128
$t_{retraso} [ms]$	90	200	430	880
$F_{pk-pk} [N]$	0,16	0,12	0,09	0,07

$F_{pk-pk} [N]$ de la señal de fuerza sin filtrar = 0,50 N

Tabla 4.4: Nivel de ruido y retraso al utilizar filtro de la media móvil

El retraso es ligeramente superior utilizando el filtro de la media móvil, excepto para un

tamaño de ventana de 32, que son iguales. En cambio, el nivel de ruido es inferior en todos los casos. La tendencia muestra que para un mismo nivel del ruido, el filtro de la media móvil tiene un menor retraso. Además, hay que tener en cuenta que se está midiendo el retraso en el punto más desfavorable para éste último filtro, por lo que en una situación de trabajo real, en los que el salto en los valores de fuerza son más pequeño, el retraso sería menor.

Por esta razón, por la facilidad de implementación y baja complejidad computacional, se ha decidido utilizar el filtro de la media móvil con un tamaño de ventana de 32.

Por lo tanto, añadiendo la etapa de filtrado, la versión final del sistema de control se muestra en la Figura 4.18.

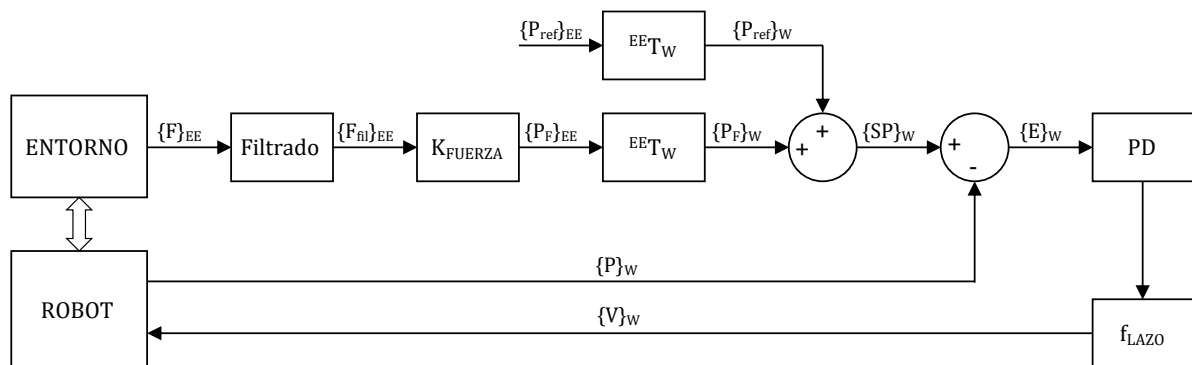


Figura 4.18: Versión final del esquema de control con realimentación de fuerza

Únicamente se ha añadido la etapa de filtrado antes de hacer la conversión entre la fuerza y la posición. La señal filtrada tiene la nomenclatura $\{F_{fil}\}_{EE}$

Capítulo 5

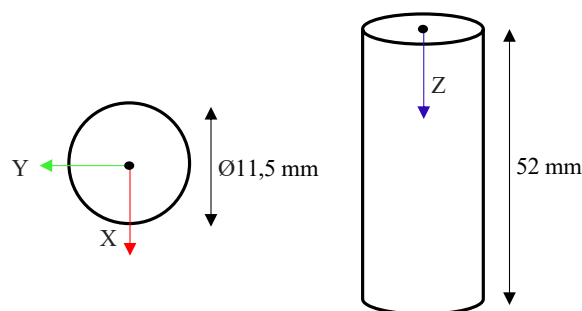
Validación sistema de control con realimentación de fuerza

En este quinto capítulo se lleva a cabo la validación del sistema de control con realimentación de fuerza a través de la simulación de escenarios de colisión y contacto. Esto permitirá terminar de ajustar los parámetros del lazo de control, así como verificar su comportamiento.

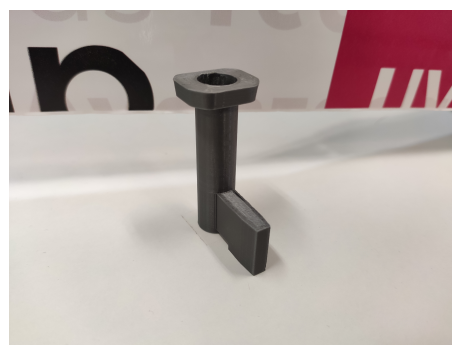
5.1. Introducción

Con el objetivo de validar el algoritmo de fuerza se han planteado diferentes escenarios de colisión y contacto que imitan posibles situaciones durante una cirugía. Esto ha permitido sintonizar el valor de la constante K_{FUERZA} , que relaciona la fuerza y el avance.

Estos escenarios se imprimirán en 3D y compartirán un mismo esqueleto que se ancla en el modelo del cráneo. En la Figura 5.1 se muestran las dimensiones de un escenario vacío y el modelo impreso en 3D.



(a) Dimensiones de los escenarios



(b) Modelo de colisión impreso en 3D

Figura 5.1: Esqueleto de escenarios de colisión

La geometría de los escenarios estará definida en el interior del cilindro. El saliente que se aprecia en la Figura 5.1b únicamente tiene el objetivo de mejorar el anclaje escenario-cráneo. El sistema de coordenadas corresponde a $/tcp$, coincidiendo con los ejes de las lecturas de fuerza.

En la Figura 5.2 se muestran dos imágenes con el escenario vacío posicionado sobre el cráneo.

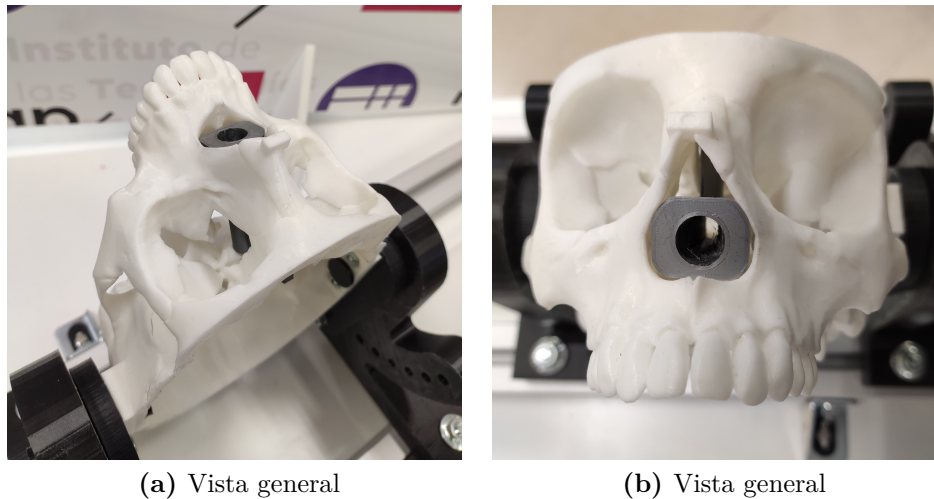


Figura 5.2: Escenario montado sobre el modelo del cráneo

Durante los ensayos se han recuperado los siguientes datos en cada uno de los 3 ejes:

- Lectura del sensor de fuerza $\{F\}_{EE}$
- Señal de fuerza filtrada, utilizando un filtro de la media móvil, con un tamaño de ventana de 32 $\{F_{fil}\}_{EE}$
- Referencia de posición externa referida al sistema de referencia del endoscopio $\{P_{ref}\}_{EE}$
- Posición relativa del endoscopio respecto al punto inicial $\{P_{rel}\}_{EE}$

Los parámetros de control utilizados se muestran la Tabla 5.1.

Valor proporcional de controlador: 0,13	Aceleración: 80 mm/s^2
Valor integral del controlador: 0	Saturación de velocidad: $\pm 200 \text{ mm/s}$
Valor derivativo del controlador: 0,032	Umbral de fuerza: 0,1 N
Constante de fuerza K_{FUERZA} : 5 mm/N	Velocidad de movimiento: $2,5 \text{ mm/s}$

Tabla 5.1: Parámetros de control utilizados en los escenarios

5.2. Escenarios

5.2.1. Escenario 1

La geometría del escenario 1 se muestra en la Figura 5.3. Se pretende simular una situación en la cual el endoscopio se ha desviado ligeramente de su trayectoria y por lo tanto está rozando con las paredes. La apertura final es de $4,5 \text{ mm}$, suficiente para que pase el endoscopio de 4 mm de diámetro. El ángulo de contacto es pequeño, por lo tanto, las fuerzas que aparecen son pequeñas también.

Para forzar el contacto con la pared, el punto inicial de la trayectoria se encuentra desviado $3,8 \text{ mm}$ en la dirección positiva del eje Y. La flecha de color blanco y trazo discontinuo muestra la referencia de posición que se va a comandar al robot, paralela al eje Z.

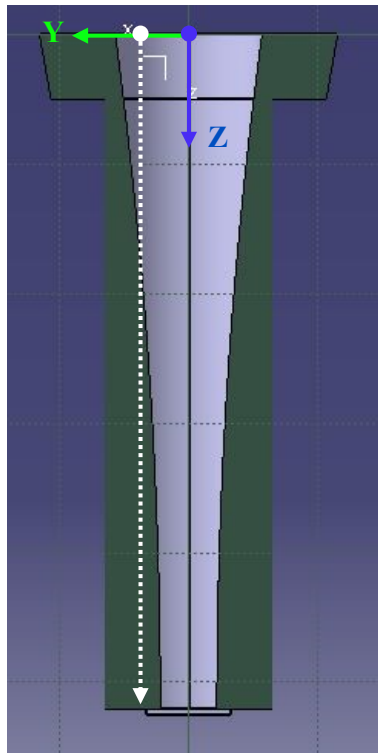


Figura 5.3: Sección del escenario 1

Las fuerzas de reacción que aparecen son negativas, tanto en el eje Y como en el eje Z, y hay fuerza nula sobre el eje X, por lo tanto, se omitirá la gráfica en ese eje por no aportar información.

En la Figura 5.4 se muestran dos gráficas, la primera en el eje Y, y la segunda en el eje Z. En azul se representa la fuerza filtrada, en color rojo y línea continua la posición relativa del robot respecto al punto inicial y en discontinua la referencia de posición.

Comenzando el análisis con el eje Y, observamos que a medida que aumenta el tiempo, el robot se desvía de su referencia de $3,8 \text{ mm}$ y llega hasta aproximadamente 3 mm . Este

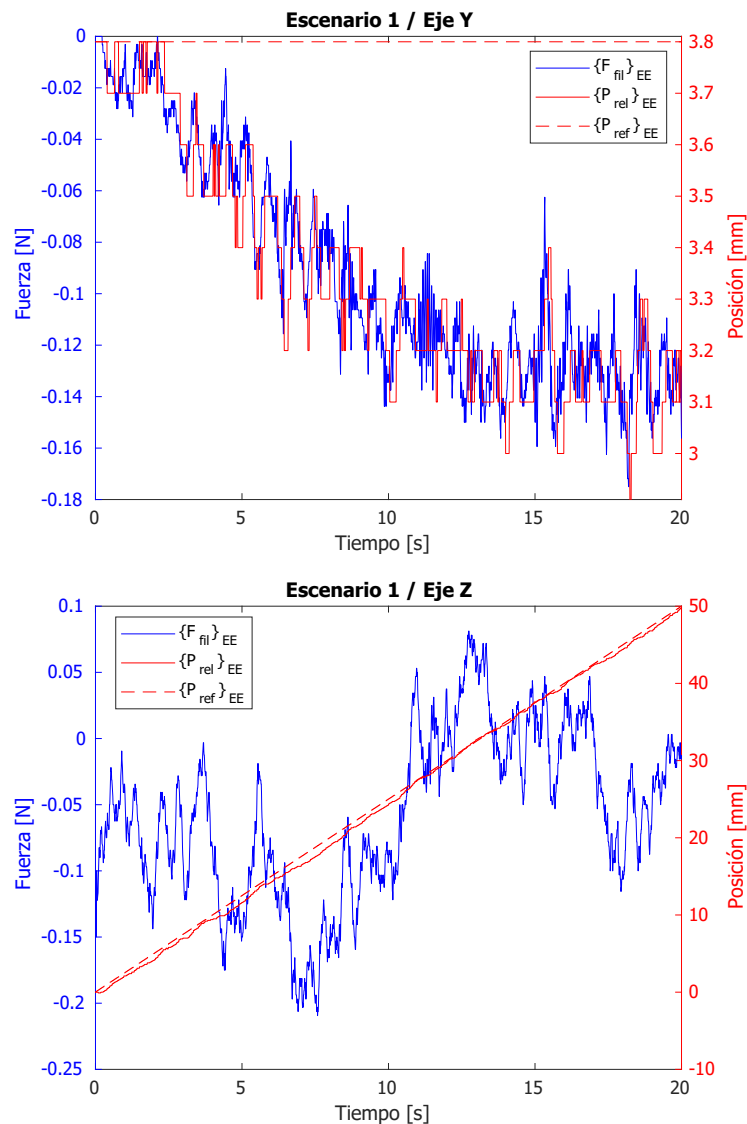


Figura 5.4: Representación de la fuerza

comportamiento es deseable y esperable, ya que muestra que se está desplazado hacia la derecha en oposición a la rampa.

También se observa que tanto la señal fuerza como de posición no es una línea de pendiente negativa constante, si no que se observa un escalonamiento y subidas y bajadas en sus valores. La razón de este comportamiento se intenta justificar en la Figura 5.5. Al igual que en las gráficas anteriores, la línea roja discontinua indica la referencia de posición, y la línea continua la posición real del robot.

Cuando el endoscopio encuentra un obstáculo, se mueve en oposición al mismo, en la dirección normal a la superficie de contacto. En el supuesto de la Figura 5.5, cuando hay contacto, se produce un movimiento hacia a la derecha y hacia arriba. Una vez desaparece la fuerza, el robot vuelve a seguir la referencia, moviéndose otra vez hacia la izquierda y abajo, colisionando con la pared.

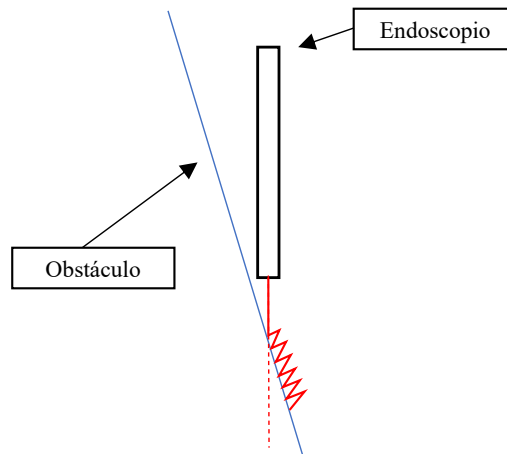


Figura 5.5: Escalonamiento de la fuerza y la posición

Este es el motivo por el cual se aprecian esas oscilaciones en los valores de fuerza y posición. En el caso real, la fuerza nunca llega a hacerse cero, sino que sigue aumentando. Esto es debido a que a medida que pasa el tiempo, el robot se desvía más de la referencia, y, por lo tanto, la acción de recuperación es más grande que la de reacción.

Respecto a la gráfica en el eje Z, los resultados no son tan evidentes. Cabría esperar una fuerza de tendencia negativa, al igual que la recogida en el eje Y. En cambio, debemos tener en cuenta los siguientes factores:

- La fuerza soportada en el eje Z es significativamente menor. Debido a la inclinación de la rampa por la que desliza el endoscopio, hace que las fuerzas en el eje Y sean 15 veces mayores a las fuerzas en Z, tal y como se demuestra en la Figura 5.6
- Las fuerzas que se están ejerciendo están muy por debajo de la resolución del sensor de fuerza en ese eje

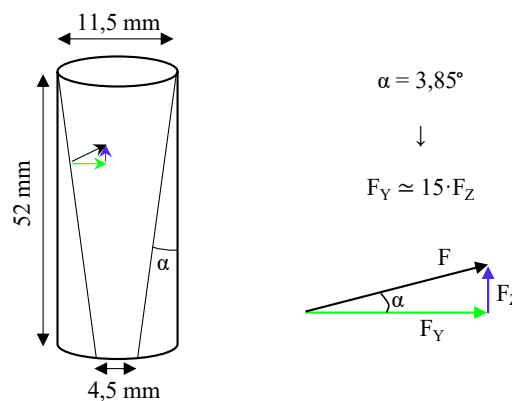


Figura 5.6: Relación entre fuerzas en el eje Y y el eje Z

Como consecuencia, no se pueden extraer demasiadas conclusiones de la gráfica en el eje Z, ya que principalmente se observa ruido.

Por último, en la Figura 5.7 se muestra una gráfica en 3D dónde se representa el cilindro por el que accede el endoscopio, la referencia de posición y la posición real del robot a lo largo del experimento.

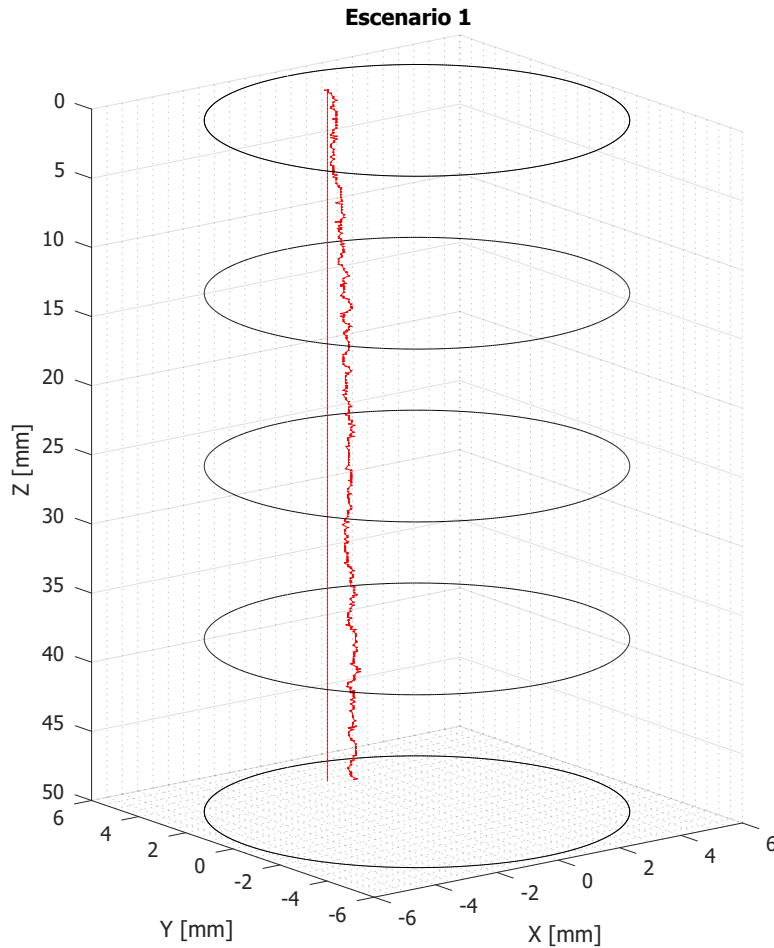


Figura 5.7: Representación de la posición en 3D en el escenario 1

5.2.2. Escenario 2

La geometría del escenario 2 se muestra en la Figura 5.8. En este caso el contacto principal se realiza en el eje X, en vez de en el eje Y con el primer escenario. La principal diferencia es que el ángulo de contacto es mayor, lo que simularía una mayor desviación, y por lo tanto las fuerzas que aparecen son mayores también. Concretamente, el ángulo que ve el endoscopio es de $6,3^\circ$.

Para forzar el contacto con la pared, la posición de inicio estará desviada 4,5 mm en la dirección negativa del eje X.

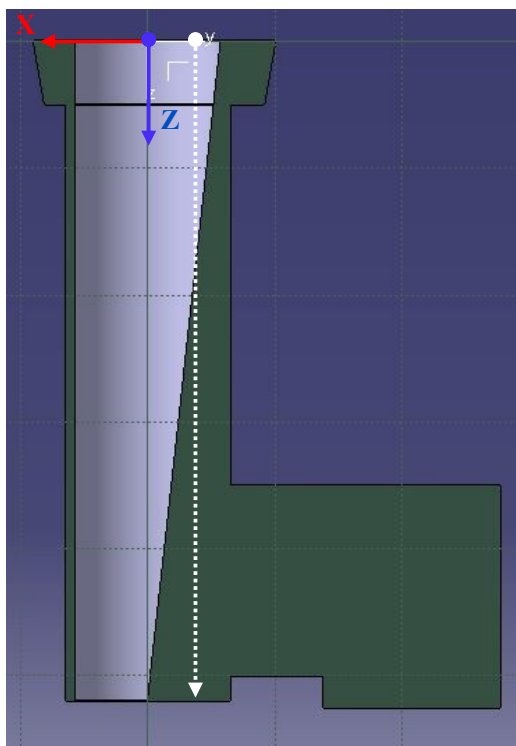


Figura 5.8: Sección del escenario 2

Se muestran los resultados en el eje X y Z en la Figura 5.9.

En el eje X observamos un comportamiento de escalera tal y como se espera. En este caso, la desviación respecto a la referencia es mayor, ya que las fuerzas que aparecen son mayores. Concretamente se desvía un total de 2 mm aproximadamente.

En este caso también cobra importancia la deformación de la varilla del endoscopio y del portaherramientas, que no es completamente rígido. Lo que nos muestran estas gráficas, es que parte de la fuerza está siendo absorbida por los elementos físicos en forma de deformación (concretamente los 2 mm restantes, ya que la posición final debería coincidir con el centro del cilindro que tiene cota cero en X).

En el eje Z observamos un comportamiento similar al primer escenario durante los primeros 10 segundos. A partir de ese instante, la fuerza tiene una tendencia negativa como

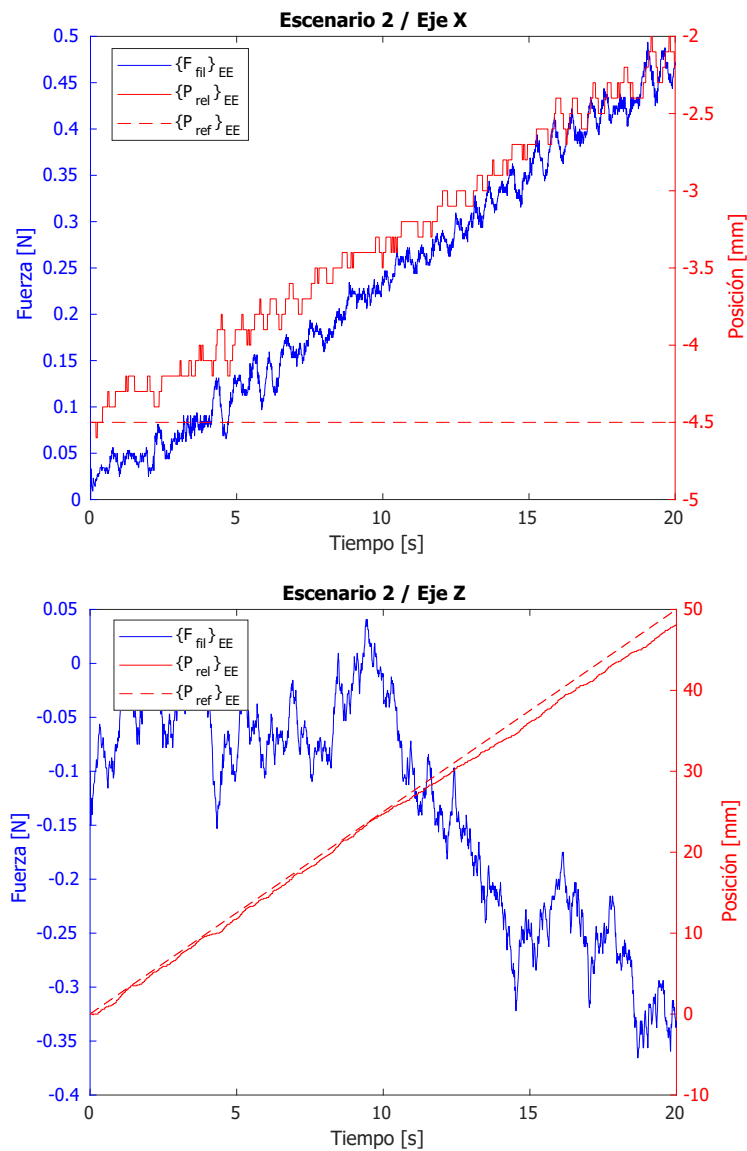


Figura 5.9: Representación de la fuerza y la posición en el escenario 2

cabría esperar, ya que está sometido a esfuerzos de compresión. También se observa una desviación de la posición respecto a la referencia, lo que indica que hay movimientos de retracción. El valor de esta retracción al final del escenario es de 2 mm aproximadamente.

Aun así, no hay que perder de vista la falta de precisión en el eje Z para este escenario en particular.

Por último, en la Figura 5.10 se muestra una visión 3D de la reacción al escenario.

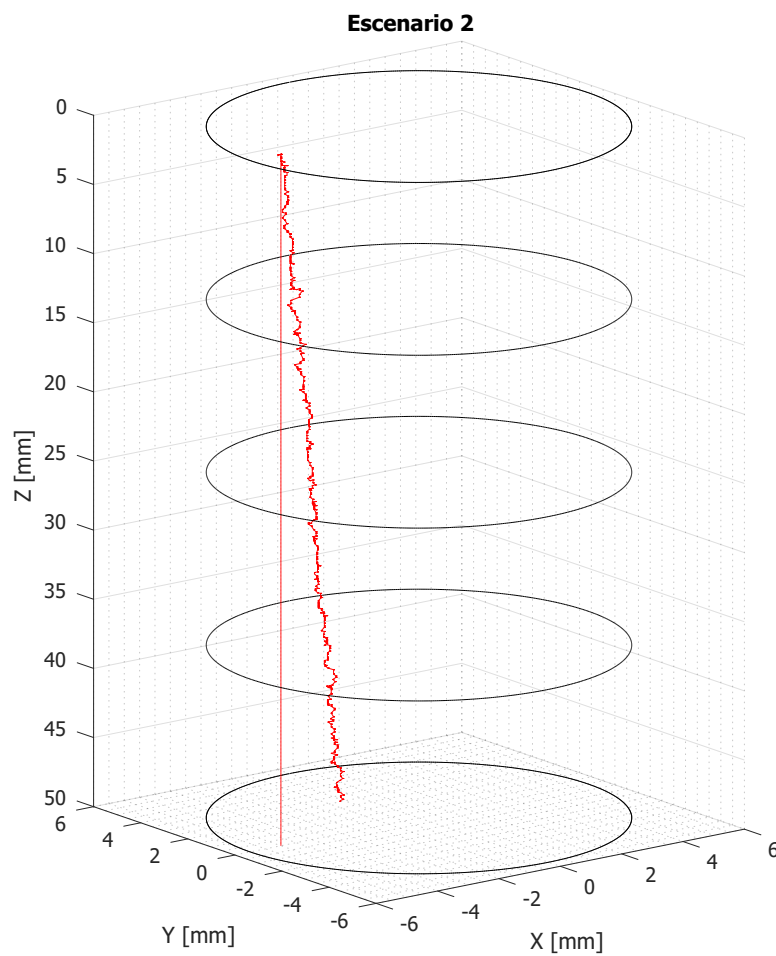


Figura 5.10: Representación de la posición en 3D en el escenario 2

5.2.3. Escenario 3

La geometría del escenario 3 se muestra en la Figura 5.11. Se plantea un camino sinuoso con rampas cortas en ambos lados del plazo XZ. El camino que seguirá el endoscopio será una diagonal, comienza a 4,5 mm en la dirección negativa del eje X y finaliza a 4,5 mm en la dirección positiva del mismo eje.

Hay que tener en cuenta que el algoritmo de fuerza únicamente trabaja con orientación constante, por lo tanto, el endoscopio siempre será paralelo al eje Z que se muestra en la sección.

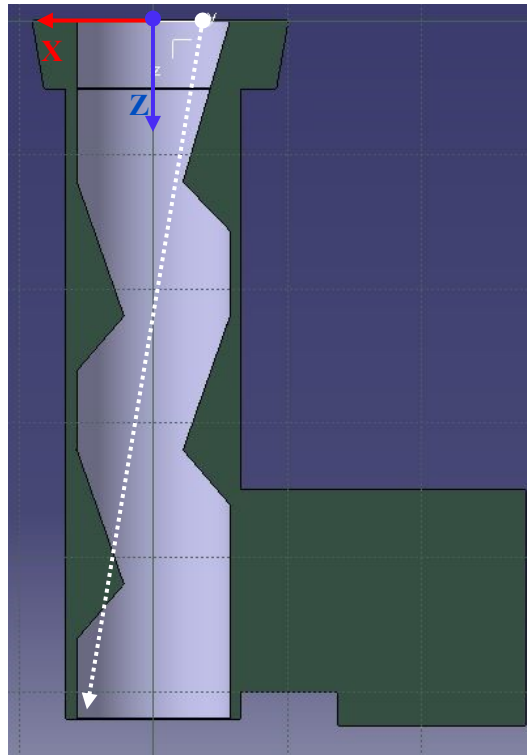


Figura 5.11: Sección del escenario 3

Se muestran los resultados en el eje X y Z en la Figura 5.12.

Inicialmente el endoscopio se encuentra con la primera rampa del lado derecho, que hace que el robot se mueva en la dirección positiva del eje X. Como coincide con la dirección de consigna, simplemente mayorará su valor.

En la gráfica se observa como aumenta la fuerza en X y gradualmente va descendiendo hasta llegar a 0. En ese instante el endoscopio se encuentra centrado en el cilindro y por lo tanto no está sometida a ninguna perturbación externa.

A medida que avanza el escenario, el endoscopio colisionará con las rampas del lado izquierdo. En cambio, no lo hará con su extremo, sino en algún punto en la dirección longitudinal del mismo.

Como se observa en la gráfica, las fuerzas son cada vez mayores a medida que la referencia

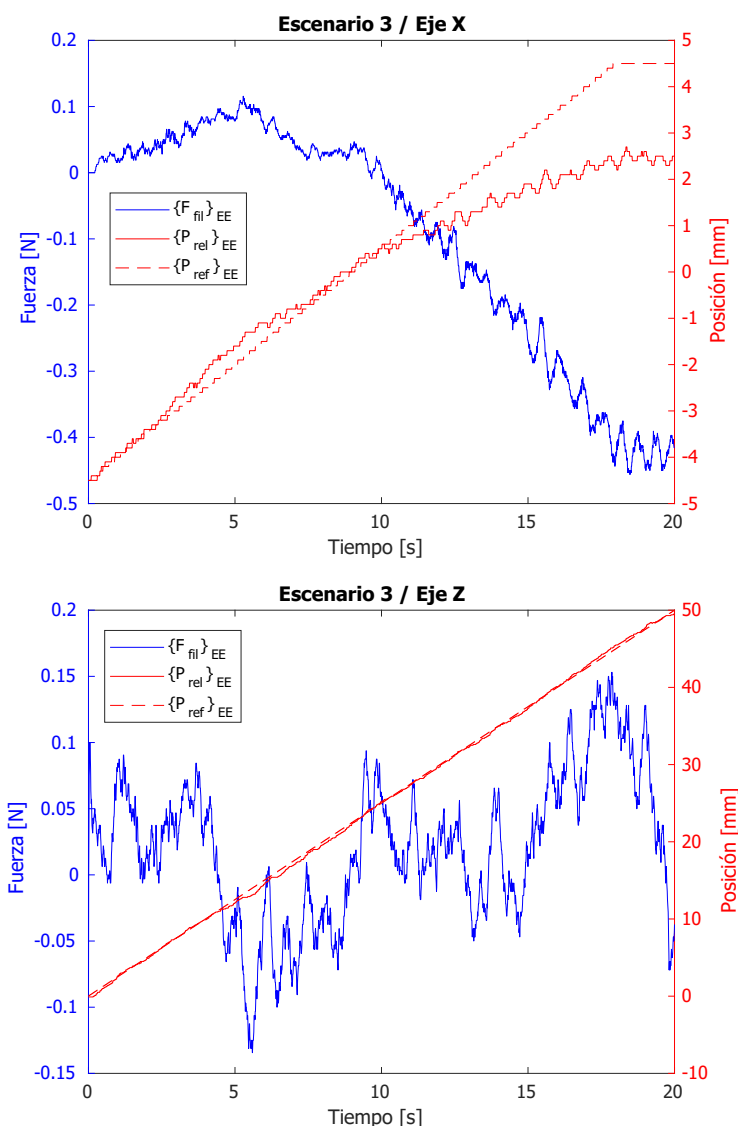


Figura 5.12: Representación de la fuerza y la posición en el escenario 3

se aleja de la posición real del endoscopio. En este caso, el robot genera una reacción en oposición a esa fuerza, que coincide con la dirección negativa del eje X.

El hecho de que el endoscopio no haya colisionado frontalmente con las rampas izquierdas, sino con el lateral, se puede corroborar observando la gráfica en Z, en la cual no aparecen apenas esfuerzos durante el transcurso del experimento.

Por último, en la Figura 5.13 se muestra una visión 3D de la trayectoria segunda durante la ejecución del escenario.

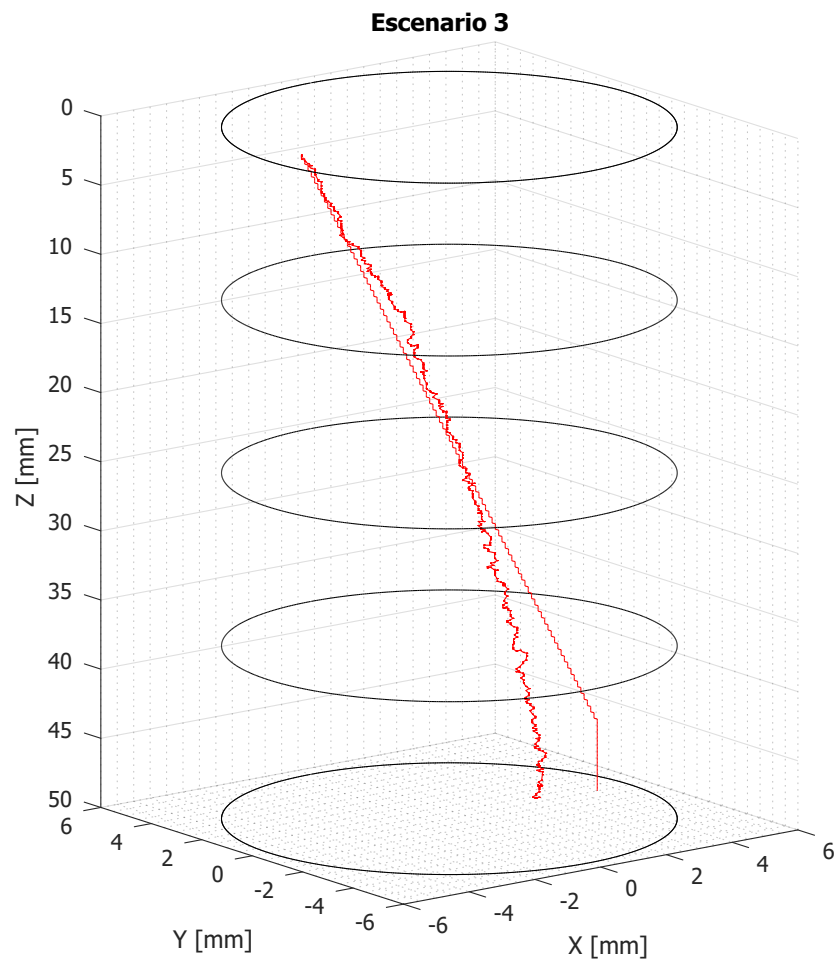


Figura 5.13: Representación de la posición en 3D en el escenario 3

5.2.4. Escenario 4

En los escenarios anteriores se han utilizado rampas para modelar los escenarios. En cambio, este escenario tiene forma de embudo y está modelado los 360°. En este escenario se simula una situación en la que el contacto se va haciendo más grande hasta que llega un punto de colisión total, en la cual no se permite el avance.

En la Figura 5.14 se muestra la geometría del escenario. La zona de colisión está a una distancia de 42 mm en la dirección positiva del eje Z. Para forzar el contacto con la pared, la posición de inicio estará desviada 4,5 mm en la dirección negativa del eje X, tal y como se muestra en línea blanca discontinua.

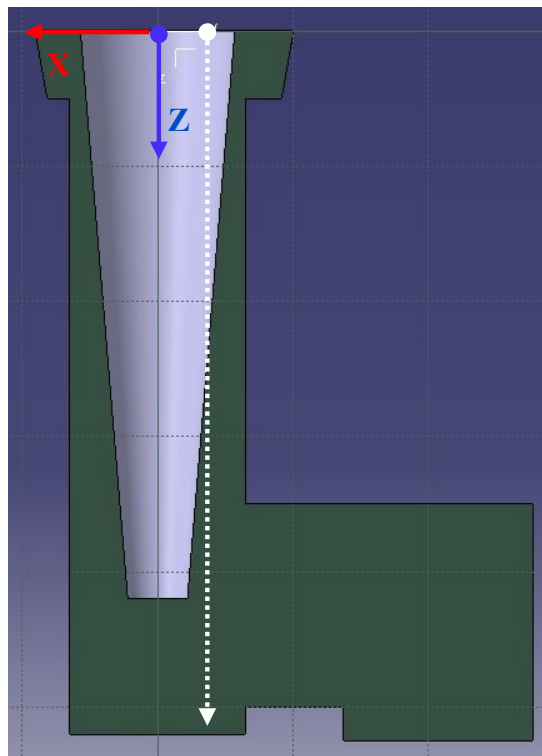


Figura 5.14: Sección del escenario 4

En la Figura 5.15 se muestran dos gráficas. En la primera de ellas se muestra la señal sin filtrar de fuerza, y en la segunda la señal filtrada, ambas en el eje Z que es el más relevante para este experimento.

Antes de comentar las gráficas, conviene comentar una característica del sistema de control. Por la propia naturaleza del filtrado, los picos de fuerza de la señal filtrada tienen una magnitud significativamente menor, además de inducir retraso.

Por motivos de seguridad, en cada iteración se comprueba cual es la fuerza no filtrada en cada uno de los ejes. Si se supera el umbral de ± 5 N, se activa una parada de protección y el robot se detiene.

Volviendo sobre las gráficas, no se observa ninguna reacción hasta que llega a los 42 mm.

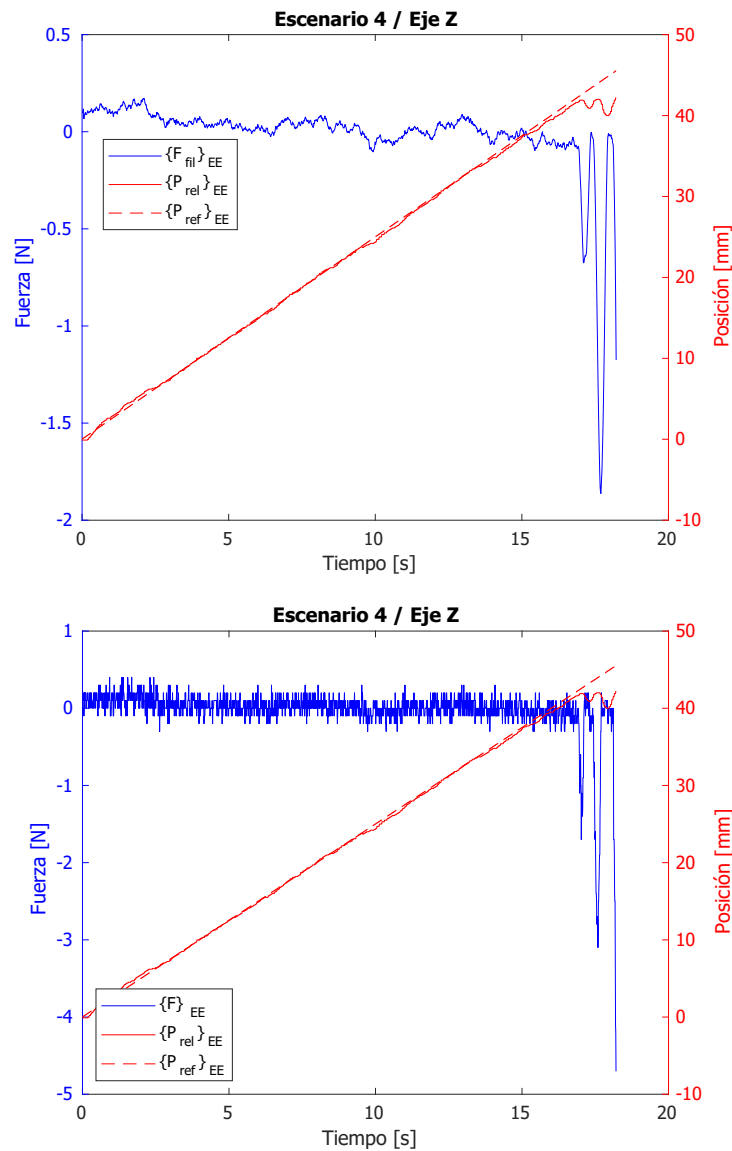


Figura 5.15: Representación de la fuerza y la posición en el escenario 4

En ese instante la fuerza aumenta, y como consecuencia, el robot retrocede. Una vez la fuerza ha disminuido, como la consigna sigue aumentando, el robot intenta alcanzar la referencia, moviéndose hacia delante. Una vez más, aparecen fuerzas, esta vez mayores, ya que, al estar más alejado de la referencia, el valor de error es mayor. Finalmente, vuelve a retirarse y cuando recupera, la fuerza sin filtrar excede los 5 N y el robot se para por seguridad.

El hecho de que aparezca esta oscilación se debe a los siguientes factores:

- Existe un retraso entre las fuerzas reales y las fuerzas filtradas, que son las utilizadas en el algoritmo de control. Por lo tanto, la salida del control no se genera en el instante exacto en el que se produce la fuerza
- También existe un retraso entre el instante en el que se manda la consigna al robot,

y éste la alcanza. Este retraso es inevitable, y estará condicionado por los valores de aceleración y velocidad consignados

- El hecho de que la consigna no deje de aumentar hace que el error de posicionamiento se haga cada vez más grande. Como consecuencia, la salida del controlador (la velocidad) será mayor y por lo tanto las fuerzas que aparecen cuando entre en contacto, serán mayores también

En la Figura 5.16 se muestra una visión 3D de la trayectoria segunda durante la ejecución del escenario.

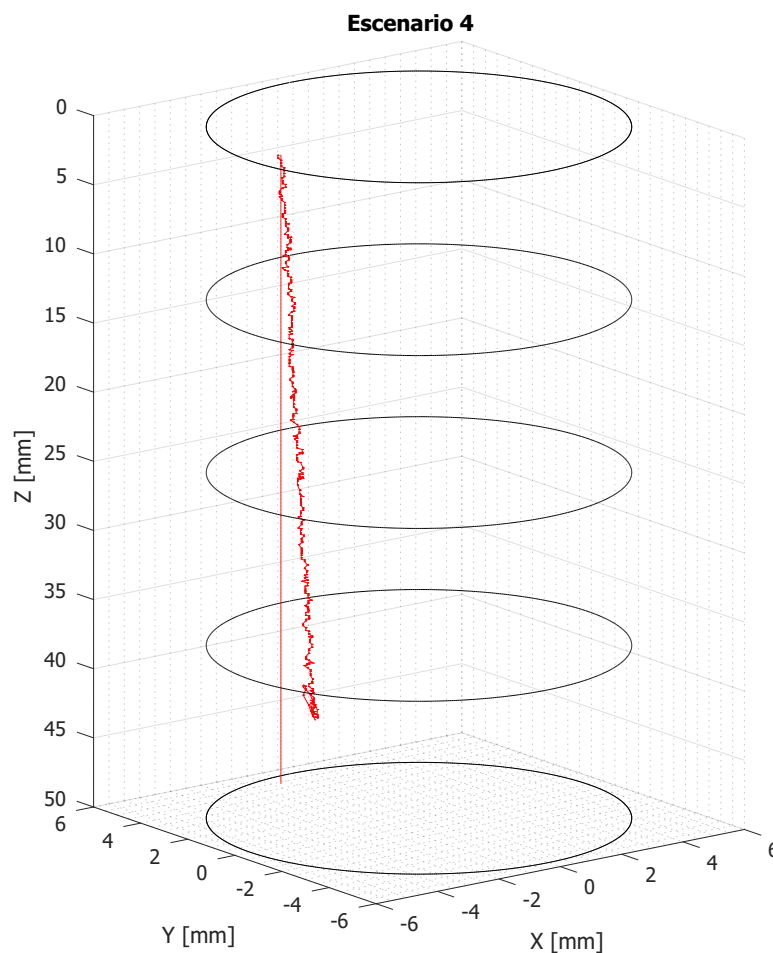


Figura 5.16: Representación de la posición en 3D en el escenario 4

5.2.5. Escenario 5

En el escenario anterior se plantaba una situación en la cual el endoscopio se encontraba con un obstáculo que no le permitía avanzar. El material con el que están impresos los escenarios es bastante rígido, por lo tanto, no es capaz de absorber ningún tipo de esfuerzo. Para intentar simular una situación más parecida a la que nos podríamos encontrar en una operación, se ha utilizado goma EVA para simular los tejidos internos. Se han utilizado 2 capas de 3,8 mm cada una. En la Figura 5.17 se muestra un esquema del escenario.

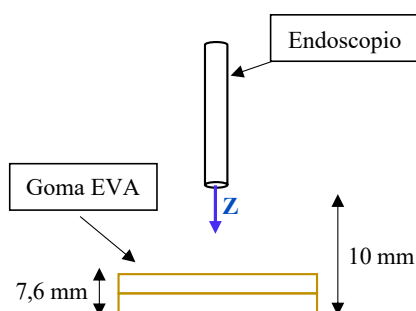


Figura 5.17: Esquema del escenario 5

La consigna del robot es 20 mm en la dirección positiva del eje Z. Cuando haya avanzado 2,4 mm entrará en contacto con la goma EVA y a una distancia de 10 mm se encontrará la mesa y por lo tanto no se permitirá más avance.

En la Figura 5.18 se muestra la gráfica de posición y fuerza en el eje Z, que es el único implicado en este experimento.

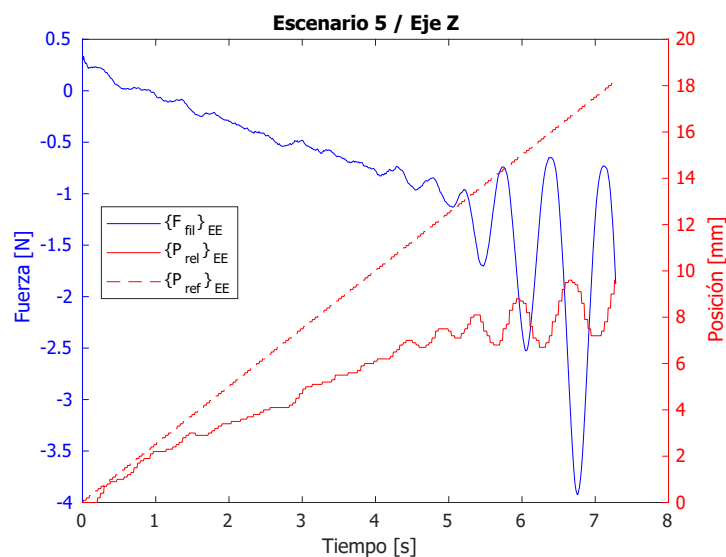


Figura 5.18: Representación de la fuerza y la posición en el escenario 5

En dicha gráfica podemos distinguir tres zonas. Durante el primer segundo, el endoscopio no ha entrado en contacto con la goma EVA, y por lo tanto las fuerzas que aparecen son

ruido y la posición no se ve muy afectada.

A partir del segundo 1 y hasta el 4,5 aproximadamente, observamos un comportamiento elástico. A medida que avanza, aumentan las fuerzas de forma proporcional y estable, ya que no se aprecian oscilaciones.

Finalmente, a partir del segundo 4,5, el robot habría alcanzado la mesa y observamos el mismo comportamiento oscilatorio del escenario anterior, en el que eventualmente se alcanza una fuerza (no filtrada) de 5 N y se detiene la ejecución.

5.3. Mejora del eje Z

El comportamiento en los ejes XY ha sido el esperado, en cambio los resultados en el eje Z no son aceptables.

Pese a contar con una etapa de filtrado previa, el nivel de ruido en conjunción con un valor de K_{FUERZA} de 5 mm/N hace que la variación en la posición respecto a la referencia pueda tomar valores de hasta 2 mm . Existen varias soluciones a este problema:

- Aumentar el tamaño de ventana del filtro, disminuyendo el nivel de ruido
- Disminuir el valor de K_{FUERZA} para que para un mismo nivel de ruido, el desplazamiento debido al mismo sea menor
- Aumentar el umbral de fuerza, de forma que la posición sea más estable cuando no se estén ejerciendo fuerzas

Además, se desean eliminar las oscilaciones encontradas en los escenarios 4 y 5.

En la Figura 5.19 se muestran un total de 6 gráficas, en las cuales se reproduce el escenario 5 (utilización de goma EVA) en los que la constante K_{FUERZA} varía entre 0 mm/N y 5 mm/N , el tamaño de ventana de filtrado ha aumentado de 32 a 64, y el umbral de fuerza ha aumentado de 0,1 N a 0,2 N .

Para valores de constantes de fuerza 0, 1 y 2 mm/N se observa cómo la gráfica se corta antes de llegar a los 8 segundos, o lo que es lo mismo, a alcanzar una consigna teórica de 20 mm . Esto es debido a que la acción de repulsión no es suficientemente grande como para contrarrestar las fuerzas que aparecen, por lo tanto, dichos valores de K_{FUERZA} se descartan.

El primer valor de K_{FUERZA} que es capaz de absorber una desviación en la referencia de 10 mm sin superar una fuerza de 5 N de forma estable es 3 mm/N . En dicha gráfica no aparece ningún tipo de oscilación, y aunque la traza de la posición no es tan limpia como en valores de K_{FUERZA} inferiores, es aceptable.

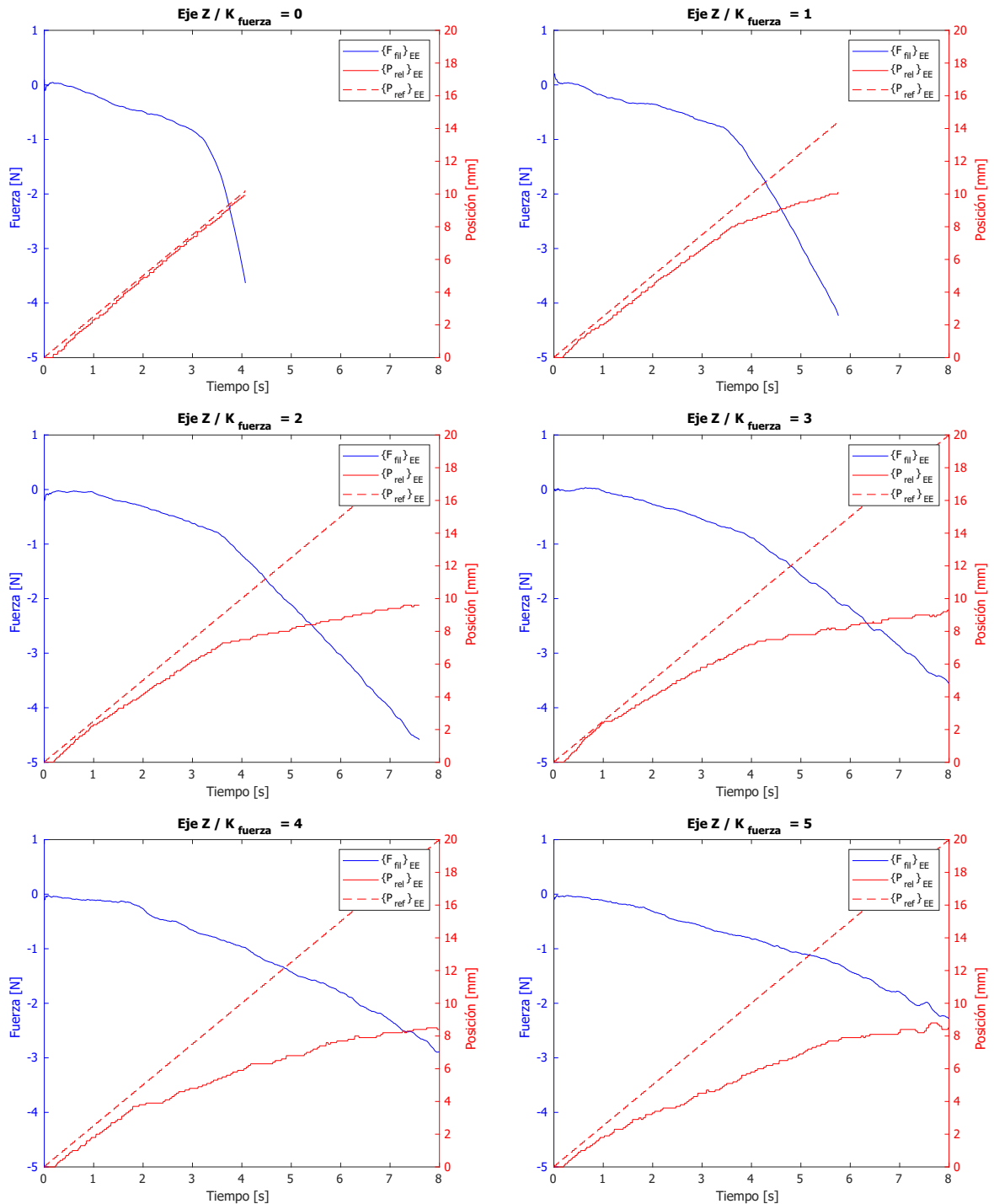


Figura 5.19: Escenario 5 con mejoras en el eje Z y distintos valores K_{FUERZA}

Para un valor de constante de 4 mm/N se consigue que se ejerza una fuerza final menor en comparación a 3 mm/N (3 N frente a $3,5 \text{ N}$), en cambio, comprometería la precisión de posicionamiento como ocurría anteriormente. Para un valor de 5 mm/N aparecen pequeñas oscilaciones, por lo tanto también se descarta.

Tras el análisis de estas gráficas se ha adoptado una K_{FUERZA} para el eje Z de un valor de 3 mm/N , eliminando satisfactoriamente las oscilaciones.

En la Figura 5.20 se muestra una gráfica en la que se ha superpuesto la señal de fuerza sin filtrar a modo de comparación.

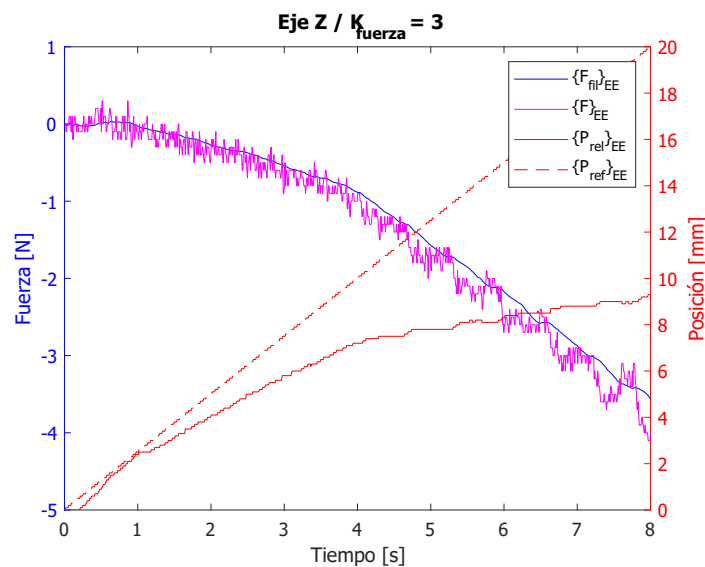


Figura 5.20: Superposición de la fuerza sin filtrar en la nueva configuración del eje Z

Se observa que el filtrado de la señal de fuerza es bastante efectivo, y además es capaz de absorber los grandes saltos en la señal, notables a partir del segundo 5, aportando más estabilidad al sistema.

5.4. Conclusiones

Una vez analizados los resultados de los cinco escenarios, se pueden extraer las siguientes conclusiones:

- Se valida el comportamiento del algoritmo de fuerza. El robot ha sido capaz de moverse en la dirección opuesta a los obstáculos, minimizando la fuerza de contacto de forma estable
- Los cambios realizados en el eje Z han permitido mejorar el comportamiento en dicha dirección, paliando los efectos del mayor nivel de ruido
- El escenario 3 muestra una debilidad del algoritmo de control, y es que únicamente tiene control sobre las coordenadas XYZ, pero no tiene control sobre la orientación. En algunos escenarios es interesante permitir el cambio de orientación para disminuir las fuerzas ejercidas de una forma más efectiva
- Se comprueba que ante una situación inesperada en la cual aparecen fuerzas grandes no esperadas, el robot es capaz de parar automáticamente la ejecución del movimiento para disminuir o eliminar los posibles daños

Capítulo 6

Conclusiones y líneas futuras

6.1. Conclusiones

Tras finalizar la redacción del presente TFM, se va a hacer una recapitulación de los objetivos propuestos en el capítulo introductorio junto con las conclusiones obtenidas.

El primer objetivo era revisar el estado actual de desarrollo de la cirugía robótica en el campo endonasal. Aunque existen un elevado número de desarrollos activos, la mayoría de estos únicamente se encuentran en fase de prototipado y muchos de ellos se centran en la creación de un soporte endoscópico, y no en una sistema robótico de navegación.

Se ha llevado a cabo un estudio sobre los métodos de control de *compliance*, que garantizan que el robot se adapte a las fuerzas de interacción, y se ha encontrado una aplicación de uno de estos métodos en un sistema de navegación diseñado para el campo endonasal llamado *SmartArm*. Se han encontrados dos aplicaciones en sistemas de control para laparoscopia y el lijado de aspas para uso aeronáutico que han sido de gran utilidad para el desarrollo del sistema de control con realimentación de fuerza.

El siguiente objetivo marcado fue el desarrollo de primitivas de movimiento para el control de robot *UR3* bajo el entorno de trabajo *ROS*. Se han desarrollado dos herramientas para controlar el robot, la primera de ellas basada en la biblioteca de movimientos *MoveIt*, que añade funcionalidades extra como detección de colisiones con el entorno, y la biblioteca *URScript* que utiliza los comandos nativos de *Universal Robots* para mandar las peticiones al controlador del robot. Ambas han resultado ser muy efectivas, cada una con sus ventajas e inconvenientes.

Gracias a la utilización de *MoveIt*, ha permitido cumplir el objetivo de crear un entorno de simulación que represente el estado del robot en el escenario de trabajo. Además, la utilización de *URSim* permite realizar pruebas de los sistemas de control sin necesidad de utilizar el brazo robótico real.

A continuación, se ha desarrollado un sistema de control que se opone al movimiento en

las direcciones en la que se detecta fuerza. El sistema de control ha pasado por varias iteraciones, en las cuales se ha sintonizado el controlador PD y se ha refinado la etapa de filtrado de la señal de fuerza para mejorar el comportamiento global del sistema.

Finalmente, el último objetivo contemplaba la validación del sistema de control mediante la creación y análisis de escenario de contacto y colisión, los cuales han permitido terminar de refinar los parámetros del sistema de control, y validar el correcto funcionamiento del mismo.

Se puede concluir, por lo tanto, que se han cumplido todos los objetivos fijados para este TFM.

6.2. Líneas futuras

Una vez finalizado el Trabajo Fin de Máster, se pueden plantear las siguientes líneas de trabajo futuras:

- Optimización de los parámetros del lazo de control con realimentación de fuerza. El método que se ha seguido para sintonizar los parámetros del lazo de control ha sido siguiente el método de prueba y error. Aunque se hayan obtenido resultados suficientemente buenos, siempre existe un margen de mejora; con la ayuda de un modelado detallado del sistema y la utilización de *software* de simulación como *Matlab* y *Simulink*, podría mejorar la estabilidad y respuesta temporal del sistema. Además, la realización de pruebas reales en entornos de simulación más realistas, aportarían más información acerca del desempeño en un escenario real
- Incluir compensación gravitatoria. Debido al peso suspendido sobre el sensor de fuerza, en función de la posición y orientación de la herramienta, se obtienen diferentes medidas de fuerza. Esto es atajado mediante la puesta a cero de todos los valores cuando se alcanza la posición de aproximación y la orientación de la herramienta no es modificada. Aunque una vez más, los resultados son suficientemente buenos, podrían mejorarse si se incluyera una compensación gravitatoria a las medidas de fuerza. Esto también mejoraría la versatilidad del sistema robótico y daría opción a realizar movimientos más complejos con cambios de orientación
- Creación de una interfaz de usuario. Actualmente, todo el control se realiza a través de *scripts* y la línea de comandos. La creación de una interfaz de usuario permitiría recibir información de una forma más clara, además de facilitar su utilización por personal no especializado
- Integración en el proyecto *Craneéal*. Como ha comentado en diferentes puntos del TFM, el trabajo se encuentra englobado dentro del proyecto *Craneéal*. El siguiente

paso es por lo tanto llevar a cabo una integración del trabajo realizado para validar el comportamiento del sistema global

Bibliografía

- [1] Robótica Médica. Proyecto CRANEEAL. Universidad de Málaga;. Available from: <https://www.uma.es/medical-robotics/info/107696/proyecto-dpi2016-80391-C3-1-R/>.
- [2] Instituto Quirúrgico de Andalucía IQA. Indicaciones de la Colectomía en pacientes de Cirugía Bariátrica;. Available from: <http://iqaquiron.com/portal/indicaciones-de-la-colectomia-en-pacientes-de-cirurgia-bariatrica>.
- [3] Messerer M, Dubourg J. Endoscopic pituitary surgery. *Journal of Neurosurgery*. 2014;121(6):1533.
- [4] Ashrafian H, Clancy O, Grover V, Darzi A. The evolution of robotic surgery: Surgical and anaesthetic aspects. In: *British Journal of Anaesthesia*; 2017. .
- [5] Intuitive Surgical. Da Vinci Surgery | Da Vinci Surgical System | Robotic Technology;. Available from: <https://www.davincisurgery.com/da-vinci-systems/about-da-vinci-systems>.
- [6] Medtronic. Capsule Endoscopy Products;. Available from: <https://www.medtronic.com/covidien/en-us/products/capsule-endoscopy.html>.
- [7] Chalongsongse S, Chumnanvej S, Suthakorn J. Analysis of Endonasal Endoscopic Transsphenoidal (EET) surgery pathway and workspace for path guiding robot design. *Asian Journal of Surgery*. 2019;42(8):814–822. Available from: <https://doi.org/10.1016/j.asjsur.2018.12.016>.
- [8] Goh H, Kim S, Kang SG, Lee DY. Kinematic Requirements of a Haptic Interface for Simulation of Endonasal Endoscopic Skull Base Surgery. In: *2018 18th International Conference on Control, Automation and Systems (ICCAS)*; 2018. p. 39–44.
- [9] Karl Storz. Telescopes, Visualization and Documentation Systems for Video-Assisted Cardiac Surgery and Open Heart Surgery with Minimal Access;. Available from: https://www.karlstorz.com/cps/rde/xbcr/karlstorz_{_}assets/ASSETS/2142150.pdf.

-
- [10] Madoglio A, Zappa F, Mattavelli D, Rampinelli V, Ferrari M, Schreiber A, et al. Robotics in endoscopic transnasal skull base surgery: Literature review and personal experience. In: Control Systems Design of Bio-Robotics and Bio-mechatronics with Advanced Applications. Elsevier; 2020. p. 221–244.
- [11] De Gea Fernández J, Kirchner F. Predictive compliance for interaction control of robot manipulators. IEEE International Conference on Intelligent Robots and Systems. 2011;(April):4134–4140.
- [12] ATI Industrial Automation: Compliance Devices;. Available from: <https://www.ati-ia.com/products/compliance/compensator{ }main.aspx>.
- [13] Xia T, Baird C, Jallo G, Hayes K, Nakajima N, Hata N, et al. An integrated system for planning, navigation and robotic assistance for skull base surgery. The International Journal of Medical Robotics and Computer Assisted Surgery. 2008 dec;4(4):321–330. Available from: <http://doi.wiley.com/10.1002/rcs.213>.
- [14] Marinho MM, Harada K, Morita A, Mitsuishi M. SmartArm: Integration and validation of a versatile surgical robotic system for constrained workspaces. The International Journal of Medical Robotics and Computer Assisted Surgery. 2020 apr;16(2). Available from: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rcs.2053>.
- [15] Rivas-Blanco I, Bauzano E, Cuevas-Rodriguez M, Del Saz-Orozco P, Munoz VF. Force-position control for a miniature camera robotic system for single-site surgery. IEEE International Conference on Intelligent Robots and Systems. 2013;p. 3065–3070.
- [16] Xu X, Zhu D, Zhang H, Yan S, Ding H. Application of novel force control strategies to enhance robotic abrasive belt grinding quality of aero-engine blades. Chinese Journal of Aeronautics. 2019 oct;32(10):2368–2382.
- [17] Universal Robots. Manual de usuario UR3/CB3. 3.5.5;. Available from: <https://www.universal-robots.com/support/>.
- [18] Pinzas 2F-85 y 2F-140 - Robotiq;. Available from: <https://robotiq.com/es/productos/pinza-adaptable-2f85-140>.
- [19] OnRobot. Our HEX force/torque sensor for robotics makes automation simple;. Available from: <https://onrobot.com/en/products/hex-6-axis-force-torque-sensor>.
- [20] Frontpage - SMARTSHIFT robotics;. Available from: <https://smartshift-robotics.com/>.
- [21] Cappabianca P, Maria Cavallo L, de Divittis O, Esposito F, Solari D, Notaris MD, et al. Endoscopic pituitary and skull base surgery. Anatomy and Surgery of the

- Endoscopic Endonasal Approach;. Available from: <https://www.karlstorz.com/cps/rde/xbcr/karlstorz{ }assets/ASSETS/3314461.pdf>.
- [22] Universal Robots - What is a singularity?;. Available from: <https://www.universal-robots.com/articles/ur-articles/what-is-a-singularity/>.
- [23] MoveIt Motion Planning Framework;. Available from: <https://moveit.ros.org/>.
- [24] OnRobot. HEX-E Sensor 2.0 Datasheet; 2018.
- [25] Muñoz García Á, Rivas I, Turiel JP, Muñoz V, Fraile Marinero JC, de la Fuente E, et al. Integration of a Surgical Robotic Co-worker in an Endoscopic Neurosurgical Assistance Platform. In: *Advances in Intelligent Systems and Computing*. vol. 1093 AISC. Springer; 2020. p. 453–464. Available from: <https://link.springer.com/chapter/10.1007/978-3-030-36150-1{ }37>.
- [26] Targarona EM, Feliu X, Salvador JL, editors. *Cirugía endoscópica. Guías clínicas de la asociación español de cirujanos*. Arán Ediciones; 2010. Available from: <https://www.aecirujanos.es/files/documentacion/documentos/cirurgia-endoscopica-2-edicion.pdf>.
- [27] Mayo Clinic. Colectistectomía;. Available from: <https://www.mayoclinic.org/es-es/tests-procedures/cholecystectomy/about/pac-20384818>.
- [28] Lanfranco AR, Castellanos AE, Desai JP, Meyers WC. *Robotic Surgery: A Current Perspective*; 2004.
- [29] Bickerton R, Nassimizadeh AK, Ahmed S. Three-dimensional endoscopy: The future of nasoendoscopic training. *The Laryngoscope*. 2019 jun;129(6):1280–1285. Available from: <https://onlinelibrary.wiley.com/doi/abs/10.1002/lary.27812>.
- [30] Tumor de la base del cráneo | MD Anderson Cancer Center;. Available from: <https://mdanderson.es/elcancer/elcancerdelaaalaz/tumor-de-la-base-del-craneo>.
- [31] Ajler P, Campero A, Hem S, Goldschmidt E, Landriel F, Carrizo A, et al. Cirugía transnasal endoscópica para tumores de hipófisis. *Surgical Neurology International*. 2012;3(7):389.
- [32] Yadav YR, Sachdev S, Parihar V, Namdev H, Bhatele PR. Endoscopic endonasal trans-sphenoid surgery of pituitary adenoma. Thieme Medical Publishers; 2012.
- [33] Kwoh YS, Hou J, Jonckheere EA, Hayati S. A Robot with Improved Absolute Positioning Accuracy for CT Guided Stereotactic Brain Surgery. *IEEE Transactions on Biomedical Engineering*. 1988;35(2):153–160.

-
- [34] Ravikiran S, Dinesh N. Evidence for Robots. SICOT-J. 2017;3. Available from: <https://pubmed.ncbi.nlm.nih.gov/28534472/>.
- [35] Imperial College London. Probot | Research groups;. Available from: <https://www.imperial.ac.uk/mechatronics-in-medicine/research/probot/>.
- [36] Marescaux J, Leroy J, Rubino F, Smith M, Vix M, Simone M, et al. Transcontinental robot-assisted remote telesurgery: Feasibility and potential applications. *Annals of Surgery*. 2002;235(4):487–492. Available from: </pmc/articles/PMC1422462/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC1422462/>.
- [37] Ota T, Degani A, Schwartzman D, Zubiato B, McGarvey J, Choset H, et al. A Highly Articulated Robotic Surgical System for Minimally Invasive Surgery. *Annals of Thoracic Surgery*. 2009 apr;87(4):1253–1256.
- [38] Khan EM, Frumkin W, Ng GA, Neelagaru S, Abi-Samra FM, Lee J, et al. First experience with a novel robotic remote catheter system: Amigo mapping trial. *J Interv Card Electrophysiol*. 2013;37:121–129.
- [39] Ciuti G, Valdastrì P, Menciassi A, Dario P. Robotic magnetic steering and locomotion of capsule endoscope for diagnostic and surgical endoluminal procedures. *Robotica*. 2010 mar;28(2):199–207.
- [40] FDA. What We Do;. Available from: <https://www.fda.gov/about-fda/what-we-do>.
- [41] Schneider JS, Burgner J, Webster RJ, Russell PT. Robotic surgery for the sinuses and skull base: What are the possibilities and what are the obstacles? NIH Public Access; 2013.
- [42] Medrobotics. Flex Robotic System: Expanding the reach of surgery;. Available from: <https://medrobotics.com/gateway/flex-robotic-system/>.
- [43] Bolzoni Villaret A, Doglietto F, Carobbio A, Schreiber A, Panni C, Piantoni E, et al. Robotic Transnasal Endoscopic Skull Base Surgery: Systematic Review of the Literature and Report of a Novel Prototype for a Hybrid System (Brescia Endoscope Assistant Robotic Holder). *World Neurosurgery*. 2017 sep;105:875–883.
- [44] Hilberg O, Jackson AC, Swift DL, Pedersen OF. Acoustic rhinometry: Evaluation of nasal cavity geometry by acoustic reflection. *Journal of Applied Physiology*. 1989;66(1):295–303. Available from: <https://journals.physiology.org/doi/abs/10.1152/jappl.1989.66.1.295>.
- [45] Paraskevopoulos D, Roth J, Constantini S. Endoscope Holders in Cranial Neurosurgery: Part i - Technology, Trends, and Implications. *World Neurosurgery*. 2016 may;89:343–354.

- [46] Nimsky C, Rachinger J, Iro H, Fahlbusch R. Adaptation of a Hexapod-Based Robotic System for Extended Endoscope-Assisted Transsphenoidal Skull Base Surgery. *Minimally Invasive Neurosurgery*. 2004 feb;47(1):41–46. Available from: <https://pubmed.ncbi.nlm.nih.gov/15100931/>.
- [47] Wurm J, Dannenmann T, Bohr C, Iro H, Bumm K. Increased safety in robotic paranasal sinus and skull base surgery with redundant navigation and automated registration. *The International Journal of Medical Robotics and Computer Assisted Surgery*. 2005 sep;1(3):42–48. Available from: <http://doi.wiley.com/10.1002/racs.26>.
- [48] Nathan CAO, Chakradeo V, Malhotra K, D’Agostino H, Patwardhan R. The voice-controlled robotic assist scope holder AESOP for the endoscopic approach to the sella. *Skull Base*. 2006 aug;16(3):123–132. Available from: <http://www.thieme-connect.de/DOI/DOI?10.1055/s-2006-939679>.
- [49] Strauss G, Hofer M, Kehrt S, Grunert R, Korb W, Trantakis C, et al. Manipulator Assisted Endoscope Guidance in Functional Endoscopic Sinus Surgery: Proof of Concept. *HNO*. 2007 mar;55(3):177–184. Available from: <https://pubmed.ncbi.nlm.nih.gov/16773352/>.
- [50] Yoon HS, Oh SM, Jeong JH, Lee SH, Tae K, Koh KC, et al. Active bending endoscope robot system for navigation through sinus area. *Institute of Electrical and Electronics Engineers (IEEE)*; 2011. p. 967–972.
- [51] Eichhorn KW, Bootz F. Clinical requirements and possible applications of robot assisted endoscopy in skull base and sinus surgery. *Acta neurochirurgica Supplement*. 2011;109:237–240. Available from: https://link.springer.com/chapter/10.1007/978-3-211-99651-5_{ }37.
- [52] Swaney PJ, Croom JM, Burgner J, Gilbert HB, Rucker DC, Webster RJ, et al. Design of a quadramanual robot for single-nostril skull base surgery. In: *ASME 2012 5th Annual Dynamic Systems and Control Conference Joint with the JSME 2012 11th Motion and Vibration Conference, DSCC 2012-MOVIC 2012*. vol. 3. American Society of Mechanical Engineers Digital Collection; 2012. p. 387–393.
- [53] Trévillot V, Sobral R, Dombre E, Poignet P, Herman B, Crampette L. Innovative endoscopic sino-nasal and anterior skull base robotics. *International Journal of Computer Assisted Radiology and Surgery*. 2013 apr;8(6):977–987. Available from: <https://link.springer.com/article/10.1007/s11548-013-0839-1>.
- [54] Schneider JS, Burgner J, Webster RJ, Russell PT. Robotic surgery for the sinuses and skull base: What are the possibilities and what are the obstacles? *Curr Opin Otolaryngol Head Neck Surg*; 2013. Available from: <https://pubmed.ncbi.nlm.nih.gov/23299117/>.

-
- [55] B C, S C, I A, M T, S K. A Haptic Guided Robotic System for Endoscope Positioning and Holding. *Turkish neurosurgery*. 2015;25(4). Available from: <https://pubmed.ncbi.nlm.nih.gov/26242338/>.
- [56] Swaney PJ, Gilbert HB, Webster RJ, Russell PT, Weaver KD. Endonasal skull base tumor removal using concentric tube continuum robots: A phantom study. *Journal of Neurological Surgery, Part B: Skull Base*. 2015 mar;76(2):145–149. Available from: <https://pubmed.ncbi.nlm.nih.gov/27054057/>.
- [57] Chan JYK, Leung I, Navarro-Alarcon D, Lin W, Li P, Lee DLY, et al. Foot-controlled robotic-enabled endoscope holder for endoscopic sinus surgery: A cadaveric feasibility study. *The Laryngoscope*. 2016 mar;126(3):566–569. Available from: <http://doi.wiley.com/10.1002/lary.25634>.
- [58] Ogiwara T, Goto T, Nagm A, Hongo K. Endoscopic endonasal transsphenoidal surgery using the iArmS operation support robot: Initial experience in 43 patients. *Neurosurgical Focus*. 2017;42(5).
- [59] Medineering. Endoscope Ropbot;. Available from: <http://www.medineering.de/{#}/en/products{%-}23endoscope-robot>.
- [60] Zappa F, Mattavelli D, Madoglio A, Rampinelli V, Ferrari M, Tampalini F, et al. Hybrid Robotics for Endoscopic Skull Base Surgery: Preclinical Evaluation and Surgeon First Impression. *World Neurosurgery*. 2020 feb;134:e572–e580.
- [61] Singh R, Baby B, Damodaran N, Srivastav V, Suri A, Banerjee S, et al. Design and Validation of an Open-Source, Partial Task Trainer for Endonasal Neuro-Endoscopic Skills Development: Indian Experience. *World Neurosurgery*. 2016 feb;86:259–269.
- [62] Renishaw. Neuromate robotic system for stereotactic neurosurgery;. Available from: <https://www.renishaw.com/en/neuromate-robotic-system-for-stereotactic-neurosurgery--10712>.
- [63] 3D Systems. Dispositivos hápticos Phantom Premium;. Available from: <https://es.3dsystems.com/haptics-devices/3d-systems-phantom-premium>.
- [64] Marinho MM, Adorno BV, Harada K, Deie K, Deguet A, Kazanzides P, et al. A unified framework for the teleoperation of surgical robots in constrained workspaces. In: *Proceedings - IEEE International Conference on Robotics and Automation*. vol. 2019-May. Institute of Electrical and Electronics Engineers Inc.; 2019. p. 2721–2727.
- [65] Marinho MM, Adorno BV, Harada K, Mitsuishi M. Active constraints using vector field inequalities for surgical robots. In: *Proceedings - IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers Inc.; 2018. p. 5364–5371.

- [66] Questions on Universal Robots Answered;. Available from: <https://blog.robotiq.com/all-your-questions-on-universal-robots-answered-here>.
- [67] ATI Tool Changer Products: QC-5;. Available from: <https://www.ati-ia.com/products/toolchanger/QC.aspx?ID=QC-5>.
- [68] ATI Tool Changer Products: MC-10;. Available from: <https://www.ati-ia.com/es-mx/products/toolchanger/QC.aspx?ID=MC-10>.
- [69] ROS.org | About ROS;. Available from: <https://www.ros.org/about-ros/>.
- [70] Topics - ROS Wiki;. Available from: <http://wiki.ros.org/Topics>.
- [71] Coleman D, Ucan ISÂ, Chitta S, Correll N. Reducing the Barrier to Entry of Complex Robotic Software: a MoveIt! Case Study. *Journal of Software Engineering for Robotics*. 2014;5(1):3–16. Available from: www.joser.org.
- [72] tf2 - ROS Wiki;. Available from: <http://wiki.ros.org/tf2>.
- [73] Packages - ROS Wiki;. Available from: <http://wiki.ros.org/Packages>.
- [74] GitHub - ros-industrial/universal_robot: ROS-Industrial Universal Robots support;. Available from: https://github.com/ros-industrial/universal_robot/tree/kinetic-devel.
- [75] GitHub - ros-industrial/ur_modern_driver;. Available from: https://github.com/ros-industrial/ur_modern_driver/tree/kinetic-devel.
- [76] GitHub - UniversalRobots/Universal_Robots_ROS_Driver;. Available from: https://github.com/UniversalRobots/Universal_Robots_ROS_Driver.
- [77] Universal Robots. The URScript Programming Language 3.5.4; 2018. Available from: <https://www.universal-robots.com/support/>.
- [78] rviz - ROS Wiki;. Available from: <http://wiki.ros.org/rviz>.
- [79] UR Download | Support Site | Universal Robots;. Available from: <https://www.universal-robots.com/download/?option=38719#section16632>.

Anexo A

Robot UR3

A.1. Especificaciones técnicas

Tipo de robot	UR3
Peso	11 kg / 24.3 lb
Carga máxima (consulte la sección 4.4)	3 kg / 6.6 lb
Alcance	500 mm / 19.7 in
Rango giro juntas	Rotación infinita de la última junta, $\pm 360^\circ$ para todas las demás juntas
Velocidad	Todas las juntas de la muñeca: Máx. 360 °/s, Otras juntas: Máx. 180 °/s. Herramienta: Aprox. 1 m/s / Aprox. 39.4 in/s.
Repetibilidad	± 0.1 mm / ± 0.0039 in (4 mils)
Espacio necesario	$\varnothing 128$ mm / 5.0 in
Grados de libertad	6 juntas giratorias
Tamaño de la caja de control (ancho \times alto \times largo)	475 mm \times 423 mm \times 268 mm / 18.7 in \times 16.7 in \times 10.6 in
Puertos de E/S de la caja de control	16 entradas digitales, 16 salidas digitales, 2 entradas analógicas, 2 salidas analógicas
Puertos de E/S de la herramienta	2 entradas digitales, 2 salidas digitales, 2 entradas analógicas
Fuente de alimentación de E/S	24 V 2 A en caja de control y 12 V/24 V 600 mA en herramienta
Comunicación	TCP/IP 100 Mbit: IEEE 802.3u, 100BASE-TX Ethernet socket, Modbus TCP & Adaptador EtherNet/IP, Profinet
Programación	Interfaz gráfica de usuario PolyScope en pantalla táctil de 12"
Ruido	70 dB(A)
Clasificación IP	IP64
Clasificación de sala blanca	Brazo robótico: ISO Clase 5 Caja de control: ISO Clase 6
Consumo de energía	Aprox. 100 W utilizando un programa típico
Funcionamiento colaborativo	15 funciones de seguridad avanzada. De acuerdo con: EN ISO 13849-1:2008, PLd y EN ISO 10218-1:2011, cláusula 5.10.5
Materiales	Aluminio, plástico PP
Temperatura	El robot puede funcionar en un intervalo de temperatura ambiente de 0-50 °C. A una velocidad de junta continuamente elevada, la especificación de temperatura ambiente máxima se reduce.
Fuente de alimentación	100-240 VAC, 50-60 Hz
Cables	Cable entre el robot y la caja de control (6 m / 236 in) Cable entre la pantalla táctil y la caja de control (4.5 m / 177 in)

Figura A.1: Especificaciones técnicas del robot UR3 [17]

A.2. Brida de salida de la herramienta

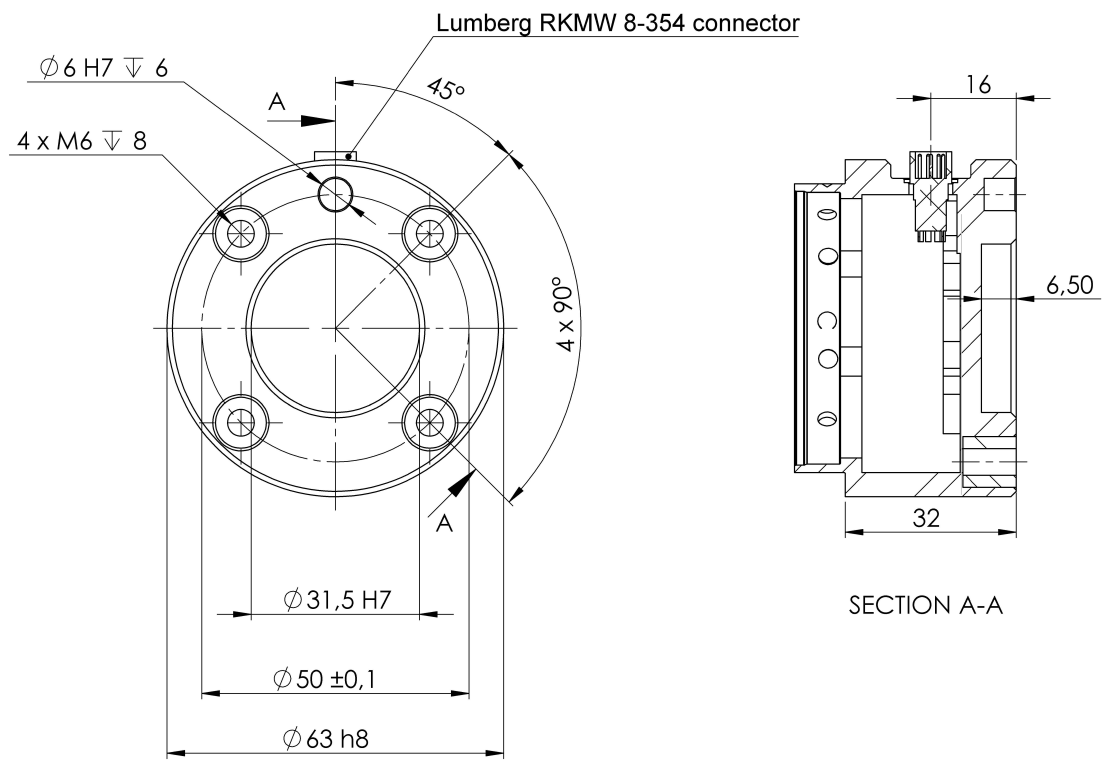


Figura A.2: Brida de salida de la herramienta. Todas las unidades están en mm [17]

Anexo B

Sensor de fuerza HEX-E

B.1. Especificaciones técnicas

Sensor Type	6-Axis Force/Torque Sensor			
Dimensions (Height x Diameter)	37.5 x 70 mm			
Weight (with built-in adapter plates)	245 g			
	Fxy	Fz	Txy	Tz
Nominal Capacity (N.C)	200 N	200 N	10 Nm	6.5 Nm
Single axis deformation at N.C (typical)	± 1.7 mm	± 0.3 mm	± 2.5 °	± 5 °
Single axis overload	500 %	500 %	500 %	500 %
Signal noise ² (typical)	0.035 N	0.15 N	0.002 Nm	0.001 Nm
Noise-free resolution (typical)	0.2 N	0.8 N	0.010 Nm	0.002 Nm
Full scale nonlinearity	< 2%	< 2%	< 2%	< 2%
Hysteresis (measured on Fz axis , typical)	< 2 %	< 2 %	< 2 %	< 2 %
Crosstalk (typical)	< 5 %	< 5 %	< 5 %	< 5 %
Working temperature range	0 C° / +55 °C			
Power requirement	DC input range 7-24V		0.8 W	
Mounting screws	5 x M4 x 6 mm 1 x M4 x 12 mm (for the Cable Holder)			ISO14581

¹ It needs protection when working in corrosive liquid environments

² Signal noise is defined as the standard deviation (1σ) of a typical one second no-load signal.

Figura B.1: Especificaciones técnicas del sensor de fuerza HEX-E [24]

B.2. Planos

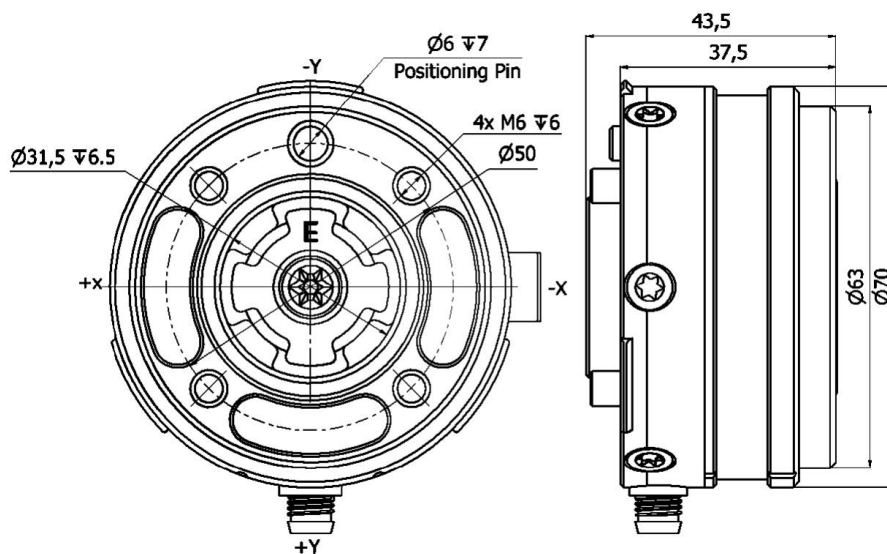


Figura B.2: Planos del sensor de fuerza HEX-E [24]