

UNIVERSIDAD AUTÓNOMA DE CIUDAD JUÁREZ

Instituto de Ingeniería y Tecnología

Departamento de Ingeniería Industrial y Manufactura



Enseñanza Robótica Colaborativa Mediante Guiado Manual con Estimación
de la Fuerza de Interacción.

Tesis presentada por:

Ing. Camilo Morales Corral, 183284

Para obtener el grado de:

MAESTRO EN TECNOLOGÍA

Dr. Ángel Flores Abad
Dr. Israel Soto Marrufo

Ciudad Juárez, Chihuahua

Julio, 2020

Ciudad Juárez, Chihuahua 23 de Julio de 2020
Asunto: Liberación de Asesoría

Dr. Delfino Cornejo Monroy
Coordinador de la Maestría en Tecnología
Presente.-

Por medio de la presente nos permitimos comunicarle que después de haber realizado las asesorías correspondientes al reporte técnico Enseanza robótica colaborativa mediante guiado manual con estimación de la fuerza de interacción, del alumno Ing. Camilo Morales Corral, de la Maestría en Tecnología, consideramos que lo ha concluido satisfactoriamente, por lo que pueden continuar con los trámites de titulación.

Sin más por el momento, reciba un cordial saludo.

Atentamente

Ángel Flores Abad
Profesor Investigador IIT

Israel Soto Marrufo
Profesor Investigador IIT

Ccp:

Coordinador del programa
Ing. Camilo Morales Corral
Archivo

Ciudad Juárez, Chihuahua 23 de Julio de 2020
Asunto: Autorización de Impresión

C. Ing. Camilo Morales Corral
Presente.-

En virtud de que cumple satisfactoriamente los requisitos solicitados, informo a usted que se autoriza la impresión del proyecto de Enseñanza Robótica Colaborativa Mediante Guiado Manual con Estimación de la Fuerza de Interacción., para presentar los resultados del proyecto de titulación con el propósito de obtener el título de Licenciado en Maestro en Tecnología.

Sin otro particular, reciba un cordial saludo.

Prof.Dr. Francisco Carrillo Pereyra
Profesor de Proyecto de Tecnología II

Declaración de Originalidad

Yo Ing. Camilo Morales Corral, declaro que el material contenido en esta publicación fue generado con la revisión de los documentos que se mencionan en la sección de referencias y que la solución desarrollada es original y no ha sido copiada de ninguna otra fuente, ni ha sido usada para obtener otro título o reconocimiento en otra Institución de Educación Superior.

Ing. Camilo Morales Corral

Agradecimientos

Agradezco a mi familia todo el apoyo, con el cual la distancia se acorta y el "gorrión" se hace más llevadero. Por soportar la lejanía y la soledad sin reclamos, conociendo que estas son el pago de crecer como profesional.

Se agradece a los maestros de la UACJ por todos los conocimientos adquiridos. En especial a los asesores del proyecto, por su dedicación y apoyo durante estos dos años. A los diferentes coordinadores del programa, por sus asesorías certeras y su colaboración.

Agradezco a mis compañeros de clases y amigos, por su apoyo dentro de los salones de clases y fuera de estos. Por acompañarme y siempre estar prestos a brindar su ayuda. Y por que no, por las "kawamas" que compartimos en los ratos libres.

Por ultimo, pero no menos importante, se agradece a CONACyT por el financiamiento recibido para desarrollar mis estudios de postgrados.

Dedicatoria

A mi familia, en especial a mi abuela Rosa.

Índice general

1. Introducción	1
1.1. Antecedentes	2
1.1.1. Métodos de enseñanza.	2
1.1.2. Interacción física entre humanos y robots.	3
1.1.3. Requerimientos de seguridad para la pHRI.	5
1.1.4. Esquemas propuestos para controlar la interacción del robot con el entorno.	6
1.1.5. Esquemas propuestos para controlar la interacción durante la programación por guiado.	8
1.2. Definición del problema.	9
1.3. Objetivos.	9
1.3.1. Objetivo General.	9
1.3.2. Objetivos específicos.	9
1.4. Justificación.	10
1.5. Metodología.	10
2. Marco Teórico	11
2.1. Modelos matemático de un robot manipulador.	12
2.1.1. Modelo cinemático directo de un robot manipulador.	12
2.1.2. Modelo cinemático inverso de un robot manipulador.	12
2.1.3. Modelo diferencial de un robot manipulador.	13
2.1.4. Modelo dinámico de un robot manipulador.	13

<i>ÍNDICE GENERAL</i>	III
2.1.5. Validación de los modelos matemáticos.	15
2.2. Observador de perturbación.	16
2.3. Estructura de control empleada durante el guiado manual.	17
2.3.1. Controlador de posición.	18
2.3.2. Controlador de admitancia.	18
3. Desarrollo del Proyecto	21
3.1. Diseño y fabricación del prototipo de pruebas.	21
3.2. Modelado matemático del prototipo de pruebas.	24
3.2.1. Modelo cinemático directo.	24
3.2.2. Modelo cinemático inverso.	25
3.2.3. Matriz Jacobiana.	27
3.2.4. Modelo dinámico.	28
3.2.5. Obtención experimental de los parámetros del modelo.	29
3.2.6. Validación del modelo dinámico del motor de DC.	33
3.3. Diseño del controlador interno de posición.	35
3.4. Diseño del DOB para estimar la fuerza de guiado.	39
3.5. Diseño del controlador de admitancia para el guiado manual.	43
4. Conclusiones	49
Bibliografía	51
A. Características de los componentes.	56
B. Comunicación entre el Arduino Mega y Simulink.	58

Índice de figuras

1.1. Marco de referencia de tres niveles para lograr la pHRI.	4
2.1. Diagrama de bloques del Observador de Perturbaciones.	16
2.2. Diagrama de bloques del Controlador Proporcional con Realimentación de Velocidad.	18
2.3. Estructura de control propuesta. Método de enseñanza por guiado manual. .	20
3.1. Manipuladores SCARA de 3 gdl comerciales.	22
3.2. Diseño CAD del prototipo de pruebas.	22
3.3. Prototipo de prueba real.	23
3.4. Definición de marcos de referencia para el prototipo de pruebas.	25
3.5. Esquema de un manipulador SCARA de 2 gdl para el ángulo θ_2	25
3.6. Esquema de un manipulador SCARA de 2 gdl para el ángulo θ_1	27
3.7. Vistas acotadas del prototipo de pruebas.	30
3.8. Gráfica característica de velocidad contra voltaje.	32
3.9. Datos de la relación fuerza/voltaje según hoja de datos.	34
3.10. Comparación del comportamiento del modelo simulado y el motor real. . . .	35
3.11. Diagrama en bloques del sistema realimentado.	36
3.12. Coordenadas articulares para trayectoria circular en simulación.	37
3.13. Señal de control para una trayectoria circular.	37
3.14. Coordenadas articulares para trayectoria circular en experimento real.	38
3.15. Error de seguimiento de las coordenadas articulares.	38

3.16. Trayectoria circular realizada en experimento real.	39
3.17. Simulación de la respuesta del DOB para $g = 30$	40
3.18. Coordenadas articulares ante la presencia de una f_{ext} en simulación.	40
3.19. Experimento para la validación del DOB.	41
3.20. Comparación entre la f_e medida y la estimada en el mecanismo de 1 dof.	42
3.21. Comparación entre la f_e medida y la estimada en el robot de 2 dof.	43
3.22. Comparación entre la f_e real y la estimada. Simulación de guiado.	44
3.23. Definición del ángulo de aplicación de la f_e	44
3.24. Simulación de guiado aplicando una f_e en un ángulo $\gamma = 80^\circ$	45
3.25. Fuerza medida y estimada. Experimento realimentando fuerza medida.	46
3.26. Señal de salida del control de posición.	47
3.27. Señal de salida del controlador de admitancia.	47
3.28. Ángulo de referencia y real durante el experimento en 1 dof.	47
3.29. Fuerza medida y estimada. Experimento realimentando fuerza estimada.	48

Índice de tablas

3.1. Parámetros de la estructura mecánica del prototipo de pruebas.	30
3.2. Experimentos para el calculo de la constante contraelectromotriz (K_a). . . .	31
3.3. Experimentos para el calculo del momento de inercia (J_m).	33
3.4. Experimentos para la caracterización de la fuerza.	34
3.5. Parámetros ajustados del modelo del motor de DC.	35

Resumen

En la actualidad las industrias manufactureras requieren de celdas de producción cada vez más modulares y flexibles, las cuales puedan adaptarse rápidamente a los requisitos de la producción. Los métodos de programación para robots tradicionales presentan como desventajas la necesidad de conocimientos específicos y largos tiempos de trabajo para poder entregar un programa funcional [1]. Para contribuir al desarrollo de métodos de programación más eficientes e intuitivos, en la presente investigación se propone el desarrollo de una estructura de control que permita el guiado manual de un manipulador. Para conseguir que el robot pueda ser guiado por un operador humano se diseñó un controlador de admitancia, demostrándose el correcto funcionamiento de la estructura de control propuesta en simulación y mediante experimentos realizados en un prototipo diseñado con este fin. Se encontraron dificultades para validar la estructura de control propuesta durante los experimentos reales cuando se realimenta la señal de fuerza estimada por el DOB. La razón de este comportamiento se asocia a la capacidad computacional de la controladora empleada.

Capítulo 1

Introducción

En la era de la globalización, las industrias manufactureras se enfrentan a una diversificación continua de sus productos y a ciclos de vida más cortos. Por lo cual en la actualidad se requiere de celdas de producción cada vez más modulares y flexibles, las cuales puedan adaptarse rápidamente a los requisitos de la producción. Para cumplir estos requerimientos, hace algunos años, se observan enfoques innovadores empleando tecnologías de la información y las comunicaciones (TICs) y robótica, que potencian el control flexible y la automatización de líneas de montaje [2]. Estas tendencias son lo que actualmente se conoce como Industria 4.0, un término acuñado por el Ministerio Federal de Educación e Investigación de Alemania, durante la Feria de Hannover en el 2011 [3]. Dado que las definiciones para este término son variadas, en el presente documento se adoptará la siguiente:

“El objetivo de la Industria 4.0 es aumentar la digitalización de los procesos de fabricación y las cadenas de suministro, facilitando las comunicaciones entre personas, máquinas y productos, permitiendo así el acceso en tiempo real a la información de los productos y la producción, así como mejorar el desempeño de procesos autónomos a lo largo de las cadenas de valor” (Santos, Mehraei, Barros, Araujo, Ares, 2017).

La definición anterior se refiere a digitalización en general, sin embargo, en realidad existen ciertas tecnologías que son claves. Siendo las que más influyen sobre este nuevo paradigma: los sistemas ciberfísicos, la internet de las cosas, la internet de los servicios, la robótica, el *big data*, manufactura en la nube y realidad aumentada [4]. Específicamente dentro de la robótica la tendencia esta encaminada a potenciar la interacción entre los robot y los humanos, ya sean como robots de servicios o realizando alguna tarea conjunta en la industria. Esto puede ser comprobado por el rápido y fuerte crecimiento del mercado de los robots colaborativos. Según datos de la Federación Internacional de Robótica este debe alcanzar el 34% de las ventas del sector de la robótica para el 2025 [5].

Según [6], el primer desafío si se quiere implementar la colaboración entre humanos y robots es la seguridad. Por lo cual en los últimos años se ha venido trabajando en muchas

soluciones, sin embargo estas aún no se han transferido por completo a la industria [7]. Identificándose dos áreas de interés en las que aún se debe trabajar [6]:

1. Métodos de diseño. Lo que incluye leyes de control, sensores y planificación de trayectoria que permitan al operador humano interactuar con el robot dentro del mismo entorno.
2. Interfaz de usuario amigables. Deben facilitar la interacción entre el robot y el humano, incluyendo los métodos de programación y los modos de entrada/salida de datos. Se deben dejar detrás las técnicas tradicionales (teclados, pantallas, y *teach pendants*) para implementar técnicas en las que se emplee el guiado manual, los comandos de voz, los gestos, o realidad aumentada, etc.

Este segundo punto es de gran relevancia si se quiere disminuir los tiempos de configuración de las celdas de producción. Los métodos de programación tradicionales presentan como desventajas la necesidad de conocimientos específicos y largos tiempos de trabajo para poder entregar un programa funcional [1]. Para ilustrar esta desventaja: la tarea de soldar la carrocería de un vehículo grande puede llevar más de ocho meses de programación, mientras que el tiempo de ciclo del proceso en sí mismo es de solo 16 horas [8].

Para contribuir al desarrollo de métodos de programación más eficientes e intuitivos, en la presente investigación se propone el desarrollo de una estructura de control que permita el guiado manual de un robot manipulador. Esta estructura debe permitir la estimación de la fuerza que el programador imprime en la estructura mediante un Observador de Perturbaciones (DOB, por sus siglas en inglés). Se deben realizar los experimentos necesarios para comprobar el correcto funcionamiento de la estructura empleando un prototipo de pruebas.

1.1. Antecedentes

1.1.1. Métodos de enseñanza.

Los métodos de programación tradicionales [6] y [9] los clasifican en programación en línea y fuera de línea. Siendo el primer método el más difundido en la actualidad en la industria [1]. En el primer método el robot es movido por el programador siguiendo la trayectoria deseada usando el *teach pendant* (método conocido como, *lead-through programming*). Durante este proceso se van almacenando los diferentes puntos de la trayectoria en el controlador del robot, para luego seguir esta secuencia cuando se conmute a modo automático. Este concepto parece sencillo, sin embargo, para realizarlo de manera eficiente se requiere de mucha experiencia y habilidades por parte del programador. Así mismo es un proceso en el que se emplea mucho tiempo, y debe ser ejecutado cada vez que se necesite cambiar la tarea a ejecutar por el robot.

El segundo método emplea un programa de ingeniería asistida por computadora (CAE, por sus siglas en inglés). Este contiene un conjunto de funciones de modelado que permiten la representación gráfica del entorno y la simulación de las trayectorias del robot en el modelo 3D de la celda de producción. Los más avanzados, ya traen módulos para procesos específicos como pintura, soldadura o pulido. Después de la simulación y prueba del programa, este debe ser descargado al robot, debiéndose realizar algunos ajustes finales por medio del *teach pendant*. Con este método los tiempos de programación siguen siendo relativamente largos, pero no interfieren de manera directa con la producción ya que se hace fuera de línea. Sin embargo, si necesitan de un gran conocimiento por parte de los ingenieros que van a realizar la programación. Además, de que cada fabricante de robots ha desarrollado su propio software, y la licencia de estos tienen un elevado valor [6, 1].

Según [6] en los últimos años se han desarrollado métodos novedosos para la programación de los robots, entre ellos: programación por guiado (*walk-through programming*), programación por demostración (*programming by demonstration*), interfaces multimodales, realidad virtual y realidad aumentada.

El método de programación guiada permite al usuario mover el efector final del robot a las posiciones deseadas libremente. Al mismo tiempo el controlador del robot graba las coordenadas de las articulaciones correspondientes, para luego reproducir la trayectoria. Según [6], esta técnica de programar el robot mediante guiado manual hace más sencillo e intuitivo el proceso, ya que no se necesita tener conocimiento de lenguajes de programación. Sin embargo, [10] reconoce que introduce un problema significativo: la seguridad durante la interacción entre el humano y el robot.

Según [6], la programación por demostración puede ser vista como una extensión de la programación guiada. Para [1] la solución ideal sería aquella en la que el robot rastreara los movimientos del operador mientras realiza una tarea y luego, utilizando métodos de aprendizaje automático, adquiere la capacidad de reproducirla.

1.1.2. Interacción física entre humanos y robots.

Según [11] los inicios del estudio de la interacción entre humanos y robots (pHRI, por sus siglas en inglés) como campo multidisciplinario comenzó a surgir a mediados de la década de 1990. Esto se debió a su contribución al aumento de la eficiencia, productividad y flexibilidad en las cadenas de montaje, así como la reducción de la carga de trabajo y el estrés en los trabajadores [2, 12]. El objetivo bajo esta concepción es permitir una estrecha colaboración entre humanos y robots, en todas las tareas industriales y de servicio, que requieren que la capacidad de adaptación de los humanos se fusione con el alto rendimiento en términos de precisión, velocidad y carga útil del robot [7].

Con el fin de conseguir este objetivo, en [13] se presenta un marco de referencia de tres

niveles que el robot debe cumplir, el cual se resume en la figura 1.1.

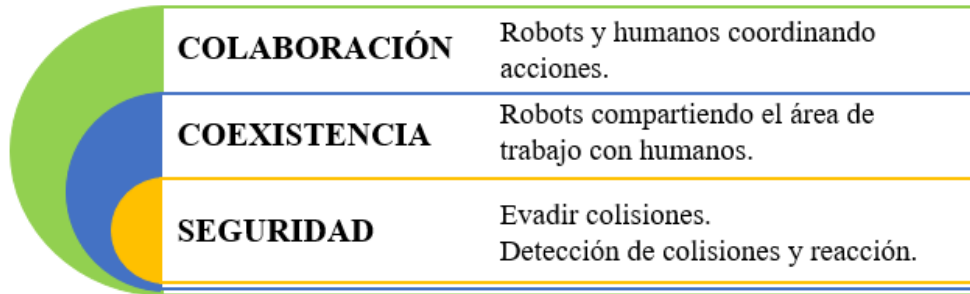


Figura 1.1: Marco de referencia de tres niveles para lograr la pHRI.

En el escenario de la pHRI, los métodos de **seguridad** tradicionales (jaulas, barreras, etc.) no son adecuados. Frente a las nuevas necesidades, en este nivel se han desarrollado técnicas para la detección de colisiones, así como la reacción después de la colisión. La **coexistencia** es la capacidad del robot de compartir el espacio de trabajo de manera segura con el humano, incluso podrían trabajar en el mismo objeto, pero sin ningún contacto mutuo o coordinación de acciones. En este nivel se implementan métodos para la evaluación de la distancia o separación, así como algoritmos para evitar las colisiones. La **colaboración** permite al robot realizar una tarea compleja con interacción humana directa. En esta interacción se coordinan las acciones en dos modalidades, no mutuamente excluyentes: estableciendo intencionalmente un contacto físico con intercambios de fuerzas entre los dos o sin contacto, donde la comunicación se logra por un comando de voz o un gesto.

Robots diseñados y fabricados bajo los preceptos de la pHRI: los Cobots.

Los inicios de la robótica colaborativa (Cobots) datan de 1995, cuando la Northwestern University y General Motor comenzaron un proyecto para la creación de un dispositivo de asistencia inteligente. El proyecto perseguía incluir tecnología robótica a los dispositivos manuales de asistencia ampliamente usados para manipular materiales [14]. El equipo de este proyecto introduce durante la Conferencia Internacional de Robótica y Automatización de 1997 el concepto de Cobots como una “clase de dispositivos robóticos pasivos que están diseñados para la colaboración directa con un operador humano dentro de un espacio de trabajo compartido” [15]. Ya para septiembre de 1999 se había otorgado la patente número USOO5952796A a favor de Colgate y Peshkin, donde describían un aparato y un método para la interacción física directa entre una persona y un manipulador [16].

En el 2005 comenzó la iniciativa robótica europea para el fortalecimiento de la competitividad de las Pymes (SME Robot), la cual perseguía introducir la robótica para hacerlas más rentables y competitivas mediante una solución económica y flexible [17]. La tecnología Cobots fue evolucionando hasta que en diciembre de 2008 se instaló el primer brazo robótico

colaborativo (UR5, Universal Robots) capaz de operar de manera segura junto con humanos [18]. Aunque en la actualidad la compañía Universal Robots cuenta con el 60 del mercado, ya son varios los fabricantes (Fanuc, Yaskawa, ABB, Kawasaki, Nachi, Kuka, Denso, Epson y Adept) que han desarrollado su dispositivo de brazo robótico colaborativo [5]. Una primera revisión de los modelos comerciales disponibles se puede encontrar en [19].

1.1.3. Requerimientos de seguridad para la pHRI.

Ninguna tecnología nueva se deja a criterio de los fabricantes, por lo que se han desarrollado una serie de estándares que proporcionan una visión unificada de los requisitos y guías generales para el diseño de nuevos dispositivos. Los principales son la norma ISO 10218-1/2 *Robots and robotic devices – Safety requirements for industrial robots*, del 2011, donde se establecen requerimientos de diseño para los fabricantes de robot y sus controladores, así como los requerimientos para la integración de un robot en un sistema industrial, y la especificación técnica ISO TS 15066 *Robots and robotic devices – Safety requirements for industrial robots – Collaborative operation* que provee de lineamientos para la operación de los robots. Un resumen de estos estándares se puede encontrar en [6].

Uno de los temas más importantes abordados dentro de la ISO 10218-1/2 son los modos de trabajo durante la colaboración, se definen cuatro [6]:

1. **Parada segura monitorizada.** El humano y el robot pueden trabajar dentro del área de colaboración, pero no al mismo tiempo. Cuando el ser humano ingresa al área de colaboración, el robot se somete a un modo de parada segura y su movimiento se detiene mediante un software redundante y una tecnología de seguridad basada en electrónica.
2. **Guiado manual.** Mientras que el operador activa el dispositivo de guía manual se activa el modo velocidad segura monitorizada, permitiendo el movimiento del robot. Cuando el operador suelta el dispositivo, el robot regresa a la parada segura monitorizada y reanuda el programa previamente interrumpido tan pronto como el operador sale del área de colaboración.
3. **Monitoreo de velocidad y distancia.** Permite la presencia humana dentro del espacio del robot haciendo uso de sensores (escáner o sistemas de visión) con clasificación de seguridad para realizar el monitoreo. La velocidad del robot es modificada según la distancia medida por los sensores que monitorizan el entorno.
4. **Limitación de fuerza y potencia.** En [20] se ha propuesto una descripción general del control de la interacción física entre robots y humanos, evitando consecuencias perjudiciales para este último. Se presentan cuatro posibles reacciones del robot en respuesta al contacto, la solución más simple es activar los frenos del robot después

de la colisión con una parada inmediata. También existen modos de control de torque que resultan en un comportamiento más seguro, como la disminución de la energía de impacto a través del movimiento en sentido contrario en la dirección de movimiento.

1.1.4. Esquemas propuestos para controlar la interacción del robot con el entorno.

Controlar la interacción del robot manipulador con el entorno es crucial para acometer una gran variedad de tareas en aplicaciones industriales. Ejemplos típicos de estas tareas son: montaje de componentes mecánicos, seguimientos de aristas, lijado, desbaste, pulimento, entre otras [21]. Por tal razón, el uso de estrategias de control de velocidad y fuerza en robot manipuladores no es algo nuevo. Muchos esquemas de control han sido propuestos para resolver este problema. Los primeros trabajos en esta área se pueden encontrar muy bien descritos en artículos de revisión de literatura como [22] y [23].

Según [24], estas estrategias pueden ser divididas en dos categorías principales: control pasivo y control activo de fuerza. En la primera, normalmente se agrega alguna flexibilidad al elemento terminal con el fin de amortiguar los impactos y aumentar la tolerancia a eventuales errores de posicionamiento. En ellas no se persigue un control explícito de la fuerza, sino que se controla la relación entre la fuerza y la posición del efector en contacto con el entorno. Por lo que se emplea una planeación cuidadosa de la trayectoria y la aproximación a los objetos. Así, el control de fuerza se realiza a través del control de posición, por lo que no se hace necesario utilizar sensores de fuerza o alteraciones en la programación. En esta clasificación son relevantes el control de rigidez y el control de impedancia, en ellas el error de posición se relaciona con la fuerza de contacto a través de la rigidez mecánica o de una impedancia de parámetros ajustables. Este esquema es fuertemente acoplado y no lineal, lo que crea dificultades en el modelado y en el control [25]. En la segunda, se busca que las fuerzas de contacto asuman un valor determinado. En ella el valor de la fuerza es realimentado, siendo usado para modificar o generar en línea la trayectoria deseada [24]. Estas estrategias se pueden analizar en dos subespacios: un controlador para el control de fuerza y otro para el control de posición [21]. La estrategia más relevante en esta categoría es el control híbrido de fuerza/posición (HMFC, por sus siglas en inglés) [26]. Esta tiene como objetivo controlar la posición en las direcciones no restringidas de movimiento, y controlar la fuerza en las direcciones donde existe restricción de movimiento [27]. El buen desempeño de esta estrategia depende de un total conocimiento de la geometría del entorno y lograr establecer el contacto con el medio ambiente de manera estable. Desafortunadamente, en la mayoría de los casos, no se cuenta con un modelo preciso del entorno, razón por la cual se han desarrollado otras estrategias. Entre ellas el control interno/externo de posición/fuerza, en el cual se cierra un lazo externo de fuerza sobre un lazo interno de posición, y el control paralelo de fuerza/posición, en las direcciones restringidas de actuación del robot se generan

fuerzas, por lo que domina el controlador de fuerza, y si no aparecen fuerzas de interacción, domina el controlador de posición [26].

En los últimos años, el control de fuerza ha tenido varias aportaciones con base en las estrategias de control inteligente. En sus inicios, controladores basados en lógica difusa fueron propuestos por [28], [29], y [24]. En estos no es necesario recurrir a modelos matemáticos detallados, ya que el control es hecho con base en un conjunto de reglas heurísticas, cuyo establecimiento se basa en la experiencia humana. Solucionándose así, el tema de las no linealidades, el alto acoplamiento y las perturbaciones existentes en los modelos. En [21] se propone un control predictivo de fuerza, donde el problema de optimización es resuelto a través del uso de una *fuzzy scaling machine*. Esta estrategia se valida mediante simulación de un manipulador de 2 DoF, considerando un modelo no lineal del entorno. Posteriormente, [30] propone un control de impedancia inteligente donde emplea una red neuronal difusa para establecer los parámetros óptimos para diferentes entornos y traslación autónoma flexible. Usa una tabla uniaxial (1 DoF) para realizar un experimento de agarre y validar el control. A su vez, [31] presenta un método de control híbrido de fuerza/posición basado en redes neuronales. Esta se emplea para compensar la parte de la dinámica del manipulador que no se conoce. Esto posibilita que el controlador vaya aprendiendo en línea. Utilizándose un robot manipulador de dos enlaces para realizar simulaciones con el fin de validar el controlador. En [32] se propone implementar el HMFC utilizando un método de control de lógica difusa. La estrategia se valida en un robot manipulador de 7 DoF.

Una de las principales limitantes de los esquemas de control donde se realimenta la fuerza es la necesidad de medirla de forma directa. Estos sensores tienen un alto costo, un ancho de banda limitado, introducen peso en el brazo y necesitan variaciones de la estructura para su instalación. Por estas razones se han desarrollado investigaciones que buscan sustituir estos sensores por estimadores de fuerzas. En [33] se estudia la relación entre las velocidades iniciales del efector y el impulso de contacto resultante para estimar la fuerza de contacto. En [34] estiman la fuerza de contacto usando la ecuación de movimiento del sistema. En [35] proponen el uso de un observador de perturbación como estimador de fuerza. Más tarde [36] lo retoman empleando un modelo lineal en su diseño, demostrando su eficiencia. Mientras que en la propuesta de [37] lo emplean basado en un modelo no lineal para la estimación de la fuerza de contacto durante la interacción háptica. Una aplicación de un observador no lineal para estimar las fuerzas ejercidas por un manipulador de 6 DoF es expuesta en [38]. Se demuestra que las fuerzas externas pueden ser bien estimadas usando solamente mediciones de los motores. En [39] se emplea un observador de perturbación para estimar las fuerzas de contacto durante la captura de un satélite por un robot de servicio espacial. Demostrándose un comportamiento casi idéntico entre la fuerza actual y la estimada por el observador.

1.1.5. Esquemas propuestos para controlar la interacción durante la programación por guiado.

Como se trato anteriormente, el método de programación guiada consiste en que el usuario mueva el efector final del robot a las posiciones deseadas, mientras el controlador del robot graba las coordenadas de las articulaciones correspondientes, para luego reproducir la trayectoria. Para que este método sea aplicable a cualquier robot, independientemente de su tamaño, de las altas relaciones en las cajas de engranaje o del efecto de la gravedad sobre los eslabones, los motores deben estar energizados. Esto es lo que va a permitir que al operador humano pueda mover fácilmente los eslabones de acuerdo con la tarea a realizar [40], pero al mismo tiempo hace que se requiera de un estricto control sobre los movimientos del robot. El operador humano puede ser visto usualmente como un tipo especial de entorno [41]. Por lo que no es extraño que, para lograr el guiado manual, se hayan implementado varios de los esquemas de control expuestos anteriormente.

En [40] se implementa un sistema de programación por guiado para un robot cuya tarea es soldar los cascos de los barcos en un astillero. En este se instala un sensor de fuerza/par en la herramienta, el cual es actuado por el operador. Un controlador de impedancia hace que el robot se mueva en función de la fuerza ejercida por el operador. Hay tareas, como la pintura, que requieren de mucha habilidad por parte del personal que las desarrolla. Para no interferir con el proceso de enseñanza es necesario que el operador sienta la misma reacción del sistema robótico como si estuviera usando la herramienta real. Por lo cual [42], rescata un concepto que ya había sido usado antes en aplicaciones donde humanos y robots debían compartir una carga pesada, el de herramienta virtual. Su contribución es presentar una formulación exacta de las ecuaciones dinámicas de una herramienta virtual, obtenidas por el método de Newton-Euler. Junto a esto el uso de un controlador de impedancia basado en las mediciones de fuerza/par, para lograr controlar la interacción entre el operador y el robot en un robot industrial.

El trabajo desarrollado por [43] tiene sus basamentos en los descritos anteriormente y su aporte esta enfocado en aumentar la seguridad durante el guiado. Para conseguirlo se propone dos estrategias: se limita el movimiento del robot, disminuyendo su velocidad cuando el centro de la herramienta alcanza un valor limite; y se introducen barreras virtuales que limitan el área de trabajo del robot. Ambas estrategias explotan las ventajas del controlador de admitancia propuesto. Aplicaciones similares se pueden encontrar en [44], donde se emplea en un robot *pick and place*, y en [9] el cual desarrolla una aplicación de programación por guiado para un robot pintor.

1.2. Definición del problema.

Según [9] las técnicas de enseñanza por guiado manual posibilitan la programación de un robot de una manera intuitiva y sencilla. Estos disminuyen los tiempos empleados para realizar esta tarea y la necesidad de poseer conocimientos de difíciles lenguajes de programación. A pesar de estas ventajas, la enseñanza por guiado manual solo ha sido implementada comercialmente en robots manipuladores ligeros (conocidos como *Lightweight*). Estos tienen la ventaja de poseer sensores de torque en cada una de sus articulaciones, lo que los hace idóneos para este tipo de enseñanza. Sin embargo, su capacidad de carga es limitada lo que hace que no puedan ser empleados en todas las aplicaciones para las cuales tradicionalmente se aplica la robótica. Comprobarse en [45], donde se analizan la experiencia de empresas alemanas que ya usan robots manipuladores ligeros para 25 aplicaciones diferentes.

Para poder implementar la enseñanza por guiado manual en un robot tradicional, las soluciones dadas hasta el momento están enfocadas en la instalación de un sensor de fuerza/torque. Esto es un procedimiento invasivo para el robot ya que no se contempló en su diseño, y además los materiales requeridos son costosos. De aquí se deriva la pregunta que guía esta investigación: ¿Como implementar la enseñanza por guiado manual sin la necesidad de constar con un sensor de fuerza/torque?

1.3. Objetivos.

1.3.1. Objetivo General.

Desarrollar una estructura de control que permita al operador humano guiar manualmente un prototipo robótico durante la fase de programación. La estructura desarrollada debe incluir un Observador de Perturbaciones para estimar la fuerza aplicada por el operador durante el guiado.

1.3.2. Objetivos específicos.

1. Realizar una revisión de literatura que permita adquirir conocimiento sobre la enseñanza por guiado manual, la interacción entre humanos y robots, así como las estrategias de control más empleadas para estas aplicaciones. Las metodologías para estimar la fuerza de contacto en aplicaciones robóticas también es objeto de esta revisión.
2. Diseñar y fabricar un prototipo robótico de bajo costo que permita realizar experimentos de validación de las estrategias propuestas.

3. Diseñar y validar un Observador de Perturbación para estimar la fuerza aplicada por el operador en la herramienta del prototipo de prueba.
4. Integrar el Observador de Perturbaciones dentro de la arquitectura de control aplicada a la enseñanza por guiado manual.
5. Validar la arquitectura propuesta mediante experimentos en el prototipo de pruebas.

1.4. Justificación.

El empleo de un Observador de Perturbaciones dentro del esquema de programación por guiado manual permitirá llevar este método a robots industriales tradicionales. Eliminando la necesidad de ejecutar modificaciones en su estructura para instalar los sensores de fuerza/torque, y permitiendo abaratar la solución al no contar con el costoso sensor. El habilitar esta forma de programación en un robot tradicional posibilita que el personal encargado de esta tarea en la fábrica requiera de menos conocimientos especializados, realizando un procedimiento intuitivo y asequible.

1.5. Metodología.

Revisión de literatura. Se realizará una revisión de la literatura para conocer el estado actual y las tendencias sobre métodos de enseñanzas en entornos colaborativos. Se pretende establecer cuáles estructuras de control son las que mejores resultados han reportado y sobre qué plataforma se realizan los experimentos de validación.

Desarrollo de la estructura de control. Con base en la teoría existente se propondrá una estructura de control que permita el guiado manual de un robot manipulador en entornos colaborativos. La fuerza de guiado ejercida por el operador se estimará empleando un Observador de Perturbaciones. La estructura será probada haciendo uso de un mecanismo de 1 DoF.

Fabricación del prototipo. Diseñar y fabricar un prototipo para validar la estructura de control propuestas. Las partes rígidas del manipulador serán fabricadas en plástico. Se empleará componentes comerciales para los motores, sensores y unidades de control.

Programación. Desarrollar mediante técnicas analíticas el modelo matemático del manipulador. Determinar mediante técnicas experimentales los parámetros del modelo que mejor ajusten el prototipo real. Desarrollar la programación del prototipo.

Validación por medio de experimentos. Se llevarán a cabo experimentos para validar el correcto funcionamiento de la estructura de control propuesta.

Capítulo 2

Marco Teórico

Las bases de los robots industriales modernos fueron establecidas por el ingeniero e inventor norteamericano George C Devol. Este comenzó a trabajar desde 1956 con Joseph F Engelberger en un dispositivo de transferencia de artículos programada. Así fundan la Consolidated Controls Corporation, que más tarde se pasó a llamar Unimation, y en 1960 instalan su primera máquina en la planta de General Motors en New Jersey. En los años posteriores este sector siguió en crecimiento, Japón aventajo a los EE.UU., mientras que Europa se retrasa en sumarse a las tendencias. No fue hasta 1973 cuando la empresa sueca ASEA, lanza su robot IR6. Sin embargo, ya para 1980 se funda la Federación Internacional de Robótica con sede en Estocolmo, Suecia. La Asociación Internacional de Estándares (ISO) en su norma 8373 de 1998 define a los **robot manipuladores** como:

Manipulador de 3 o más ejes, con control automático, reprogramable, multiaplicación, móvil o no, destinado a ser utilizado en aplicaciones de automatización industrial. Incluye al manipulador (sistema mecánico y actuadores) y al sistema de control (software y hardware de control y potencia).

Mecánicamente, un robot manipulador está formado por una serie de elementos o eslabones unidos mediante articulaciones que permiten un movimiento relativo entre dos eslabones consecutivos. El movimiento de cada articulación puede ser de desplazamiento (articulación prismática), de giro (articulación rotacional), o una combinación de ambos. Cada uno de los movimientos independientes que puede realizar una articulación con respecto a la anterior, se denomina grado de libertad (GDL). La estructura mecánica de un robot manipulador constituye una cadena cinemática. En caso de que sólo haya un camino posible para llegar desde cualquier eslabón a cualquier otro, se dirá que se trata de una cadena cinemática abierta. Los robots con cadena cinemática cerrada y en particular los robots denominados de estructura paralela, son menos frecuentes [46].

2.1. Modelos matemático de un robot manipulador.

2.1.1. Modelo cinemático directo de un robot manipulador.

El modelo cinemático directo permite conocer la posición y orientación que adopta la herramienta cuando se conocen las variables que fijan la posición y orientación de sus articulaciones. Para obtener dicho modelo existen dos métodos:

1. Método geométrico: Consiste en encontrar las relaciones que permiten conocer la localización espacial de la herramienta a partir de sus coordenadas articulares. No existe un procedimiento específico para su aplicación por lo cual no puede sistematizarse. Su aplicación esta limitada a robots con pocos grados de libertad, volviéndose inoperativo con el aumento de los grados.
2. Método basado en cambios de sistemas de referencia: Si se considera el manipulador como una cadena cinemáticas formada por objetos rígidos (eslabones) unidos entre sí mediante articulaciones, el problema se reduce a encontrar una matriz de transformación homogénea que relaciones la posición y orientación de la herramienta respecto a la base. Esta matriz sera función de las coordenadas articulares y dependerá de las constantes geométricas del eslabón.

$$\mathbf{T}(\theta_1 \dots \theta_n) = {}^0 \mathbf{A}_1(\theta_1) \dots {}^{n-1} \mathbf{A}_n(\theta_n) \quad (2.1)$$

Para poder sistematizar la obtención de las ecuaciones cinemáticas de la cadena completa, Denavit-Hartenberg propusieron un método para localizar cada sistema de coordenadas ligado al eslabón.

2.1.2. Modelo cinemático inverso de un robot manipulador.

Para una cadena abierta general de n grados de libertad con cinemática directa $\mathbf{T}(\boldsymbol{\theta})$, donde $\boldsymbol{\theta} \in \mathbb{R}^n$, el problema de cinemática inversa queda definido de la siguiente manera: dada una transformación homogénea $\mathbf{X} \in SE(3)$, encontrar las soluciones de $\boldsymbol{\theta}$ que satisfacen $\mathbf{T}(\boldsymbol{\theta}) = \mathbf{X}$ [47]. Lo que se entiende como encontrar los valores de las variables articulares $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_n]^T$ para que la herramienta se posicione y oriente en una localización espacial dada [46].

Los métodos más empleados para la obtención de este modelo son [46, 48]:

1. Método numérico: Este método es iterativo, por lo que su velocidad de convergencia es lenta, e incluso no siempre esta garantizada. Este método también es sensible a los

sistemas operativos o a los propios *software* usados para resolver el sistema. Además, en muchas ocasiones las soluciones del problema de cinemática inversa no es única, lo cual hace que existan varias combinaciones de θ que posicionen y orienten el efector final. Por estas razones es más conveniente emplear métodos que proporcionen una solución cerrada, los cuales aseguren la solución más adecuada entre las posibles.

2. Método geométrico: Es un método gráfico, que en ocasiones suele ser difícil de visualizar para robots con más de 3 gdl. Esta basado en relaciones geométricas y trigonométricas sobre los eslabones y articulaciones del robot, para lo cual se recurre a formar triángulos rectángulos. Su mayor ventaja es que entrega todas las soluciones del sistema.
3. Método analítico: O basado en la estructura de la matriz de transformación homogénea.

2.1.3. Modelo diferencial de un robot manipulador.

El modelo diferencial de un robot manipulador es expresado en la **Matriz Jacobiana**. Existen dos formas de formular dicho modelo:

1. Matriz Jacobiana analítica: Establece la relación de las velocidades articulares con las velocidades de la localización del efector final, siendo esta la posición y orientación expresada en base a sus coordenadas cartesianas y ángulos de Euler $(\dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi})$.
2. Matriz Jacobiana geométrica: Establece la relación de las velocidades articulares con los vectores de velocidad lineal y angular $(v_x, v_y, v_z, w_x, w_y, w_z)$ con que se mueve el efector final, expresados en un sistema de referencia determinado.

La principal ventaja de la **Matriz Jacobiana** es convertir las velocidades y las fuerzas entre el espacio articular y el espacio operacional del robot, para lo cual se expresan las relaciones presentadas en 2.2 y 2.3.

$$\dot{\mathbf{x}}_e(t) = \mathbf{J}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}(t) \quad (2.2)$$

$$\boldsymbol{\tau}(t) = \mathbf{J}^T(\boldsymbol{\theta})\mathbf{f}_e(t) \quad (2.3)$$

2.1.4. Modelo dinámico de un robot manipulador.

El modelo dinámico de un manipulador robótico es muy útil por varias razones: simulación del movimiento del robot, análisis de su diseño y desempeño, diseño y evaluación de los controladores, o formar parte del mismo controlador, entre otras. El método empleado para generarlo depende entonces del uso que se le vaya a dar al modelo [49].

Las ecuaciones dinámicas del robot son comúnmente derivadas empleando el método de Euler-Lagrange o el de Newton-Euler, pero también son empleados otros como el D Alembert Generalizado y el Lagrangiano Recursivo. El primero es bastante efectivo para robots con estructuras simples, con tres o menos grados de libertad. Sin embargo, para los robots con más grados de libertad, genera ecuaciones complejas que no tienen buenos resultados si se emplean para control en tiempo real. El método de Newton-Euler proporciona un conjunto de ecuaciones cuya complejidad varía linealmente con el número de articulaciones del manipulador, pero cuya falta de estructura complica el desarrollo de técnicas de control avanzadas[47, 49].

El empleo de las ecuaciones de Euler-Lagrange para el modelado dinámico de un manipulador comienza por elegir un conjunto de coordenadas independientes $\boldsymbol{\theta} \in \mathbb{R}^n$ que describan la configuración del sistema.

Luego se define una función lagrangiana $\mathcal{L}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ como la energía cinética general del sistema $K(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ menos la energía potencial $P(\boldsymbol{\theta})$:

$$\mathcal{L}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = K(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - P(\boldsymbol{\theta}) \quad (2.4)$$

Por consiguiente las ecuaciones de movimiento de Euler-Lagrange para un robot manipulador de n GDL, viene dada por la ecuación 2.5:

$$\tau_i = \frac{d}{dt} \frac{\partial \mathcal{L}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})}{\partial \dot{\theta}_i} - \frac{\partial \mathcal{L}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})}{\partial \theta_i}, i = 1, \dots, n \quad (2.5)$$

El modelo matemático que describe la dinámica de un robot manipulador, escrito en su forma matricial, puede ser expresada por un conjunto de ecuaciones de segundo orden de la forma:

$$\boldsymbol{\tau} = \mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{g}(\boldsymbol{\theta}) \quad (2.6)$$

Donde:

$\boldsymbol{\tau} \in \mathbb{R}^n$: es el vector de los torques o las fuerzas aplicadas en las articulaciones.

$\mathbf{M}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \in \mathbb{R}^{n \times n}$: es la matriz de inercia, es una matriz simétrica y definida positiva.

$\boldsymbol{\theta} \in \mathbb{R}^n$: es el vector que representa la posición de las articulaciones.

$\mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \in \mathbb{R}^n$: es el vector de las fuerzas centrífuga y de Coriolis.

$\mathbf{g}(\boldsymbol{\theta}) \in \mathbb{R}^n$: es el vector de las fuerzas gravitacionales [47].

En el caso de un robot que solo se mueva en el plano horizontal, como el robot SCARA de 2 GDL, o aquellos en cuyo diseño mecánico lo consideraran, su modelo dinámico no contiene el termino gravitacional $\mathbf{g}(\boldsymbol{\theta})$. Según [50] para este caso su modelo pasa a ser:

$$\tau = \mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \quad (2.7)$$

2.1.5. Validación de los modelos matemáticos.

Una vez definidos los parámetros del modelo se debe validar si este reproduce de manera aceptable el comportamiento del sistema. En caso contrario, se debe volver a estimar los parámetros y volver a validar hasta que se encuentre el modelo más simple y que mejor capture la dinámica del sistema. En [51], este problema lo dividen en tres preguntas: ¿El modelo coincide de manera aceptable con los datos obtenidos del sistema real? ¿El modelo es suficientemente bueno para el propósito que se va a emplear? ¿El modelo describe al sistema real? Las técnicas de validación desarrolladas se centran en la primera pregunta, para dar respuesta a la segunda que es lo que realmente importa, ya que la tercera no puede ser contestada de manera objetiva.

En [52], presentan varias herramientas que se utilizan para validar modelos matemáticos. Una de las más sencillas es comparar la salida del modelo simulado con la salida medida del sistema real para una misma señal de entrada. En esta prueba no basta con graficar las dos salidas y compararlas de manera visual, sino que se han establecido ciertos indicadores.

1. Error cuadrático medio:

$$\%Ajuste = \frac{\|m - m_{ref}\|^2}{N_s} \quad (2.8)$$

2. Raíz normalizada del error cuadrático medio:

$$\%Ajuste = 1 - \frac{\|m_{ref} - m\|}{\|m_{ref} - \bar{m}_{ref}\|} \quad (2.9)$$

3. Error cuadrático medio normalizado:

$$\%Ajuste = 1 - \frac{\|m_{ref} - m\|^2}{\|m_{ref} - \bar{m}_{ref}\|^2} \quad (2.10)$$

En las ecuaciones anteriores: m son los datos de prueba, m_{ref} los datos de referencia, N_s el número de muestras y $\|$ representa la norma-2 del vector. Un 100% corresponde a un ajuste perfecto y un 0%, que el modelo no es mejor que estimar la salida usando una constante. Estos indicadores también pueden ser valores negativos. En este caso el algoritmo de estimación fallo en converger o los datos de estimación y validación no fueron procesados de la misma manera.

2.2. Observador de perturbación.

El Observador de Perturbaciones fue introducido en [53], como una vía de hacer más sencillo el control de velocidad de motores de DC empleando técnicas de control moderno. El objetivo fue encontrar un método alternativo a la acción integral del controlador PID clásico, para así, compensar el efecto de lo que denomino como “perturbación equivalente”. Esta no solo esta compuesta por el torque de carga, sino que incluye los efectos de la fricción y la incertidumbre en la constante de torque e inercia [54].

El torque impuesto por la carga (τ_{ext}) al motor es desconocido, este puede ser estimado mediante un *zero-observer* con error de estado estacionario nulo. Este observador puede ser diseñado como:

$$\hat{\tau}_{ext} = \frac{g}{S+g}(Ku(t) + gJ\dot{\theta}) - gJ\dot{\theta} \quad (2.11)$$

En el observador de la ecuación 2.11, $\hat{\tau}_{ext}$ esta compuesto por un filtro pasa bajos y el modelo dinámico del motor, presentado en las ecuaciones 3.24 y 3.25. La constante del filtro, g debe ser definida como un número positivo, y debe seleccionarse lo más grande posible. Esta condición hace que el DOB sea insensible a variación en los parámetros [55, 56]. En la figura 2.1 se representa el diagrama en bloques del DOB.

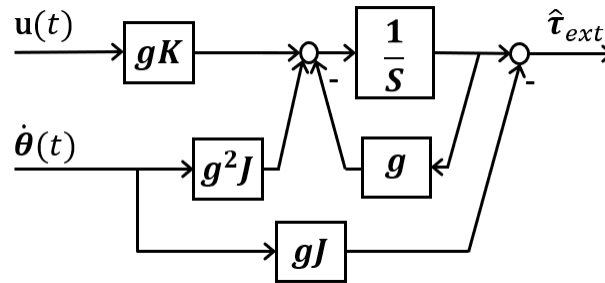


Figura 2.1: Diagrama de bloques del Observador de Perturbaciones.

Poco tiempo después de su introducción, se comenzó a emplear en el campo del control de robots manipuladores, donde su uso se ha extendido y diversificado, encontrando varias aplicaciones en [57]:

1. Cuando para controlar el manipulador se emplean un controlador independiente en cada articulación, el uso del DOB mejora su desempeño. Esto se debe a que el torque estimado considera el acoplamiento con otro eslabón y lo compensa.

2.3. ESTRUCTURA DE CONTROL EMPLEADA DURANTE EL GUIADO MANUAL.17

2. También ha sido empleado para la estimación de la fricción.
3. Se ha empleado para el control de fuerza cuando se requiere realimentación, donde el DOB actúa como sensor de torque. Esto simplifica la estructura del sistema, reduce costos, y mejora la fiabilidad.
4. Cuando el manipulador trabaja en entornos poco conocidos, el DOB puede ser empleado para monitoreo y planificación de trayectoria en lugar de control.

Los modelos para estimar los torques también han sido mejorados, o adaptados a una aplicación dada, con el de cursar de los años. Así en [39] presentan la ecuación 2.13:

$$\hat{\mathbf{f}}_{\text{ext}} = \mathbf{M}(\mathbf{x})\ddot{\mathbf{x}} + \mathbf{h}(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}} - \mathbf{f} \quad (2.12)$$

o lo que es lo mismo:

$$\hat{\mathbf{f}}_{\text{ext}} = \mathbf{J}^{-T}(\boldsymbol{\theta}) \left(\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \boldsymbol{\tau}(t) \right) \quad (2.13)$$

2.3. Estructura de control empleada durante el guiado manual.

Las tareas a realizar por un robot en una celda de manufactura habitualmente están especificadas como una trayectoria de la herramienta. Sin embargo, las acciones de control necesarias para realizarla se deben ejecutar a nivel de las articulaciones actuadas. El primer paso es convertir la trayectoria deseada, que se encuentra en términos de coordenadas de la herramienta, a las correspondientes coordenadas articulares deseadas. Para conseguir este objetivo se emplea el modelo cinemático inverso del robot. Luego un controlador determina el torque necesario para mover el manipulador sobre la trayectoria deseada, realimentando la posición y velocidad de los brazos. Para finalizar las coordenadas de la herramienta son calculadas partiendo de las variables articulares haciendo uso del modelo cinemático directo del robot.

En entornos poco conocidos, el movimiento de la herramienta esta sujeto a modificaciones en línea, con el objetivo de responder a eventos inesperados o responder a la medición de un sensor. Esto se vuelve esencial cuando se necesita controlar la interacción entre el manipulador y el entorno [58].

2.3.1. Controlador de posición.

Cuando se quiere controlar la posición de un robot cuyo modelo no contenga componentes gravitacionales es suficiente el empleo de un Controlador Proporcional-Derivativo (PD) [50]. El controlador PD es una extensión inmediata del Controlador Proporcional con Realimentación de Velocidad. Este a su vez es el controlador a lazo cerrado más simple que puede ser empleado en el control de posición de las articulaciones de un manipulador actuado con motores de DC.

La ecuación que describe el Controlador Proporcional con Realimentación de Velocidad es presentada en 2.2.

$$u(t) = \mathbf{K}_p \tilde{\boldsymbol{\theta}} - \mathbf{K}_v \dot{\boldsymbol{\theta}} \quad (2.14)$$

Donde:

$\mathbf{K}_p, \mathbf{K}_v \in \mathbb{R}^{n \times n}$: Son matrices positivas definidas que representan las ganancias de posición y de velocidad respectivamente.

$\tilde{\boldsymbol{\theta}} = (\boldsymbol{\theta}_{ref} - \boldsymbol{\theta}) \in \mathbb{R}^n$: Es el error de posición, donde $\boldsymbol{\theta}_{ref}$ es la posición deseada y $\boldsymbol{\theta}$ es la posición real medida por los *encoder*.

El diagrama de bloques de un robot manipulador controlado por un Controlador Proporcional con Realimentación de Velocidad es mostrado en la figura 2.2.

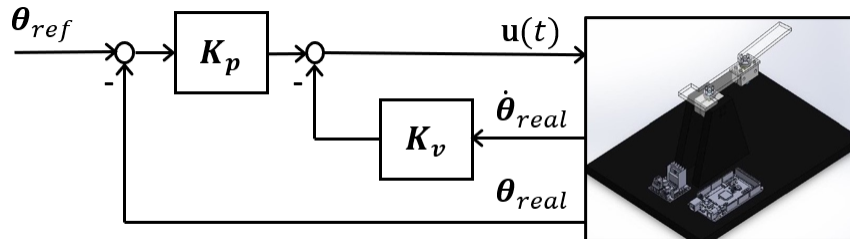


Figura 2.2: Diagrama de bloques del Controlador Proporcional con Realimentación de Velocidad.

2.3.2. Controlador de admitancia.

La manera más simple de implementar el controlador que permita el guiado manual es cerrar un lazo por fuera del lazo de control de posición, cuya entrada sea la medición de un sensor de fuerza/torque montado entre la herramienta y el robot, que permita calcular la

2.3. ESTRUCTURA DE CONTROL EMPLEADA DURANTE EL GUIADO MANUAL.19

corrección de posición y orientación de la misma. Luego la salida de ese controlador debe ser convertida a posiciones en el espacio articular por medio de la cinemática inversa del robot. Estas posiciones pasan a ser las referencias de los controladores internos de posición. Los cuales se asume que fueron diseñados para minimizar el error de seguimiento y optimizar la respuesta dinámica. Lo que se traduce en que la posición actual de la herramienta sea igual a la referencia de posición calculada por el controlador [44, 9].

Según [40] para que el controlador haga que el robot se mueva en respuesta a la fuerza medida, es importante poder controlar la dinámica del manipulador, o controlar su “impedancia”.

Con base en el modelo dinámico de un robot manipulador descrito en 2.7, y considerando que se encuentra bajo la acción de una fuerza (\mathbf{f}_{ext}) producida por el operador humano:

$$\mathbf{f} + \mathbf{f}_{\text{ext}} = \mathbf{M}(x)\ddot{\mathbf{x}} + \mathbf{h}(x, \dot{x})\dot{\mathbf{x}} \quad (2.15)$$

Lo que se persigue con el controlador de admitancia es establecer una relación dinámica entre el movimiento del robot y la fuerza aplicada por el operador. Para que el robot obedezca a la fuerza, esta relación puede ser establecida como un sistema masa-resorte-amortiguador, cuya ecuación es descrita en 2.16:

$$\mathbf{f}_{\text{ext}} = \mathbf{M}_d\ddot{\tilde{\mathbf{x}}} + \mathbf{D}_d\dot{\tilde{\mathbf{x}}} + \mathbf{K}_d\tilde{\mathbf{x}} \quad (2.16)$$

Donde:

\mathbf{M}_d , \mathbf{D}_d y \mathbf{K}_d : son las matrices de inercia, amortiguamiento y rigidez, respectivamente. Las mismas deben ser matrices simétricas y definidas positivas. Pueden ser ajustadas a valores diferentes a los parámetros físicos del robot y así obtener el comportamiento deseado [40].

$\tilde{\mathbf{x}}(t) = \mathbf{x}(t) - \mathbf{x}_d(t)$: es el delta de posición calculado por el controlador de admitancia.

En esta aplicación el operador debe guiar el robot a través de la fuerza aplicada sin especificar directamente la posición requerida. Además se debe facilitar el manejo del robot, como si trabajara en un entorno de gravedad cero. Para conseguir estos requerimientos $\mathbf{K}_d = 0$, mientras que \mathbf{M}_d y \mathbf{D}_d son especificados según las preferencias del operador. La ecuación que describe este caso particular de interacción, donde la herramienta se comporta como un sistema de masa-amortiguador, se reescribe como:

$$\mathbf{f}_{\text{ext}} = \mathbf{M}_d\ddot{\tilde{\mathbf{x}}} + \mathbf{D}_d\dot{\tilde{\mathbf{x}}} \quad (2.17)$$

Si el robot se siente muy pesado, se deben disminuir los valores de \mathbf{M}_d . Si el robot se mueve demasiado rápido, se deben aumentar los valores de \mathbf{D}_d . Para un posicionamiento preciso se requiere que los valores de \mathbf{D}_d sean relativamente altos [40, 9].

La arquitectura de control resultante para el Método de enseñanza por guiado manual de un robot es mostrado en la figura 2.3.

2.3. ESTRUCTURA DE CONTROL EMPLEADA DURANTE EL GUIADO MANUAL.20

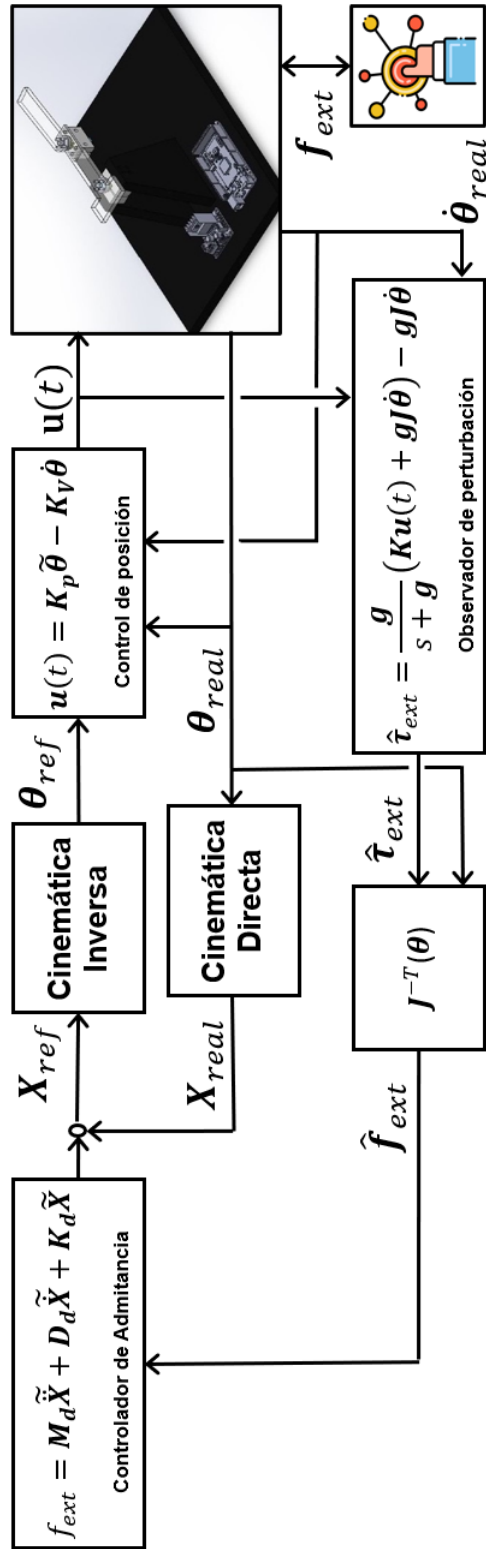


Figura 2.3: Estructura de control propuesta. Método de enseñanza por guiado manual.

Capítulo 3

Desarrollo del Proyecto

3.1. Diseño y fabricación del prototipo de pruebas.

El robot SCARA de tres articulaciones, como los mostrados en la figura 3.1, es el tipo de manipulador más simple que demuestra las características no lineales de un manipulador robótico como son las fuerzas centrífugas y de Coriolis [49]. Por esta razón se eligió como la configuración para el prototipo de pruebas que se empleara en el presente trabajo. Para los experimentos que se desarrollaran no se considera ninguna herramienta en el extremo del robot, por lo que se eliminara la tercera articulación. La estructura del prototipo debe ser de pequeñas dimensiones, de materiales ligeros y fácil maquinado. Para la base se selecciono y los brazos fueron fabricados de acrílico de 5 mm de espesor. El diseño mecánico se realizó utilizando un programa CAD que permitió comprobar la funcionalidad del ensamble, el diseño realizado es mostrado en la figura 3.2. Las no linealidades del sistema son causadas por las fuerzas centrífugas y de Coriolis que actúan en los brazos, y las fuerzas que se transmiten a través de las articulaciones. Existe una relación de incremento cuadrático entre estas fuerzas y la velocidad angular. Por esta razón [49] propone el uso de motores con alto torque y que sean capaces de acelerar los brazos a altas velocidades. Otros factores a considerar son: cuan fácil se pueda realizar el control de posición (continuidad, realimentación, etc) y los costos. Debido a estos requerimientos el prototipo será actuado por motores de corriente directa de imanes permanentes. Como característica específica poseerán un reductor mecánico que garantice el torque requerido, lo que permite reducir el tamaño del motor y por tal razón su masa. Al igual deberán contar con un *encoder* que permita realimentar de manera continua la posición del eje del motor.

Los motores seleccionados son del fabricante Polulu modelo 3076. Estos motores deben ser alimentados con 6 VDC y consumen una corriente máxima de 1.5 A. Al motor viene acoplado un reductor mecánico con una relación de 150:1, lo que le permite ejercer un torque de 0.41 kgcm y una velocidad de 220 rpm. Como características extras se agrego un *encoder*



Figura 3.1: Manipuladores SCARA de 3 gdl comerciales.

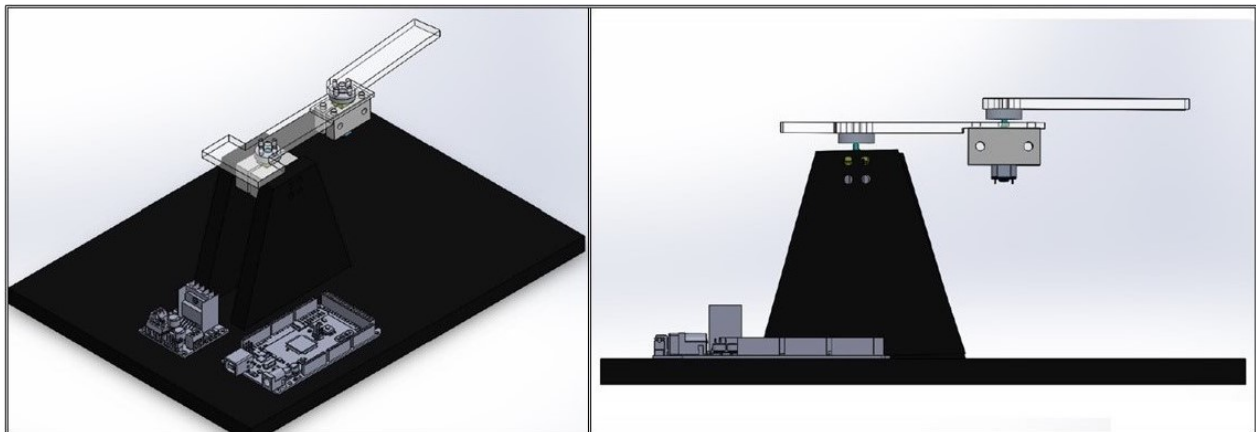


Figura 3.2: Diseño CAD del prototipo de pruebas.

magnético por Efecto Hall que trabaja en cuadratura para proporcionar una salida de 12 CPR.

Para la etapa de potencia del sistema se emplea un módulo controlador de motores que cuenta con un circuito integrado L298N que posee dos canales de Puente H. Estos son utilizados para manipular el sentido de giro y la velocidad de los dos motores de DC del prototipo. Para controlar el prototipo se emplea una placa controladora de Arduino modelo MEGA 2560. Esta posee 54 pines de entrada/salida digitales, 16 analógicos y 4 UARTs (puertos serial). De los pines digitales se pueden emplear 14 como salidas de señales con modulación por ancho de pulso (PWM, por sus siglas en inglés) que permiten manipular el L298N. También dispone de 6 entradas de interrupciones, las cuales son empleadas para conectar las señales proveniente de los *encoder*. Información más detallada de los componentes puede ser consultada en el Apéndice A. El prototipo de pruebas es mostrado en la figura 3.3.



Figura 3.3: Prototipo de prueba real.

Con el objetivo de validar el comportamiento de la fuerza estimada por el Observador de Perturbación al prototipo de pruebas se le acopla un sensor de fuerza. En base a las experiencias expuestas en [59] el dispositivo empleado es un sensor de fuerza resistivo (FSR, por sus siglas en inglés), el modelo seleccionado es el FSR 400 del fabricante Interlink Electronics. Este tipo de sensor están conformados por una película de polímero y un material semiconductor en su interior. Estos responden con una disminución de su resistencia con el aumento de la fuerza aplicada a la superficie activa del sensor. Según el fabricante su diseño está optimizado para el uso en dispositivos de interfaz hombre-máquina, incluyendo electrónica automotriz, sistemas médicos, controles industriales y robótica [60]. Tiene un rango de medición de 0.2 a 2 N y un área activa de diámetro igual a $5mm^2$. Su conexión a una entrada analógica de controlador se realizó empleando un divisor de voltaje con una $R = 10K\Omega$.

Después de adquirir la señal se convierte de voltaje a fuerza, y se realiza un filtrado de la misma.

El *software* empleado para la programación del sistema es el paquete Simulink, perteneciente a MATLAB 2018a. Para la comunicación con el prototipo se emplearon las cajas de herramientas: *Simulink Support Package for Arduino Hardware* y el *Rensselaer Arduino Support Package Library (RASPLib)*. Estas cuentan con los bloques necesarios para leer la señal analógica del sensor de fuerza y las señales de pulsos provenientes de los *encoders*, así como los bloques para enviar las señales de PWM para controlar los motores. La velocidad de ambos motores se obtuvo calculando la derivada de la posición entregada por los *encoders*. El valor obtenido por este método presenta mucho ruido por lo que se empleó un filtro digital con respuesta de impulso de duración finita (FIR, por sus siglas en inglés), para suavizar la señal y permitir mejorar los cálculos posteriores. El código desarrollado para la comunicación es proporcionado en el Apéndice B.

3.2. Modelado matemático del prototipo de pruebas.

3.2.1. Modelo cinemático directo.

La configuración SCARA 2 gdl mostrada en la figura 3.3, posee dos articulaciones rotacionales cuyos ejes de rotación son paralelos entre si, lo que causa que el movimiento sea en un solo plano. El mecanismo posee dos grados de libertad, por lo que se necesitan dos variables para definir su posición. Las variables seleccionadas son el ángulo formado entre el primer eslabón y el marco de referencia (θ_1), y el ángulo formado entre el primer eslabón y el segundo (θ_2). Debido a la sencillez de este prototipo se empleara el método geométrico para determinar el modelo cinemático directo.

En la figura 3.4 se define el marco de referencia $\{M_0\}$, un sistema cartesiano de dos dimensiones con origen en el eje de rotación del primer motor. Las coordenadas cartesianas (p_x y p_y) denotan la posición del extremo final de la segunda articulación, con respecto al marco de referencia. Las ecuaciones matemáticas que describen la dependencia que existe entre las coordenadas (p_x y p_y) y las coordenadas articulares (θ_1 y θ_2) son:

$$p_x = L_1 \cos \theta_1 + L_2 \cos(\theta_1 + \theta_2) \quad (3.1)$$

$$p_y = L_1 \sin \theta_1 + L_2 \sin(\theta_1 + \theta_2) \quad (3.2)$$

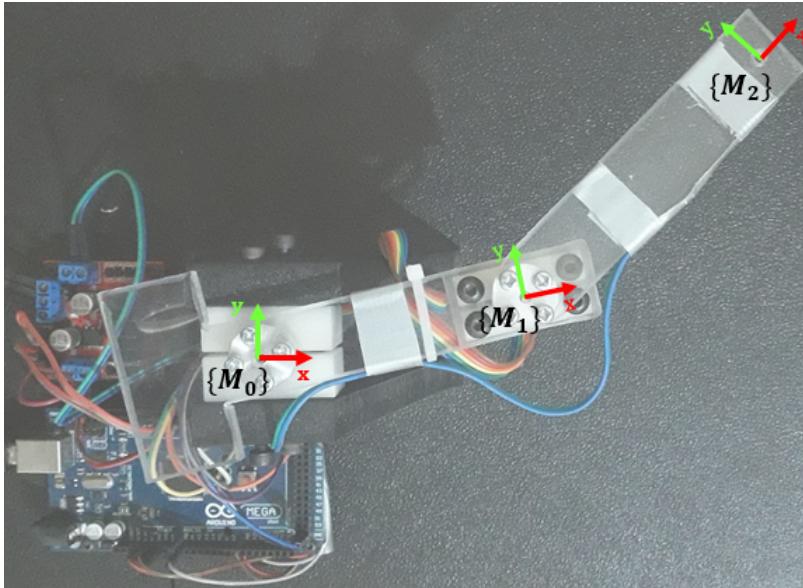


Figura 3.4: Definición de marcos de referencia para el prototipo de pruebas.

3.2.2. Modelo cinemático inverso.

Empleando el método geométrico y partiendo de las coordenadas (p_x, p_y) referidas a $\{M_0\}$ en las que se quiere posicionar el extremo del robot, se representan las dos articulaciones como se muestra en la figura 3.5.

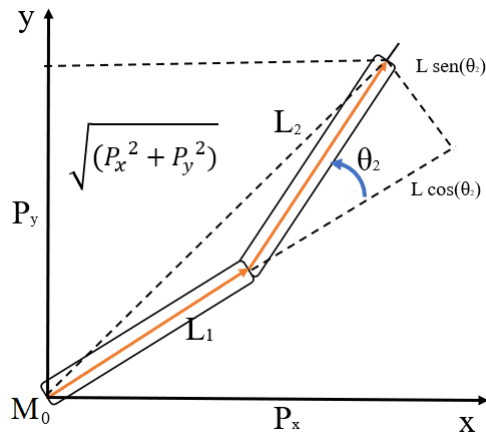


Figura 3.5: Esquema de un manipulador SCARA de 2 gdl para el ángulo θ_2 .

Definiendo el triángulo rectángulo representado y haciendo uso del Teorema de Pitágoras:

$$\left(\sqrt{p_x^2 + p_y^2}\right)^2 = [L_1 + L_2 \cos(\theta_2)]^2 + [L_2 \sin(\theta_2)]^2 \quad (3.3)$$

$$p_x^2 + p_y^2 = L_1^2 + L_2^2 \cos^2(\theta_2) + 2L_1 L_2 \cos(\theta_2) + L_2^2 \sin^2(\theta_2) \quad (3.4)$$

$$p_x^2 + p_y^2 = L_1^2 + L_2^2 \cos^2(\theta_2) + \sin^2(\theta_2) + 2L_1 L_2 \cos(\theta_2) \quad (3.5)$$

$$\cos(\theta_2) = \frac{p_x^2 + p_y^2 - L_1^2 - L_2^2}{2L_1 L_2} \quad (3.6)$$

Despejando θ_2 de 3.6:

$$\theta_2 = \arccos\left(\frac{p_x^2 + p_y^2 - L_1^2 - L_2^2}{2L_1 L_2}\right) \quad (3.7)$$

Con el objetivo de evitar las dos soluciones que resultan de la función arco coseno, se puede realizar una conversión para usar la arco tangente. Esta función tiene una precisión uniforme en todo el rango de sus argumentos y devuelve un valor único.

$$\theta_2 = \operatorname{atan}\left(\frac{\sqrt{1 - \left(\frac{p_x^2 + p_y^2 - L_1^2 - L_2^2}{2L_1 L_2}\right)^2}}{\frac{p_x^2 + p_y^2 - L_1^2 - L_2^2}{2L_1 L_2}}\right) \quad (3.8)$$

Para el ángulo θ_1 , en la figura 3.6 se puede observar que:

$$\phi = \operatorname{atan}\left(\frac{L_2 \sin(\theta_2)}{L_1 + L_2 \cos(\theta_2)}\right) \quad (3.9)$$

$$\alpha = \theta_1 + \phi = \operatorname{atan}\left(\frac{p_y}{p_x}\right) \quad (3.10)$$

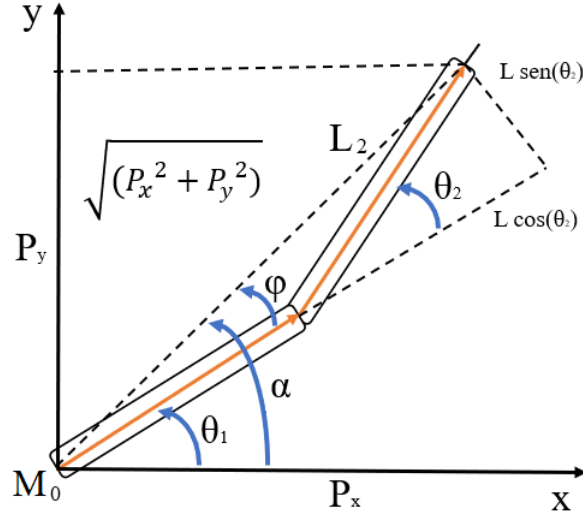


Figura 3.6: Esquema de un manipulador SCARA de 2 gdl para el ángulo θ_1 .

despejando en la ecuación 3.10:

$$\theta_1 = \alpha - \phi \quad (3.11)$$

sustituyendo 3.9 y 3.10 en 3.11:

$$\theta_1 = \text{artan} \left(\frac{p_y}{p_x} \right) - \text{artan} \left(\frac{L_2 \text{sen}(\theta_2)}{L_1 + L_2 \text{cos}(\theta_2)} \right) \quad (3.12)$$

3.2.3. Matriz Jacobiana.

Diferenciando las ecuaciones 3.1 y 3.2 podemos encontrar la relación entre las velocidades de las articulaciones y la velocidad de la herramienta, o lo que es lo mismo $\mathbf{J}(\boldsymbol{\theta})$. Aplicando la regla de la cadena, la derivación queda expresada como:

$$\dot{p}_x = \frac{\partial p_x}{\partial t} = \frac{\partial p_x}{\partial \theta_1} \frac{\partial \theta_1}{\partial t} + \frac{\partial p_x}{\partial \theta_2} \frac{\partial \theta_2}{\partial t} = \frac{\partial p_x}{\partial \theta_1} \dot{\theta}_1 + \frac{\partial p_x}{\partial \theta_2} \dot{\theta}_2 \quad (3.13)$$

$$\dot{p}_y = \frac{\partial p_y}{\partial t} = \frac{\partial p_y}{\partial \theta_1} \frac{\partial \theta_1}{\partial t} + \frac{\partial p_y}{\partial \theta_2} \frac{\partial \theta_2}{\partial t} = \frac{\partial p_y}{\partial \theta_1} \dot{\theta}_1 + \frac{\partial p_y}{\partial \theta_2} \dot{\theta}_2 \quad (3.14)$$

Calculando estas derivadas y expresándolas en forma matricial, se llega al modelo 3.15.

$$\mathbf{J}(\boldsymbol{\theta}) = \begin{bmatrix} -L_1 \text{sen}(\theta_1) - L_2 \text{sen}(\theta_1 + \theta_2) & -L_2 \text{sen}(\theta_1 + \theta_2) \\ L_1 \text{cos}(\theta_1) + L_2 \text{cos}(\theta_1 + \theta_2) & L_2 \text{cos}(\theta_1 + \theta_2) \end{bmatrix} \quad (3.15)$$

3.2.4. Modelo dinámico.

En [48, 50] se encuentra el procedimiento para plantear al modelo dinámico de un mecanismo de eslabones rígidos con dos articulaciones rotacionales. La energía cinética se define directamente considerando las componentes de la velocidad del centro de masa de cada eslabón (l_1, l_2) , el cual para mayor simplicidad se supone en el centro geométrico del eslabón. Y los momentos de inercia de los eslabones con respecto al eje que pasa por sus centros de masa y perpendicular al plano (I_1, I_2) . Primero se definen las coordenadas generalizadas de cada centro de masa, y luego por derivación directa se determinan las velocidades.

$$\dot{x}_1 = -l_1 \text{sen} \theta_1 \dot{\theta}_1 \quad (3.16)$$

$$\dot{y}_1 = l_1 \text{cos} \theta_1 \dot{\theta}_1 \quad (3.17)$$

$$\dot{x}_2 = -L_1 \text{sen} \theta_1 \dot{\theta}_1 - l_1 \text{sen} (\theta_1 + \theta_2) (\dot{\theta}_1 + \dot{\theta}_2) \quad (3.18)$$

$$\dot{y}_2 = L_1 \text{cos} \theta_1 \dot{\theta}_1 + l_2 \text{cos} (\theta_1 + \theta_2) (\dot{\theta}_1 + \dot{\theta}_2) \quad (3.19)$$

Las expresiones para el cálculo de la energía cinética son:

$$\begin{aligned} K_1(\dot{\theta}) &= \frac{1}{2} m_1 (\dot{x}_1^2 + \dot{y}_1^2) + \frac{1}{2} I_1 \dot{\theta}_1^2 \\ &= \frac{1}{2} m_1 l_1^2 \dot{\theta}_1^2 + \frac{1}{2} I_1 \dot{\theta}_1^2 \end{aligned} \quad (3.20)$$

$$\begin{aligned} K_2(\dot{\theta}) &= \frac{1}{2} m_2 (\dot{x}_2^2 + \dot{y}_2^2) \\ &= \frac{1}{2} m_2 (L_1^2 + l_2^2 + 2L_1 l_2 \text{cos} \theta_2) \dot{\theta}_1^2 \\ &\quad + \frac{1}{2} m_2 \left((2l_2^2 + 2L_1 l_2 \text{cos} \theta_2) \dot{\theta}_1 \dot{\theta}_2 + l_2^2 \dot{\theta}_2^2 \right) \\ &\quad + \frac{1}{2} I_1 (\dot{\theta}_1 + \dot{\theta}_2)^2 \end{aligned} \quad (3.21)$$

así la energía cinética total es $K(\dot{\theta}) = K_1(\dot{\theta}) + K_2(\dot{\theta})$ y la energía potencial es $P(\theta) = 0$, ya que el manipulador se desplaza solo en la horizontal.

El lagrangiano es $\mathcal{L}(\theta, \dot{\theta}) = K(\dot{\theta}) - P(\theta)$, el cual es derivado parcialmente respecto a $\theta_1, \theta_2, \dot{\theta}_1$ y $\dot{\theta}_2$, para encontrar las ecuaciones de movimiento del sistema. Las cuales son presentadas en 3.22 y 3.23.

$$\begin{aligned} \tau_1 &= (m_1 l_1^2 + m_2 L_1^2 + m_2 l_2^2 + 2m_2 L_1 l_2 \text{cos} \theta_2 + I_1 + I_2) \ddot{\theta}_1 \\ &\quad + (m_2 l_2^2 + m_2 L_1 l_2 \text{cos} \theta_2 + I_2) \ddot{\theta}_2 \\ &\quad - 2m_2 L_1 l_2 \text{sen} \theta_2 \dot{\theta}_1 \dot{\theta}_2 - m_2 L_1 l_2 \text{sen} \theta_2 \dot{\theta}_2^2 \end{aligned} \quad (3.22)$$

$$\begin{aligned}\tau_2 = & (m_2 l_2^2 + m_2 L_1 + I_2) \ddot{\theta}_1 \\ & + (m_2 l_2^2 + I_2) \ddot{\theta}_2 \\ & + m_2 L_1 l_2 \text{sen} \theta_2 \dot{\theta}_1^2\end{aligned}\quad (3.23)$$

Como el prototipo es actuado con motores de DC es razonable incluir la dinámica de los motores dentro del modelo dinámico del manipulador. El modelo dinámico de un motor de DC de imanes permanentes es presentado en [50], las ecuaciones que lo describen son:

$$\frac{di_a(t)}{dt} = \frac{1}{L_a} V_a(t) - \frac{R_a}{L_a} i_a(t) - \frac{K}{L_a} \dot{\theta}_m(t) \quad (3.24)$$

$$\frac{d\dot{\theta}_m(t)}{dt} = \frac{K}{J_m} i_a(t) - \frac{f_v}{J_m} \dot{\theta}_m(t) - \frac{1}{J_m} \tau_c(t) \quad (3.25)$$

Donde:

$i_a(t)$: Corriente eléctrica de entrada al motor.

$V_a(t)$: Tensión de entrada al motor.

$\dot{\theta}_m(t)$: Velocidad angular del eje del motor.

$\tau_c(t)$: Torque entregado a la carga.

L_a : Inductancia del motor.

R_a : Resistencia de la bobina del rotor.

K : $K = K_f = K_d$, constante proporcional de voltaje y de torque.

J_m : Momento de inercia del rotor.

f_v : Constante de fricción viscosa.

A manera de simplificación [61] plantea que es razonable ignorar la inercia, la fricción y la inducción del motor, de forma que el modelo empleado tiene la forma:

$$V_a(t) = \frac{R_a}{K} \tau_c(t) + K \dot{\theta}_m(t) \quad (3.26)$$

3.2.5. Obtención experimental de los parámetros del modelo.

Estructura mecánica: Los parámetros de la estructura física necesarios para el modelo son las masas de los eslabones, sus longitudes, así como la posición de los centros de masas. Las primeras se obtuvieron pesando los eslabones en una báscula digital con resolución de 1 g. Los parámetros dimensionales se obtuvieron empleando el *software* CAD que se empleo en el diseño. La figura 3.7 muestra las vistas superior y trasera del prototipo de pruebas donde se especifican las cotas necesarias. Los valores numéricos de los mismos son presentados en la tabla 3.2.5.

Parámetro		Valor	Unidad
Masa del primer eslabón.	m_1	0.07	Kg
Masa del segundo eslabón.	m_2	0.02	Kg
Longitud del primer eslabón.	L_1	0.075	m
Longitud del segundo eslabón.	L_2	0.085	m
Distancia al centro de masa del primer eslabón.	l_1	0.064	m
Distancia al centro de masa del segundo eslabón.	l_2	0.042	m

Tabla 3.1: Parámetros de la estructura mecánica del prototipo de pruebas.

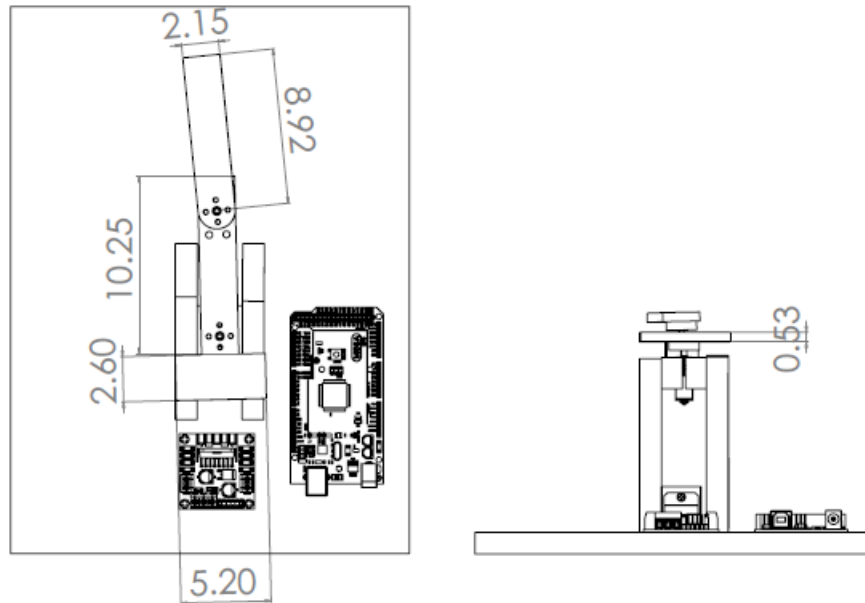


Figura 3.7: Vistas acotadas del prototipo de pruebas.

Motores de corriente directa: El motor que se empleara en el prototipo es el modelo 3076 de Polulu, cuyos datos de chapa son: $V_{in} = 6$ V, relación de transmisión de 150:1, velocidad en vacío 170 rpm, el mismo tiene acoplado un *encoder* por efecto Hall de 12 CPR.

1. Determinación de R_a .

Para determinar el valor de la resistencia de armadura (R_a) existen dos métodos: medirla de forma directa en los terminales del motor o realizar varias mediciones de tensión

de alimentación y corriente consumida, para luego calcular R_a empleando la Ley de Ohm. El método empleado sera el primero, para lo cual se realizaron mediciones en varios motores del mismo modelo y se promediaron, obteniendo un valor de $R_a = 3,2\Omega$.

2. Determinación de L_a .

La inductancia de armadura (L_a), al igual que la R_a , es medida de manera directa en el motor empleando un *LCR meter* (Keysight U1733C). El valor que arrojo el equipo de medición es $L_a = 3.32$ mH.

3. Determinación de las constantes proporcionales (K).

Para determinar el valor de la constante contraelectromotriz K_a se empleara la ecuación 3.27.

$$K_a = \frac{V_a - R_a i_a}{\omega} \quad (3.27)$$

Igual que anteriormente, para minimizar el error se realizaran seis mediciones variando el voltaje, los resultados se muestran en la tabla 3.

V_a (V)	i_a (mA)	ω (rad/seg)	K_a (Vseg/rad)
6.00	116.00	213.00	0.264
5.00	115.60	180.70	0.256
4.00	114.20	148.70	0.244
3.00	113.10	112.80	0.234
2.00	104.00	81.50	0.205
1.00	83.50	28.50	0.257

Tabla 3.2: Experimentos para el calculo de la constante contraelectromotriz (K_a).

Posteriormente se emplea la ecuación 3.27 y se promediaran los valores calculados. El resultado obtenido es $K_a = 0.0243$ Vseg/rad. Como se menciono anteriormente el valor numérico de esta constante corresponde con el de K_m , solo hay que sustituir las unidades de medida, así $K_m = 0.0243$ Nm/A.

4. Determinación de la fricción viscosa (f_v) y la fricción de Coulomb (f_c).

Para obtener los valores de la fricción viscosa se construyó la curva característica de tensión contra velocidad. Primero se tomaron valores de voltaje con un paso de 0.4 V, ya cercanos a cero se cambio el paso a 0.1 V. Como se puede observar en la figura 3.8 el comportamiento del motor es bastante lineal en todo el rango de trabajo. Utilizando

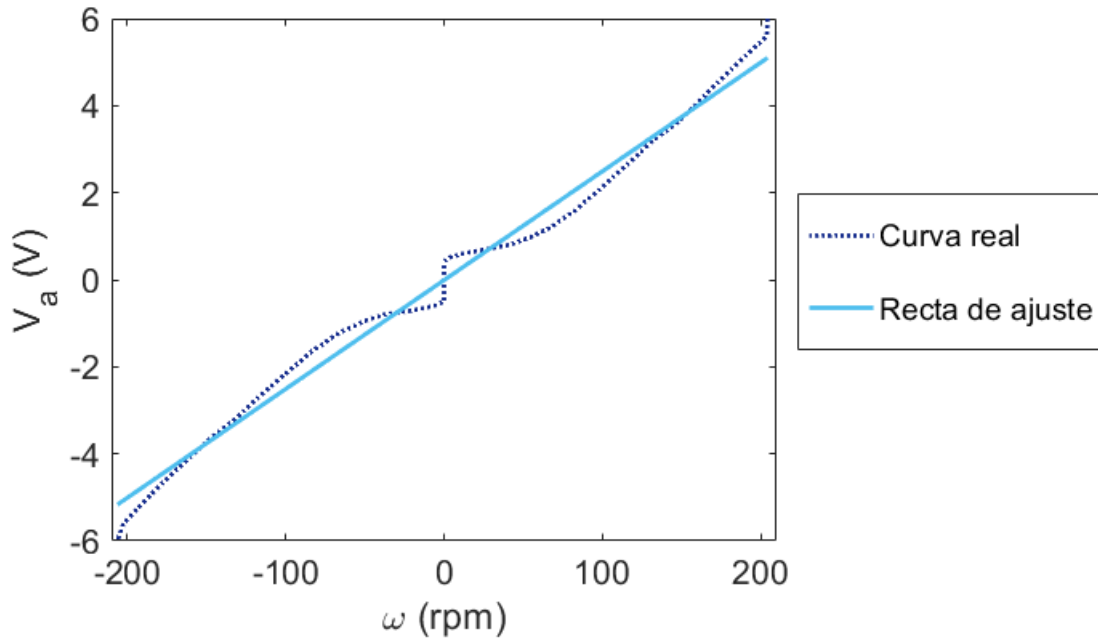


Figura 3.8: Gráfica característica de velocidad contra voltaje.

el método de regresión lineal, se ajusta una recta a los valores antes obtenidos. En ella el valor de la pendiente es la fricción viscosa y el intercepto en el eje del voltaje, es la fricción de Coulomb. Como se puede observar en la gráfica 3.8 los valores son $f_v = 0.0238 \text{ NmS}$ y $f_c = 0.4489 \text{ NmS}$.

5. Determinación del momento de inercia (J_m).

Para obtener el momento de inercia se parte de la ecuación:

$$J_m = \frac{-f_c \Delta t}{\ln \left(\frac{V_f - \frac{f_v}{f_c}}{V_i + \frac{f_v}{f_c}} \right)} \quad (3.28)$$

Se realizaron varios experimentos donde se aplicó un voltaje al motor, luego se llevó a cero y se gráfico su comportamiento. De estos experimentos se obtuvo el tiempo inicial y el tiempo final (Δt), así como la velocidad inicial y la velocidad final. Luego usando la ecuación 3.28 se calculó el valor del momento de inercia. En la tabla 5 se muestran los valores obtenidos para doce experimentos, seis para voltajes negativos y seis para

$\Delta t(mseg)$	$\omega_i(rad/seg)$	$\omega_f(rad/seg)$	$J_m(Kgm^2)$
300	215.40	21.70	0.0586
300	183.00	11.00	0.0478
300	157.30	16.00	0.0588
300	122.60	6.20	0.0449
250	94.20	11.60	0.0535
250	55.20	4.00	0.0425
250	-51.40	-2.80	0.0388
250	-91.60	-10.00	0.0508
300	-120.90	-5.00	0.0424
300	-149.30	-6.80	0.0437
300	-179.10	-8.70	0.0446
300	-204.60	-10.00	0.0447

Tabla 3.3: Experimentos para el calculo del momento de inercia (J_m).

voltajes positivos. Luego se promedian los experimentos y se obtiene el valor de $J_m = 0.0476 \text{ Kgm}^2$.

Sensor de fuerza FSR 400: Las curvas características que relacionan la variación de la resistencia con la fuerza aplicada al sensor son proporcionadas por el fabricante en la hoja de datos, se presenta en la figura 3.2.5. Aunque se cuenta con esta información, y con el objetivo de caracterizar el sensor empleado con una mayor exactitud, se realizaron una serie de experimentos. En estos se le aplico una fuerza determinada y normal a la superficie activa del sensor, mientras se alimentaba con una fuente de 5 V y una resistencia $R = 10K\Omega$ en el divisor de voltaje. Con los resultados obtenidos, tabla 3.2.5, se ajusto una curva polinómica de orden tres, ecuación 3.29, con un ajuste del 99.1%, que permitió establecer la relación fuerza/voltaje que se programo en el sistema.

$$F = 73 \times 10^{-3}V^3 - 293 \times 10^{-3}V^2 + 397 \times 10^{-3}V - 96 \times 10^{-3} \quad (3.29)$$

3.2.6. Validación del modelo dinámico del motor de DC.

Luego de obtener los parámetros del motor por separado, estos fueron sustituidos en las ecuaciones 3.24 y 3.25. Posteriormente se realizaron varias simulaciones cambiando el voltaje de entrada y graficando la velocidad calculada por el modelo y la velocidad real del motor. Se realizaron varios ajustes en los parámetros hasta obtener la mejor respuesta, la cual se presenta en la figura 3.10.

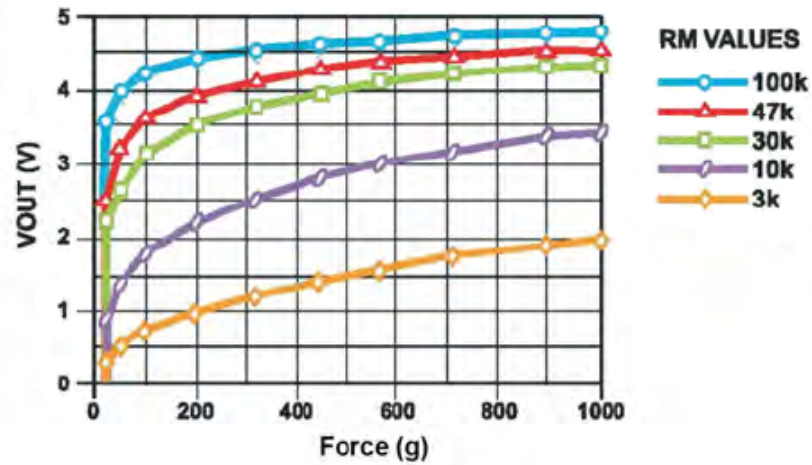


Figura 3.9: Datos de la relación fuerza/voltaje según hoja de datos.

Voltaje (V)	Fuerza (N)	Voltaje (V)	Fuerza (N)
1.01	0.047	2.69	0.302
1.16	0.064	2.89	0.333
1.33	0.071	2.92	0.387
1.51	0.087	3.08	0.461
1.80	0.108	3.17	0.541
1.91	0.127	3.30	0.629
2.14	0.146	3.37	0.726
2.29	0.195	3.50	0.786
2.50	0.227	3.60	0.986
2.63	0.274	3.70	1.129

Tabla 3.4: Experimentos para la caracterización de la fuerza.

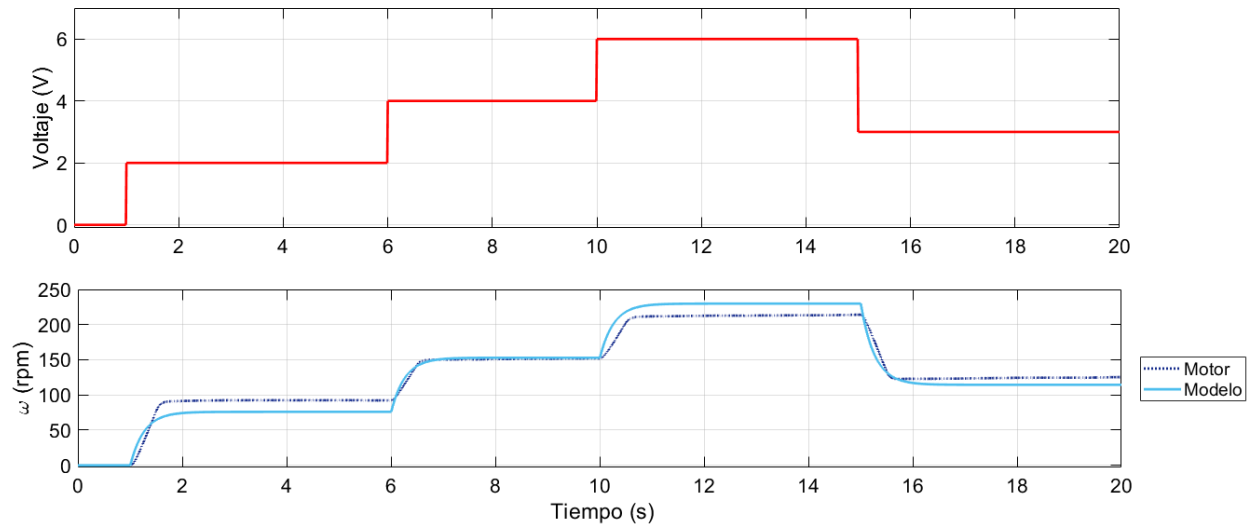


Figura 3.10: Comparación del comportamiento del modelo simulado y el motor real.

La respuesta de la figura 3.10 se obtiene con los parámetros mostrados en la tabla 3.2.6. Para validar la respuesta del modelo se calculo el error cuadrático medio normalizado, tomando como referencia los valores reales. El resultado obtenido, $ECMN = 0.95$, es indicativo de un buen ajuste del modelo.

Parámetros		Valor	Unidad
Resistencia de la bobina del rotor.	R_a	3,2000	Ω
Inductancia del motor.	L_a	3,3200	mH
Momento de inercia del rotor.	J_m	0,0700	Kgm^2
Constante de fricción viscosa.	f_v	0,0238	Nms
Constante de fricción de Coulomb.	C	0,2600	Nms
Constante proporcional de voltaje.	K_d	0,0243	$V/rad/s$
Constante proporcional de torque.	K_f	0,0243	Nm/A

Tabla 3.5: Parámetros ajustados del modelo del motor de DC.

3.3. Diseño del controlador interno de posición.

Como se vio anteriormente, el diseño del controlador interno debe garantizar el correcto seguimiento de la referencia. Para la señal de prueba, las trayectorias más simples que pueden ejecutarse por el prototipo es una línea o un círculo. La trayectoria seleccionada es un círculo

de $r=0.05$ m y centro en $C(0.01,0.05)$, la cual esta descrita por las ecuaciones que se presentan en 3.30.

$$\begin{aligned} p_x(t) &= r \cos(2\pi ft) + C_x \\ p_y(t) &= r \sin(2\pi ft) + C_y \end{aligned} \quad (3.30)$$

El controlador empleado para garantizar el seguimiento de la trayectoria es un Controlador Proporcional con Realimentación de Velocidad, cuya estructura se presento en la sección 2.3.1. El diagrama en bloques del sistema realimentado es mostrado en la figura 3.11.

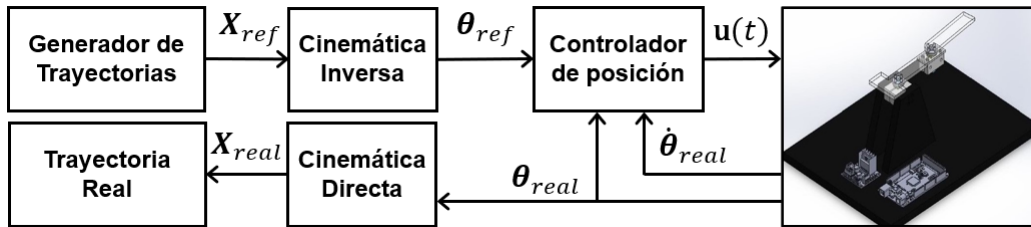


Figura 3.11: Diagrama en bloques del sistema realimentado.

Para realizar el ajuste de las ganancias se realizaron simulaciones en Simulink empleando el modelo dinámico del robot formulado en la sección 3.2.4. Las ganancias fueron obtenidas con la herramienta *PID Tuner*, ya que permite ajustar la salida del sistema realimentado de una manera visual manteniendo una relación entre desempeño y rapidez. El algoritmo elige una frecuencia de cruce basada en la dinámica de la planta, y ajusta las ganancias para un margen de fase por defecto de $MF = 60^\circ$. Luego las varia en relación con la respuesta a lazo cerrado seleccionada para el sistema. Los valores obtenidos mediante este procedimiento son $K_p = 20.4$ y $K_v = 0.42$, el correcto seguimiento de la trayectoria se presenta en la figura 3.12, donde se muestra los valores de las coordenadas articulares para la trayectoria circular. La acción de control es presentada en la figura 3.13.

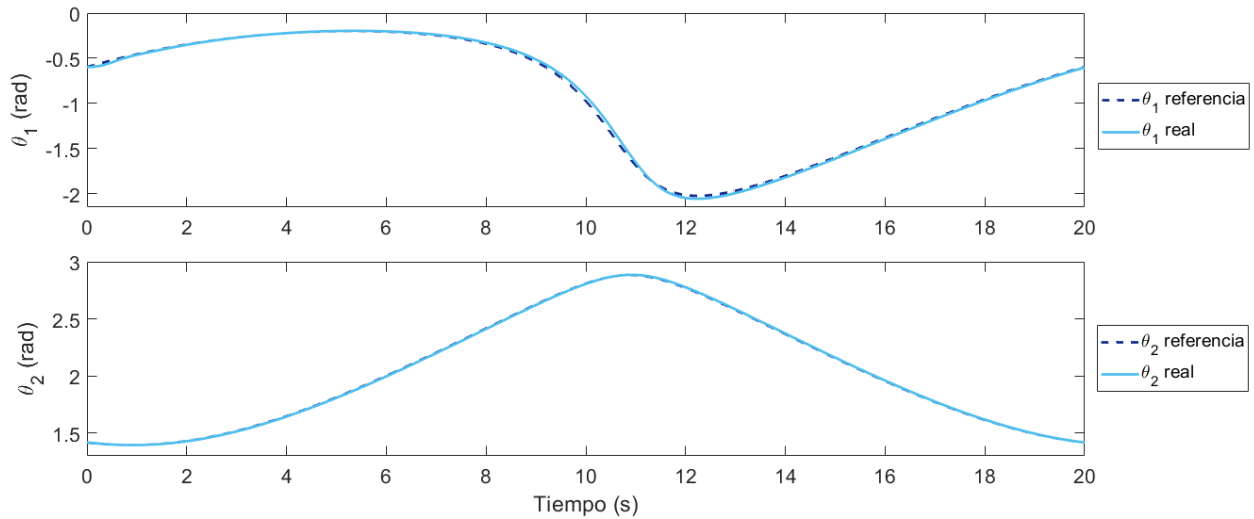


Figura 3.12: Coordenadas articulares para trayectoria circular en simulación.

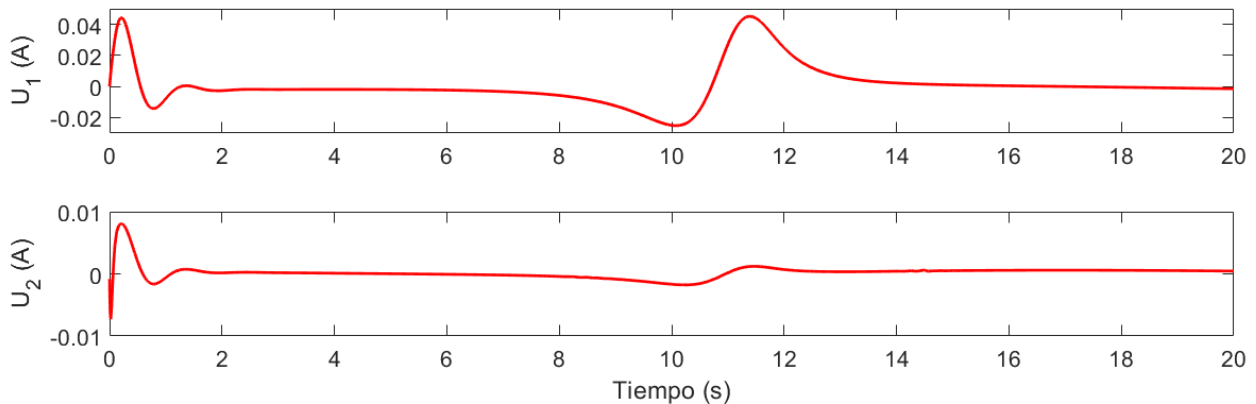


Figura 3.13: Señal de control para una trayectoria circular.

Después de comprobar el ajuste de las ganancias del controlador mediante simulación, se realizaron experimentos que permitieron comprobar el seguimiento de las coordenadas articulares del prototipo. Se realizaron pequeños ajustes en las ganancias siguiendo reglas empíricas, obteniéndose los resultados mostrados en la figura 3.14 para una $K_p = 25$ y una $K_v = 0.2$. En ella la línea discontinua representa la posición de referencia y la línea continua la posición real medida por los *encoder*.

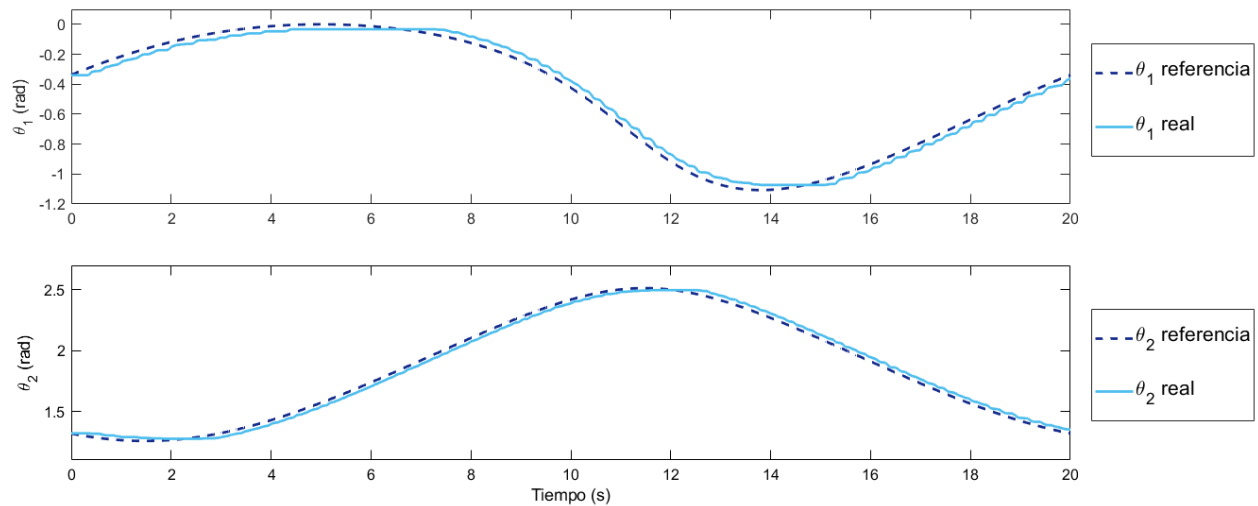


Figura 3.14: Coordenadas articulares para trayectoria circular en experimento real.

En la figura 3.15 se muestra el error de posicionamiento para ambas articulaciones, comprobándose que se mantiene en un rango de $-0,05 \leq \tilde{\theta} \leq 0,05$. Con el objetivo de confirmar el buen seguimiento de la referencia en el sistema realimentado se calculó el error cuadrático medio de ambas coordenadas articulares, $ECM_{\theta_1} = 13 \times 10^{-4}$ rad y $ECM_{\theta_2} = 8,9 \times 10^{-4}$ rad.

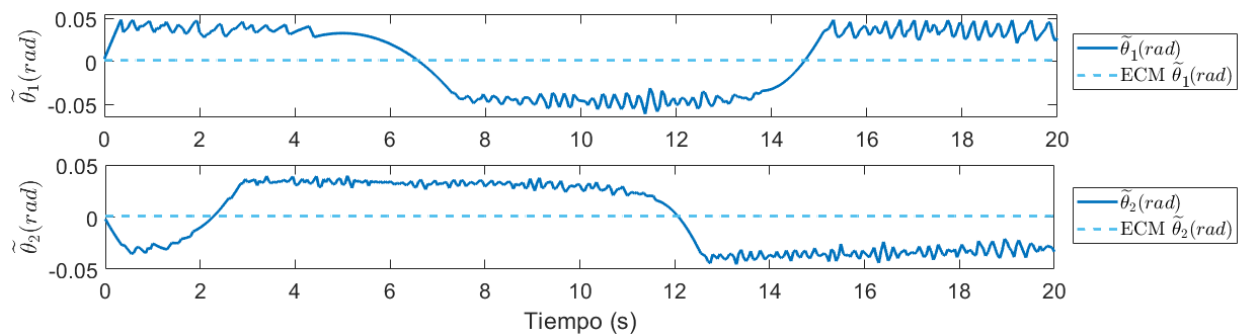


Figura 3.15: Error de seguimiento de las coordenadas articulares.

El seguimiento de la trayectoria en coordenadas cartesianas se muestra en la figura 3.16. Calculando el error cuadrático medio para esta trayectoria, $ECM_X = 9,8 \times 10^{-4}$ m y $ECM_Y = 14,1 \times 10^{-4}$ m, se demuestra un comportamiento aceptable. En [43] se plantea que para las técnicas de programación por guiado manual es aceptable un error en el rango de unos milímetros.

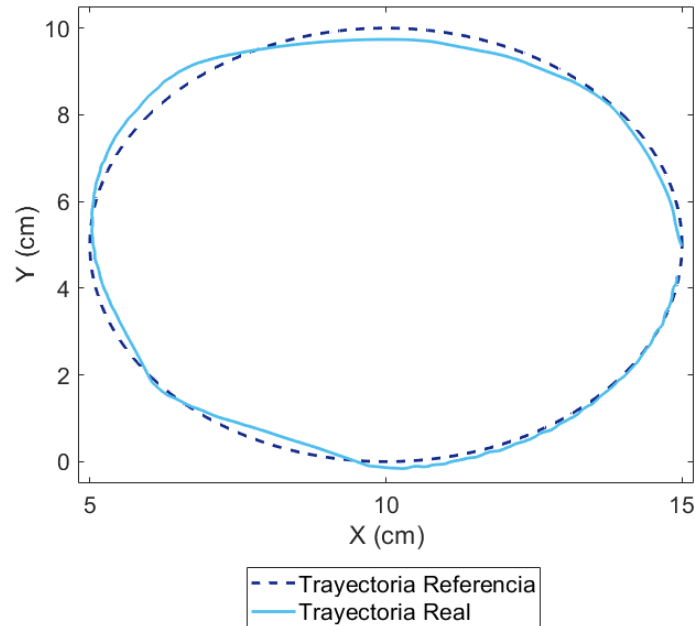


Figura 3.16: Trayectoria circular realizada en experimento real.

3.4. Diseño del DOB para estimar la fuerza de guiado.

En el diseño del Observador de Perturbaciones están involucrados los parámetros que caracterizan los motores de DC que actúan el prototipo, los cuales fueron presentados en la Tabla 3.2.6, y el valor de la frecuencia de corte del filtro pasa bajos (g). Para realizar la selección de este parámetro se realizaron simulaciones en las cuales se incorporó la presencia de una fuerza externa durante el seguimiento de la trayectoria circular. El valor de g se incrementó hasta obtener el valor más grande posible, en la figura 3.17 se muestran la fuerza externa y la fuerza estimada con $g = 30$. La respuesta obtenida muestra un correcto funcionamiento de la respuesta estacionaria del DOB, en contraste con un retardo que aparece en la respuesta transitoria.

En la figura 3.18 se presenta el comportamiento de las coordenadas articulares durante la simulación. Se puede observar en el intervalo de 0.5 a 12 s, en el cual está presente la fuerza externa, como el controlador es capaz de alcanzar nuevamente la referencia. Esta es una de las características que se incorporan con la inclusión del DOB dentro de la estructura de control, un mejor rechazo a las perturbaciones.

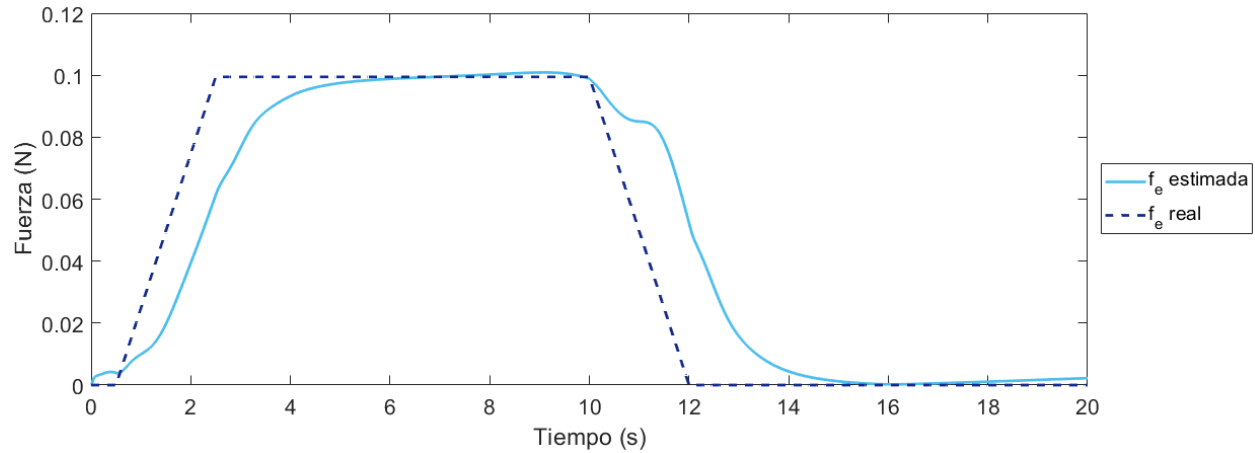


Figura 3.17: Simulación de la respuesta del DOB para $g = 30$.

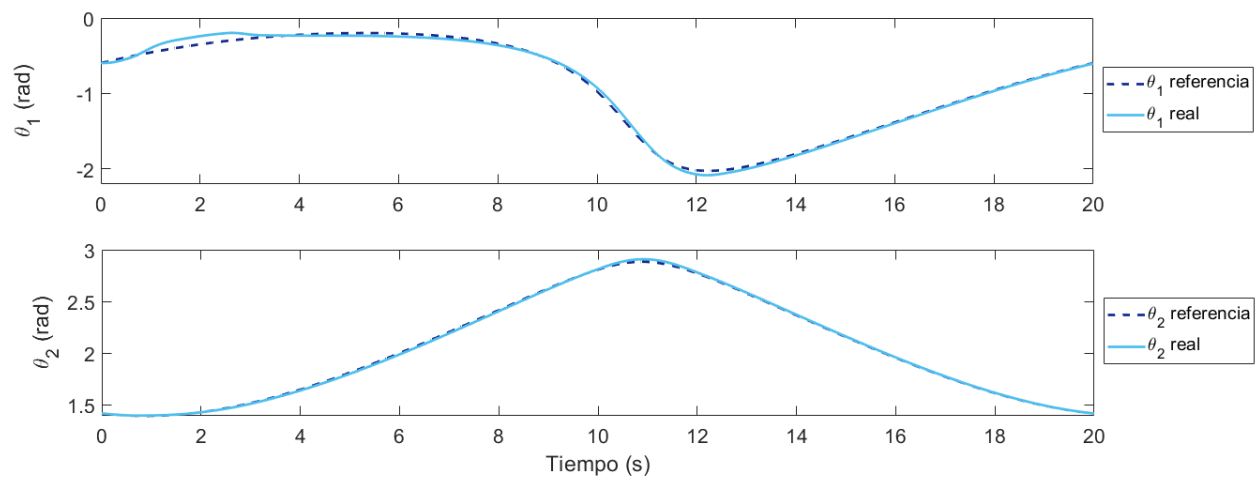


Figura 3.18: Coordenadas articulares ante la presencia de una f_{ext} en simulación.

Para validar el buen funcionamiento del Observador de Perturbaciones se realizaron experimentos con el objetivo de comparar la fuerza estimada con la fuerza medida por el sensor FSR 400. Los experimentos se realizaron considerando que el sensor empleado es uniaxial, por lo cual la f_{ext} se aplicó perpendicular a la superficie activa del mismo como se muestra en la figura 3.19.

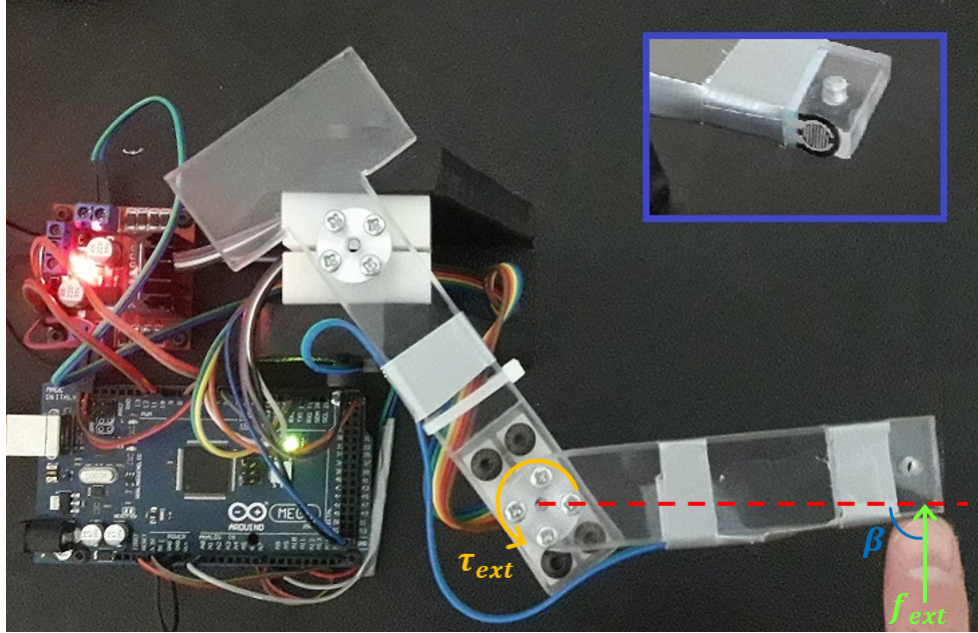


Figura 3.19: Experimento para la validación del DOB.

En un inicio se considero un mecanismo de 1 dof, para lo cual se empleo la segunda articulación. Es conocido que el torque generado por una fuerza aplicada en una palanca es igual al producto de esta ultima por brazo de palanca, lo que se formaliza en la ecuación 3.32.

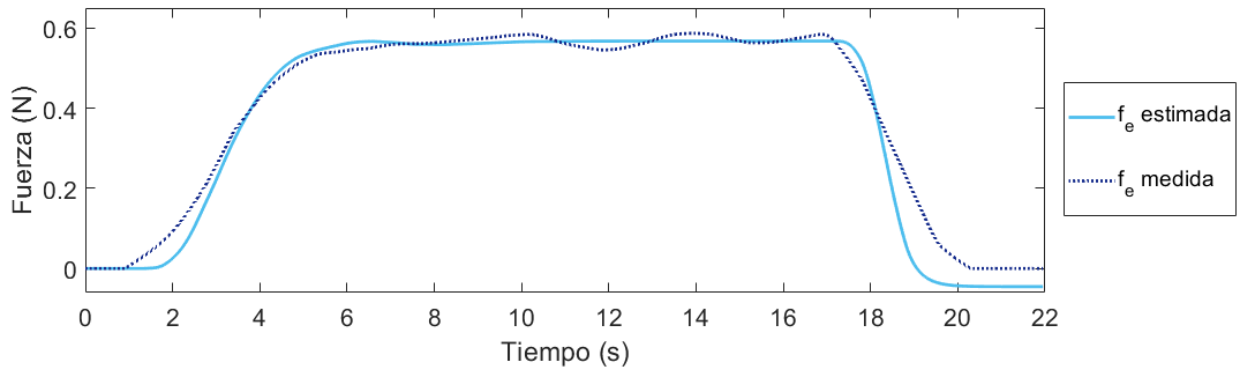
$$\tau = f_{ext}L_p \text{sen}\beta \quad (3.31)$$

Donde: f_{ext} es la fuerza externa aplicada a la palanca, L_p es la longitud de la palanca, y β es el ángulo entre la fuerza y la palanca.

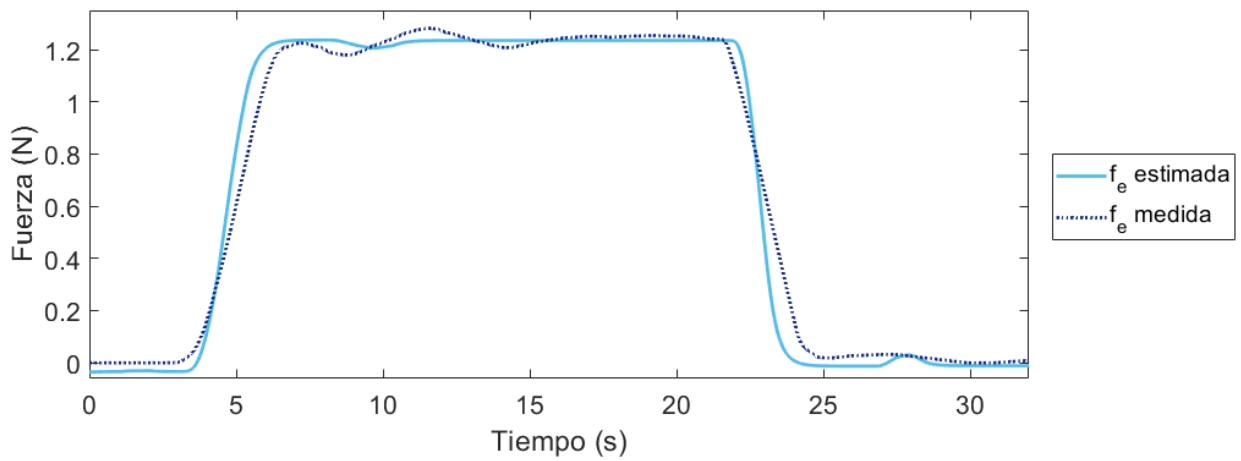
Considerando el supuesto de perpendicularidad a la hora de aplicar la fuerza, el ángulo $\beta = 90^\circ$ y $\text{sen}\beta = 1$, por lo cual la fuerza estimada se puede calcular como:

$$\hat{f}_{ext} = \frac{\hat{\tau}_{ext}}{L_p} \quad (3.32)$$

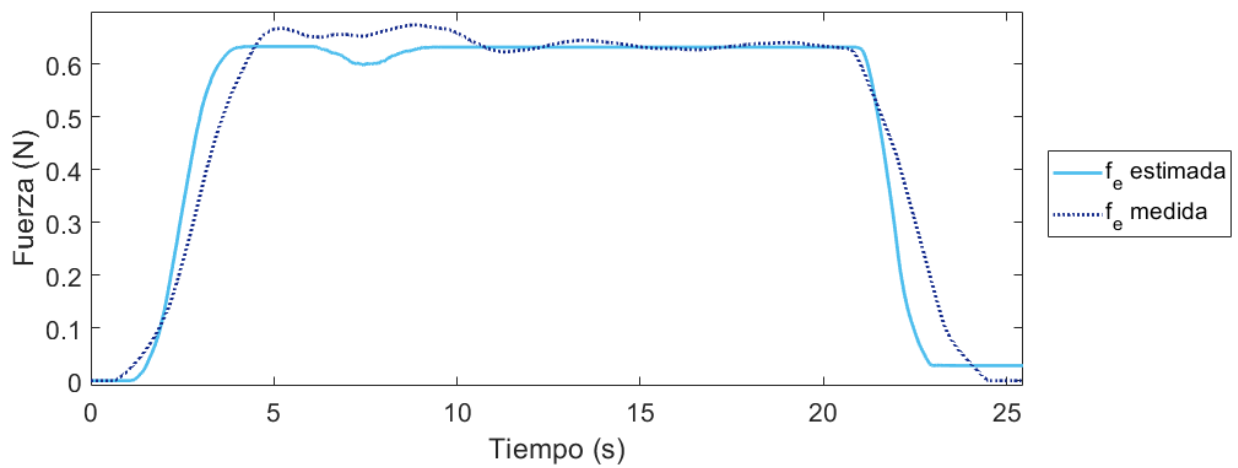
La figura 3.4 muestra la fuerza medida y la fuerza estimada obtenidas en tres experimento. Para validar el correcto funcionamiento del DOB se calcula el error cuadrático medio normalizado de las estimaciones, $ECMN_1 = 0,98$, $ECMN_2 = 0,99$ y $ECMN_3 = 0,96$. Los valores obtenidos al ser muy cercanos a la unidad demuestran que el funcionamiento del observador es correcto. Se debe resaltar que en estos experimentos se logro mejorar la respuesta transitoria, sin embargo, en todos los experimentos realizados se presenta un retardo ($\Delta t = 0,2s$) entre la medición del sensor y la fuerza estimada por el DOB.



(a) Experimento 1.



(b) Experimento 2.



(c) Experimento 3.

Figura 3.20: Comparación entre la f_e medida y la estimada en el mecanismo de 1 dof.

Para extender los resultados al manipulador SCARA de 2 DoF, la fuerza externa estimada debe calcularse empleando la ecuación 2.3, de esta forma se obtienen las componentes del vector de fuerza estimado. Como el sensor no permite una medición de las componentes en cada eje, se empleó el operador norma para calcular la magnitud de la fuerza estimada en el extremo del manipulador. Los resultados de la comparación entre la fuerza medida y estimada son mostrados en la figura 3.21. Como se aprecia, el valor de la fuerza estimada corresponde con valor de la media aritmética de la fuerza medida. Se tomó como referencia este estadígrafo ya que la alta sensibilidad del sensor provoca variaciones en la medición.

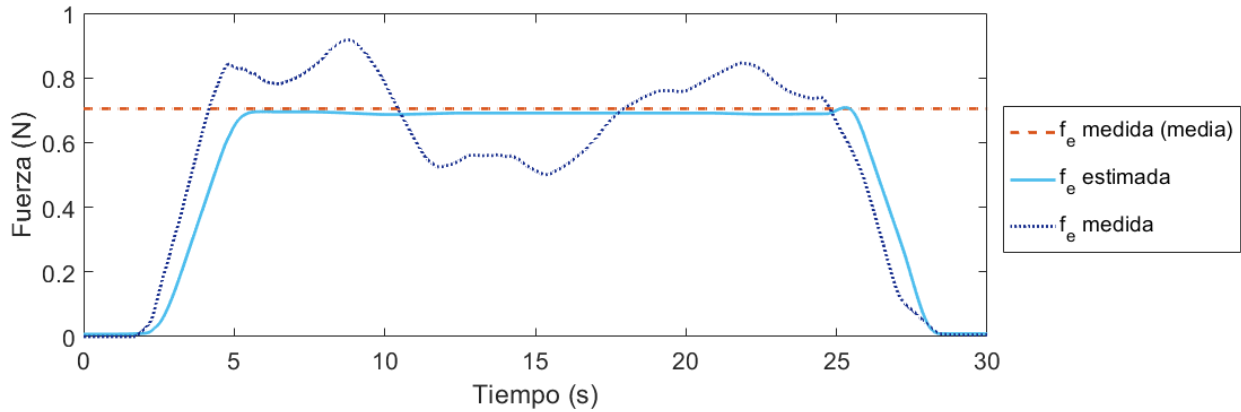


Figura 3.21: Comparación entre la f_e medida y la estimada en el robot de 2 dof.

3.5. Diseño del controlador de admitancia para el guiado manual.

El diseño del controlador de admitancia comprende el correcto ajuste de las matrices \mathbf{M}_d , \mathbf{D}_d y \mathbf{K}_d . Para el ajuste de estos parámetros se realizaron varias simulaciones siguiendo el esquema en bloques presentado en la figura 2.3, y se siguieron las recomendaciones descritas en [40], para conseguir el mejor ajuste. Los resultados presentados en la figura se alcanzaron con:

$$\mathbf{M}_d = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$$

$$\mathbf{D}_d = \begin{bmatrix} 4,5 & 0 \\ 0 & 4,5 \end{bmatrix}$$

$$\mathbf{K}_d = \begin{bmatrix} 70 & 0 \\ 0 & 70 \end{bmatrix}$$

3.5. DISEÑO DEL CONTROLADOR DE ADMITANCIA PARA EL GUIADO MANUAL.44

La figura 3.22 muestra el comportamiento de la fuerza real y la fuerza estimada durante la simulación. Como se observa la respuesta del DOB presenta características iguales a las ya descritas anteriormente.

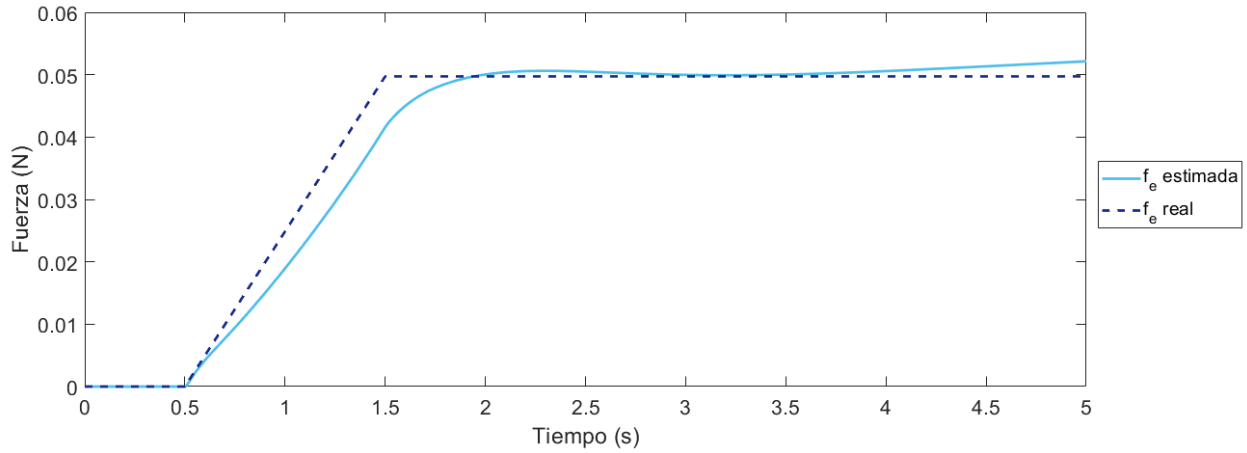


Figura 3.22: Comparación entre la f_e real y la estimada. Simulación de guiado.

Para mostrar el correcto funcionamiento del controlador de admitancia se presentara la trayectoria obtenida cuando se aplica la fuerza el extremo del manipulador , como queda definido en la figura 3.23. Esto corresponde a una componente de fuerza positiva en el eje Y y otra componente negativa y más pequeña en magnitud en el eje X.

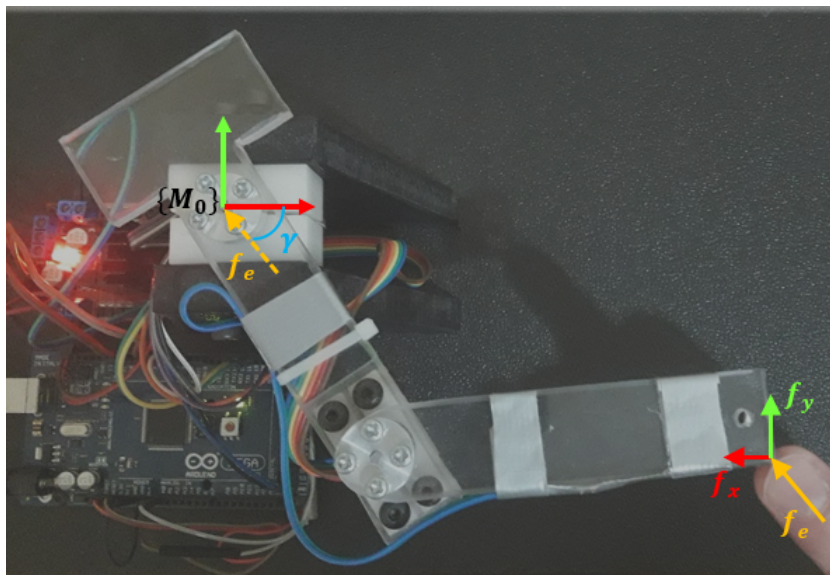
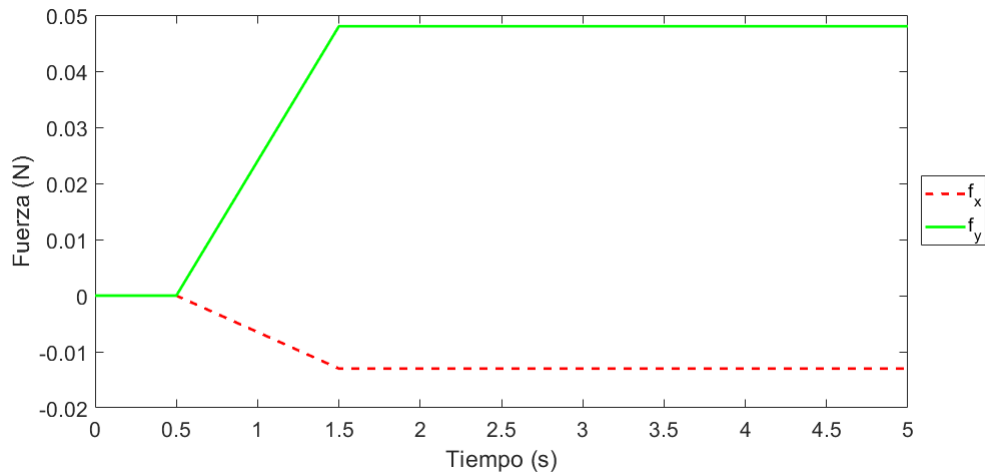


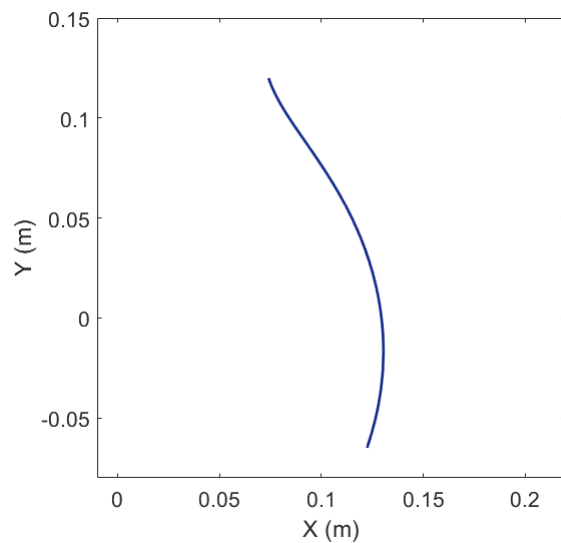
Figura 3.23: Definición del ángulo de aplicación de la f_e .

3.5. DISEÑO DEL CONTROLADOR DE ADMITANCIA PARA EL GUIADO MANUAL.45

Los resultados son mostrados en la figura 3.5, donde se aprecia que la herramienta sigue una trayectoria cercana a una línea recta vertical. No se alcanza la línea recta ya que el ángulo de aplicación de la fuerza permanece fijo durante todo el experimento.



(a) Componentes de f_e .



(b) Trayectoria vertical.

Figura 3.24: Simulación de guiado aplicando una f_e en un ángulo $\gamma = 80^\circ$.

Para experimentar con el controlador de admitancia en el prototipo real se comenzara haciendo uso del mecanismo de 1 dof y realimentando la señal de sensor. Estas condiciones

3.5. DISEÑO DEL CONTROLADOR DE ADMITANCIA PARA EL GUIADO MANUAL.46

dan como resultado un problema más sencillo donde no interfiere la configuración geométrica del robot ni errores de estimación. En la figura 3.25 se muestra la fuerza medida y la fuerza estimada durante el experimento.

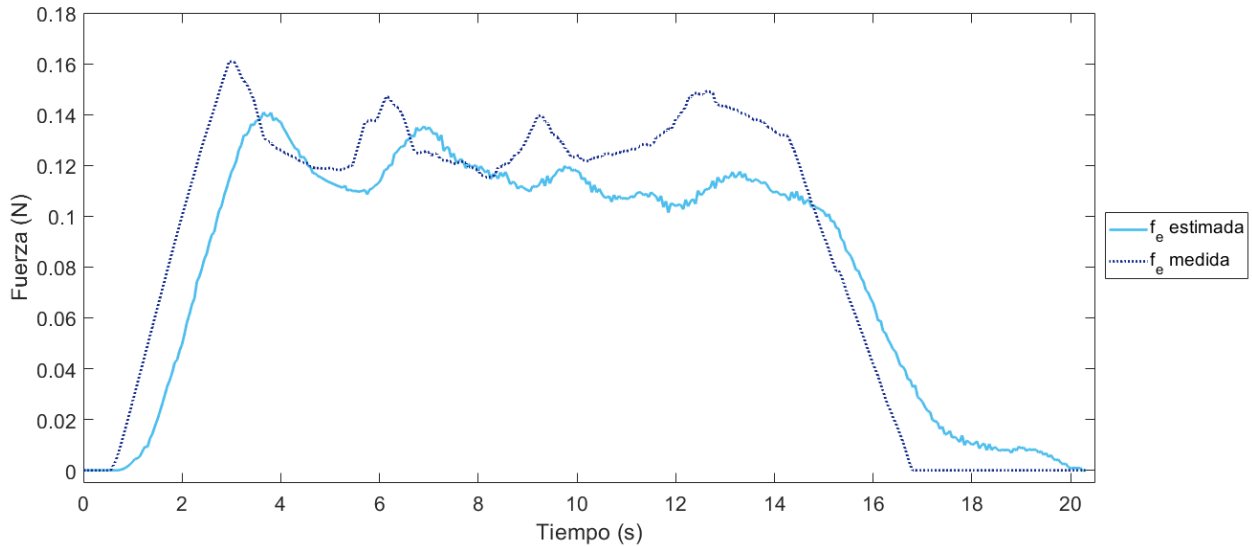


Figura 3.25: Fuerza medida y estimada. Experimento realimentando fuerza medida.

Analizando la curva podemos encontrar un primer pico el cual es el valor máximo alcanzado durante todo el experimento, esto se corresponde con la fuerza necesaria para vencer la inercia del mecanismo y comenzar el movimiento. Los siguientes picos se corresponden con los momentos en los cuales la señal de control supera el voltaje de barrera y el motor comienza a colaborar con el movimiento, lo cual se muestra en la figura 3.26. Esto provoca que la fuerza necesaria para mover el brazo disminuya, lo cual a su vez causa que disminuya la señal de control. El pequeño valor de momento de inercia del brazo ($I = 50,7 \times 10^{-6} \text{ Kg m}^2$), hace que se trabaje en las fronteras del voltaje de barrera, siendo muy difícil conseguir un equilibrio durante el guiado.

En la figura 3.27 se muestra la salida del controlador de admitancia, la cual se interpreta como un delta de posición en coordenadas cartesianas de la herramienta. Este error, sumado a la posición real, resultan en la posición de referencia que debe alcanzar la herramienta según la fuerza con la cual el humano la mueve. Este valor se traduce en un error en el valor del ángulo de posicionamiento, diferencia que se puede observar en la figura 3.28.

3.5. DISEÑO DEL CONTROLADOR DE ADMITANCIA PARA EL GUIADO MANUAL.47

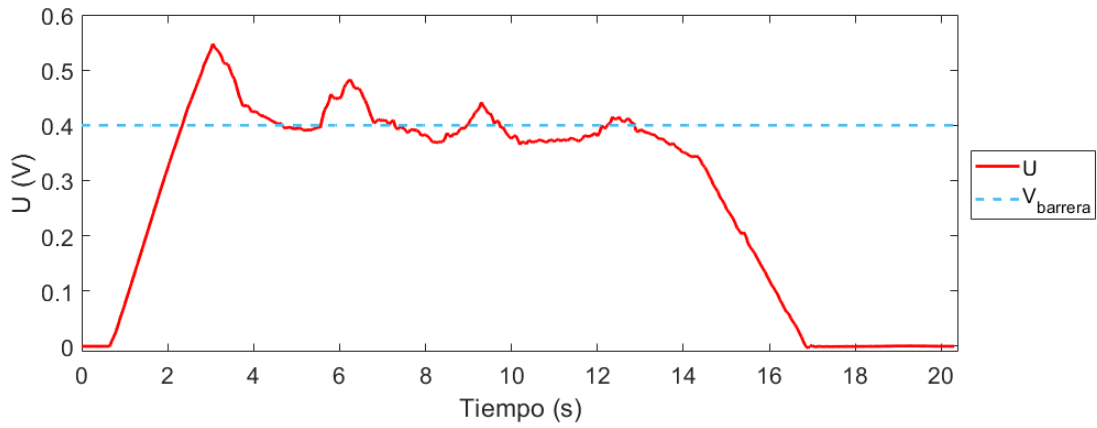


Figura 3.26: Señal de salida del control de posición.

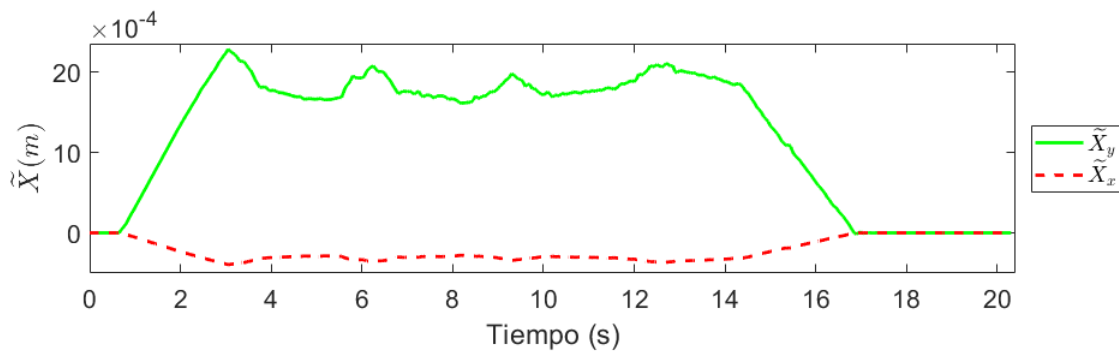


Figura 3.27: Señal de salida del controlador de admitancia.

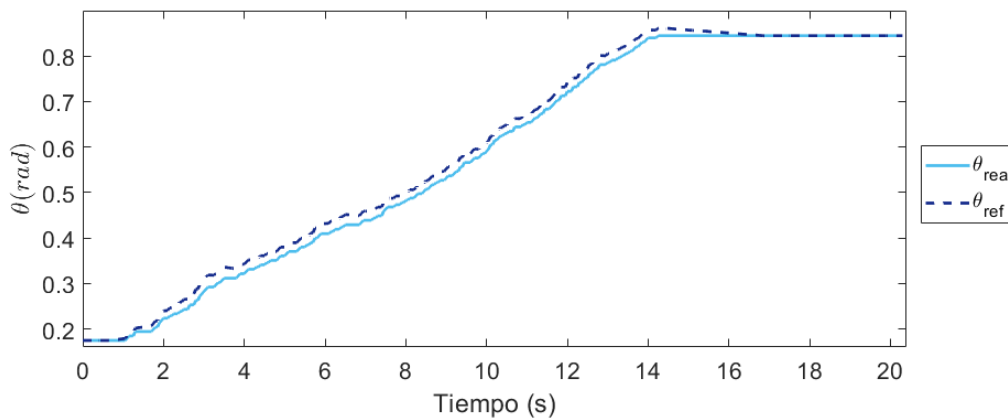


Figura 3.28: Ángulo de referencia y real durante el experimento en 1 dof.

3.5. DISEÑO DEL CONTROLADOR DE ADMITANCIA PARA EL GUIADO MANUAL.48

Una vez validado el funcionamiento del Observador de Perturbaciones y del Controlador de Admitancia por separado, se integran para comprobar su funcionamiento conjunto. Para esto se realimentó la señal de fuerza estimada por el DOB, y se realizaron nuevamente los experimentos. Los resultados obtenidos son presentados en la figura 3.29, los valores de la fuerza estimada no se corresponden con la fuerza medida. Tras realizar variaciones en los ajustes sin lograr resultados significativos, se concluye que el retardo de tiempo presente en la señal de fuerza estimada está afectando el desempeño global del sistema. La naturaleza de este retardo está relacionada con los métodos numéricos empleados en la programación del sistema y la capacidad de cálculo del controlador del prototipo.

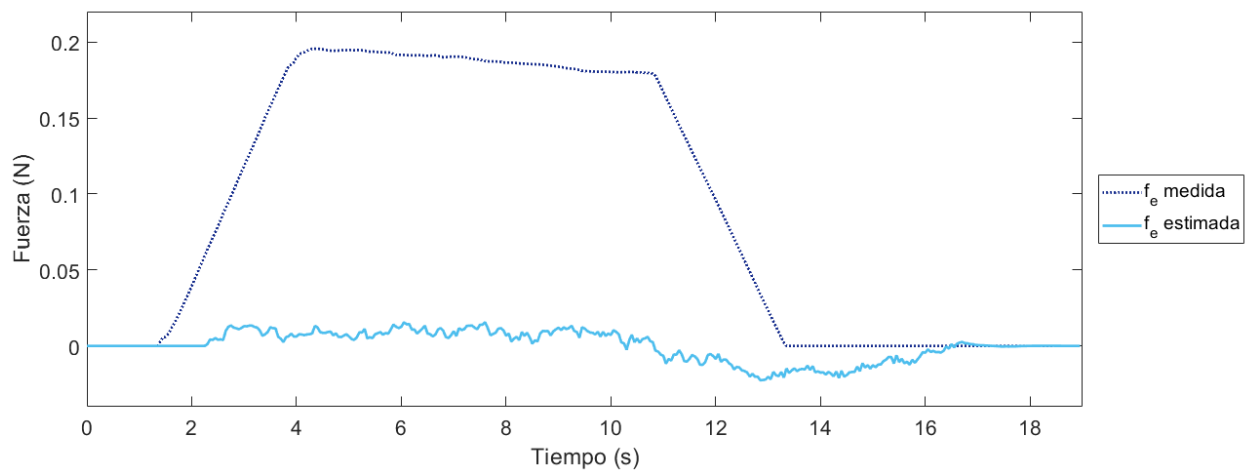


Figura 3.29: Fuerza medida y estimada. Experimento realimentando fuerza estimada.

Capítulo 4

Conclusiones

La dinámica del mercado actual reta a las empresas productoras a implementar celdas de producción flexibles, cuya reconfiguración pueda desarrollarse en cortos periodos de tiempo y a un bajo costo. Los métodos de programación de los robots tradicionales requieren de conocimientos especializados y de grandes tiempos de desarrollo y depuración para la configuración de una nueva tarea. Por esta razón en los últimos años se han desarrollado métodos de programación más intuitivos, que requieren de menos conocimientos especializados y de menos tiempos de desarrollo. Las aplicaciones que se han llevado a la industria se encuentran en el marco de los brazos robóticos colaborativos. Estos poseen la ventaja de contar con sensores de torque/par en sus articulaciones lo que los hace candidatos para la implementación de estrategias de enseñanza por guiado manual. Sin embargo, presentan la desventaja de estar limitados en cuanto a la carga que pueden manejar, reduciendo las aplicaciones en las cuales se puede emplear a nivel industrial. Por esta razón en la presente investigación se desarrolló una estructura de control que permite implementar el control de un prototipo robótico mientras un operador humano lo guía durante la fase de enseñanza. La estructura de control busca incluir un Observador de Perturbaciones para estimar la fuerza aplicada por el operador durante el guiado, y así hacer extensible este método de programación a los robots tradicionales.

Se diseñó el prototipo de un robot SCARA de dos grados de libertad, el cual fue fabricado en materiales de plástico. Se emplearon como actuadores motores de DC a los cuales se le acoplaron *encoders* para conocer la posición articular del robot. La tarjeta que se empleó para controlar el prototipo fue una Arduino Mega 2560 y la fase de potencia para manejar los motores se implementó con un L298. El *software* empleado para la programación del sistema es el paquete Simulink, perteneciente a MATLAB 2018a. Para la comunicación con el prototipo se emplearon las cajas de herramientas: *Simulink Support Package for Arduino Hardware* y el *Rensselaer Arduino Support Package Library (RASPLib)*. Estas cuentan con los bloques necesarios para leer la señal analógica del sensor de fuerza y las señales de pulsos provenientes de los *encoders*, así como los bloques para enviar las señales de PWM para

controlar los motores. El diseño del prototipo presenta ciertas desventajas que fueron apareciendo durante el desarrollo del proyecto: las reducidas dimensiones de los eslabones hacen que el área de trabajo del robot sea pequeña; a lo que se une que los materiales empleados en su fabricación son ligeros, lo que provoca que la fuerza requerida para moverlos sea pequeña. Estas limitantes hacen que se trabaje con valores cercanos a cero, o cercanos a los límites de voltaje de barrera para los motores, provocando que los resultados obtenidos sean más sensibles a ruidos externos. La capacidad de cálculo de la tarjeta controladora empleada fue la mayor desventaja, ya que comprometió el buen funcionamiento del programa, introduciendo retardos de tiempos que conllevaron al cumplimiento parcial del objetivo general del proyecto.

Se realizaron experimentos que permitieron parametrizar los modelos cinemáticos y dinámico del prototipo de pruebas fabricado. Empleando estos modelos se diseñó un Controlador Proporcional con Realimentación de Velocidad que permitió al prototipo ejecutar varias trayectorias con un error de posicionamiento del orden de un milímetro. Posteriormente al prototipo se le agregó un sensor de fuerza en el extremo del segundo eslabón, el objetivo de esta instrumentación es conseguir la medición de fuerza durante los experimentos para la validación del Observador de Perturbaciones diseñado. Los resultados obtenidos durante los experimentos de validación del DOB para un mecanismo de 1 DoF y para el prototipo de pruebas fueron satisfactorios, presentando un porcentaje de ajuste promedio de 97.5 %.

Para conseguir que el robot pueda ser guiado por un operador humano se diseñó un controlador de admitancia, demostrándose el correcto funcionamiento de la estructura de control propuesta en simulación y en experimentos realizados con el prototipo. Se encontraron dificultades para validar la estructura de control propuesta durante los experimentos realizados con el prototipo realimentando la señal de fuerza estimada por el DOB. Como se mencionó anteriormente, este resultado negativo se debe a la capacidad de cálculo de la controladora empleada y al costo computacional del programa realizado. Esto hace que se introduzca un retardo de cuatro periodos de muestreo en la señal estimada, error que se arrastra durante todo el experimento. Como trabajo futuro se propone la sustitución de la tarjeta controladora por otra de mayor capacidad de cálculo.

Bibliografía

- [1] D. Massa, M. Callegari, and C. Cristalli, “Manual guidance for industrial robot programming,” *Industrial Robot: An International Journal*, 2015.
- [2] J. Krüger, L. Wang, A. Verl, T. Bauernhansl, E. Carpanzano, S. Makris, J. Fleischer, G. Reinhart, J. Franke, and S. Pellegrinelli, “Innovative control of assembly systems and lines,” *Manufacturing Technology*, vol. 66, pp. 707–730, 2017.
- [3] T. B. Losada, “La Industria 4.0: Aplicaciones e Implicaciones,” Ph.D. dissertation, España, 2017.
- [4] A. C. Pereira and F. Romero, “A review of the meanings and the implications of the Industry 4.0 concept,” *Procedia Manufacturing*, vol. 13, pp. 1206–1214, 2017.
- [5] V. Cabezudo, “Robots colaborativos para encontrar una oportunidad en la Industria 4.0.” 2018. [Online]. Available: <https://www.muycanal.com/2018/09/11/robots-colaborativos-cobot-automatizacion>
- [6] V. Villani, F. Pini, F. Leali, and C. Secchi, “Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications,” *Mechatronics*, vol. 55, pp. 248–266, 2018.
- [7] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, and P. Fraisse, “Collaborative manufacturing with physical human–robot interaction,” *Robotics and Computer-Integrated Manufacturing*, vol. 40, pp. 1–13, 2016.
- [8] Z. Pan, J. Polden, N. Larkin, S. Van Duin, and J. Norrish, “Recent progress on programming methods for industrial robots,” in *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*. VDE, 2010, pp. 1–8.
- [9] F. Ferraguti, C. T. Landi, C. Secchi, C. Fantuzzi, M. Nolli, and M. Pesamosca, “Walk-through programming for industrial applications,” *Procedia Manufacturing*, vol. 11, pp. 31–38, 2017.

- [10] A. De Santis, V. Lippiello, B. Siciliano, and L. Villani, “Human-robot interaction control using force and vision,” in *Advances in Control Theory and Applications*. Springer, 2007, pp. 51–70.
- [11] M. A. Goodrich, A. C. Schultz *et al.*, “Human–robot interaction: a survey,” *Foundations and Trends® in Human–Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2008.
- [12] G. Michalos, S. Makris, P. Tsarouchi, T. Guasch, D. Kontovrakis, and G. Chryssolouris, “Design considerations for safe human-robot collaborative workplaces,” *Procedia CIRP*, vol. 37, pp. 248–253, 2015.
- [13] A. De Luca and F. Flacco, “Integrated control for phri: Collision avoidance, detection, reaction and collaboration,” in *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. IEEE, 2012, pp. 288–295.
- [14] M. Peshkin, J. E. Colgate, P. Akella, and W. Wannasuphprasit, “Cobots in Materials Handling,” 2000.
- [15] W. Wannasuphprasit, B. Gillespie, J. E. Colgate, and M. A. Peshkin, “Cobot Control,” Albuquerque, NM, 1997.
- [16] J. E. Colgate and M. A. Peshkin, “COBOTS,” Estados Unidos, p. 19, 1999.
- [17] InfoPLC, “La Robótica Colaborativa crecerá hasta 10 veces en 2020.” 2016. [Online]. Available: <http://www.infopl.net>
- [18] A. P. Calitz, P. Poisat, and M. Cullen, “The future African workplace: The use of collaborative robots in manufacturing,” *SA Journal of Human Resource Management*, 2017.
- [19] A. M. Djuric, R. J. Urbanic, and J. L. Rickli, “A Framework for Collaborative Robot (CoBot) Integration in Advanced Manufacturing Systems,” *SAE Int. J. Mater. Manf.*, 2016.
- [20] S. Haddadin, “Physical Safety in Robotics,” in *Formal Modeling and Verification of Cyber-Physical Systems*, R. Drechsler and U. Kühne, Eds. Germany: Springer, 2015.
- [21] L. F. Baptista, J. M. C. Sousa, and J. M. G. S. da Costa, *Predictive Force Control of Robot Manipulators in Nonrigid Environments.*, S. Cubero, Ed., Alemania, 2006.
- [22] S. Chiaverini, B. Siciliano, and L. Villani, “A survey of robot interaction control schemes with experimental comparison,” *IEEE/ASME Transactions on mechatronics*, vol. 4, no. 3, pp. 273–285, 1999.


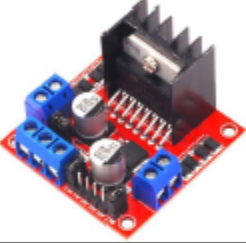



- [23] T. Yoshikawa, “Force control of robot manipulators.” in *IEEE international conference on robotics and automation.*, 2000.
- [24] J. N. Pires and T. Godinho, “Control difuso de fuerza en robotica industrial,” *Ingenierias*, vol. XI, no. 41, 2008.
- [25] H. Montes, “Análisis, diseño y evaluación de estrategias de control de fuerza en robots caminantes,” Ph.D. dissertation, Madrid, 2006.
- [26] Z. Jelacic, “Contact force problem in the rehabilitation robot control design,” in *International Conference on Medical and Biological Engineering*, A. Badnjevic, Ed., vol. 62. Singapur: Springer, 2017, pp. 193–204.
- [27] V. Ortenzia, R. Stolkina, J. Kuob, and M. Mistryc, “Hybrid motion/force control: a review.” *Advanced Robotics*, pp. 1–12, 2019.
- [28] K. T. Song and H. P. Li, “A Fuzzy Adaptative Control Design for Compliant Motion of a Manipulator,” in *IEEE International Conference on Robotics and Automation*, Minneapolis, 1996.
- [29] S. T. Lin and A. K. Huang, “Hierarchical fuzzy force control for industrial robots,” *IEEE Transactions on Industrial Electronics*, vol. 45, no. 4, pp. 646–653, 1998.
- [30] Y. Sato, “Study of intelligent impedance control for robot arm.” in *International Conference on Artificial Intelligence and Software Engineering*, S. K. Chen and Y. H. Chang, Eds., Taiwan, 2014, p. 655.
- [31] K. Rani and N. Kumar, “Design of intelligent Hibrid Force and Position Control of Robot Manipulation.” *Procedia Computer Science*, vol. 125, pp. 42–49, 2018.
- [32] M. A. Llama, W. Z. Castañón, and R. Garcia-Hernandez, “Force and Position Fuzzy Control: A Case Study in a Mitsubishi PA10-7CE Robot Arm.” in *Advanced Topics on Computer Vision, Control and Robotics in Mechatronics*, O. O. V. Villegas, M. Nandayapa, and I. Soto, Eds. México: Springer, 2018.
- [33] S. Yoshikawa and K. Yamada, “Impact estimation of a space robot at capturing a target,” *IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, vol. 3, pp. 1570–1577, 1994.
- [34] S. A. A. Moosavian and E. Papadopoulos, “On the control of space free-flyers using multiple impedance control,” *IEEE International Conference on Robotics and Automation*, vol. 1, pp. 853–858, 1997.
- [35] T. Murakami, F. Yu, and K. Ohnishi, “Torque Sensorless Control in Multidegree-of-Freedom Manipulator,” *IEEE Trans. Ind. Electron*, vol. 40, no. 2, pp. 259–265, 1993.

- [36] S. Katsura, Y. Matsumoto, and K. Ohnishi, “Analysis and Experimental Validation of Force Bandwidth for Force Control,” *IEEE Trans. Ind. Electron.*, vol. 53, no. 3, pp. 922–928, 2006.
- [37] A. Gupta and M. K. O’Malley, “Disturbance-Observer-Based Force Estimation for Haptic Feedback,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 133, 2011.
- [38] J. Qin, F. Leonard, and G. Abba, “Experimental External Force Estimation Using a Non-Linear Observer for 6 axes Flexible-Joint Industrial Manipulators.” Istanbul; Turkey, pp. 1–6, 2013.
- [39] A. Flores-Abad, M. Nandayapa, and M. A. Garcia-Teran, “Force Sensorless Impedance Control for a Space Robot to Capture a Satellite for On-orbit Servicing,” in *IEEE Aerospace Conference*, no. 1-7. IEEE, 2018.
- [40] M. H. Ang, W. Lin, and S.-Y. Lim, “A walk-through programmed robot for welding in shipyards,” *Industrial Robot: An International Journal*, 1999.
- [41] A. Q. Keemink, H. van der Kooij, and A. H. Stienen, “Admittance control for physical human–robot interaction,” *The International Journal of Robotics Research*, vol. 37, no. 11, pp. 1421–1444, 2018.
- [42] G. Ferretti, G. Magnani, and P. Rocco, “Assigning virtual tool dynamics to an industrial robot through an admittance controller,” in *2009 International Conference on Advanced Robotics*. IEEE, 2009, pp. 1–6.
- [43] L. Bascetta, G. Ferretti, G. Magnani, and P. Rocco, “Walk-through programming for robotic manipulators based on admittance control,” *Robotica*, vol. 31, no. 7, pp. 1143–1153, 2013.
- [44] C. T. Landi, F. Ferraguti, C. Secchi, and C. Fantuzzi, “Tool compensation in walk-through programming for admittance-controlled robots,” in *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2016, pp. 5335–5340.
- [45] W. Bauer, M. Bender, M. Braun, P. Rally, and O. Scholtz, “Lightweight robots in manual assembly – best to start simply!” 2016.
- [46] A. Barrientos, L. F. Peñin, C. Balaguer, and R. Aracil, *Fundamentos de robótica*, 2nd ed. Madrid, España: McGraw-Hill, 2007.
- [47] K. M. Lynch and F. C. Park, *Modern robotics. mechanics, planning, and control*. Cambridge University Press, 2017.
- [48] M. H. Ordoñez, M. B. O. Moctezuma, C. A. C. Arriaga, and J. C. R. Portillo, *Robótica: Análisis, modelado, control e implementación*. OmniaScience, 2015.

- [49] D. W. Roberts, “Bond Graph Model Based Control of Robotic Manipulators,” Ph.D. dissertation, 1993.
- [50] R. Kelly, V. S. Davila, and J. A. L. Perez, *Control of robot manipulators in joint space*. Springer Science & Business Media, 2006.
- [51] L. Ljung, *System identification. Theory for the user*. Prentice Hall, 1999.
- [52] —, *System identification toolbox: User’s guide*. Citeseer, 1995.
- [53] K. Ohishi, K. Ohnishi, and K. Moyachi, “Torque-Speed regulation of DC motor based on load torque estimation method,” in *IEEJ International Power Electronics Conference*, vol. 2, 1983, pp. 1209–1216.
- [54] R. Oboe, “How Disturbance Observer Changed my Life ,” 2018.
- [55] S. Komada and K. Ohnishi, “Force feedback control of robot manipulator by the acceleration tracing orientation method,” *IEEE Transactions on Industrial Electronics*, vol. 37, no. 1, pp. 6–12, 1990.
- [56] A. Orozco, A. F. Abad, O. O. V. Villegas, V. G. C. Sanchez, and M. Nandayapa, “Interaction Control Based on Position/Velocity Control by means of Disturbance Observer,” pp. 75–80, 2014.
- [57] W.-H. Chen, D. J. Ballance, P. J. Gawthrop, and J. O’Reilly, “A nonlinear disturbance observer for robotic manipulators,” *IEEE Transactions on industrial Electronics*, vol. 47, no. 4, pp. 932–938, 2000.
- [58] B. Siciliano and O. Khatib, *Springer handbook of robotics*. Springer, 2016.
- [59] M. H. Choi and W. W. Lee, “A force/moment sensor for intuitive robot teaching application,” in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, vol. 4. IEEE, 2001, pp. 4011–4016.
- [60] F. Datasheet, “Model 400, interlink electronics.”
- [61] W. Chen, D. J. Ballance, and P. J. Gawthrop, “Nonlinear Generalised Predictive Control and Optimal Dynamical Inversion Control,” in *IFAC Proceedings Volumes*, vol. 32, no. 2, 1999, pp. 2540–2545.

Apéndice A

Características de los componentes.

Componentes	Imagen	Características
Actuadores: Motor DC (Polulu 3076)		$V_{in} = 6 \text{ V}$ $I_{sin \text{ carga}} = 0.1 \text{ A}$ $\omega_{sin \text{ carga}} = 220 \text{ rpm}$ $I_{stall} = 1.5 \text{ A}$ $\tau_{stall} = 2 \text{ Kgcm}$ GB 150:1
Etapas de potencia: Puente H (Módulo L298N)		$V_{out} = 7\sim 46 \text{ V}$ $I_{m\acute{a}x} = 2 \text{ A}$ $P_{out} = 25 \text{ W}$ $V_{ctrl} = 5 \text{ V}$ $I_{ctrl} = 36 \text{ mA}$ "1" $2.3 \leq V_{ctrl} \leq V_{cc}$ "0" $-0.3 \leq V_{ctrl} \leq 1.5 \text{ V}$
Controladora: Arduino (Mega 2650)		$\mu\text{Ctrl: ATmega2560}$ $V_{in} = 7\sim 12 \text{ V}$ Flash: 256 Kb SRAM: 8 Kb EEPROM: 4 Kb Clock: 16 MHz I/O digital: 54 I/O <i>análoga</i> : 16
Sensor (Posición): <i>Encoder</i> magnético (Polulu 3081)		<i>Encoder</i> en cuadratura. Disco magnético y sensores de efecto Hall. $V_{in} = 2.7\sim 18 \text{ V}$ 12 CPR
Sensor (Fuerza): FSR (FSR 400 Interlink Electronics)		Rango: 0.2 ~ 2 N Resolución: Continuo Repetibilidad: $\pm 2 \%$ $R_{sin \text{ carga}} \geq 10 \text{ M}\Omega$ $T_{subida} < 3 \mu\text{s}$

Apéndice B

Comunicación entre el Arduino Mega y Simulink.

