



**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS DEL  
INSTITUTO POLITÉCNICO NACIONAL**

**Unidad Zacatenco  
Departamento de Ingeniería Eléctrica  
Sección de Mecatrónica**

**SISTEMA COOPERATIVO DE ROBOTS INDUSTRIALES PARA  
TAREAS CON INTERACCIÓN HUMANA**

**T E S I S**

Que presenta:

**ING. ERICK GABINO MARTÍNEZ REYES**

Para obtener el grado de:

**MAESTRO EN CIENCIAS**

En la especialidad de:

**INGENIERÍA ELÉCTRICA**

Director de Tesis:

**Dr. ALEJANDRO RODRÍGUEZ ÁNGELES**

Ciudad de México

Febrero 2022

# Agradecimientos

**A mis padres Alejandro y Patricia:** Por su apoyo, por su comprensión, por sus consejos y porque ustedes fueron los cimientos de mis logros. Todos sus sacrificios nunca los olvidaré, sus enseñanzas y regaños que me forjaron como soy en la actualidad. Quiero que este logro en mi vida sea también de ustedes. Gracias por su motivación.

**A mis hermanos:** Por su apoyo, porque nunca me han dejado solo, gracias por su motivación y ayuda que me han brindado siempre, sepan hermanos que jamás los dejare de apoyar.

**A mis compañeros Enrique, Alan, Miguel, Felipe e Ivan:** por que sin su ayuda y apoyo la maestría hubiera sido diferente, quiero que sepan que siempre contarán conmigo.

**A mi asesor Dr. Alejandro Rodríguez Ángeles:** Por sus consejos, por la disposición y paciencia de asesorar este trabajo de tesis, por las revisiones realizadas, por transmitir su conocimiento, y por último por su aporte en mi formación académica.

**Al CINVESTAV:** Por darme la oportunidad de realizar mis estudios de maestría.

**Al CONACYT:** Por otorgar la beca de tiempo completo que me dio la oportunidad de realizar mis estudios de maestría y por el apoyo brindado a través del proyecto CONACYT de Ciencia Básica número CB-2017-2018-A1-S-26123.

# Resumen

En el presente trabajo de tesis se muestra el desarrollo de un sistema de robots cooperativos, conformado por dos robots industriales de la marca ABB. Este sistema tiene como objetivo realizar tareas de manipulación y seguimiento de trayectorias que, en este caso el humano debe imponer interactuando directamente con el objeto a manipular, sin perder de vista que el sistema debe cumplir con la fuerza y posición que se le imponga.

Debido a la pandemia, las pruebas del sistema serán de manera virtual, por esta razón, no pudieron ser completadas las pruebas con interacción humana, debido a la evacuación de las instalaciones y el alcance del software utilizado.

Se describen los modelos de cada elemento usado en el sistema cooperativo, se describe la cinemática y dinámica de los manipuladores, el modelo de impedancia para realizar la interacción con el entorno, el modelado del objeto y del agarre, etc.

Los esquemas de control que se implementan en los manipuladores son del tipo híbrido posición-fuerza/par en los dos robots, de esta manera el sistema cooperativo se encuentra en reacción garantizando la fuerza de agarre del objeto.

Las tareas programadas al sistema se realizan por los robots cuando toman el objeto, para posteriormente agregar la interacción humana, sin embargo, resulta complicado introducir la presencia humana dentro del sistema cooperativo en la plataforma virtual RobotStudio, debido a esto, se agrega un tercer robot que es el encargado de imponer la trayectoria a seguir por el sistema cooperativo. De esta manera los dos robots sostienen el objeto, mientras que el tercer robot ejerce fuerza sobre él, en la dirección del movimiento deseado, se consideran tareas en las que los robots manipulan un objeto de forma coordinada, esto sin que exista interacción directa entre robots, es decir que estos tienen interacción solo a través del objeto.

Los robots industriales son de arquitectura cerrada, por esta razón no se tiene acceso directo a los algoritmos de control, por ello se utiliza el lenguaje RAPID propio de ABB con la plataforma virtual RobotStudio, de esta manera el esquema de control mencionado permite el buen funcionamiento de los robots manipuladores de arquitectura cerrada, teniendo el control de la posición y de la fuerza en el efector final. Cabe mencionar que no se utiliza ningún kit de fuerza o sincronización que sea proporcionado por la marca ABB.

Las pruebas realizadas se muestran en tres etapas, la primera consiste en un solo manipulador, para mostrar que el control híbrido posición-fuerza/par y el control híbrido posición-fuerza/par extendido funcionan adecuadamente combinado con el modelo de impedancia, la segunda etapa es el sistema cooperativo compuesto por dos manipuladores al mover un objeto de manera coordinada al aplicar un diferencial de fuerza, y por último el sistema

cooperativo compuesto por dos manipuladores agregando a éste un tercer robot, para realizar la manipulación de un objeto en común en un espacio bidimensional.

Al utilizar RobotStudio y ser uno de los primeros trabajos en la Sección en trabajar con los robots de manera virtual se realizaron experimentos que de primera instancia eran complicados, ya que al no tener de manera física los robots manipuladores y los sensores de fuerza, y además no se cuenta con las librerías donde se tienen los sensores de fuerza propios de ABB, se opta por implementar el modelo de impedancia para hacer las simulaciones pertinentes. De esta manera, en los resultados de simulación se observa que las mediciones de fuerza a través del modelo antes mencionado son aceptables, por ello se muestran los posibles alcances que se tiene al programar los robots en dicha plataforma, incluyendo sus limitaciones.

# Abstract

This thesis work presents the development of a cooperative robots system made up of two industrial robots of the ABB brand. The objective of this system is to carry out tasks of manipulation and tracking of trajectories that in this case the human must impose them by directly interacting with the object to be manipulated, without losing sight of the fact that the system must comply with the force and position imposed on it.

Due to the pandemic, the system tests will be carried out virtually, for this reason, the tests with human interaction could not be completed, due to the evacuation of the facilities and the scope of the software used.

Likewise, the models of each element used in the simulations are described, taking into account the kinematics and dynamics of the manipulators, the model of impedance presented of the interaction with the environment, the modelling of the object and grip, etc.

The control schemes that are implemented in the manipulators are identical, in this case the hybrid position-force / torque control is used in the two robots, in this way the cooperative system is in reaction guaranteeing the grasping force of the object.

The programmed tasks to the system are carried out by the robots when they take the object, to later add human interaction, however, it is difficult to introduce the human presence within the cooperative system in the RobotStudio virtual platform, due to this, a third robot is added which is in charge of imposing the path to be followed by the cooperative system. In this way the two robots hold the object, while the third robot exerts force on it in the direction of the desired movement, tasks are considered in which the robots manipulate an object in a coordinated way, this without direct interaction between robots, in other words, these have interaction only through the object.

Industrial robots are closed architecture, for this reason there is no direct access to the control algorithms, which is why ABB's own RAPID language is considered using the RobotStudio virtual platform. The aforementioned control scheme allows the proper functioning of the closed architecture manipulator robots, having the position and force control at the end effector. It should be mentioned that no force or synchronization kit is used that are provided by the brand.

The tests carried out are shown in 3 stages, the first consists of a single manipulator, to show that the hybrid control position-force/torque and extended hybrid position-force/torque work properly combined with the impedance model, the second stage is the cooperative system composed of two manipulators moving an object in a coordinated way while applying a differential force, and finally the cooperative system composed of two manipulators adding

a third one robot, to perform the manipulation of the object in a two-dimensional space.

By using RobotStudio and being one of the first jobs in the Mechatronics Group to work with virtual robots, complications arose when trying out the experiments, due to not having physically the robots and the force sensors, and also not having the libraries with the proper force sensors of ABB, then is decided to implement the impedance model to make the relevant simulations. In this way, in the simulation results is observed that measures of force through the aforementioned model are acceptable, for this reason, the possible scope of programming the robots on said platform is shown, including their limitations.

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Estado del arte . . . . .	5
1.2. Motivación . . . . .	7
1.3. Objetivos . . . . .	8
1.4. Contribuciones . . . . .	9
1.5. Organización . . . . .	9
<b>2. Modelos matemáticos</b>	<b>11</b>
2.1. Modelado del entorno . . . . .	11
2.2. Modelo del Robot Manipulador . . . . .	13
2.2.1. Modelo cinemático . . . . .	13
2.2.2. Modelo dinámico . . . . .	15
2.3. Sistema cooperativo de robots industriales . . . . .	16
2.3.1. Modelo dinámico del sistema cooperativo . . . . .	16
2.3.2. Humano-sistema cooperativo . . . . .	16
2.4. Modelo dinámico del objeto . . . . .	18
2.4.1. Base matemática para la tarea de agarre . . . . .	18
<b>3. Esquemas de control</b>	<b>20</b>
3.1. Control híbrido posición-fuerza/par . . . . .	20
3.1.1. Control de posición interno del robot . . . . .	21
3.1.2. Control de posición en robots ABB . . . . .	22
3.1.3. Control de fuerza-par . . . . .	22
3.2. Control híbrido posición-fuerza/par en robots ABB . . . . .	24
3.3. Control híbrido posición-fuerza/par extendido. . . . .	24
3.4. Análisis de estabilidad en lazo cerrado . . . . .	25
3.4.1. Modelo dinámico de los actuadores . . . . .	26
<b>4. Plataforma experimental</b>	<b>32</b>
4.1. RobotStudio . . . . .	32
4.1.1. Creación de una estación e interfaz de trabajo. . . . .	33
4.2. Plataforma experimental RoboStudio . . . . .	35
4.3. Descripción Robot manipulador IRB 2400/16 . . . . .	36
4.3.1. Comunicación de controladores virtuales . . . . .	37
4.4. RAPID . . . . .	40
4.5. Sensor de fuerza y tarjeta de adquisición de datos . . . . .	41
4.5.1. Creación de herramienta. . . . .	42
4.6. Interacción humana . . . . .	43

<b>5. Resultados de simulación</b>	<b>45</b>
5.1. Experimento 1: Aplicación del control híbrido posición-fuerza. . . . .	45
5.2. Experimento 2: Aplicación del control híbrido posición-fuerza extendido . . . .	50
5.3. Experimento 3: Sistema cooperativo. . . . .	53
5.3.1. Movimiento horizontal del objeto a partir de un diferencial de fuerza. . .	53
5.4. Experimento 4: Sistema cooperativo con interacción humana . . . . .	60
<b>6. Conclusiones y trabajo futuro</b>	<b>76</b>
6.1. Conclusiones . . . . .	76
6.2. Trabajo futuro . . . . .	77
<b>Apéndice A</b>	<b>82</b>
<b>Apéndice B</b>	<b>85</b>
<b>Apéndice C</b>	<b>92</b>
<b>Apéndice D</b>	<b>101</b>



# Índice de figuras

1.1. Sistema robotico GoFa para trabajos cooperativos. [1] . . . . .	2
1.2. Robot terrestre Perseverance con instrumentos y tecnología de vanguardia [2] .	3
1.3. Robot ABB realizando corte en un punto específico . . . . .	4
2.1. Sistema cooperativo humano . . . . .	17
2.2. Esquema de agarre . . . . .	19
3.1. Esquema de control híbrido posición fuerza/par . . . . .	21
3.2. Esquema de control de posición en los robots industriales . . . . .	22
3.3. Esquema de control híbrido posición-fuerza/par para robots industriales ABB	24
3.4. Esquema de control híbrido posición fuerza/par extendido . . . . .	25
3.5. Diagrama de motor DC. . . . .	26
4.1. Interfaz RobotStudio . . . . .	34
4.2. Plataforma experimental . . . . .	35
4.3. Robot IRB 2400/16 . . . . .	36
4.4. Controlador IRC5 . . . . .	36
4.5. Creación de señales digitales virtuales . . . . .	37
4.6. Lógica de estación . . . . .	38
4.7. Diagrama de conexiones entre controladores . . . . .	38
4.8. Estructura de un programa en RAPID. . . . .	40
4.9. Herramienta como sensor de fuerza . . . . .	41
4.10. Posicionar TCP de una herraamienta . . . . .	42
4.11. Navegador de objetos . . . . .	43
5.1. Sistema robótico en posición Home . . . . .	46
5.2. Sistema robótico en posición de acercamiento a la mesa . . . . .	47
5.3. Sistema robótico cuando alcanza el valor de referencia de fuerza deseada . . .	47
5.4. Propiedades del objeto experimento 1 . . . . .	48
5.5. Posición del efector final del robot en z (acercamiento) . . . . .	48
5.6. Posición del efector final del robot, experimento 1 . . . . .	49
5.7. Seguimiento de trayectoria de fuerza en z . . . . .	49
5.8. Error de fuerza en el eje z, experimento 1 . . . . .	50
5.9. Penetración del manipulador sobre la superficie de contacto . . . . .	50
5.10. Sistema robótico posicionado sobre la mesa . . . . .	50
5.11. Seguimiento de fuerza constante . . . . .	51
5.12. Posición del efector final en los 3 ejes . . . . .	52
5.13. Error de fuerza . . . . .	52
5.14. Penetración sobre la superficie . . . . .	53

5.15. Seguimiento de trayectoria del manipulador . . . . .	53
5.16. Propiedades del objeto a manipular . . . . .	54
5.17. Sistema cooperativo en RobotStudio . . . . .	54
5.18. Posición $x, y, z$ del robot 1 (izquierdo), experimento 3 . . . . .	57
5.19. Posición del efector final robot 1 en $x$ (acercamiento) . . . . .	57
5.20. Posición $x, y, z$ del robot 2 (derecho), experimento 3 . . . . .	58
5.21. Posición del efector final robot 2 en $x$ (acercamiento) . . . . .	58
5.22. Seguimiento de fuerza robot 1 (izquierdo) . . . . .	59
5.23. Error de fuerza Robot 1 (izquierdo) . . . . .	59
5.24. Seguimiento de fuerza Robot 2 (derecho) . . . . .	59
5.25. Error de fuerza Robot 2 (derecho) . . . . .	60
5.26. Sistema cooperativo con interacción de un tercer robot . . . . .	60
5.27. Propiedades del objeto . . . . .	61
5.28. Diagrama del agarre experimento 4 . . . . .	61
5.29. Seguimiento de fuerza en $x$ , robot 1 . . . . .	63
5.30. Seguimiento de fuerza en $y$ , robot 1 . . . . .	63
5.31. Seguimiento de fuerza en $x$ , robot 1 (acercamiento) . . . . .	64
5.32. Seguimiento de fuerza en $y$ , robot 1 (acercamiento) . . . . .	64
5.33. Posición del efector final en $y$ , robot 1 (acercamiento) . . . . .	64
5.34. Posición del efector final en los 3 ejes, robot 1 . . . . .	65
5.35. Error de fuerza en $x$ , robot 1 . . . . .	65
5.36. Error de fuerza en $y$ , robot 1 . . . . .	66
5.37. Penetración sobre el objeto en el eje $x$ , robot 1 . . . . .	66
5.38. Penetración sobre el objeto en el eje $y$ , robot 1 . . . . .	66
5.39. Seguimiento de fuerza en $x$ , robot 2 . . . . .	68
5.40. Seguimiento de fuerza en $y$ , robot 2 . . . . .	68
5.41. Seguimiento de fuerza en $x$ , robot 2 (acercamiento) . . . . .	69
5.42. Seguimiento de fuerza en $y$ , robot 2 (acercamiento) . . . . .	69
5.43. Posición del efector final en $y$ , robot 2 (acercamiento) . . . . .	69
5.44. Posición del efector final en los 3 ejes, robot 2 . . . . .	70
5.45. Error de fuerza en $x$ , robot 2 . . . . .	70
5.46. Error de fuerza en $y$ , robot 2 . . . . .	71
5.47. Penetración sobre el objeto en el eje $x$ , robot 2 . . . . .	71
5.48. Penetración sobre el objeto en el eje $y$ , robot 2 . . . . .	71
5.49. Seguimiento de fuerza en $y$ , robot 3 . . . . .	73
5.50. Error de fuerza en $y$ , robot 3 . . . . .	74
5.51. Penetración sobre el objeto en $y$ , robot 3 . . . . .	74

# Índice de tablas

5.1. Posición y orientación del efector final antes de la penetración experimento 1 .	47
5.2. Posición y orientación del efector final experimento 2 (home) . . . . .	51
5.3. Posición del efector final experimento 2 (contacto) . . . . .	51
5.4. Posición del efector final experimento 3 (home 1) . . . . .	55
5.5. Posición del efector final experimento 3 (contacto 1) . . . . .	55
5.6. Posición del efector final experimento 3 (home 2) . . . . .	55
5.7. Posición del efector final experimento 3 (contacto 2) . . . . .	56
5.8. Posición del efector final experimento 4 (home 1) . . . . .	62
5.9. Posición del efector final experimento 4 (acercamiento 1) . . . . .	62
5.10. Posición del efector final experimento 4 (contacto 1) . . . . .	62
5.11. Posición del efector final experimento 4 (home 2) . . . . .	67
5.12. Posición del efector final experimento 4 (acercamiento 2) . . . . .	67
5.13. Posición del efector final experimento 4 (contacto 2) . . . . .	67
5.14. Posición del efector final experimento 4 (home 3) . . . . .	72
5.15. Posición del efector final experimento 4 (acercamiento 3) . . . . .	72
5.16. Posición del efector final experimento 4 (contacto 3) . . . . .	72



# Capítulo 1

## Introducción

El avance de la tecnología aumenta día con día, éste desarrollo tecnológico es necesario para un país que sigue en constante crecimiento, la trascendencia del desarrollo científico favorece las posibilidades para que la población obtenga beneficios como mejorar la salud y calidad de vida en general. En la actualidad los sistemas robóticos tienen diferentes aplicaciones en distintas áreas, estos sistemas son cada vez más indispensables en líneas de ensamblaje o en aplicaciones médicas, teleoperación, juegos, agricultura, operaciones espaciales, entre otras, de igual manera el progreso de la ciencia y tecnología han contribuido a transformar el concepto de robot, el cual hoy en día para la sociedad es un sinónimo de automatización

La modernidad y los avances tecnológicos han tenido gran impacto en la sociedad, han cambiado de manera rápida las formas en las que se realizan las tareas propias del trabajo para sistemas de producción a nivel industrial. El avance de la tecnología y la evolución de la industria y sus procesos, han permitido que la sociedad pueda comprender hoy la relación entre el mejoramiento de los procesos industriales y los beneficios en cuanto a costo y tiempo.

La robótica es una disciplina que aborda entre otros temas, la investigación y desarrollo de una clase particular de sistemas mecánicos, denominados robots manipuladores, diseñados para realizar una amplia variedad de aplicaciones industriales, científicas, domésticas y comerciales [3].

La industria siempre ha buscado alcanzar mejoras en los procesos para la producción, pues en cuanto mayor ahorro en tiempo y gasto, mejor es el resultado para la empresa y para el sector social que depende de la actividad de la empresa. Desde la revolución industrial, la industria pudo pasar de ser una producción, en su mayoría integrada por el ser humano, a ser compartida entre las máquinas y el ser humano en colaboración.

La relación entre el ser humano y el robot se ha vuelto una constante necesaria dentro de la industria, pues al poder unir al ente controlador y al asistente en la realización de las funciones, los resultados y capacidades para la operación se magnifican en cifras realmente considerables y beneficiosas para el sector.

En [14] se menciona la importancia de destacar las tareas donde se precisa la intervención del ser humano que son por lo general, el tipo de aplicaciones donde la persona posee

mayor sensibilidad para realizarlas; entonces sería muy complicado reemplazarla por un robot manipulador ya que carece de ciertas habilidades para las cuales las personas están más dotadas. Lo que se busca entonces es que los manipuladores se transformen en asistentes inteligentes capaces de asistir al humano en todo tipo de tareas, aprovechando los recursos y características positivas, minimizando con la colaboración entre sí las características negativas.

Los sistemas cooperativos en la actualidad han tomado mayor relevancia, ya que hay tareas que un solo robot manipulador no puede realizar, para este tipo de sistemas se ha realizado investigación y se han establecido normas de seguridad que permitan el trabajo cooperativo con humanos. Algunas de las empresas que ofrecen sistemas cooperativos son ABB con el robot GoFa, como se observa en la Figura 1.1 que incorpora características que permiten el uso y operación de forma segura, sin la necesidad de construir barreras o vallas voluminosas, el sistema permite que el robot y el humano puedan compartir el espacio de trabajo y puedan realizar tareas cooperativas sin poner en peligro la productividad ni seguridad como se explica en [1].



Figura 1.1: Sistema robotico GoFa para trabajos cooperativos. [1]

Los robots manipuladores están diseñados para cumplir tareas programables, como mover materiales, herramientas, o dispositivos especializados siguiendo trayectorias dentro de su área de trabajo, sin embargo para otros tipos de manipulación los robots deben ser equipados con sensores, cámaras de visión, etc, ya que actividades como sujeción de un objeto entre dos o más robots necesitan controlar la fuerza que ejercen sobre el área de trabajo, por lo que un control de posición no sería suficiente, de este modo se implementan esquemas de control que permitan controlar la fuerza y posición del robot.

La cooperación entre el ser humano y el robot dentro del proceso industrial, forman una relación natural y necesaria para poder transitar a un mejoramiento en los procesos ligados a la producción. Esta relación es importante, pues aún con el gran avance tecnológico, no existe un robot que sea capaz de poder pensar por sí mismo, por lo que es necesario que el ser humano determine las funciones y operaciones a realizar.

Es por esto que, partiendo de la evolución tecnológica y el crecimiento en la demanda de la sociedad, los robots y el ser humano necesitan trabajar en cooperación para poder satisfacer las necesidades del mercado, y obtener resultados positivos. La cooperación entre

robots y el ser humano, es más una necesidad, pues solo así se podrán alcanzar las metas y el mejoramiento productivo, entendiendo a los robots como una herramienta que parte de la inteligencia humana para poder realizar una mejora en las tareas. En [14] refiere que una de las características de la cooperación es, tener en cuenta que cuando se efectúan tareas en forma cooperativa y coordinada, existe además una intersección entre las áreas de trabajo de los manipuladores y de las personas, ya que se encuentran en la misma celda de trabajo ejecutando acciones simultáneamente en un mismo producto.

La cooperación entonces se hace de manera constante y bajo un mismo espacio de trabajo, pues debe existir una coordinación entre el individuo y el robot para la ejecución de las actividades o la orientación de las funciones a realizar. Es por esto que hay que comprender de antemano, que los robots dependen directamente del individuo para la ejecución de sus funciones, y para la dirección y coordinación de las actividades a realizar. Este contexto en general se conoce como planificación de tareas.

Es por esto que, en el presente trabajo de investigación, se buscará analizar de primera mano el proceso de cooperación entre el robot y el ser humano para la realización de tareas industriales, y cómo esa relación desencadena una serie de conceptos que son necesarios de entender para poder avanzar hacia los nuevos procesos de producción industrial, se buscará comprender la estructura de la relación de cooperación entre el robot y el individuo, y se desglosarán los conceptos más importantes para su comprensión.

En general los robots pueden clasificarse como móviles, humanoides e industriales, por ejemplo, los robots móviles que se desplazan mediante ruedas o patas, un ejemplo es el robot perseverance como se observa en la Figura 1.2, el cual es el más sofisticado de la NASA que será enviado a marte, este robot está equipado con cámaras de las más avanzadas como en la actualidad Mastcam-Z. Los robots industriales son los de mayor importancia, debido a la utilidad que se le da en el sector industrial como parte principal de modernización de empresas, estos robots en su mayoría son brazos mecánicos con diferentes articulaciones para hombro y codo, la principal ventaja al incorporar robots en la industria es la repetitividad, ya que estos no se cansan ni se fatigan como sucede con los humanos.

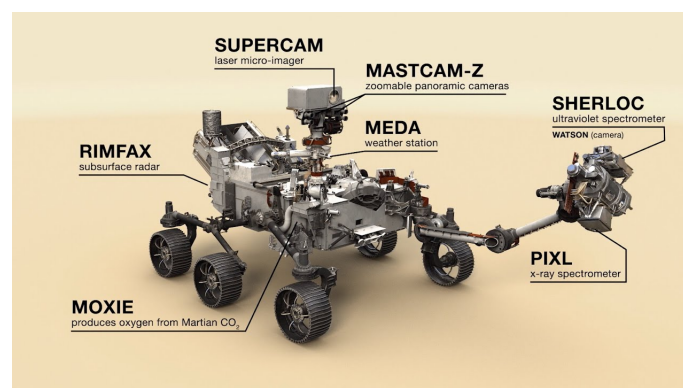


Figura 1.2: Robot terrestre Perseverance con instrumentos y tecnología de vanguardia [2]

Dentro de las principales aplicaciones que se tienen en los robots industriales están los procesos de pintado en cubetas, carrocerías automotrices, tinas, soldadura de punto en diversas piezas industriales, traslado de herramientas, empaquetado de materiales, entre

otras.

Los robots están equipados con sensores para proporcionar información del estado del robot, de igual manera en aplicaciones específicas utilizan sensores de fuerza para conocer la interacción con el ambiente, se equipan con cámaras de video para localización de objetos en el espacio de trabajo.

Los robots industriales cuentan con un control interno de posición que, permite situar el manipulador en un punto dentro de su espacio de trabajo como se muestra en la Figura 1.3, donde se presenta un robot realizando un corte en un punto específico, como éste se tienen más ejemplos de robots realizando trabajos de *"pick and place"* o trabajos de soldadura.

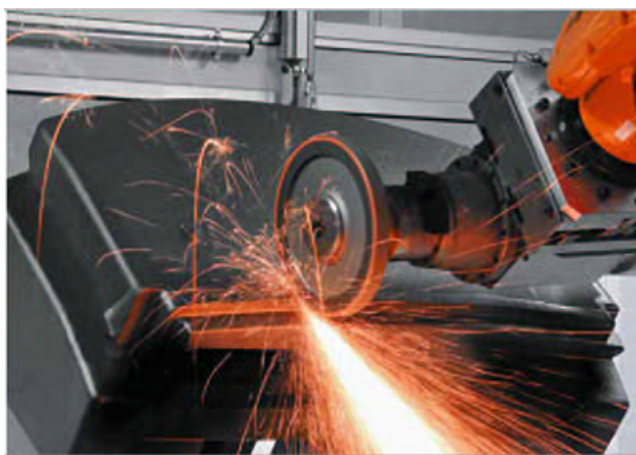


Figura 1.3: Robot ABB realizando corte en un punto específico

La cooperación entre el humano y el robot dentro del proceso industrial, forman una relación natural y necesaria para poder transitar a un mejoramiento en los procesos ligados a la producción. Esta relación es importante, pues aún con el gran avance tecnológico, no existe un robot que sea capaz de poder pensar por si mismo, para ello existe gran cantidad de esquemas de control, mismos que se encuentran en investigaciones como en [38] tratando el tema de control de impedancia y en [3] con esquemas de control de un enfoque alternativo de impedancia y trabajos como modelos de dinámica inversa con aplicación al control de impedancia como se explica en [20]

Los sistemas de robots cooperativos con interacción humana representan una nueva generación de robots, este sistema permite coordinar múltiples robots para establecer o comunicar la posición de cada robot en el espacio de trabajo. En el presente trabajo el sistema cooperativo se conforma de dos robots industriales ABB los cuales son de arquitectura cerrada, en otras palabras no se tiene acceso a los algoritmos de control internos del robot, de tal manera que los esquemas de control que se implementarán se realizarán por medio del lenguaje RAPID a través de RobotStudio, mismo software que se utilizará para simulaciones de los robots y del esquema de control de fuerza. La idea principal es instalar en el efector final del sistema cooperativo un sensor de fuerza en cada robot, que para efectos de simulación utiliza el modelo de impedancia que se explica en el Capítulo 2, y en lugar de la interacción humana, se agrega al sistema cooperativo un manipulador más.



## 1.1. Estado del arte

La robótica cuenta con un antecedente muy reciente, sobre todo si se analiza desde la óptica industrial, y el concepto que engloba a lo que hoy se conoce como robots. La robótica según lo refiere [25], inicia en la época de 1960, en donde se comenzaron a realizar los primeros prototipos de máquinas operadas por programas.

Ahora bien, los programas que dieron el inicio al movimiento de lo que se conoce como la robótica en la actualidad, iniciaron en la década de 1950, con la programación para cálculos matemáticos y manipulación de símbolos. Pues bien, estos antecedentes proporcionaron la base para lo que en la actualidad se conoce como la inteligencia artificial. Los algoritmos que se utilizaban en el pasado, eran incapaces de poder comprender o resolver situaciones, por lo que su programación era orientada a funciones no cognitivas.

La relación actual entre el ser humano y el robot debe entenderse desde el punto de vista de la cooperación, pues los avances han permitido que la máquina como comúnmente se conoce, pueda entender y analizar un medio y así tomar una decisión con base a las funciones o programaciones asignadas por el individuo. La cooperación no se basa simplemente en asignar funciones, sino en poder planificar un programa determinado de tareas en donde el ser humano pueda indicarle al robot las funciones a realizar para obtener el resultado esperado.

En [21] se hace uso del control de impedancia para manipulaciones robóticas, se introduce el concepto llamado target-impedance reference trajectory (trayectoria de referencia de impedancia-objetivo) (TIRT) por sus siglas en inglés, que caracteriza una relación dinámica deseada del punto final con el ambiente y un enfoque de Lyapunov, en él se presentan dos algoritmos de control de impedancia adaptativa, se realizan suposiciones en que algunos parámetros en la dinámica del manipulador pueden ser desconocidos o inciertos y que las mediciones del sensor de fuerza utilizados son imprecisas, se observa que los mecanismos de adaptación pueden adecuarse al método propuesto por N. Hogan [38] con el esquema de control de impedancia.

Otra contribución a los sistemas cooperativos se presenta en [12], en donde se muestra un enfoque adaptativo para el control de movimiento y fuerza de múltiples brazos robóticos coordinados. En esta ley de adaptación se utiliza la teoría de hiperestabilidad de Popov para estimar en línea los parámetros inciertos de múltiples brazos y carga útil, éste enfoque se utiliza para controlar el movimiento de un objeto sostenido por los brazos, las fuerzas de contacto en el objeto y las fuerzas internas. Los parámetros desconocidos de los múltiples brazos robóticos coordinados y el objeto se estiman en términos de tres ecuaciones de subsistema de error, las cuales son un subsistema de error de posición, un subsistema de error de fuerza de contacto y un subsistema de error de fuerza interna. Se muestra de igual manera que el esquema de control adaptativo propuesto mejora la posición y la precisión del seguimiento de la fuerza interna y de contacto de una clase de sistemas con un conocimiento incierto del modelo dinámico.

En [13] se realiza una manipulación de un objeto con un sistema cooperativo, realizando un control tanto de movimiento del objeto como de las fuerzas ejercidas por los manipuladores, el enfoque utilizado se denomina formulación simétrica, en él se diseña un control

que utiliza un método híbrido modificado de par calculado, basado en el principio de ortogonalización. Además la masa del objeto se estima calculando la fuerza aplicada por los efectores finales de cada robot para sostener el objeto. Para tratar el problema de controlar robots con restricciones que sostiene un objeto rígido, se definen trayectorias y fuerzas deseadas por cada robot, para ello se retoman trabajos como en [26] y [27], en los cuales proponen el ya conocido control híbrido de posición/fuerza, introduciendo una matriz de selección de cumplimiento para distinguir el control de posición de los componentes de control de fuerza en coordenadas cartesianas; es decir, un apartado para el control de posición y el otro para el control de fuerza, suponiendo en éste método que la dinámica del manipulador es bien conocida y por lo tanto los términos no lineales pueden compensarse.

De igual manera se encuentra en la literatura el trabajo [9], donde se aborda la implementación del control de fuerza en robots industriales utilizando un robot IRB 140 con el controlador S4Cplus M2000 de ABB, montando en el efector final del manipulador un sensor de fuerza/par de 6 grados ATI SI-65-5 conectado al controlador del robot a través de una interfaz analógica. El sistema de control del mencionado robot es de arquitectura cerrada, esto quiere decir que no se tiene acceso al código de los algoritmos que están basados en un control de posición en el cual se asume una estrategia PID con información de las posiciones y velocidades medidas y deseadas. De tal manera que si se intenta desviar al robot de su trayectoria mediante la aplicación de fuerzas externas, el control incrementa la acción de los actuadores tratando de mantener la trayectoria. Los comandos se implementan en el controlador del robot en lenguaje RAPID, se desarrollaron funciones para reducir la constante elástica del sistema, implementando la idea del eje libre con la función SoftServo, que sirve para aflojar uno o más ejes que se comportan como si estuvieran soportados por un resorte de rigidez ajustable.

En [10] se presentan conceptos relacionados al control de fuerza en robótica industrial, mostrando una aplicación de control de fuerza basado en lógica difusa. Se proponen dos tipos de controles explícitos. El control explícito directo, el cual está basado en fuerza, y el control explícito indirecto basado en posición. Por otro lado para poder compensar la incertidumbre del ambiente, resultante de los modelos de contacto imprecisos o simplificados y de la formulación dinámica de contacto adoptada. Se opta por utilizar el controlador basado en lógica difusa, de esta forma no es necesario recurrir a modelos matemáticos detallados y así el control es desarrollado en base a reglas heurísticas cuyo establecimiento se basa en la experiencia humana. Así los controladores difusos son más robustos a pequeñas variaciones de los parámetros del sistema.

En [17] se presenta un estimador de la fuerza de contacto consistente en la integración de la información de sensores de fuerza, posición y aceleración, se describe un procedimiento de calibración automático llamado "plug-and-play" para la identificación y ajuste de los parámetros de este observador, se realizan trabajos experimentales con un robot industrial ABB con arquitectura abierta, se propone un observador dinámico de la fuerza de contacto con propiedades de filtrado, en el que para conseguir una estimación apropiada de la fuerza de contacto, las ganancias deberían ser lo suficientemente elevadas para atenuar el ruido del sistema. Para el proceso de auto-calibración se utiliza una secuencia predefinida de movimientos que permitan estimar el conjunto de parámetros utilizado por el observador de la fuerza de contacto.

## 1.2. Motivación

La robótica industrial es actualmente uno de los campos más importantes en el desarrollo de la industria moderna. La demanda social es alta, y la competencia por satisfacerla es agresiva. Las empresas industriales que manejan procesos de producción, necesitan adaptar mecanismos modernos a los sistemas para maximizar sus operaciones y ahorrar en cuanto a tiempo de fabricación y costos de operación, para poder obtener mejores utilidades y competir dentro del mercado global.

El mercado internacional se caracteriza por una ardua competencia empresarial, incluso los países que son denominados del tercer mundo, compiten con su mano de obra ante la falta de tecnología para el desarrollo industrial. [15], refiere que: esta situación está acorde o incluso por encima de la situación económica de España y demuestra la importancia de la robótica en la economía española. Este hecho nos lleva a la conclusión que nuestras empresas deben apostar por los robots para poder competir en productividad y calidad con los nuevos competidores emergentes del tercer mundo que, actualmente lo único que ofrecen es mano de obra barata. La robotización y la automatización, el conocimiento de tecnologías, y la formación de ingenieros en automatización son la clave de la supervivencia de muchas de nuestras empresas.

La apuesta por la robotización en los procesos industriales y el contar con personal capacitado en el área, tales como los ingenieros, son presentadas como una necesidad para el mejoramiento de los sistemas de producción. Las industrias deben buscar la innovación en los procesos de producción, lo que significa conocer a la competencia, las características del mercado y la población objetivo para llevar los productos. Gran parte de los países desarrollados, han invertido en la robótica industrial como una alianza cooperativa entre el ser humano y el robot, para asistir y mejorar los procesos de fabricación y atención técnica.

En [15], presenta a España como una nación ubicada en los primeros diez puestos en el desarrollo de programas de robotización industrial, sin embargo, países como Estados Unidos y Canadá, han apostado fuertemente a la industria robótica como cooperación para exportar asistencia y producción al mercado mundial. Canadá es un país que por excelencia se puede denominar como pionero en este campo, pues la robótica hace parte importante del proceso de las empresas para poder producir y brindar asistencia técnica al mercado objetivo a quienes exportan.

Esta motivación del desarrollo de la robótica industrial para las empresas y para los estados, es afín al concepto de la misma. Para poder entender la importancia o la motivación de su desarrollo, es necesario primeramente entender la base conceptual. En [25], refiere sobre la robótica que: se considera la robótica como la ciencia que estudia la evolución, el diseño, la construcción y aplicaciones de los robots, los cuales son dispositivos mecánicos dotados de articulaciones móviles, destinados a efectuar una manipulación o un proceso continuo y se encuentra controlado por un ser humano o por un sistema lógico, que en muchos casos puede ser un computador.

Partiendo de la definición propia de la robótica, se encuentran frases que definen el objeto de la relación entre el individuo y el robot, y el foco de estudio del presente trabajo. La robótica se concibe entonces como la ciencia que busca diseñar, construir y evolucionar las

herramientas tecnológicas que cuentan con articulaciones para la realización de funciones, las cuales se denominan como robots. Los sistemas de control presentados por la definición son dos: 1) el sistema lógico que es el programa que permite dotar al sistema de unas indicaciones para la realización de tareas, el cual va ligado a lo que se registraba en las décadas de los 50, 60 y 70, y 2) el sistema que es controlado por el ser humano, quien, de forma cooperativa, le indica al robot las actividades, comandos y funciones a realizar.

De esta manera estos argumentos presentados, configuran una base esencial para el desarrollo de la robótica industrial y su importancia dentro de un sistema de producción moderno. El ser humano cada día avanza con los desarrollos de la tecnología, y la relación entre el individuo y el robot es cada vez más cercana. Entender de primera mano la cooperación entre individuo y robot, es el primer paso para transitar a esta nueva era. Se necesita de personal altamente capacitado para que el funcionamiento de los sistemas de producción mejoren y den rentabilidad para la competencia en el mercado internacional.

### 1.3. Objetivos

**Objetivo general:** Implementar un sistema cooperativo conformado por dos robots industriales ABB de arquitectura cerrada, que sea capaz de seguir trayectorias impuestas por un tercer robot o un humano, interactuando directamente en el mismo espacio de trabajo.

**Objetivos específicos:** Partiendo de la base del concepto de la robótica y la importancia que su desarrollo implica para la sociedad y para la industria, se necesita definir los objetivos que son claves para el análisis y la comprensión de este sistema. Debe de entenderse que la robótica moderna implica una serie de factores que son necesarios como la incursión de la internet y las TIC como herramientas para el trabajo y el acceso al desarrollo de la robótica industrial.

- Estudiar el estado del arte de sistemas robóticos y sistemas cooperativos además de conocer estrategias de control que se pueden aplicar a este tipo de sistemas.
- Implementar el control híbrido posición-fuerza/par en el robot industrial con el modelo de impedancia.
- Implementar el control híbrido posición-fuerza/par extendido en el robot industrial con el modelo de impedancia.
- Validar experimentalmente el esquema de control híbrido posición/fuerza implementado en el sistema cooperativo para la tarea de agarre.
- Realizar las tareas de manipulación con el sistema cooperativo compuesto por dos robots para realizar el movimiento traslacional del objeto con el comportamiento reactivo en uno de los manipuladores.
- Implementar el sistema cooperativo con un comportamiento reactivo a la interacción de un tercer robot.

Estos objetivos describen a la perfección el aporte y la meta de la robótica industrial para mostrar el desarrollo y evolución de la misma. Los robots se describen como herramientas

de trabajo, los cuales son empleados cuando se requiere una gran precisión.

Para terminar con este apartado, los objetivos van orientados al concepto general que parte de la robótica, y a su función dentro del campo en el que se desempeñen, a partir de lo anterior, es que se puede analizar el impacto y los efectos en que la cooperación entre el humano y el robot resulta ser determinante para el mundo moderno.

## 1.4. Contribuciones

Para empezar, se estudiará la implementación del control de fuerza en los robots industriales. El control de fuerza fundamenta su posición en las trayectorias en las que el robot ha de moverse para poder realizar su actividad, las cuales, por lo general, son definidas a través del controlador. En [29], refieren que: los movimientos que realiza un robot industrial, resultan de la ejecución de un programa de usuario por el controlador del robot. El programa define la posición y orientación de los puntos hacia los que el robot debe moverse y el tipo de trayectoria a seguir. Este método funciona muy bien para aplicaciones en que las trayectorias son totalmente predecibles. Sin embargo, muchas aplicaciones involucran cierto grado de incertidumbre en las posiciones/orientaciones y movimientos. En estas aplicaciones, pueden utilizarse entradas de sensores para proveer información al robot de manera que pueda adaptarse a las condiciones cambiantes.

El control de fuerza y la interacción que realiza el individuo para el control o coordinación de las tareas, se basan en las especificaciones propuestas al robot. El controlador es la base por medio del cual se determina la trayectoria y el cambio de la misma, y con ello se determina la fuerza y la certeza para la realización de determinada función.

En resumen, la aportación de este tema de tesis es la implementación del control de fuerza en robots industriales para manipulación de un objeto; de ésta forma, y aprovechando el comportamiento reactivo de los robots, poder generar tareas de manipulación traslacional y rotacional dadas por el humano, interactuando directamente en su espacio de trabajo de los manipuladores.

## 1.5. Organización

La estructura del presente trabajo de tesis explica en el Capítulo 1 el estado del arte, en el cual se describen trabajos que sirven de apoyo como un antecedente para la elaboración de dicho trabajo, se muestra la motivación de realizar simulaciones que validen el trabajo, y las contribuciones que se realizan en este campo de la robótica.

En el Capítulo 2 se detallan los modelos dinámicos y cinemáticos de los manipuladores, así como del sistema cooperativo, se generaliza el modelo de la tarea de agarre y el modelo dinámico del objeto a manipular, de igual forma se tiene el modelo matemático para realizar las simulaciones con el entorno y así obtener las fuerzas debidas a la interacción con el entorno.

El tercer capítulo muestra los esquemas de control que se implementan en las simulaciones, y un análisis de estabilidad en cuanto al modelo dinámico de los actuadores. Se agrega como un esquema extra el control híbrido posición-fuerza/par extendido para conocer el comportamiento del manipulador en la plataforma virtual RobotStudio.

En el capítulo 4 se presenta una breve explicación de RobotStudio y como crear una estación nueva para poder realizar las simulaciones pertinentes, así como la descripción del robot manipulador IRB 2400, se explica como realizar la comunicación de dos o mas controladores virtuales para la sincronización, y la creación de E/S virtuales. Por último se explica brevemente los sensores que generalmente se usan en la Sección de Mecatrónica, mismos que a causa del COVID-19 no se utilizarón al no estar en los laboratorios del Cinvestav, y en vez de ellos se diseña una herramienta desde RobotStudio que será instalada en la brida del robot y se menciona el cambio que se hace al integrar un tercer robot al sistema cooperativo.

Los resultados de las simulaciones se muestran en el Capítulo 5, para ello se describen diferentes casos de estudio, las condiciones y los pasos que se llevaron a cabo para cada experimento.

Por último se tienen en el Capítulo 6 las conclusiones realizadas a partir de los resultados obtenidos, y los alcances de dicho trabajo de tesis. De igual manera se mencionan propuestas de trabajos futuros tomando como antecedente el presente trabajo de tesis.

# Capítulo 2

## Modelos matemáticos

En el presente capítulo se presentan los modelos cinemáticos y dinámicos que describen el comportamiento del sistema cooperativo con interacción humana, así como el modelo de impedancia para la interacción con el entorno y el objeto a manipular.

Se puede entender a los modelos matemáticos como una descripción o representación de un fenómeno en la que intervienen una serie de procesos y factores que condicionan la forma en la que se construye un modelo final, por lo que son tan importantes para el avance en la simulación de lo que ocurre en la realidad.

### 2.1. Modelado del entorno

El problema principal cuando el robot comienza a realizar tareas de manipulación de objetos es el control de fuerzas y pares que se generan por el contacto. En otras palabras, el entorno impone restricciones en las trayectorias programadas al robot en su efector final. De tal manera que al implementar un control de posición puro para controlar la interacción robot-entorno éste seguramente fallará. Por tal motivo, se busca minimizar las desviaciones ocasionadas por el entorno, al tener este tipo de interacción surgen fuerzas de reacción con las cuales se logra minimizar las desviaciones aumentando la deformación sobre la superficie sobre la cual el robot trabaja.

En lo que refiere al modelo del ambiente o entorno se pueden clasificar en dos grupos:

Entorno rígido (restringido): se refiere a la interacción mecánica entre el robot y su entorno, este incluye los modelos de impacto, fricción de Coulomb y restricciones asociadas.

Entorno dócil (compliant): se refiere a un modelo simple pero significativo del contacto, por esto es considerado un entorno elástico dócil el cual se describe por el modelo descrito en (2.1).

En [38] Hogan concluye el modelado del entorno como un sistema masa-resorte-amortiguador generalizado, teniendo la primer aproximación al modelado de la interacción robot-entorno.

$$h_R = M(\ddot{X} - \ddot{X}_s) + B(\dot{X} - \dot{X}_s) + K(X - X_s) \quad (2.1)$$

Donde:

$h_R \in \mathbb{R}^{6 \times 1}$ : Vector de fuerzas y pares de reacción.

$K \in \mathbb{R}^{6 \times 6}$ : Rigidez de la superficie la cual representa una relación entre fuerza y deformación.

$M \in \mathbb{R}^{6 \times 6}$ : Matriz de masas.

$B \in \mathbb{R}^{6 \times 6}$ : Matriz de amortiguamiento del ambiente.

$X_s \in \mathbb{R}^{6 \times 1}$ : Vector de posición de la superficie hasta antes de la deformación.

$X \in \mathbb{R}^{6 \times 1}$ : Vector de posición del efector final.

Con:

$$h_R = [F_R \quad T_R]^T$$

Donde:

$$F_R = \begin{bmatrix} F_{R_x} \\ F_{R_y} \\ F_{R_z} \end{bmatrix}$$

$$T_R = \begin{bmatrix} T_{R_x} \\ T_{R_y} \\ T_{R_z} \end{bmatrix}$$

Donde  $F_{R_x}, F_{R_y}, F_{R_z}$  representan las componentes cartesianas de la fuerza que se efectúa en el efector final y  $T_{R_x}, T_{R_y}, T_{R_z}$  representan los momentos en el efector final.

El modelo de impedancia está basado en la respuesta dinámica del sistema robótico ante la presencia de fuerzas de contacto, de acuerdo con esto, la respuesta está caracterizada por tres parámetros la rigidez, el amortiguamiento y la inercia. La sintonía de dichos parámetros puede llevar al sistema a tres tipos de comportamientos: sub-amortiguado, críticamente amortiguado y sobre-amortiguado, y en función de esto el robot puede comportarse, tal como se realiza el saludo entre seres humanos, de una manera dócil o brusca.

Finalmente el modelo anterior dado por la ecuación (2.1) relaciona la fuerza y posición, que está dada por la impedancia mecánica, para describir y poder simplificar esta ecuación se deben mencionar los términos impedancia y admitancia, se menciona en [3] que son términos equivalentes e intercambiables para sistemas lineales que operan en frecuencia finita. Sin embargo, la manipulación (interacción mecánica) es un problema no lineal y por lo tanto la equivalencia para sistemas lineales no se conserva. El término impedancia es un concepto generalmente usado en ingeniería eléctrica, la cual se define como la razón entre el voltaje y el flujo de corriente alterna, en otras palabras, es la relación entre la fuerza (diferencia de potencial) y el flujo de corriente y por otra parte el inverso de la impedancia se le



denomina admitancia.

Estos conceptos tienen analogías en otro tipo de sistemas, como por ejemplo, el sistema mecánico con la relación fuerza/velocidad llamada impedancia mecánica.

De esta manera realizando una analogía de un sistema eléctrico a un sistema mecánico y tomando en consideración que el robot se desplaza a baja velocidad y se tiene un ambiente no viscoso, el modelo correspondiente a la ecuación (2.1) se puede reescribir como en la ecuación (2.2)

$$h_R = K(X - X_s) \quad (2.2)$$

Siendo esta última la ecuación equivalente a un sistema puramente capacitivo, el cual será utilizado para el presente trabajo.

## 2.2. Modelo del Robot Manipulador

A continuación se presenta el modelo cinemático directo, el modelo cinemático inverso y el diferencial, así como como la dinámica de movimiento libre y restringido.

### 2.2.1. Modelo cinemático

La cinemática directa se entiende como el estudio de movimiento de robots manipuladores, relacionando las coordenadas cartesianas en función de las coordenadas generalizadas o articulares según se explica en [3].

El estudio de la cinemática directa de los robots manipuladores proporciona elementos para analizar sus trayectorias y el desplazamiento del robot, el posicionamiento del robot en el espacio tridimensional requiere de seis coordenadas, tres de ellas para definir la posición cartesiana y tres para la orientación, a la relación que existe entre las coordenadas cartesianas y las articulares se conoce como cinemática directa, las articulaciones pueden producir movimientos lineales o giratorios, a estas se les denomina articulaciones prismáticas y rotacionales respectivamente, para obtener el modelo cinemático se puede hacer uso de transformaciones homogéneas, sin embargo el método usado comúnmente es la metodología Denavit-Hartenberg, el cual hace uso de las longitudes y ángulos para establecer de manera concreta el modelo matemático de las transformaciones homogéneas, obteniendo de esta manera el modelo cinemático directo. Se pueden encontrar diferentes trabajos que explican otras metodologías como en [3], [4], [8], [11].

De tal manera que la dimensión es  $X \in \mathbb{R}^{6 \times 1}$  y  $q \in \mathbb{R}^{n \times 1}$ , donde  $n$  representa el número de grados de libertad.

$$X = F_{CD}(q) \quad (2.3)$$

De esta forma la cinemática directa convierte las coordenadas articulares a coordenadas cartesianas, a través de una función vectorial como:

$$[x \ y \ z \ \theta \ \phi \ \psi]^T = X \quad (2.4)$$

Donde  $q \in \mathbb{R}^n$  representa el vector de posiciones articulares del robot,  $\theta, \phi, \psi \in \mathbb{R}$  representan la orientación final del robot y  $[x, y, z]$  representan las coordenadas cartesianas.

Por otra parte la cinemática inversa relaciona las coordenadas articulares del robot en función de las coordenadas cartesianas.

$$q = F_{CI}(X) \quad (2.5)$$

Por último se tiene la cinemática diferencial, que se encarga de establecer la relación que existe entre velocidades lineales y angulares del efector final, con velocidades articulares  $\dot{q}$ , a esto se le conoce generalmente como el Jacobiano  $J(q)$ .

$$J(q) = \begin{bmatrix} J_v(q) \\ J_w(q) \end{bmatrix} \quad (2.6)$$

Donde  $J_v(q) \in \mathbb{R}^{3 \times n}$  relaciona la velocidad articular  $\dot{q} \in \mathbb{R}^n$  con la velocidad lineal  $v \in \mathbb{R}^3$ , mientras que  $J_w(q) \in \mathbb{R}^{3 \times n}$  relaciona la velocidad angular  $w \in \mathbb{R}^3$  con la velocidad articular  $\dot{q} \in \mathbb{R}^n$ .

$$\dot{X} = J(q) \quad (2.7)$$

El Jacobiano sirve no solo para mapear dichas velocidades, también es útil para encontrar configuraciones singulares, y la más importante para este tema de tesis es describir la relación existente entre las fuerzas aplicadas y los pares resultantes  $F_e$  del efector final del robot, debidos a la interacción con el entorno.

$$\tau_R = J^T(q) h_R \quad (2.8)$$

$h_R$  se compone de un vector de fuerzas  $F \in \mathbb{R}^{3 \times 1}$  y  $T \in \mathbb{R}^{3 \times 1}$  correspondiente al vector de momentos y los pares aplicados en las articulaciones  $\tau \in \mathbb{R}^{n \times 1}$ .

### 2.2.2. Modelo dinámico

A continuación, se presenta el modelado dinámico de robots manipuladores el cual se puede obtener mediante diferentes métodos, el más representativo es por medio de las ecuaciones de movimiento de Euler-Lagrange, debido a las propiedades matemáticas que se deducen de manera natural que facilita el análisis y diseño de algoritmos de control. La dinámica establece la relación entre las fuerzas y pares aplicados en los actuadores y el movimiento del manipulador, actualmente se tiene bastante información para el desarrollo de la dinámica de robots manipuladores, se puede encontrar en trabajos como en [3], [4], [40]

En el área de simulación el modelo dinámico es la parte clave debido a que se puede reproducir todos los fenómenos físicos del robot sin la necesidad de tener un robot real, de igual manera resulta útil para evaluar estrategias de control y planeación de trayectorias.

**Dinámica de movimiento libre:** Es el modelo utilizado cuando el manipulador no tiene contacto con ningún entorno, dicho de otra manera, no existen fuerzas externas que impidan o modifiquen su movimiento, la ecuación que describe su comportamiento es la ecuación (2.9).

$$\tau_a = D(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) \quad (2.9)$$

**Dinámica de movimiento restringido:** Este modelo representa al manipulador cuando éste tiene contacto con el entorno, de esta manera se generan fuerzas y pares que modifican el movimiento del manipulador, observese la ecuación (2.10).

$$\tau_a + \tau_R = D(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) \quad (2.10)$$

El modelo dinámico en (2.10) representa la base matemática para analizar los fenómenos físicos presentes en la estructura de un robot manipulador de  $n$  grados de libertad con eslabones rígidos. Se tiene  $\ddot{q} \in \mathbb{R}^n$  que representa vector de aceleraciones articulares,  $D(q) \in \mathbb{R}^{n \times n}$  es la matriz de inercias simétrica y definida positiva que representa el cambio de estado de movimiento del robot manipulador,  $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$  es la matriz de fuerzas centrípetas y de Coriolis, las fuerzas centrípetas son fuerzas radiales y las fuerzas de Coriolis representan la desviación del movimiento de traslación debido a su componente de rotación,  $g(q) \in \mathbb{R}^n$  es el vector de fuerzas gravitacionales,  $\tau_a \in \mathbb{R}^{n \times 1}$ , vector de pares aplicados a cada articulación por los actuadores,  $\tau_R \in \mathbb{R}^{n \times 1}$ , representa el vector de pares generados debido a las fuerzas de reacción en las articulaciones del manipulador .

## 2.3. Sistema cooperativo de robots industriales

Actualmente la robótica avanza de manera rápida, uno de los avances son los robots cooperativos, ya que hay bastantes tareas que un robot no puede realizar o en su caso son demasiado difíciles resolver para un solo robot, la idea fundamental es diseñar sistemas compuestos de varios robots capaces de resolver problemas conjuntamente, esto se hace con la finalidad de reducir costos con sistemas de un solo robot especializado.

### 2.3.1. Modelo dinámico del sistema cooperativo

Al trabajar con sistemas cooperativos resulta importante conocer la dinámica del sistema, en este caso se tiene una dinámica restringida al trabajar con un objeto en común.

Sabemos que la ecuación dinámica restringida de un robot manipulador está dada como en (2.10), por otra parte la ecuación dinámica compacta de dos robots manipuladores que conforman el sistema cooperativo es como en (2.11)

$$\tau_{asc} + J_{sc}^T H_{sc} = D_{sc}(q) \ddot{q}_{sc} + C_{sc}(q, \dot{q}) \dot{q}_{sc} + B_{sc} \dot{q}_{sc} + g_{sc}(q) \quad (2.11)$$

Donde:

$$D_{sc}(q_{sc}) = \text{blockdiag}(D_1, D_2)$$

$$C_{sc}(Q_{sc}, \dot{q}_{sc}) = \text{blockdiag}(C_1, C_2)$$

$$J_{sc} = \text{blockdiag}(J_1, J_2)$$

$$B_{sc} = \text{blockdiag}(B_1, B_2)$$

$$g_{sc}(q_{sc}) = [g_1^T, g_2^T]^T$$

$$\tau_{sc} = [\tau_1^T, \tau_2^T]^T$$

$$h_{sc} = [h_1^T, h_2^T]^T$$

$$q_{sc} = [q_1^T, q_2^T]^T$$

De esta forma, el sistema cooperativo tiene la misma configuración cinemática con  $n$  grados de libertad, este modelo toma en cuenta la interacción con el entorno, en este caso el objeto, cabe mencionar que cada robot ABB tiene su propio marco de referencia, así como el objeto.

### 2.3.2. Humano-sistema cooperativo

Un escenario donde diferentes tipos de robots comparten espacio con los seres humanos, interactuando activamente con ellos, e incluso colaborando en actividades muy propias del quehacer humano, parece ser un futuro cercano, sin embargo, la mayor parte de

los sistemas robotizados que existen en la actualidad carecen de las capacidades necesarias para poder asumir con éxito tan ambiciosa tarea. Hoy en día, las aplicaciones robóticas más comunes están ligadas al sector industrial, y consideran que los robots deben operar alejados de los seres humanos. Por lo tanto y favoreciendo criterios de seguridad y productividad propios del sector, el diseño de los robots industriales no contempla la interacción física directa con humanos.

De esta manera los robots que pretendan interactuar con los seres humanos tendrán que ser diseñados para desenvolverse apropiadamente en un entorno acondicionado física, cognitiva y socialmente por la presencia del hombre. Deberán asumir que el ser humano no sólo es un cuerpo físico, dotado de órganos que le permite cumplir las funciones vitales y que le facilitan el contacto con el entorno; sino que también tiene una mente, que le permite pensar, sentir, recordar e interactuar con otros seres humanos. Un enfoque refiere al estudio del intercambio de acciones físicas entre humano y robot, el cual contempla la coexistencia útil de robots y humanos en un espacio común, donde el contacto físico permanente o eventual se hace presente.

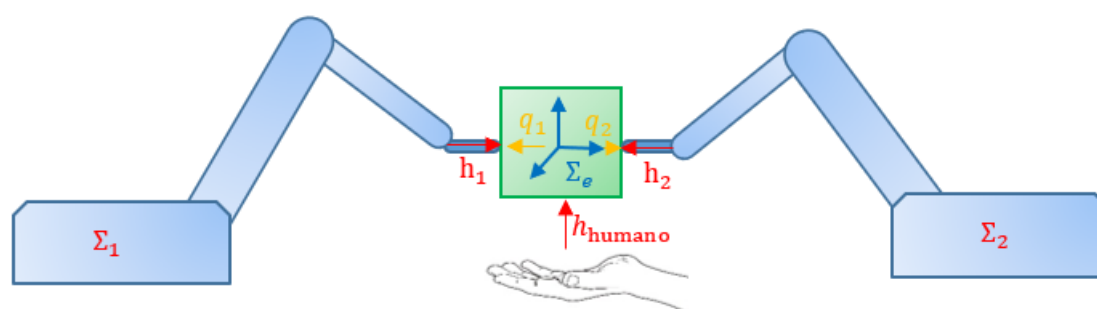


Figura 2.1: Sistema cooperativo humano

En un sistema cooperativo como se observa en la Figura 2.1 donde se contemplan las fuerzas que ejerce cada robot, denotadas como  $h_1$ ,  $h_2$  y  $h_{humano}$  de esta manera se ejerce fuerza sobre el objeto generando el agarre de éste, como se ha mencionado, cada robot ABB tiene su propio marco de referencia local, de igual manera se tiene un marco local para el objeto  $\Sigma_e$ , al utilizar la plataforma RobotStudio se hace uso de los marcos de referencia globales y locales, resulta útil trabajar con un marco global, para de esta manera poder monitorear la posición y orientación del efector final de cada robot, y con esto las fuerzas generalizadas  $h_1$ ,  $h_2$  y  $h_{humano}$ .

De esta manera una investigación como la que se presenta en este trabajo, abre las puertas a nuevos horizontes, tanto para la robótica industrial como para otro tipo de servicios y aplicaciones, donde es posible apreciar robots de manipulación cooperativa para trabajos industriales, robots para el apoyo de actividades domésticas, robots para la asistencia a personas con discapacidades físicas, etc. Todas las posibles aplicaciones comparten la necesaria interacción física, segura y fiable entre robots y humanos como se explica en [16].

## 2.4. Modelo dinámico del objeto

El sistema cooperativo logra interactuar con el objeto a través de las fuerzas que se aplican en los puntos de contacto, por esta razón resulta necesario conocer y establecer la dinámica del objeto como se expresa en [35], la cual está expresada por (2.12)

$$M_e \dot{v}_e^e + C(v_e^e) v_e^e + g_e^e = h_e^e \quad (2.12)$$

Donde:

$$\begin{aligned} M_e &= \begin{bmatrix} m_e I & 0 \\ 0 & J_e^e \end{bmatrix} & v_e^e &= \begin{bmatrix} \dot{p}_e^e \\ w_e^e \end{bmatrix} \\ g_e^e &= \begin{bmatrix} -m_e g_0^e \\ 0 \end{bmatrix} & h_e^e &= \begin{bmatrix} f_e^e \\ \tau_e^e \end{bmatrix} \\ C(v_e^e) &= \begin{bmatrix} m_e S(w_e^e) & 0 \\ 0 & S(w_e^e) J_e^e \end{bmatrix} \end{aligned}$$

$v_e^e$  es el vector de velocidades lineales y angulares del marco de referencia  $\Sigma_e$ ,  $m_e$  la masa del objeto,  $g_0^e$  es el vector de fuerzas gravitacionales,  $J_e^e$  es el tensor de inercia del objeto,  $h_e^e$  el vector de fuerzas generalizadas que ejercen los robots sobre el objeto, dicho de otro modo son las fuerzas externas,  $O$  es la matriz nula,  $I$  denota la matriz identidad,  $S(\cdot)$  es un operador matricial antisimétrico generado por el producto cruz.

### 2.4.1. Base matemática para la tarea de agarre

En [18] se define el fenómeno de contacto como aquel que se establece entre dos cuerpos mediante una fuerza o carga de compresión perpendicular a las superficies de estos, señala también dos relaciones importantes como se menciona a continuación:

- La relación entre la fuerza de carga de compresión y el desplazamiento normal del cuerpo, que a su vez determina la rigidez del contacto y las propiedades dinámicas del sistema.
- La relación entre las tensiones o esfuerzos generados en la zona de contacto y la resistencia mecánica del sistema.

De esta manera, para que exista el fenómeno de contacto, dos o más cuerpos deben estar en constante interacción, de tal manera que para poder mover el objeto desde una posición inicial a una posición final los manipuladores necesitan aplicar fuerza en los puntos de contacto de la superficie, los cuales se mencionan en el Capítulo 5, donde los puntos de contacto se seleccionan de tal manera que al realizar la etapa de agarre el objeto no presente movimientos no deseados, garantizando así el agarre del objeto.

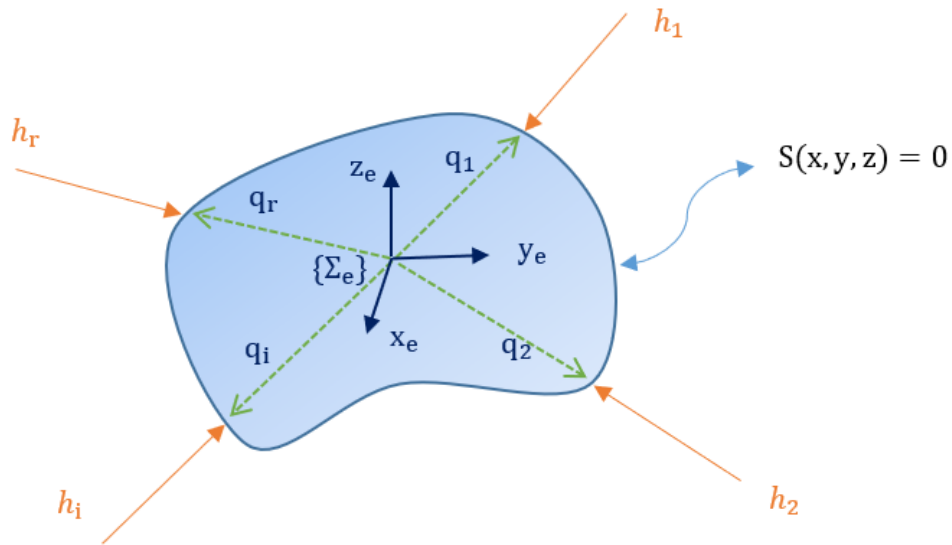


Figura 2.2: Esquema de agarre

Se considera la superficie del objeto como  $S(x, y, z) = 0$  y un marco de referencia  $\Sigma_e$  ubicado en el centro de masa del objeto. Por otra parte el vector  $q_1 = (q_{x1}, q_{y1}, q_{z1})$  con  $r = 1, 2, \dots, r$  que representa la ubicación espacial de los puntos de contacto tomando en cuenta el centro de masa del objeto con  $r$  el número total de puntos. Por último las fuerzas y momentos  $f_i$  y  $\tau_i$  respectivamente denotados como la fuerza generalizada  $h_i$  como se observa en la Figura 2.2 y en la ecuación (2.13).

$$h_i = [f_{xi}, f_{yi}, f_{zi}, \tau_{xi}, \tau_{yi}, \tau_{zi}]^T \quad (2.13)$$

Las fuerzas generalizadas denotadas como  $h_i$  se aplican de manera puntual hacia el centro de masa de la superficie de contacto generando fuerzas de reacción  $f_r$  que provoca movimientos traslacionales y tensiones mecánicas. De manera similar, si las fuerzas antes mencionadas tienen una dirección diferente se produce un par resultante  $\tau_r$ , generando rotaciones sobre el objeto.

# Capítulo 3

## Esquemas de control

En el presente trabajo se muestran los esquemas de control utilizados, como se ha mencionado en el Capítulo 1, los sistemas de control de robots industriales ABB son de arquitectura cerrada, por esta razón no se puede acceder al código de los algoritmos, estos están basados en un control de posición que utiliza la estrategia de control PID o par computado como se menciona en [9].

Se muestran los fundamentos y conceptos necesarios del control de posición, y control de fuerza, así como los esquemas de control utilizados en el desarrollo del presente trabajo, con esto surge el concepto de control de interacción, el cual consiste en la regulación de las fuerzas y/o pares generados durante la ejecución de una tarea en la que el efector final del robot tiene contacto físico con el entorno. El objetivo principal es mantener las fuerzas de contacto con el entorno. Cabe mencionar que el control de fuerza inicia en cuanto el efector final tiene contacto con el entorno, de esta manera al obtener lecturas de fuerza se realiza la retroalimentación.

### 3.1. Control híbrido posición-fuerza/par

La regulación de fuerzas se clasifica en *control directo de fuerza* y *control indirecto de fuerza*, para el primer caso se regula el contacto a un valor de fuerza deseado y constante utilizando un lazo de control de fuerza. Para el segundo caso se regula la interacción indirectamente a través del control de posición y/o velocidad.

En tareas donde el robot interactúa con el entorno éste impone restricciones en las trayectorias. Esta situación se conoce como movimiento restringido, de esta manera una estrategia de control de posición es candidata para fallar, debido a la desviación de la trayectoria de referencia, por esta razón un control de posición no es suficiente para controlar la interacción con el entorno. Por lo tanto, si la trayectoria impuesta al robot se ve afectada por el entorno, surgen fuerzas de reacción, así la ley de control minimiza dicha desviación aumentando en consecuencia la fuerza de contacto y con ellos la deformación de la superficie sobre la que el robot realiza la tarea.

La fuerza de contacto describe la interacción entre el robot y su entorno, de tal manera que, con el control de fuerza se espera una interacción provista de fuerza, de aquí se tiene que para un manejo apropiado de la interacción es necesario considerar estrategias de



control de fuerza. El control tiene como objetivo seguir lo más cerca posible al valor de referencia.

En el esquema de control que se observa en la Figura 3.1 se tiene que, al hacer contacto el robot manipulador con el ambiente se genera una fuerza de reacción, misma que alimenta al control de fuerza/par, obteniendo de este una posición  $X_r$  que nuevamente regresa al control PID de posición.

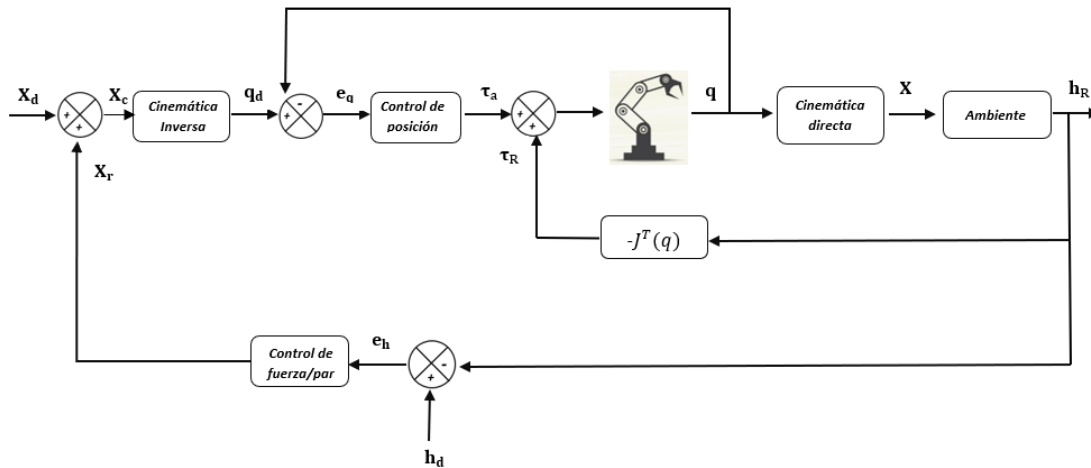


Figura 3.1: Esquema de control híbrido posición fuerza/par

### 3.1.1. Control de posición interno del robot

La tarea principal de un control de posición consiste en determinar los pares aplicados a los servomotores (actuador rotativo que permite un control preciso en términos de posición angular), de tal forma que estos actuadores sigan lo más cercano posible a la posición deseada  $q_d$ , de tal modo que, el objetivo de control se cumple cuando  $\lim_{t \rightarrow \infty} \tilde{q} = 0$  siendo  $\tilde{q}(t)$  el error entre la posición deseada y la actual como en la Figura 3.1

$$\tilde{q}(t) = q_d - q(t) \tag{3.1}$$

La implementación en el control de posición se realiza por medio de variables articulares, enviando primeramente a la cinemática inversa el valor  $X_d$ , ésta se encargará de transformar la posición en términos de variables articulares  $q_d$ . Con la ecuación (3.1) se establece la ley de control PID de posición, la cual, en la mayoría de los casos es la que tienen los robots manipuladores, dicho control se expresa como en la ecuación (3.2)

$$\tau = K_p e_q + K_i \int e_q dt + K_d \dot{e}_q \tag{3.2}$$

Donde:

$K_p \in \mathbb{R}^{n \times 1}$  Matriz diagonal de ganancia proporcional.

$K_i \in \mathbb{R}^{n \times 1}$  Matriz diagonal de ganancia integral.

$K_d \in \mathbb{R}^{n \times 1}$  Matriz diagonal de ganancia derivativa.

$e_q \in \mathbb{R}^{n \times 1}$  Error de posición en variables articulares.

La sintonización de dichas ganancias se realiza de tal manera que permita obtener una respuesta eficiente, en robots industriales estos valores están ya definidos y no hay manera de acceder a ellos, como se mencionó anteriormente son de arquitectura cerrada.

### 3.1.2. Control de posición en robots ABB

Los robots industriales son de arquitectura cerrada, por lo cual no se tiene acceso a los códigos del control de bajo nivel, de tal manera que en el presente trabajo se desarrolla el sistema cooperativo con dos robots industriales, a los cuales se implementará el esquema de control de fuerza para de esta forma generar una trayectoria que el control de posición interno del robot pueda realizar correctamente, cumpliendo el esquema de la Figura 3.2

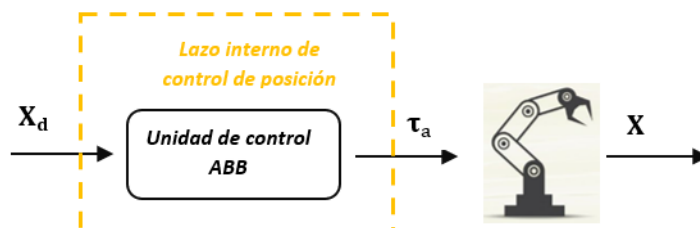


Figura 3.2: Esquema de control de posición en los robots industriales

### 3.1.3. Control de fuerza-par

El control de fuerza es primordial cuando se necesita que el robot interactúe con algún objeto o entorno. Se selecciona un control PI ya que al tener un sensor de fuerza analógico la parte derivativa amplificaría notablemente el ruido que pudiera presentarse en las mediciones. En este tipo de aplicaciones es necesario tener un control de fuerza que sea lo suficientemente preciso de tal manera que permita especificar una fuerza de interacción deseada.

De tal forma que el control de posición (PID) está activo en todo momento, y el control de fuerza es conocido como el lazo de control externo del esquema de control híbrido, según el modelo de impedancia propuesto por Hogan en [38], cuando el robot manipulador está en contacto con una superficie u objeto se activa el control de fuerza, siendo así que el

efector final realiza una deformación sobre la superficie, la cual corresponde a una fuerza de reacción como se observa en la ecuación (2.2) donde se muestra un modelo de impedancia capacitivo.

El control PI de fuerza es alimentado por el error de fuerzas generalizadas, como se muestra en la ecuación (3.3).

$$e_h = h_d - h_R \quad (3.3)$$

La ley de control se puede expresar como en (3.4):

$$X_r = K_{ph}e_h + K_{ih} \int e_h dt \quad (3.4)$$

Donde:  $e_h \in \mathbb{R}^{6 \times 1}$  es el error de fuerza y par,  $K_{ih} \in \mathbb{R}^{6 \times 6}$  representa la matriz diagonal de ganancias integrales,  $K_{ph} \in \mathbb{R}^{6 \times 6}$  representa la matriz diagonal de ganancias derivativas.

La ley de control entrega a la salida  $x_r$ , la cual se interpreta como una posición, ésta posición sumada con la posición deseada, da como resultado una posición comandada como en (3.5), representando de esta manera la posición y orientación del efector final del robot manipulador.

$$X_c = X_d + X_r \quad (3.5)$$

El lazo de control de fuerza permite realizar la etapa de agarre, ya que al enviar al control PI el error de fuerza en cada instante de tiempo permite actualizar constantemente el valor de  $X_c$ , esto se puede apreciar en la Figura 3.1, donde después de la interacción con el ambiente se tienen fuerzas de reacción, mismas que al obtener el error con un valor de referencia se obtiene el error de fuerza que alimenta en cada instante de tiempo al control de fuerza, y este a su vez entrega a la salida una trayectoria que será seguida por el control de posición.

### 3.2. Control híbrido posición-fuerza/par en robots ABB

Al no tener acceso a los algoritmos de control de los robots industriales se añade un lazo de control externo, el cual tiene la tarea principal de agregar una trayectoria la cual se envía al control interno PID del manipulador, de esta manera el esquema de la Figura 3.1 se puede generalizar como se muestra en la Figura 3.3.

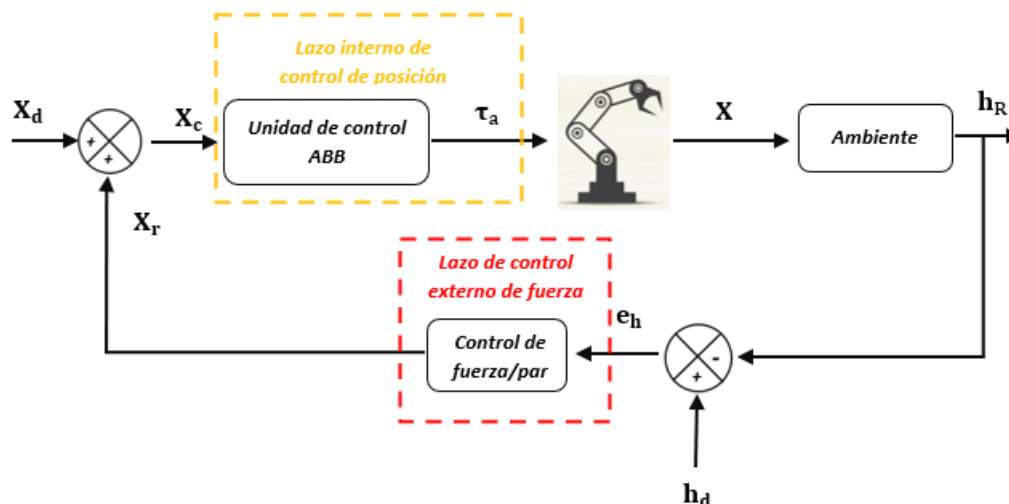


Figura 3.3: Esquema de control híbrido posición-fuerza/par para robots industriales ABB

En base al esquema anterior se puede observar que el control híbrido posición-fuerza/par se compone de un lazo de control interno del manipulador, y el lazo de control externo de fuerza/par. De esta forma es posible realizar tareas en las que el robot manipulador pueda ejercer alguna trayectoria de fuerza y el control interno del ABB realiza la tarea de posicionamiento.

### 3.3. Control híbrido posición-fuerza/par extendido.

Cuando se implementa el esquema de control híbrido posición/fuerza-par en ambos manipuladores se entiende que la referencia de fuerza es constante y de igual manera la de posición. De esta manera se genera el agarre del objeto, así el objeto no presenta cambios en su posición y orientación, esto quiere decir que el sistema se encuentra en equilibrio estático.

Como se menciona en [23] se puede realizar trayectorias dentro del control de posición interno del robot, pero al existir un control de fuerza externo se prioriza la etapa de control externo, por lo que intentar programar una trayectoria más después de la etapa de agarre no se obtendrían los resultados esperados, por esta razón se propone un lazo de control de posición que, dependiendo de la trayectoria deseada, este entrega a la salida un vector de fuerza/par ( $h_{PI}$ ), que sumado con el valor de referencia de fuerza deseada ( $h_d$ ) se obtiene un comportamiento diferente del manipulador en su efector final.

Si existe una desigualdad en la magnitud de las fuerzas aplicadas por los manipuladores se generan fuerzas y pares externos ( $h_r$ ) en el objeto, lo cual se refleja como un desplazamiento o rotación de éste.

$$h_{PI} = K_{pPI}e_{PI} + K_{iPI} \int e_{PI} dt + K_{dPI}\dot{e}_{PI} \quad (3.6)$$

Se utiliza un control PID de posición, el cual se expresa como en la ecuación (3.6). Donde  $K_{pPI}, K_{iPI}, K_{dPI} \in \mathbb{R}^{6 \times 6}$  son matrices diagonales de ganancias proporcional, integral y derivativa respectivamente,  $e_{PI}$  es el error de posición dado por (3.7)

$$e_{PI} = \tilde{X}_d(t) - X \quad (3.7)$$

Donde  $\tilde{X}_d(t)$  representan la posición y orientación deseadas del control de posición-I y  $X$  representa la posición y orientación del efector final.

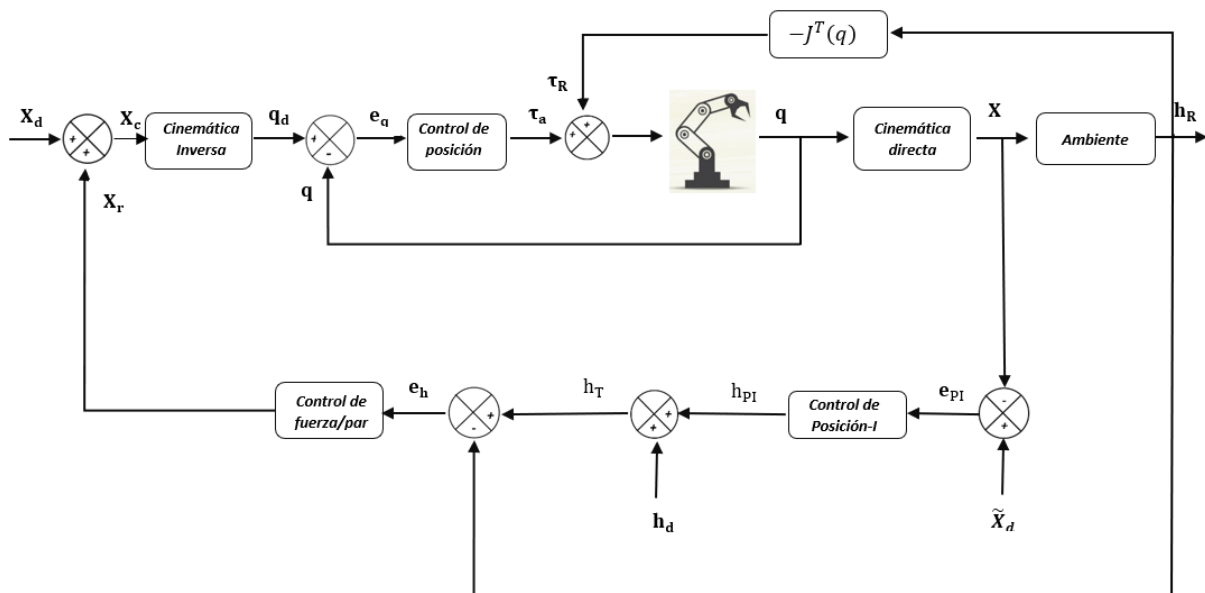


Figura 3.4: Esquema de control híbrido posición fuerza/par extendido

De esta manera el esquema de control que se muestra en la Figura 3.4 se implementa en un solo robot manipulador, mientras que al otro robot se implementa el esquema de la Figura 3.1

### 3.4. Análisis de estabilidad en lazo cerrado

En la presente sección se muestra el análisis del sistema en lazo cerrado. En trabajos como en [23] se desarrolla un análisis del diagrama a bloques con un sistema teleoperado de

múltiples entradas. Los manipuladores que se tienen en la sección y con los que se han realizado las simulaciones son de arquitectura cerrada. De esta manera se analiza la estabilidad del manipulador tomando en cuenta la dinámica de los actuadores.

### 3.4.1. Modelo dinámico de los actuadores

El modelo dinámico de un robot manipulador se compone por una parte del modelo de su estructura mecánica, que relaciona su movimiento con las fuerzas y pares que lo originan, y por otro lado con el sistema de accionamiento, que relaciona las órdenes de mando generadas en la unidad de control con las fuerzas y pares utilizados para generar movimiento en los manipuladores.

La ecuación que describe la dinámica no lineal de un manipulador se describe en la ecuación (2.10), la cual es válida si se considera que los eslabones son rígidos, es decir que no incluye el fenómeno de elasticidad, por esta razón es necesario obtener un modelo que relacione el voltaje  $v$  de entrada con los torques generados por los motores  $\tau$ .

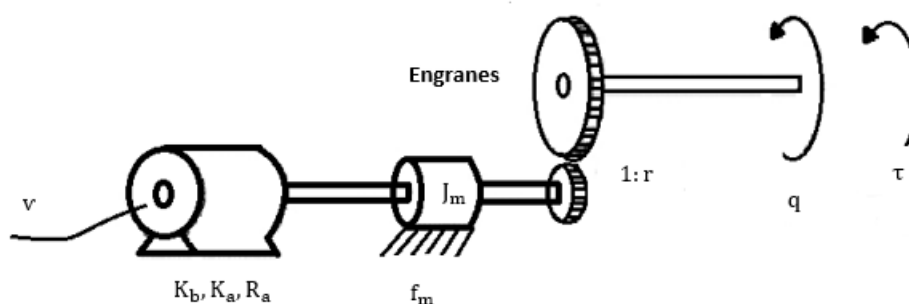


Figura 3.5: Diagrama de motor DC.

Si se considera que cada articulación del robot manipulador contiene motores de corriente directa que convierten la energía eléctrica en energía mecánica, y se tiene que el diagrama de la Figura 3.5 representa el modelo dinámico lineal correspondiente a la ecuación (3.8)

$$J_m \ddot{q} + f_m \dot{q} + \frac{K_a K_b}{R_a} \dot{q} + \frac{\tau}{r R_a} = \frac{K_a}{r R_a} v \quad (3.8)$$

Donde:

$J_m$ , Inercia del rotor [ $kg \cdot m$ ]

$K_a$ , Constante de torque [ $N \cdot m / A$ ]

$R_a$ , Resistencia de armadura [ $\Omega$ ]

$K_b$ , Constante de fuerza contra-electromotriz [ $V \cdot s / rad$ ]

$f_m$ , Coeficiente de fricción del rotor [ $N \cdot m$ ]

$r$ : Relación de reducción del engranaje  $r \ll 1$

Entonces si partimos de la idea dónde cada articulación se acciona por medio de un actuador se tiene lo siguiente:

$$J\ddot{q} + B_a\dot{q} + R\tau = KV \quad (3.9)$$

Donde cada motor desde  $i = 1, \dots, n$  forman bloques de matrices diagonales como se observa a continuación.

$$J = \text{diag}\{J_{m_i}\} \quad (3.10)$$

$$B_a = \text{diag}\left\{f_{m_i} + \left(\frac{K_a K_b}{R_a}\right)_i\right\} \quad (3.11)$$

$$R = \text{diag}\left\{\frac{1}{r_i^2}\right\} \quad (3.12)$$

$$K = \text{diag}\left\{\left(\frac{K_a}{R_a}\right)_i \frac{1}{r_i}\right\} \quad (3.13)$$

Despejando  $\tau$  de la ecuación (3.9), obtenemos la ecuación (3.14).

$$\tau = \frac{KV - B_a\dot{q} - J\ddot{q}}{R} \quad (3.14)$$

Tomando el modelo completo del manipulador, la fricción y la dinámica de los manipuladores en cada articulación y sustituyendo el respectivo valor de  $\tau$  de la ecuación (3.14) en la ecuación (2.11) obtenemos la ecuación (3.15)

$$\frac{KV - B_a\dot{q} - J\ddot{q}}{R} + J_{sc}^T H_{sc} = D_{sc}(q)\ddot{q}_{sc} + C_{sc}(q, \dot{q})\dot{q}_{sc} + B_{sc}\dot{q}_{sc} + g_{sc}(q) \quad (3.15)$$

Agrupando términos, se obtiene la ecuación (3.16)

$$(RD_{sc}(q) + J)\ddot{q}_{sc} + RC_{sc}(q, \dot{q})\dot{q}_{sc} + (RB_{sc} + B_a)\dot{q}_{sc} + Rg_{sc}(q) = KV + RJ_{sc}^T(q)H_{sc} \quad (3.16)$$

Sabemos que existen determinados sistemas reductores usados en robots industriales, esto se debe a que los reductores utilizados en robótica necesitan condiciones de funcionamiento muy exigentes, esto es por la alta precisión y velocidad de posicionamiento que

ofrecen o se exige al robot manipulador. De esta manera, al utilizar robots industriales de arquitectura cerrada se asume que, la relación de reducción del tren de engranaje es más grande que uno  $r_i \gg 1 \forall i$ , entonces la matriz  $R$  se aproxima a cero ( $R \approx 0$ ) como se puede ver en la ecuación (3.12), por esta razón se puede escribir la ecuación (3.16) como en la ecuación (3.17).

$$J\ddot{q}_{sc} + B_a\dot{q}_{sc} = KV \quad (3.17)$$

Como se ha mencionado anteriormente en los robots manipuladores de arquitectura cerrada se asume un control PID, por esta razón se considera lo siguiente.

**Suposición 1:** Las ganancias  $K_p$ ,  $K_i$  y  $K_d$  son matrices diagonales de ganancias definidas positivas, las ganancias proporcional, integral y derivativa respectivamente han sido sintonizadas de tal manera que el error de seguimiento de trayectoria variante en el tiempo  $q_d(t)$  es prácticamente cero  $e_q \approx 0$  y en consecuencia  $\dot{e}_q \approx 0$ . De manera similar para una trayectoria constante  $q_d$  se tiene  $e_q = 0$  y  $\dot{e}_q = 0$ . Como se explica en [22].

$$V_{PID} = K_p e_q + K_d \dot{e}_q + K_i \int e_q dt \quad (3.18)$$

Si se sustituye la ecuación (3.18) correspondiente al esquema del control PID interno del manipulador en la ecuación (3.17) se obtiene la ecuación (3.19).

$$K \left[ K_p e_q + K_d \dot{e}_q + K_i \int e_q dt \right] = J\ddot{q}_{sc} + B_a\dot{q}_{sc} \quad (3.19)$$

Al desarrollar la ecuación (3.19) se obtiene la ecuación (3.20).

$$\tilde{K}_P e_q + \tilde{K}_D \dot{e}_q + \tilde{K}_I \int e_q dt = J\ddot{q}_{sc} + B_a\dot{q}_{sc} \quad (3.20)$$

Con:

$$\tilde{K}_P = K_p K$$

$$\tilde{K}_D = K_d K$$

$$\tilde{K}_I = K_i K$$

**Observación:** En la ecuación (3.20) se observa que para una alta reducción del tren de engranaje, las no linealidades de la dinámica del robot llegan a ser no solo aproximadamente lineales, sino que también desacopladas. El vector  $q$  en la misma ecuación se encuentra restringido por la fuerza  $h$  a pesar de no aparecer explícitamente. La utilidad de la ecuación



(3.20) se basa en el hecho de que permite implementar una acción de control de fuerza de forma indirecta solo eligiendo adecuadamente las trayectorias de posición deseadas [28]

Ahora bien, en trabajos como [22] y [24] se ha realizado el análisis de estabilidad del sistema cooperativo en dos casos, el primero para superficies deformables y el segundo para superficies rígidas, en el presente trabajo se realizará solamente para el caso deformable, ya que el modelo usado, que se muestra en el Capítulo 2 es el modelo de impedancia capacitiva, este modelo es el utilizado generalmente para superficies deformables y el que se utilizó para el presente trabajo.

El modelo de impedancia se describe en la ecuación (2.2), el modelo para una trayectoria deseada se describe en la ecuación (3.5), y por último el control PI de fuerza está descrito por la ecuación (3.4). Según la Suposición 1 se establece que  $\Delta x = X_c - X \approx 0$  logrando de esta manera un pequeño error. De esta manera obtenemos la ecuación (3.21).

$$X_d + X_r - X \approx 0 \quad (3.21)$$

Si despejamos y sustituimos la ecuación (3.4) correspondiente al control PI en la ecuación (3.21) y se multiplica en ambos lados por el término  $K_R$  que viene dado del modelo de impedancia obtenemos la ecuación (3.22).

$$\begin{aligned} X - X_d &\approx X_r \\ h_r &= X - X_d \approx K_{ph}e_h + K_{ih} \int e_h dt \\ h_r &= K_R (X - X_d) \approx K_R \left( K_{ph}e_h + K_{ih} \int e_h dt \right) \end{aligned} \quad (3.22)$$

Por otro lado, sabemos que el error está dado por la ecuación (3.23).

$$e_h = h_d - h_r \quad (3.23)$$

Entonces la ecuación (3.23) se puede reescribir como en la ecuación (3.24)

$$e_h \approx h_d - K_R K_{ph} e_h + K_R K_{ih} \int e_h dt \quad (3.24)$$

De acuerdo a lo anterior se tiene:

$$\begin{aligned}\dot{e}_h + K_R K_{ph} \dot{e}_h &\approx \dot{h}_d - K_R K_{ih} e_h \\ (1 + K_R K_{ph}) \dot{e}_h &\approx \dot{h}_d - K_R K_{ih} e_h\end{aligned}\quad (3.25)$$

De la ecuación (3.25) se tiene  $K_R$ ,  $K_{ph}$ ,  $K_{ih}$  matrices diagonales, además la ecuación antes mencionada puede descomponerse en  $n$  ecuaciones de la forma siguiente.

$$(1 + k_{rj} k_{phj}) \dot{e}_{hj} \approx \dot{h}_{dj} - k_{rj} k_{ihj} e_{hj} \quad (3.26)$$

Donde para  $j = 1, \dots, n$ , para el caso tridimensional, considerando que  $n = 6$  ya que  $h \in \mathbb{R}^{6 \times 1}$ , de igual manera  $\dot{e}_{hj}$ ,  $e_{hj}$ ,  $\dot{h}_{dj}$  indican el  $j$ -ésimo elemento de  $\dot{e}_h$ ,  $e_h$ ,  $\dot{h}_d$  respectivamente. Por otra parte  $k_{rj}$ ,  $k_{phj}$ ,  $k_{ihj}$  indican el  $j$ -ésimo elemento de la diagonal  $K_R$ ,  $K_{ph}$ ,  $K_{ih}$ .

Ahora bien, al analizar la ecuación (3.26) se observa que si  $k_{rj} > 0$  se presentan dos casos para la fuerza deseada  $h_{dj}$ .

1.  $h_{dj}$  es constante

Si  $h_{dj}$  es constante, entonces  $\dot{h}_{dj} = 0$ , por lo tanto se tiene la ecuación 3.27.

$$\dot{e}_{hj} \approx -\frac{k_{rj} k_{ihj} e_{hj}}{1 + k_{rj} k_{phj}} \quad (3.27)$$

Ahora bien, sabemos que el error dado por  $e_{hj}$  tiende a cero cuando el tiempo tiende a infinito, por lo que  $\dot{e}_{hj} \approx 0$ .

2.  $\dot{h}_{dj}$  variante en el tiempo

Entonces se tiene que  $\dot{h}_{dj} \neq 0$ , por esta razón la ecuación (3.26) se puede reescribir como sigue.

$$\dot{e}_{hj} \approx -\frac{k_{rj} k_{ihj}}{1 + k_{rj} k_{phj}} e_{hj} \quad (3.28)$$

Al tomar la transformada de Laplace de la ecuación (3.26) se obtiene.

$$\begin{aligned}
& [(1 + k_{rj}k_{phj})s + k_{rj}k_{ihj}] E_{hj} \approx s\dot{H}_{dj}(s) \\
& E_{hj} \approx \frac{s}{(1 + k_{rj}k_{phj})s + k_{rj}k_{ihj}} \dot{H}_{dj}(s) \\
& E_{hj}(s) \approx \frac{s(1 + k_{rj}k_{phj})}{s + (k_{rj}k_{ihj}) / (1 + k_{rj}k_{phj})} \dot{H}_{dj}(s) \tag{3.29}
\end{aligned}$$

Al analizar la ecuación (3.29) observamos que, al aumentar el término  $k_{ihj}$  se puede reducir el error debido a que provoca una ganancia menor. Por otro lado, si se disminuye el término  $\dot{H}_{dj}$  el error de fuerza también se puede reducir.

# Capítulo 4

## Plataforma experimental

En el presente capítulo se muestra la plataforma que se utiliza y la implementación para realizar los experimentos de manera virtual, de igual manera se muestran las configuraciones generales que se deben considerar para trabajar con el sistema cooperativo.

### 4.1. RobotStudio

La programación y desarrollo de aplicaciones de manera remota o virtual es una forma de trabajo que se ha desarrollado en la actualidad con mucho éxito, de esta manera para desarrollar las simulaciones pertinentes en el presente trabajo se hace uso del software de simulación y programación offline de ABB, RobotStudio, que cuenta con una gran variedad de herramientas y paquetes que permiten la programación del robot o de un sistema cooperativo sin tener de manera física los robots. El software se basa en ABB VirtualController, este crea una copia exacta del software real que ejecutan los robots en producción o en laboratorios de investigación. Gracias al controlador virtual es que se pueden realizar simulaciones de diferentes tipos de robots, es importante mencionar que al trabajar en modo offline los programas creados para los robots virtuales pueden ser igualmente utilizados para el caso en que se desee trabajar con los robots de manera física, ya que se crean archivos de configuración idénticos.

A continuación, se enumeran diferentes características y capacidades del software.

1. Colaboración en soluciones de robots con participantes de cualquier lugar del mundo.
2. Creación de un Digital Twin de la fábrica para predecir los resultados de las modificaciones.
3. Verificar los sistemas del robot digitalmente para minimizar el tiempo de arranque o acelerar la producción.
4. Cuenta con un visualizador con hologramas 3D para propuestas de sistemas en salas de conferencias.
5. Creación de diferentes tipos de estación.

6. Importación de geometrías y modelos 3D de cualquier formato (RobotStudio trabaja sobre CATIA).
7. Programación y simulación cinemática de las estaciones.
8. Facilidad de creación de células robóticas.
9. Permite realizar la exportación de los resultados obtenidos a una estación real.
10. Permite guardar datos de simulaciones para graficar en diferentes Software como MATLAB.

Siendo así, RobotStudio permite la programación y simulación de sistemas robóticos desde una PC. Cabe mencionar que en la Sección de Mecatrónica no se cuenta con el kit para el control de fuerza que ofrece la empresa ABB, el cual consta del sensor de fuerza, placa del adaptador, placa de medición y cables, al obtener dicho kit se proporciona de igual manera las librerías necesarias para poder utilizar dicho kit en RobotStudio, cabe mencionar que los manipuladores son compatibles con otros sensores de fuerza, por lo que en trabajos como en [23], [22] y [24] se han realizado con sensores de fuerza de 6 grados de libertad, los cuales son: Sensor de fuerza/par JR3 y el sensor de fuerza/par ATI, ambos sensores cuentan con galgas extensiométricas para la medición de fuerza en tres ejes y mediciones de par torsor en tres ejes. En dichos trabajos, un punto importante es que realizan el sistema cooperativo sin contar con el kit de fuerza que ofrece ABB, acoplando los sensores en el efector final de cada robot. Se puede encontrar información relacionada a estos sensores en [30], [32].

En el presente trabajo y por la situación que agobia al planeta con la pandemia del SARS-CoV-2 (covid-19), se desarrollan las simulaciones en la plataforma virtual RobotStudio sin el kit de fuerza de ABB, dentro de la plataforma no se tiene disponible ningún tipo de sensor de fuerza, por lo que las simulaciones posteriores se realizaron por medio del modelo de impedancia de Hogan, descrito en el Capítulo 2 con la ecuación (2.2).

A continuación, se muestra una descripción general del software y la configuración básica para realizar las simulaciones del sistema cooperativo.

#### **4.1.1. Creación de una estación e interfaz de trabajo.**

La creación de una nueva estación es similar a la creación de un nuevo archivo o documento de cualquier otro software, una vez realizada la creación de una nueva estación se crea de manera predefinida una ruta donde se guardan todos los controladores de RobotWare.

La interfaz de usuario cuenta con una gran variedad de elementos como se observa en la Figura 4.1, la mayoría son de gran ayuda, ya que al trabajar de manera virtual se vuelve necesario el uso de estos para tener mejor precisión al momento de crear las trayectorias para los robots. De igual manera sus herramientas hacen posible crear o cambiar las propiedades de objetos creados desde la sección de modelado.

1. Contiene todas las barras de navegación como son: Posición inicial, Modelado, Simulación, Controlador, Rapid, Complementos. Esta barra de navegación sirve de igual

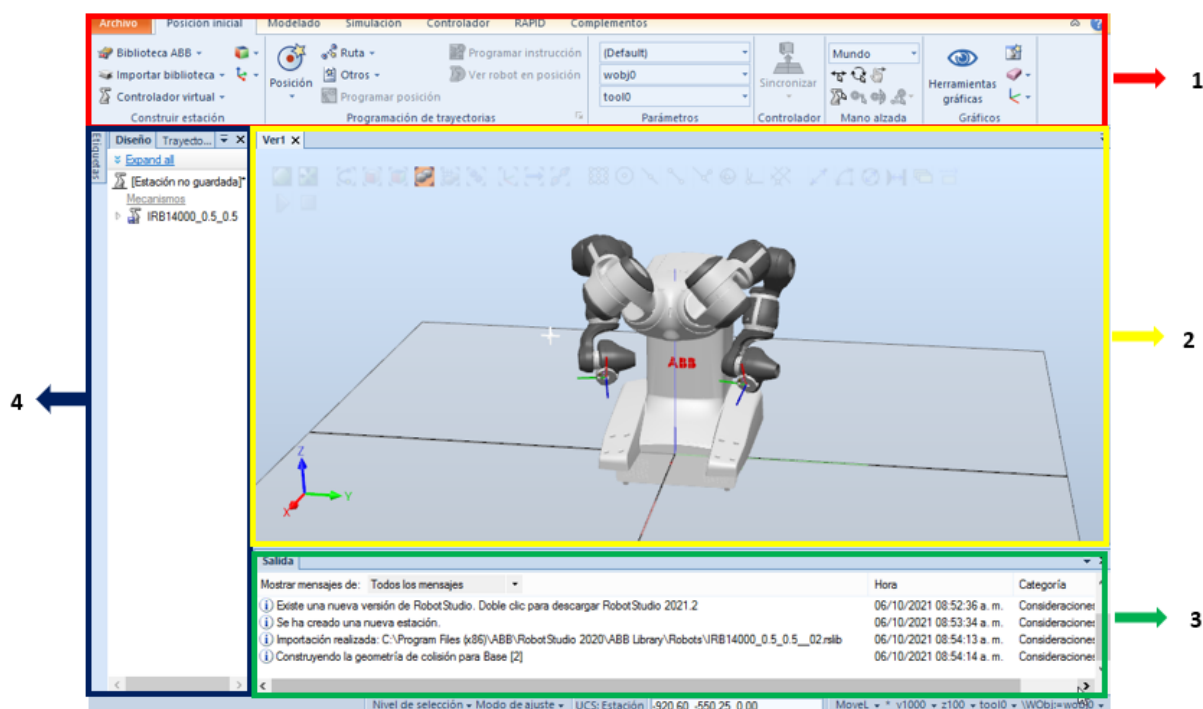


Figura 4.1: Interfaz RobotStudio

manera para programar trayectorias de los robots y manipulación de cada eje de los manipuladores.

2. Área gráfica donde se visualizan las simulaciones de los robots y componentes.
3. Ventana de salida donde RobotStudio envía mensajes al usuario.
4. Navegador de objetos, en este se pueden encontrar componentes físicos, como robots y herramientas, también se pueden visualizar trayectorias y objetos creados.

Para el desarrollo de aplicaciones en RobotStudio la empresa ABB cuenta con un manual de operación, el cual se encuentra disponible en [5], en él se explican detalladamente las herramientas del software, así como la configuración de los robots, configuración de los controladores, creación de objetos de trabajo y de trayectorias o puntos, etc. Cabe mencionar que en dicho manual se encuentra una sección dedicada a la lógica de estación, donde solo se encuentra la descripción general, sin ejemplificar, ni ampliar el uso de cada componente. Por otra parte, ABB igual cuenta con un manual de referencia técnica [6], en éste puede encontrar instrucciones en RAPID y la sintaxis, de igual manera se encuentran funciones como son operaciones matemáticas, y por último se muestran los tipos de datos que se pueden procesar en lenguaje RAPID en RobotStudio.

## 4.2. Plataforma experimental RoboStudio

En el esquema que se observa en la Figura 4.2 se tiene un arreglo de la plataforma experimental que se desarrolla en el presente trabajo.

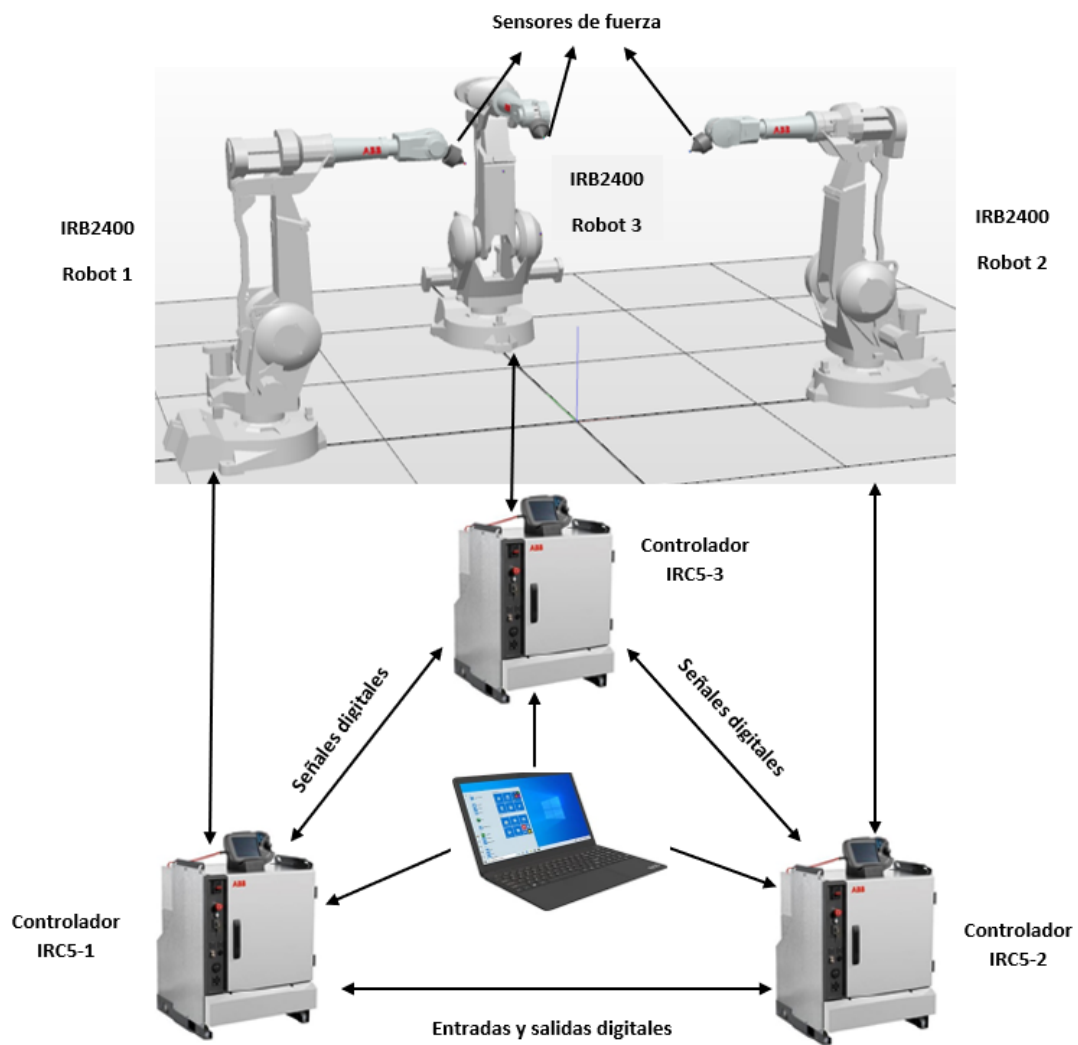


Figura 4.2: Plataforma experimental

### 4.3. Descripción Robot manipulador IRB 2400/16

Un componente necesario de la plataforma experimental es el robot IRB 2400/16 de ABB, es un robot industrial de 6 ejes, diseñado específicamente para industrias de fabricación que utilizan automatización flexible basada en robots, cuenta con una carga máxima de 20 kg y un alcance máximo de 1.5 m y una exactitud de 0.03 mm.



Figura 4.3: Robot IRB 2400/16

En RobotStudio se tiene una gran variedad de robots manipuladores, por lo que se utilizan los mismos robots que se tienen en la Sección de Mecatrónica en el Cinvestav, de igual manera el controlador usado por cada robot es el IRC5 mostrado en la Figura 4.4, este controlador tiene la tarea de enviar energía a los motores de los robots. De manera física este controlador cuenta con puertos de comunicación serial RS-232, LAN y entradas/salidas digitales, este es el encargado de leer las variables para realizar exitosamente el control de posición y orientación del robot. En él se realiza la creación de entradas y salidas digitales.



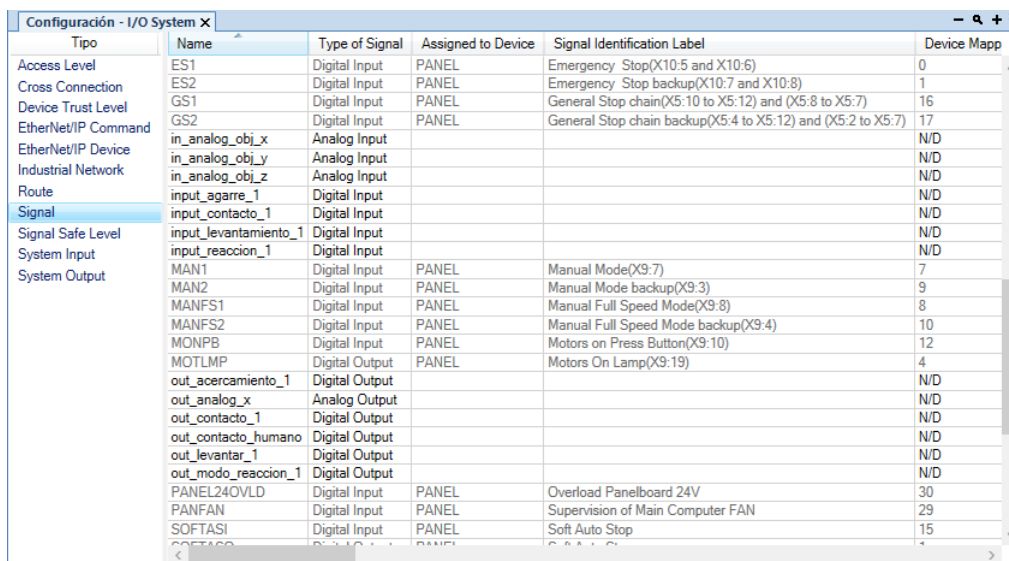
Figura 4.4: Controlador IRC5



### 4.3.1. Comunicación de controladores virtuales

La comunicación entre los controladores de manera física se realiza mediante un cable, entre el controlador del robot 1 y del robot 2, éste sirve para enviar señales digitales para sincronizar el inicio de rutinas programadas en los robots. En RobotStudio no se puede realizar la conexión entre controladores con un cable, para ello se hace uso de la lógica de estación como se detalla a continuación.

Como primer etapa se crean las señales de entradas y salidas digitales virtuales, ya que hay otra configuración para crear las señales en los controladores si se trabaja con ellos de manera física.



Tipo	Name	Type of Signal	Assigned to Device	Signal Identification Label	Device Mapp
Access Level	ES1	Digital Input	PANEL	Emergency Stop(X10:5 and X10:6)	0
Cross Connection	ES2	Digital Input	PANEL	Emergency Stop backup(X10:7 and X10:8)	1
Device Trust Level	GS1	Digital Input	PANEL	General Stop chain(X5:10 to X5:12) and (X5:8 to X5:7)	16
EtherNet/IP Command	GS2	Digital Input	PANEL	General Stop chain backup(X5:4 to X5:12) and (X5:2 to X5:7)	17
EtherNet/IP Device	in_analog_obj_x	Analog Input			N/D
Industrial Network	in_analog_obj_y	Analog Input			N/D
Route	in_analog_obj_z	Analog Input			N/D
Signal	input_agarre_1	Digital Input			N/D
Signal Safe Level	input_contacto_1	Digital Input			N/D
System Input	input_levantamiento_1	Digital Input			N/D
System Output	input_reaccion_1	Digital Input			N/D
	MAN1	Digital Input	PANEL	Manual Mode(X9:7)	7
	MAN2	Digital Input	PANEL	Manual Mode backup(X9:3)	9
	MANFS1	Digital Input	PANEL	Manual Full Speed Mode(X9:8)	8
	MANFS2	Digital Input	PANEL	Manual Full Speed Mode backup(X9:4)	10
	MONPB	Digital Input	PANEL	Motors on Press Button(X9:10)	12
	MOTLMP	Digital Output	PANEL	Motors On Lamp(X9:19)	4
	out_acercamiento_1	Digital Output			N/D
	out_analog_x	Analog Output			N/D
	out_contacto_1	Digital Output			N/D
	out_contacto_humano	Digital Output			N/D
	out_levantar_1	Digital Output			N/D
	out_modos_reaccion_1	Digital Output			N/D
	PANEL24OVL	Digital Input	PANEL	Overload Panelboard 24V	30
	PANFAN	Digital Input	PANEL	Supervision of Main Computer FAN	29
	SOFTASI	Digital Input	PANEL	Soft Auto Stop	15
	SOFTASO	Digital Input	PANEL	Soft Auto Stop	14

Figura 4.5: Creación de señales digitales virtuales

Para crear señales digitales se ingresa a la configuración interna del controlador, en este caso en el controlador del robot 1 como se ve en la Figura 4.5, en este apartado se dan de alta las señales deseadas, de igual manera se pueden crear señales analógicas como salidas o entradas, para la sincronización del sistema, basta con crear banderas de sincronización para realizar exitosamente la tarea de agarre de los robots con el objeto.

La segunda etapa consiste en comunicar los controladores virtuales para enviar y recibir señales digitales, ésta se realiza en la lógica de estación en RobotStudio, en ella se realiza la analogía del cable que interconecta los controladores físicamente. Para ingresar a la lógica de estación, se selecciona desde la pestaña simulación y aparece la ventana que se observa en la Figura 4.6

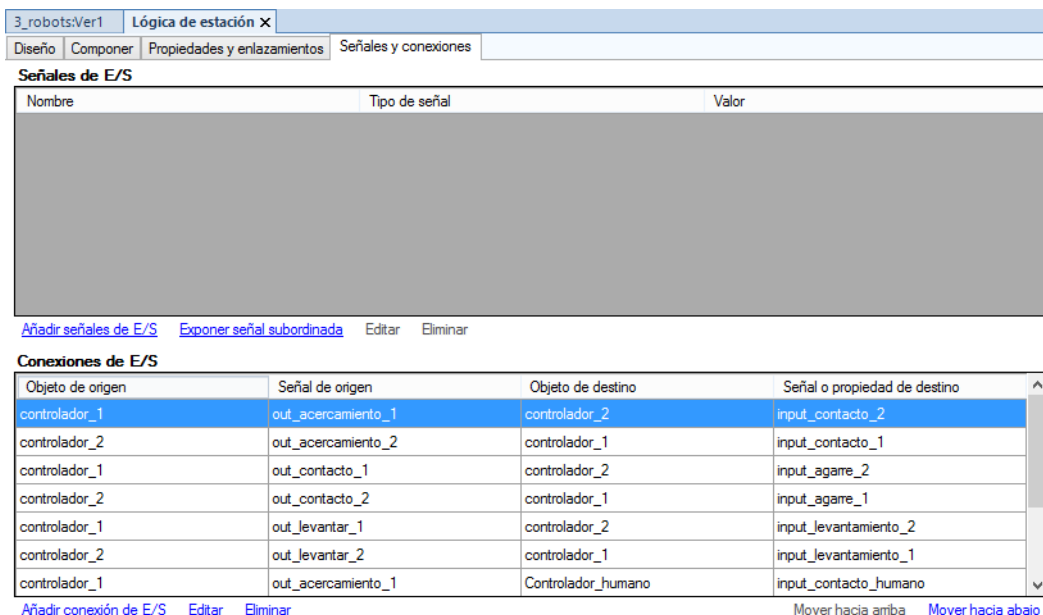


Figura 4.6: Lógica de estación

De esta manera se realizan las asignaciones de entradas y salidas digitales de los controladores, seleccionando el controlador de origen y el controlador de destino con los respectivos nombres de cada señal. En la Figura 4.7 se tienen las conexiones que se realizan entre los controladores.

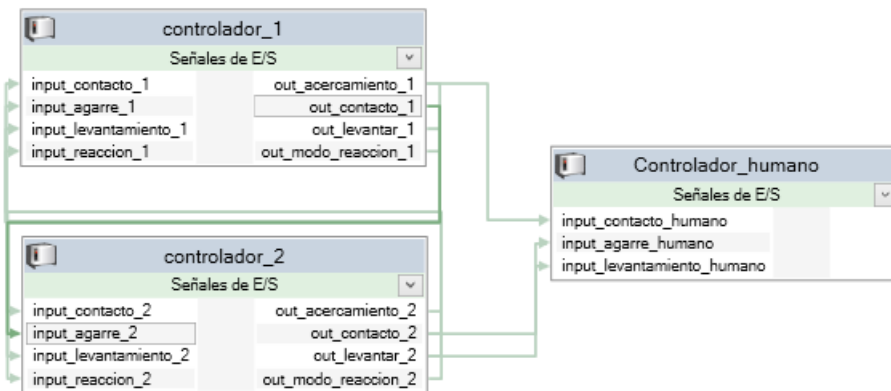


Figura 4.7: Diagrama de conexiones entre controladores

Así con la lógica de estación se realiza la conexión y comunicación entre los controladores del sistema cooperativo, así se pueden crear conexiones entre señales de E/S, cuando se trabaja con añadir o editar conexiones de E/S, se tiene la opción de seleccionar entre los controladores de la estación en las listas de objeto de origen y objeto de destino.

Cuando se realiza la simulación del sistema se tienen la siguientes tareas para los robots manipuladores.

- En cada situación los robots manipuladores tienen la instrucción de ir a la posición

*home*, esta es una posición segura donde el robot no tiene contacto con alguna superficie o entorno.

- Los robots manipuladores realizarán la primer trayectoria para posicionarse en un punto cerca del objeto, en este punto se realiza la primer bandera de sincronización donde ningún robot realizará la siguiente rutina o trayectoria hasta que los dos robots estén en el mismo punto.
- Los robots manipuladores ejecutarán la tarea de contacto con el objeto (sin aplicar fuerza), de esta manera si algún robot no ha llegado al punto de contacto, entonces no realizarán la etapa de agarre, esta tarea solo será realizada hasta que los manipuladores hagan contacto con el objeto y que cada controlador envíe las señales digitales para verificar que cada uno está bien posicionado.
- Se realiza la etapa de agarre, en este caso cada robot ejerce una fuerza total de 20 N, se tiene la misma trayectoria de fuerza deseada en cada robot, de esta manera se realiza exitosamente la etapa de agarre. Esta trayectoria se observa en la ecuación (4.1), la cual es una función suave que permite llegar al valor de referencia sin dañar los sensores de fuerza.  $t$  es el tiempo del reloj del robot o en este caso de cuando se inicia la simulación,  $t_i$  es el tiempo de inicio que tiene el robot al inicio de la rutina y  $f_d$  es el valor de referencia deseado.

$$f_{dx} = f_d * (1 - e^{-(t-t_i)/3}) \quad (4.1)$$

- Después de la etapa de agarre, se puede realizar manipulación del objeto, en el último experimento que se muestra en el Capítulo 5, se realiza la tarea de agarre para posteriormente manipular el objeto sin perder contacto.

Las unidades de control IRC5 trabajan con el sistema operativo RobotWare-Os. este permite la programación de tareas y trayectorias en los robots, esto se realiza mediante el lenguaje RAPID propio de ABB. El controlador cuenta con una unidad portátil FlexPendant, éste permite activar o mover los motores del robot, existen varias formas de usarlo, se puede usar para mover u orientar el punto final de la brida del robot, o en su caso mover cada eje por separado.

El FlexPendant es de gran ayuda cuando se trabaja con los robots de manera física, este se utiliza para llevar al robot a un punto deseado y al llegar a ese punto se genera en automático la trayectoria que el robot debe seguir para llegar a dicho punto. De esta manera se programa una posición segura *home* para los robots, y de igual manera la trayectoria que nos permite acercarnos al objeto y por último la trayectoria en donde se realiza el contacto con el entorno.

ABB proporciona el software RobotStudio para simulación y programación fuera de línea, este software permite trabajar eficientemente con los datos del controlador IRC5 virtual. RobotStudio cuenta con la consola portátil FlexPendant virtual. En este trabajo se opta por usar las herramientas que ofrece el software, que para este caso es muy útil y con una gran exactitud, ya que tiene apartados muy similares a los que se usan en diferentes programas de CAD, como se mencionó en la Sección 4.1.1 y en la Figura 4.1

## 4.4. RAPID

RAPID es un lenguaje de programación de alto nivel desarrollado por la empresa ABB. Una aplicación en RAPID consta de un programa y una serie de módulos del sistema como se muestra en el esquema de la Figura 4.8. El programa que se genera en RAPID es una secuencia de instrucciones que generan movimientos en el robot, de manera general un programa en RAPID consta de:

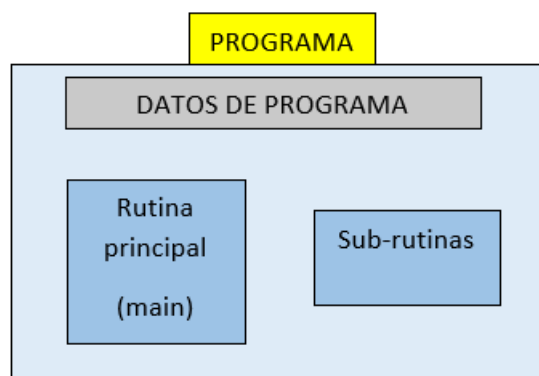


Figura 4.8: Estructura de un programa en RAPID.

- **Rutina principal (main):** rutina donde se inicia la ejecución
- **Conjunto de sub-rutinas:** Sirven para dividir el programa en partes más pequeñas, con la intención de tener un programa bien estructurado.
- **Los datos del programa:** Donde se definen posiciones, valores numéricos, sistemas de coordenadas, etc.

Existen bastantes elementos básicos, tipos de datos, tipos de registros, se puede encontrar más información en [6], donde se explica detalladamente la sintaxis correcta en cada instrucción y el tipo de dato que se utiliza en cada caso.

Dentro de lo más relevante en las características principales del lenguaje antes mencionado se tiene:

- Expresiones de lenguaje para evaluar un valor y así poder asignarlo a una variable o utilizarlo como un argumento de una instrucción de rutina.
- Asignación de tipos de datos como son; registros, datos atómicos, persistentes, etc.
- Programas estructurados por la creación de sub-rutinas.
- Expresiones aritméticas y lógicas de alto nivel.

## 4.5. Sensor de fuerza y tarjeta de adquisición de datos

En la Sección de Mecatrónica en el Cinvestav se cuenta con un par de sensores de fuerza, como se mencionó con anterioridad, al no tener acceso a los laboratorios y no contar con el kit de fuerza y de sincronización que ofrece ABB, se utilizan dos sensores que se mencionan a continuación.

- Sensor de fuerza/par de 6 GDL de la marca JR3, constituido de un arreglo de galgas extensiométricas para la medición de fuerzas aplicadas en tres ejes y el par torsor en tres ejes, en otras palabras, el sensor es el encargado de medir la deformación en cada uno de sus ejes. En [36] se muestran las características del sensor JR3, en el cual se menciona la matriz de calibración y desacoplamiento que es utilizada para convertir los voltajes de salida en una fuerza o un momento. Dependiendo del modelo que se tenga de dicho sensor es que se tienen diferentes esfuerzos de carga. Las mediciones de fuerza en  $x$ ,  $y$  es de un rango de  $\pm 111[N]$  y en  $z$  un rango de  $\pm 222[N]$ , dependiendo del modelo.
- Sensor de fuerza/par de 6 GDL de la marca ATI, constituido por un arreglo de galgas extensiométricas para la medición de fuerzas en tres ejes y de par torsor en tres ejes. En sus ejes  $x$ ,  $y$  tiene un rango de fuerza de  $\pm 80[N]$ , mientras que para el eje  $z$  tiene un rango de  $\pm 240[N]$ . En los sensores de marca ATI se tiene gran variedad y con ellos diferentes tipos de cargas que soporta cada sensor.

En el presente trabajo de tesis, al no tener el kit de ABB y no poder realizar pruebas físicas con los robots y sensores en el laboratorio, se hace uso del modelo de impedancia mostrado en el Capítulo 2. La ecuación (2.2) sirve para realizar las respectivas simulaciones que al implementar en la plataforma virtual muestra resultados congruentes con los que se obtendrían con los sensores antes mencionados.

Por esta razón se vuelve necesario diseñar una herramienta que simule el sensor de fuerza que se instala en la brida del robot como se observa en la Figura 4.9

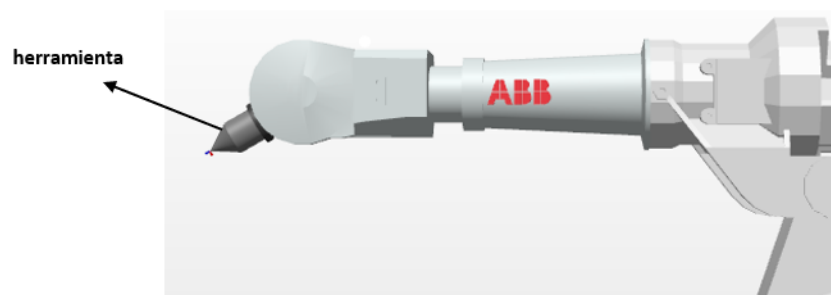


Figura 4.9: Herramienta como sensor de fuerza

Dicha herramienta está diseñada en el mismo software RobotStudio, cuenta con una masa de  $1\text{ kg}$  y con la característica de que la ubicación del TCP por sus siglas en inglés (tool center point) de la herramienta se traslada al punto final de la misma al instalarlo en la brida, al realizarlo se debe tomar en cuenta que, el punto central de la herramienta TCP ahora

se encuentra en el punto final de la herramienta, por lo que las trayectorias que se programan se realizarán tomando en cuenta la herramienta antes mencionada. En la Figura 4.10 se observa cómo se agrega y se explica la configuración de la herramienta para poder realizar exitosamente la instalación con el robot y la definición del TCP.

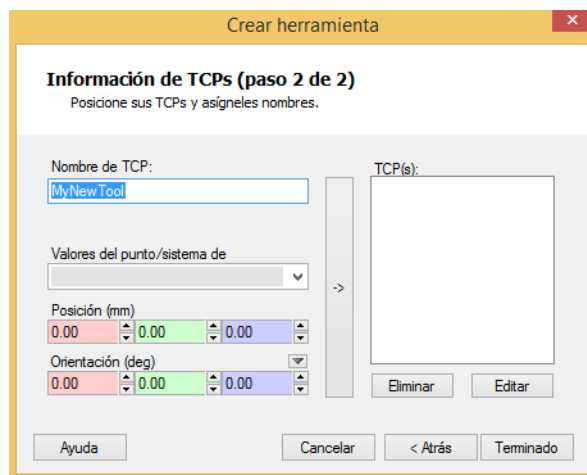


Figura 4.10: Posicionar TCP de una herramienta

De esta manera se pone la posición en la que se desea trasladar el TCP sobre la herramienta. El software ofrece una barra de opciones para poder seleccionar con mayor precisión puntos específicos de elementos diseñados o de elementos exportados de otros programas.

#### 4.5.1. Creación de herramienta.

Para simular la herramienta del robot, se necesitan sus respectivos datos, para así partir de una pieza que ya existe o una pieza creada desde la pestaña de modelado y crear una herramienta.

A continuación se enumeran brevemente los pasos para crear una herramienta.

1. Ingresar a la pestaña de modelado.
2. Crear herramienta.
3. Nombre de la herramienta.
4. Introducir la masa de la herramienta, su centro de gravedad y el momento de inercia  $I_x$ ,  $I_y$ ,  $I_z$ , si estos valores no se conocen con exactitud se puede seguir usando la herramienta sin ningún problema, pero se deben corregir estos parámetros antes de ejecutar el programa en robots reales.
5. Nombre del TCP, se debe introducir el nombre para el punto central de la herramienta (TCP).

6. Introducir la posición del TCP respecto al sistema de coordenadas mundo, que representa el punto de montaje de la herramienta.
7. Terminar la creación de la herramienta, de esta manera aparece en el navegador de objetos y en la ventana de gráficos, observese la Figura 4.11.

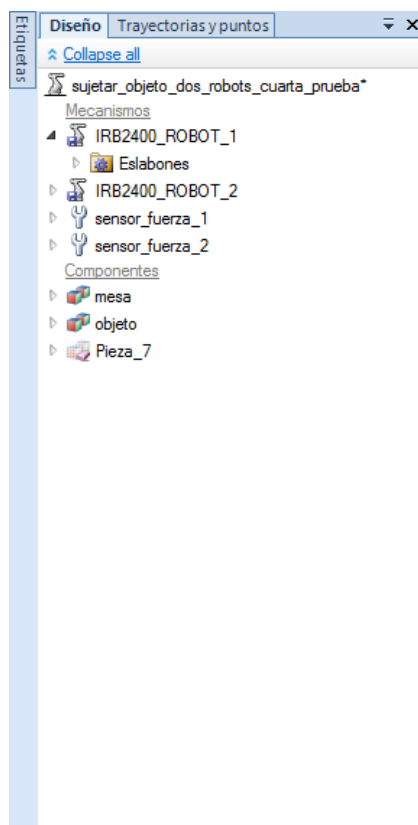


Figura 4.11: Navegador de objetos

En la Figura 4.9 se observa la herramienta instalada en el robot manipulador. Mientras que en la Figura 4.11 se muestra cómo se agrega exitosamente la herramienta, la cual por último se debe instalar en la brida del robot, para hacer esto, basta con arrastrar el objeto sobre el robot en el cual se desea instalar la herramienta.

## 4.6. Interacción humana

Los seres humanos son capaces de realizar infinidad de tareas que requieren manipulación de objetos, sin la necesidad de conocer a la perfección su entorno, esto es porque el sistema sensorial que posee el humano le permite adaptarse a las características y necesidades de la tarea a realizar. Dicho de otra forma, no es necesario conocer la dinámica del objeto (entorno) cuando se cuenta con un sistema de procesamiento y control con cierto grado de inteligencia. Por esta razón la robótica es una ciencia que a lo largo de los años ha intentado imitar el comportamiento del humano.

La robótica avanza en sus diferentes aplicaciones, desde mejoras computacionales, la electrónica, sensores y comunicaciones, debido a la gran demanda que tienen los robots

manipuladores en la industria y la complejidad de las tareas que requieren contacto directo con el robot, hacen que sea cada vez más frecuente la interacción física con seres humanos. Durante la interacción, el entorno puede establecer limitaciones geométricas en las trayectorias, las cuales se conocen como restricciones cinemáticas, esto hace referencia al movimiento restringido, el cual presenta diferentes casos.

**Contacto de tipo inercial:** Se podría ejemplificar como empujar un bloque

**Contacto disipativo:** Deslizamiento de un objeto sobre una superficie con fricción.

**Contacto elástico:** Representa el empujar o ejercer fuerza sobre una pared elástica.

En todos los casos, el uso de una estrategia de control de movimiento en el robot manipulador y que involucra la interacción con su entorno es propensa a fracasar, por esta razón se propone el esquema de control por retroalimentación que actualiza la posición y las fuerzas de contacto, que se generan al tener contacto con el entorno.

Al desarrollar los experimentos virtualmente se vuelve complicado desarrollar la interacción humana, por lo que se ha planeado sustituir la tarea del humano por un robot IRB 2400/16, mismo que se describe en la Sección 4.3, este robot de igual manera cuenta con un control de fuerza/par, que está basado en la respuesta dinámica del sistema robótico ante la presencia de fuerzas de contacto.

Siendo así, el tercer robot es el encargado de imponer trayectorias sobre el sistema cooperativo, de esta forma el robot derecho (robot 1) y robot izquierdo (robot 2) estarán en modo reacción, esto quiere decir que los robots están a la espera de fuerzas que modifiquen su posición y fuerza de referencia y tengan que desplazarse para seguir cumpliendo con dichos valores.

Por último se utiliza el controlador virtual IRC5 descrito en la Sección 4.3, ya que al no tener una forma directa de introducir trayectorias al sistema se instala en el tercer robot una herramienta que simulará el sensor de fuerza, ya que dicho manipulador imprime fuerza sobre el objeto para su interacción con el sistema, y se debe tener un buen control de fuerza en este último.



# Capítulo 5

## Resultados de simulación

En el presente capítulo se muestran resultados de experimentos realizados en la plataforma virtual RobotStudio, dichos experimentos son el resultado de implementar los esquemas de control mencionados en el Capítulo 3. La principal tarea es mostrar el correcto funcionamiento de los esquemas de control tanto para un solo robot manipulador como para el sistema cooperativo y por último para el sistema cooperativo con la interacción de un tercer robot, encargado de dar trayectorias al sistema cooperativo a través de valores de fuerza de referencia.

Una observación importante es que el software RobotStudio por sí mismo no es capaz de graficar las señales de interés, por esta razón se hace uso de MATLAB, ya que solo se pueden obtener vectores de datos de RobotStudio y MATLAB para graficar dichos datos.

Como se mencionó en el Capítulo 4, para realizar las simulaciones sin tener físicamente acceso a los robots manipuladores, ni a los sensores de fuerza que se tienen en la Sección de Mecatrónica, se utiliza el modelo de impedancia para las lecturas de fuerza en el efector final de cada robot. Un dato importante es que dicho modelo impone un valor fijo sobre el valor  $X$ , lo que se interpreta como una posición de un muro (fijo), por esta razón se vuelve complicado el correcto funcionamiento de cada simulación al tratar de manipular un objeto en común (movible).

Al ser uno de los primeros trabajos que se realizan en dicha plataforma, es interesante y relevante mostrar que el control de fuerza que se explica en el Capítulo 3 es funcional en la plataforma virtual. De igual forma se desarrolla el sistema cooperativo en RobotStudio con sus respectivo controladores como se tiene el en los laboratorios de la Sección de Mecatrónica.

### **5.1. Experimento 1: Aplicación del control híbrido posición-fuerza.**

Para la implementación del control de fuerza en la plataforma virtual RobotStudio se consideran las fuerzas medidas sobre el eje que se desea actuar, de esta forma se realiza la aplicación de fuerza sobre una mesa, la cual sirve de referencia para observar la penetración sobre la misma, ya que como se describió en el Capítulo 4 la herramienta que se instala sobre la brida del robot tiene la finalidad de representar o simular el sensor de fuerza.

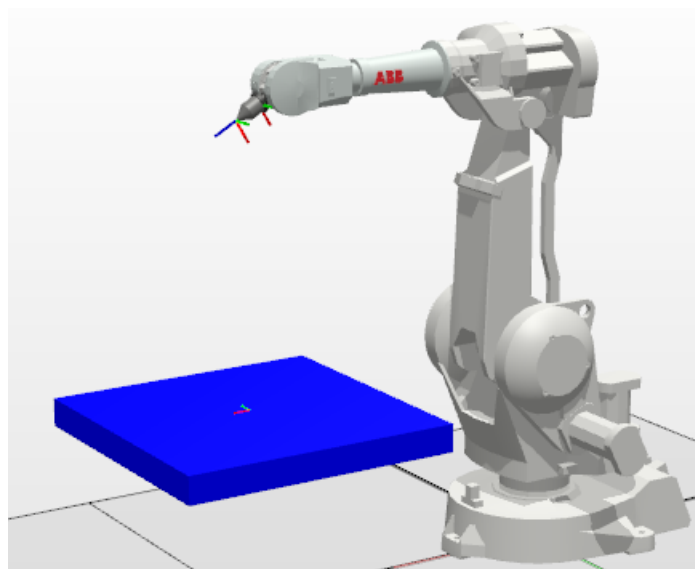


Figura 5.1: Sistema robótico en posición Home

En la figura 5.1 se muestra el sistema, el cual en este primer experimento consiste en mostrar que el modelo de impedancia funciona en la plataforma virtual.

La rutina consiste en:

1. Configuración del robot y el controlador IRC5, la cual se realiza de acuerdo con el manual del operador de RobotStudio [5].
2. Llevar al robot manipulador a posición "home", esta es una posición segura para el robot, donde no está en contacto con ningún objeto o manipulador. La creación de dicha trayectoria se realiza en el work object 0 (wobj0). Cabe mencionar que se puede realizar en cualquier otro objeto de trabajo.
3. Crear la trayectoria para acercarse a la superficie de contacto (mesa). Al crear dicha trayectoria es necesario crear un wobj diferente. Una característica importante es que las instrucciones de movimiento de moveL y moveJ tienen como argumento la precisión con la que se desea realizar dicha tarea, en este caso se utiliza "fine" la cual da la más alta precisión que ofrece ABB en sus robots manipuladores. Dicho acercamiento se observa en la Figura 5.2
4. Iniciar la trayectoria de fuerza deseada en  $z$  sobre la superficie de contacto (mesa), una vez realizado el contacto, el robot avanza sobre el eje  $z$  hasta que llegue a una penetración deseada que se cumplirá hasta que alcance el valor de referencia de fuerza deseado. En la Figura 5.3 se observa un acercamiento del sistema cuando ha concluido la tarea asignada.

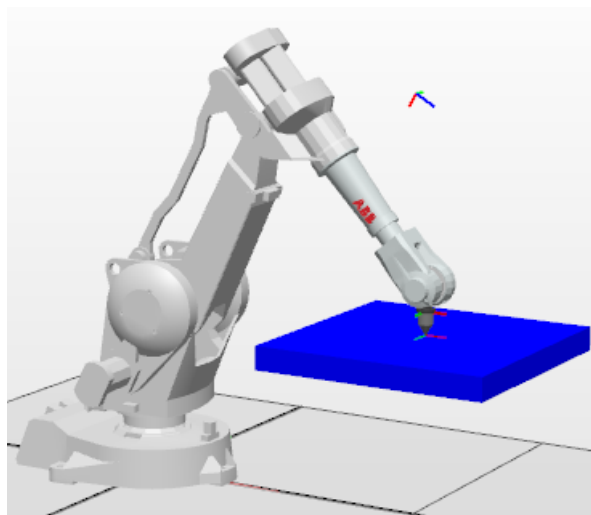


Figura 5.2: Sistema robótico en posición de acercamiento a la mesa

En dicho experimento se fijó la fuerza deseada en  $7N$ , utilizando la herramienta que se observa en la figura 5.3, donde la parte final (punta) ha penetrado sobre la superficie de la mesa, de esta manera se tiene que, ésta es la penetración necesaria para poder ejercer la fuerza de referencia. La herramienta instalada en la brida del robot como se mencionó anteriormente tiene el TCP desplazado de tal manera que ahora todas las trayectorias o movimientos que realice el robot serán a partir del punto final de éste.

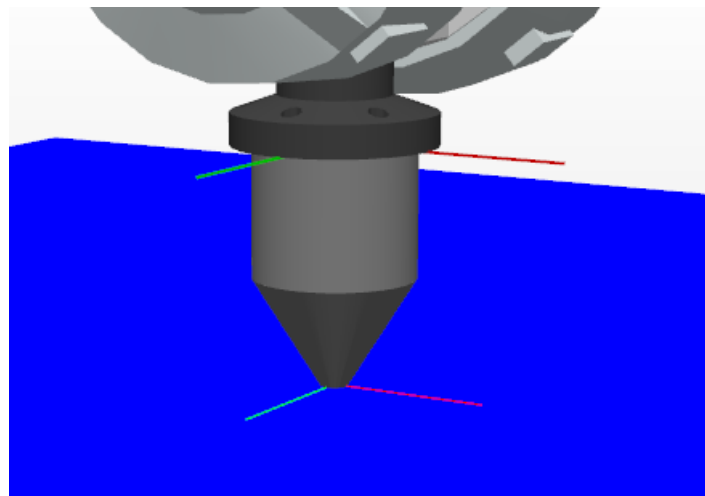


Figura 5.3: Sistema robótico cuando alcanza el valor de referencia de fuerza deseada

La posición y orientación del robot en la superficie de contacto se describe en la Tabla 5.1, estos datos son relevantes, ya que en gráficas posteriores se observa el comportamiento del efector final.

Posición $[x, y, z]$	$[1006.143, 65.832, 580] \text{ mm}$
Orientación $[q_1, q_2, q_3, q_4]$	$[0, 1, 0, 0]$

Tabla 5.1: Posición y orientación del efector final antes de la penetración experimento 1

Para las gráficas que se muestran a continuación se utilizaron las ganancias  $k_{hp} = 0.03$  y  $k_{hi} = 0.8$  para el control de fuerza. Cabe mencionar que los valores de las ganancias se obtuvieron heurísticamente, ya que al trabajar en un ambiente virtual los aspectos como la rigidez del ambiente, la superficie de contacto, materiales, etc. influyen en el cálculo directo de dichos parámetros.

Para la presente simulación se tienen los datos de la mesa usando en este caso acero y con los datos que se muestran en la Figura 5.4.

Steel	
▲ <b>Propiedades de superficie</b>	
Rugosidad (fricción)	0.7
▲ <b>Propiedades masivas</b>	
Densidad (Kg/m <sup>3</sup> )	7800
Módulo de Young (MPa)	200000
Coefficiente de Poisson	0.3
Coefficiente de restitución	0.5
Amortiguaciones de contacto	0.05

Figura 5.4: Propiedades del objeto experimento 1

Resultados obtenidos:

Como se puede apreciar en las Figuras 5.6 y 5.5 el experimento se desarrolló sobre el eje  $z$ , por lo que una vez posicionado el robot en la superficie de contacto, el eje  $x$  y el eje  $y$  no presentan ningún movimiento. En el caso del eje  $z$  se observa el desplazamiento necesario (penetración) para que el manipulador alcance el valor de fuerza de referencia. De esta manera, en las gráficas de posiciones se observa una rampa, la cual representa el movimiento del manipulador para posicionarse en la superficie de contacto.

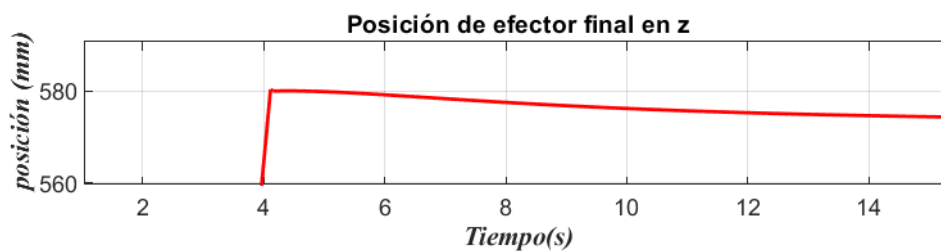


Figura 5.5: Posición del efector final del robot en  $z$  (acercamiento)

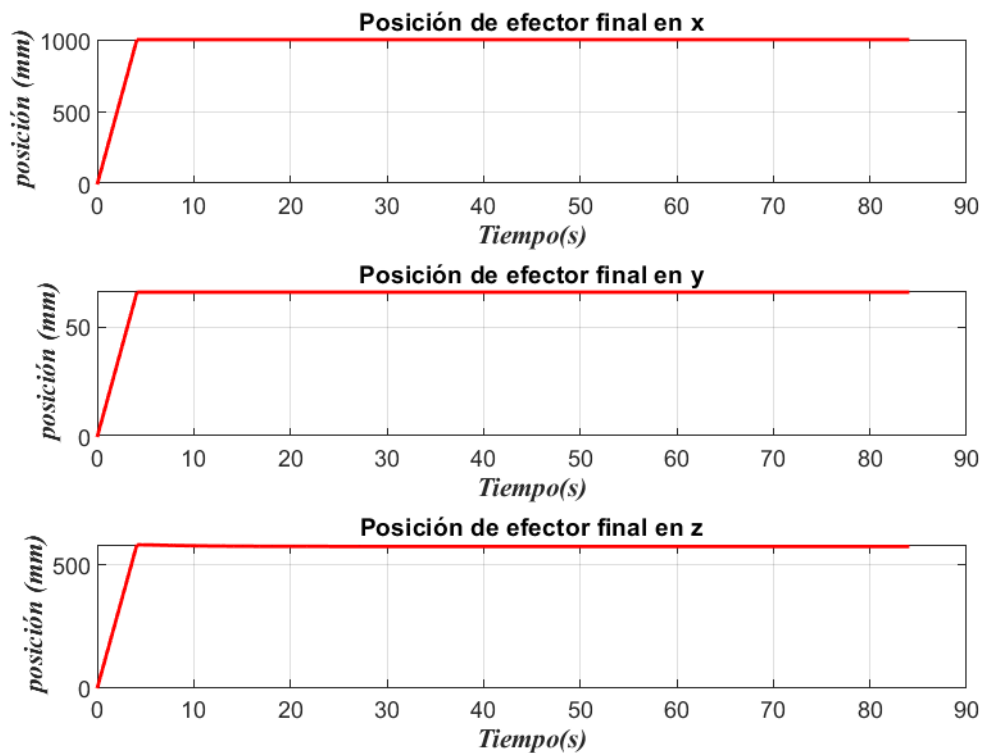


Figura 5.6: Posición del efector final del robot, experimento 1

El control de fuerza es aplicado al robot, el cual en este caso, el contacto con la superficie se realiza a los 4 segundos, posteriormente el robot aplica fuerza según la ecuación (4.1), Debido a esto la trayectoria de fuerza empieza hasta los 4 segundos, que es cuando se hace contacto con la superficie, se aprecia en la Figura 5.7 que el manipulador alcanza la fuerza deseada a los 30 segundos, y posterior a ese tiempo su valor no cambia.

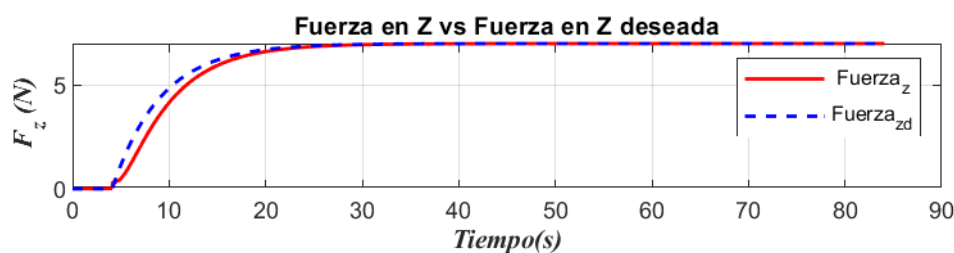


Figura 5.7: Seguimiento de trayectoria de fuerza en z

La diferencia entre la fuerza deseada y la fuerza medida se denomina error de fuerza, obsérvese la Figura 5.8, dicha señal es enviada al control de fuerza, el cual a la salida entrega una posición de referencia en el eje z y ésta es enviada nuevamente al control de posición interno del robot para generar un movimiento vertical como se observa en la Figura 5.3

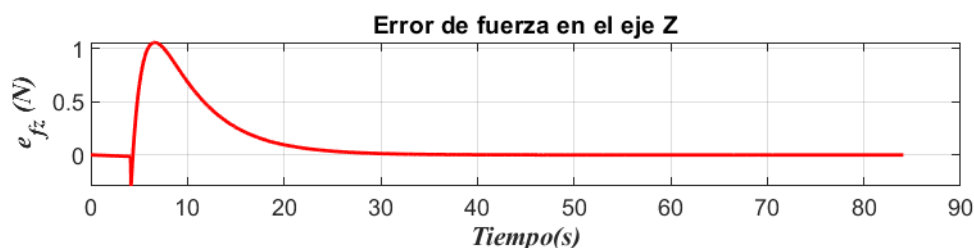


Figura 5.8: Error de fuerza en el eje z, experimento 1

Si se observa la figura 5.9 se tiene que el manipulador empieza a realizar la penetración sobre la superficie a los 4 segundos. Si se realiza una comparación con la figura 5.5 el desplazamiento del robot en  $z$  y la penetración sobre la superficie en  $z$  tiene una correspondencia directa.

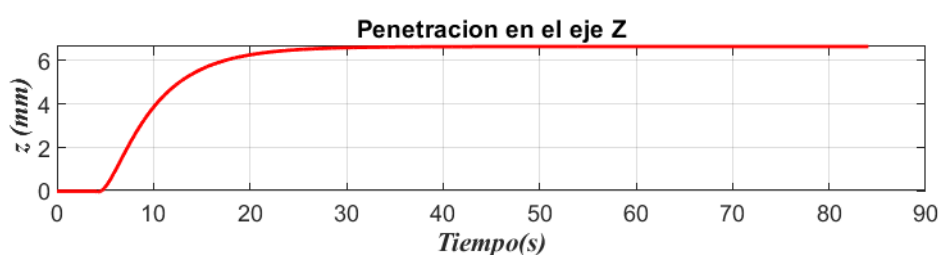


Figura 5.9: Penetración del manipulador sobre la superficie de contacto

## 5.2. Experimento 2: Aplicación del control híbrido posición-fuerza extendido

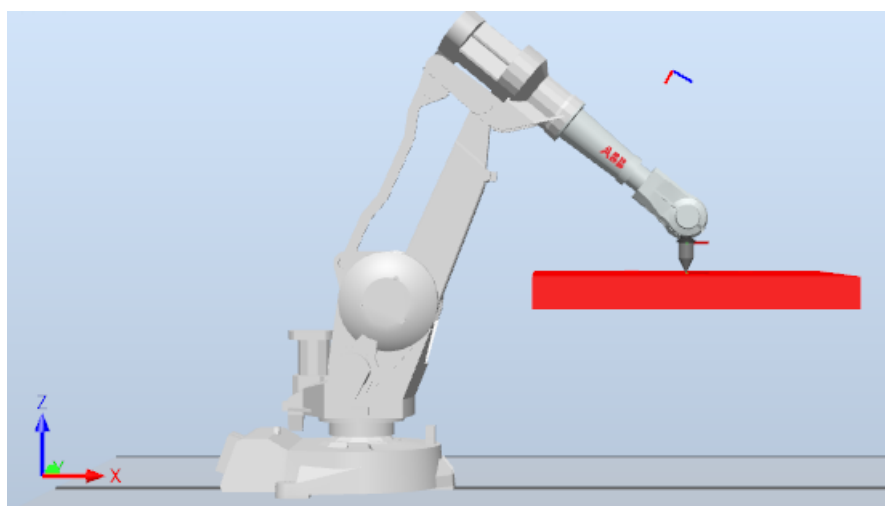


Figura 5.10: Sistema robótico posicionado sobre la mesa

Una aplicación importante es que el manipulador pueda realizar trayectorias mientras aplica fuerza constante en un eje. En la simulación que se presenta en esta sección se enumeran las tareas que el manipulador debe realizar.

1. Rutina de inicio para limpiar variables y enviar al robot a posición HOME, véase la Tabla 5.2.

Posición $[x, y, z]$	$[1015.21, 0, 1362.5] \text{ mm}$
Orientación $[q_1, q_2, q_3, q_4]$	$[0.5, 0, 0.8660, 0]$

Tabla 5.2: Posición y orientación del efector final experimento 2 (home)

2. Realizar un acercamiento a la mesa, ya que servirá de referencia para observar la penetración de la herramienta sobre la mesa, el robot se posiciona y orienta como se observa en la Tabla 5.3, de esta manera el manipulador solo aplica fuerza sobre el eje  $z$  (*tiempo de ejecución 8 segundos*)

Posición $[x, y, z]$	$[1059.75, 4.918, 700]$
Orientación $[q_1, q_2, q_3, q_4]$	$[0, 1, 0, 0]$

Tabla 5.3: Posición del efector final experimento 2 (contacto)

3. Con la ley de control activa (ya que se realizó el contacto) se realiza la penetración sobre la mesa, en este caso se debe aplicar  $10N$  sobre el eje  $z$ . La orientación y posición del efector final es la misma que se observa en la Tabla 5.3 (*Tiempo de ejecución de 8 a 35 segundos*)
4. Una vez se llegue al valor de referencia, el robot iniciará una trayectoria circular de radio  $80mm$ , sin dejar de ejercer fuerza sobre la mesa en  $z$  y manteniendo la misma posición y orientación sobre el mismo eje que se mencionó anteriormente, de esta manera el manipulador realiza la trayectoria sobre el plano  $xy$  pero en  $z$  toma un valor constante. (*Tiempo de ejecución de 48 a 108 segundos*)

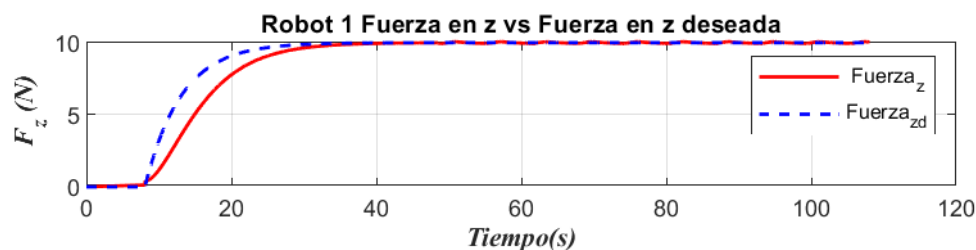


Figura 5.11: Seguimiento de fuerza constante

En la Figura 5.11 se aprecia que el robot hace contacto con la mesa a los 8 segundos de simulación. De los 8 a los 40 segundos se alcanza el valor de referencia, de esta manera se ejercen los  $10N$ , a los 48 segundos se observan ligeras variaciones, esto indica que el robot ha comenzado a moverse de manera circular, por eso las oscilaciones.

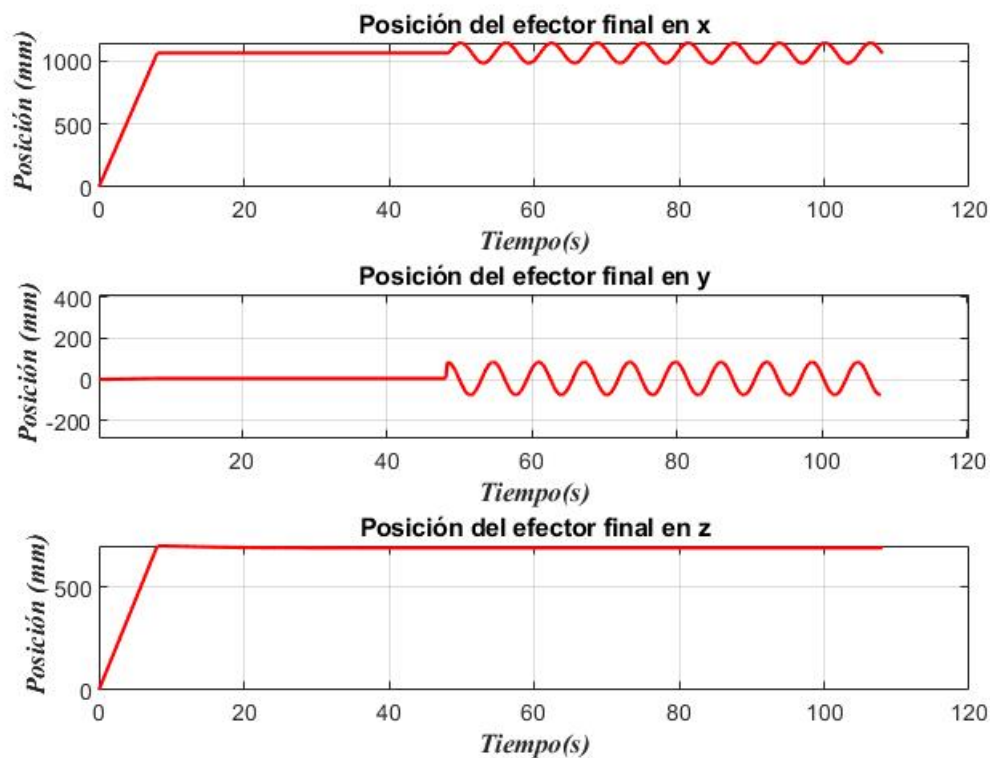


Figura 5.12: Posición del efector final en los 3 ejes

En la Figura 5.12 se muestra la posición del efector final del manipulador, en ella se puede apreciar que se desplaza en el plano  $xz$  hasta realizar la penetración sobre la mesa en  $z$ , esta tarea se realiza de los 0 a los 48 segundos. Después de dicha penetración se muestran oscilaciones en el eje  $x$  y en eje  $y$ , ya que pasando los 48 segundos el manipulador debe moverse de manera circular.

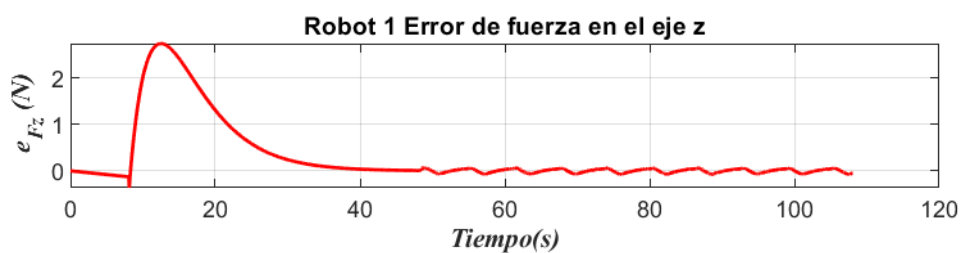


Figura 5.13: Error de fuerza

Las variaciones de fuerza se aprecian en la Figura 5.13. Por último se tiene la penetración sobre la mesa, la cual se observa en la Figura 5.14.



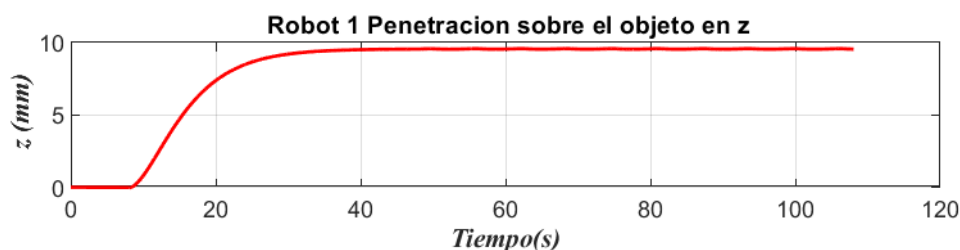


Figura 5.14: Penetración sobre la superficie

El seguimiento de trayectoria deseado vs el seguimiento que realiza el robot se muestra en la Figura 5.15 en él se aprecia que el robot se desplaza sobre el eje  $x$  para posteriormente empezar a realizar la trayectoria que se realiza en el plano  $xy$ , se tiene un ligero error que se podría reducir si se modifican las ganancias del control PI de fuerza.

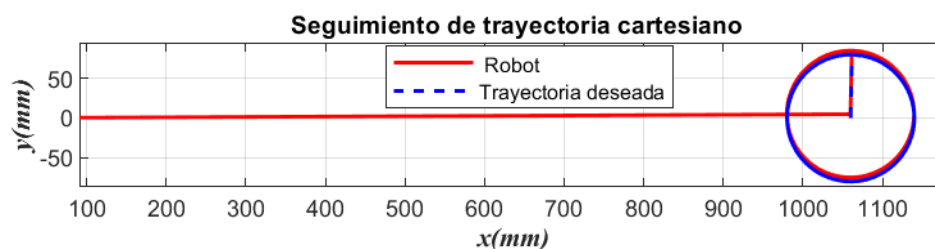


Figura 5.15: Seguimiento de trayectoria del manipulador

### 5.3. Experimento 3: Sistema cooperativo.

A continuación se muestra el sistema cooperativo compuesto por dos robots manipuladores IRB 2400, en el cual se lleva a cabo la manipulación de un objeto como se observa en la Figura 5.17.

#### 5.3.1. Movimiento horizontal del objeto a partir de un diferencial de fuerza.

Para esta sección se pretende manipular un objeto de manera horizontal a partir de un diferencial de fuerza que se genera en el robot 1 (izquierdo), dicho manipulador realiza el seguimiento de trayectoria sinusoidal sobre el eje  $x$  respecto al marco coordenado global. Al variar la referencia de fuerza se observa el movimiento en el objeto a manipular, mientras que el robot 2 (derecho) intenta mantener la fuerza en un valor de referencia constante a lo largo del experimento.

Los robots industriales tienen que realizar una tarea de manipulación de un objeto, en este caso se maneja un cubo de dimensiones  $200\text{mm} \times 200\text{mm} \times 250\text{mm}$  y el material que se utiliza gracias a la biblioteca que ofrece RobotStudio es como se observa en la Figura 5.16.

Rubber	
<b>Propiedades de superficie</b>	
Rugosidad (fricción)	1
<b>Propiedades masivas</b>	
Densidad (Kg/m <sup>3</sup> )	1500
Módulo de Young (MPa)	50
Coefficiente de Poisson	0.5
Coefficiente de restitución	0.9
Amortiguaciones de contacto	0.2

Figura 5.16: Propiedades del objeto a manipular

En la Figura 5.17 se muestra el sistema cooperativo realizado en la plataforma virtual RobotStudio, en este caso ambos robots tienen instalado en su efector final una herramienta que simula el sensor de fuerza como el que se utiliza en trabajos como [22] y [24]. De esta forma el robot izquierdo impone una trayectoria de fuerza, mientras que el robot derecho queda en reacción. El objeto a manipular se encuentra entre los dos robots, en una mesa que sirve de apoyo, cabe mencionar que RobotStudio ofrece diferentes características para los objetos que se pueden diseñar en la misma plataforma, siendo así que al objeto se le dieron propiedades físicas dinámicas y a la mesa se le dieron propiedades físicas de comportamiento anclado, esto quiere decir que la mesa interactúa con otros objetos simulados, pero sigue en una posición fija, mientras que, para el objeto el movimiento se controla mediante la simulación física.

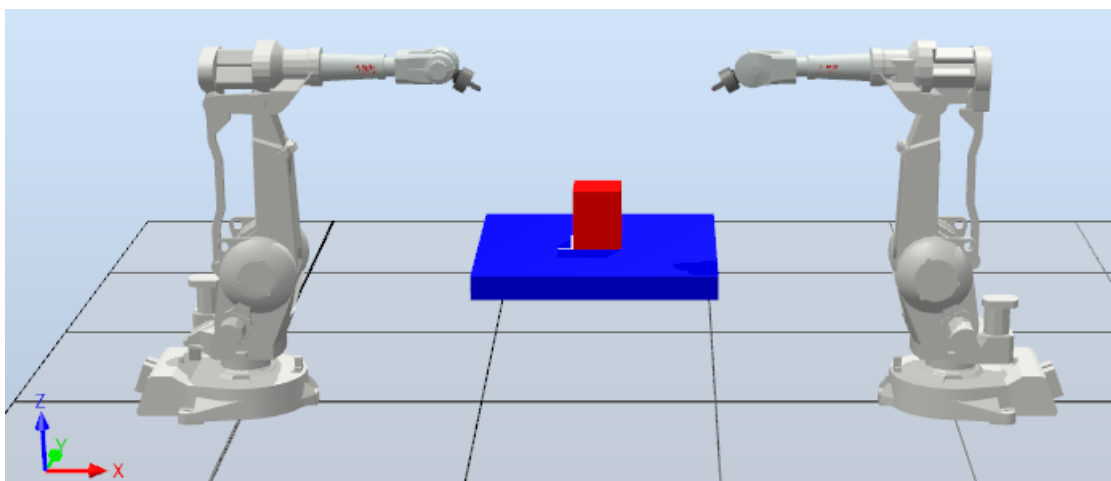


Figura 5.17: Sistema cooperativo en RobotStudio

Las tareas para el desarrollo del experimento consisten en los siguientes pasos:

#### **Robot 1 (izquierdo)**

1. Rutina de inicio donde se limpian variables y se envía al robot a la posición denominada HOME, véase la Tabla 5.4.

Posición $[x, y, z]$	[1015.2146, 0, 1362.5]
Orientación $[q_1, q_2, q_3, q_4]$	[0.5, 0, 0.8660, 0]

Tabla 5.4: Posición del efector final experimento 3 (home 1)

2. Mover el efector final del manipulador a una posición muy cercana al objeto, una vez posicionado se envía una señal al controlador del robot 2 para realizar coordinadamente la etapa de agarre, véase la Tabla 5.5. (*Tiempo de ejecución 4 seg*)

Posición $[x, y, z]$	[1400, 0, 825]
Orientación $[q_1, q_2, q_3, q_4]$	[0.7071, 0, 0.7071, 0]

Tabla 5.5: Posición del efector final experimento 3 (contacto 1)

3. Se recibe la señal del controlador 2 para dar inicio a la tarea de agarre, la cual se realiza con la ley de control híbrido posición-fuerza descrita en el Capítulo 3. El valor de referencia de la fuerza deseada es  $20N$  para asegurar el agarre del objeto. La trayectoria de fuerza se observa en la ecuación (5.1), misma que se mencionó en el Capítulo 4. Una vez realizada la tarea de agarre se envía nuevamente una señal al controlador 2 para indicar que dicho manipulador ha terminado la tarea. (*Tiempo de ejecución de 4 a 30 segundos*)

$$f_{dx} = f_d (1 - e^{-0.2(t-t_i)}) \quad (5.1)$$

4. Se activa la ley de control híbrido posición-fuerza/par extendido, correspondiente a la ecuación (3.6), para lograr una posición deseada ( $x_d$ ) y una fuerza deseada en el eje  $x$ . La trayectoria de posición deseada se observa en la ecuación (5.2)

$$x_d = x_0 + 10 \sin(0.2(t - t_{i2})) \quad (5.2)$$

Donde  $x_0$  es la posición en el eje  $x$  que se toma al inicio de este paso. Cabe mencionar que dicha trayectoria genera un error de posición que posteriormente se convierte en un diferencial de fuerza.

### Robot 2 (derecho)

1. Rutina de inicio donde se limpian variables y se envía al robot a la posición denominada HOME, véase la Tabla 5.6.

Posición $[x, y, z]$	[1015.2146, 0, 1362.5]
Orientación $[q_1, q_2, q_3, q_4]$	[0.5, 0, 0.8660, 0]

Tabla 5.6: Posición del efector final experimento 3 (home 2)

2. Mover el efector final del manipulador a una posición muy cercana al objeto, una vez posicionado se envía una señal al controlador del robot 1 para realizar coordinadamente la etapa de agarre, véase la Tabla 5.7. (*Tiempo de ejecución 4 seg*)

Posición $[x, y, z]$	$[1399.212, 0, 825]$
Orientación $[q_1, q_2, q_3, q_4]$	$[0, 0.7071, 0, 0.7071]$

Tabla 5.7: Posición del efector final experimento 3 (contacto 2)

3. Se recibe la señal del controlador 1 para dar inicio a la tarea de agarre, la cual se realiza con la ley de control híbrido posición-fuerza descrita en el Capítulo 3. El valor de referencia de la fuerza deseada es  $20N$  para asegurar el agarre del objeto. La trayectoria de fuerza se observa en la ecuación (5.3), misma que se mencionó en el Capítulo 4. Una vez realizada la tarea de agarre se envía nuevamente una señal al controlador 1 para indicar que dicho manipulador ha terminado la tarea. (*Tiempo de ejecución de 4 a 30 segundos*)

$$f_{dx} = f_d(1 - e^{-0.2(t-t_i)}) \quad (5.3)$$

4. Una vez que se recibe la última señal del controlador 1 el manipulador 2 está en reacción lo cual indica que dicho manipulador debe mantener el punto de contacto en todo momento y el valor de fuerza deseada, ya que este contiene solamente el control híbrido posición fuerza.

Las ganancias para el robot 1 (izquierdo) para el control PI de fuerza son  $k_p = 0.0000001$  y  $k_i = 1.2$ . Para el robot 2 se tienen ganancias  $k_p = 0.000000009$  y  $K_i = 0.94$ . Por otra parte, el manipulador izquierdo tiene el control PID de posición-I, el cual tiene ganancias  $k_p = 0.03$ ,  $k_i = 0.015$  y  $k_d = 0.08$ .

Resultados obtenidos:

Como se puede apreciar en la Figura 5.18 se observan las gráficas correspondientes al robot 1, en el presente experimento se trabaja en el plano  $xz$ , por lo que de esta manera la gráfica en  $y$  permanece constante.

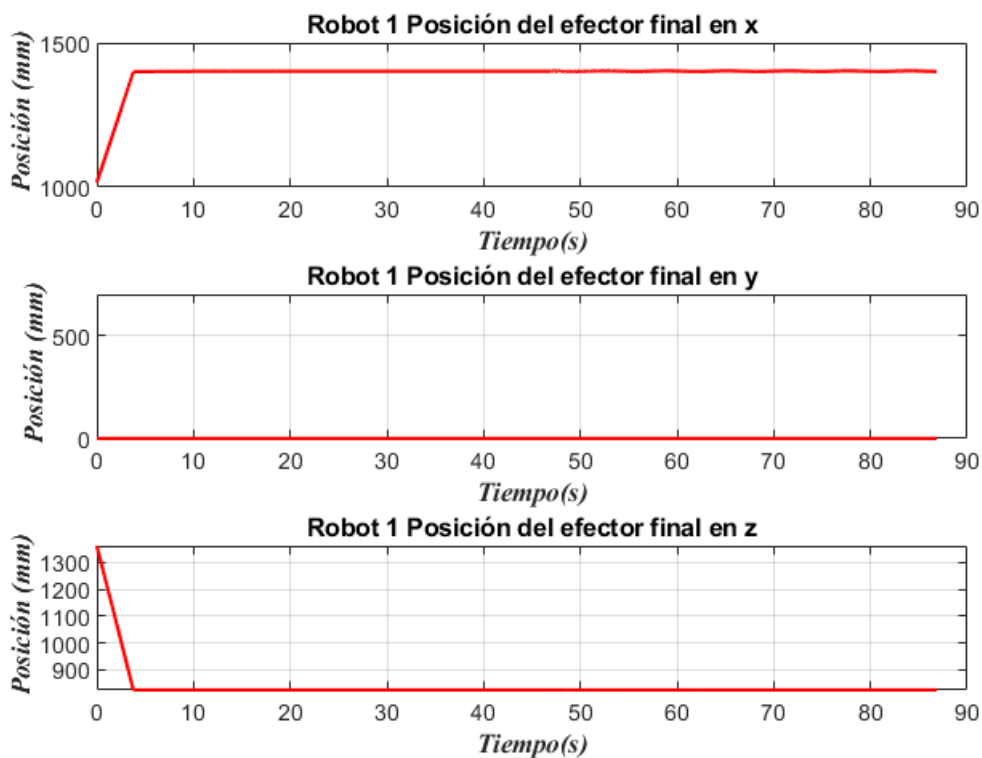


Figura 5.18: Posición x, y, z del robot 1 (izquierdo), experimento 3

Por otra parte la gráfica correspondiente a  $x$  debe oscilar de manera senoidal, por ello en la Figura 5.19 se realiza un acercamiento de tal manera que se note la oscilación del sistema, la cual ocurre a los 47 segundos. Esto se realiza con la intención de generar un movimiento horizontal sobre el objeto. En la Figura 5.18 el eje  $z$  tiene un desplazamiento que realiza el robot para llegar hasta el objeto y posteriormente permanece de manera constante en esa posición.

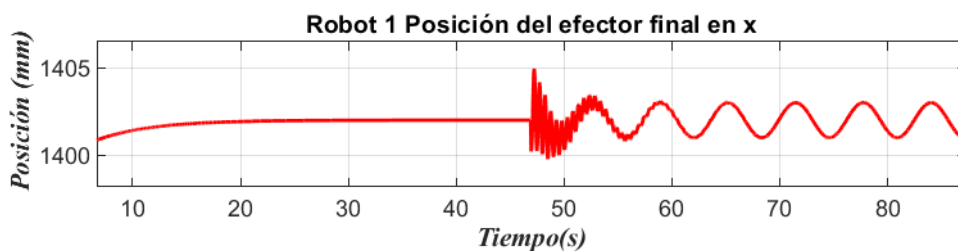


Figura 5.19: Posición del efector final robot 1 en  $x$  (acercamiento)

De manera similar se observan las gráficas correspondientes al robot 2 en la Figura 5.20, en la que de igual manera se observa que el sistema se desplaza en el plano  $xz$ , por lo que nuevamente el desplazamiento en  $y$  es constante.

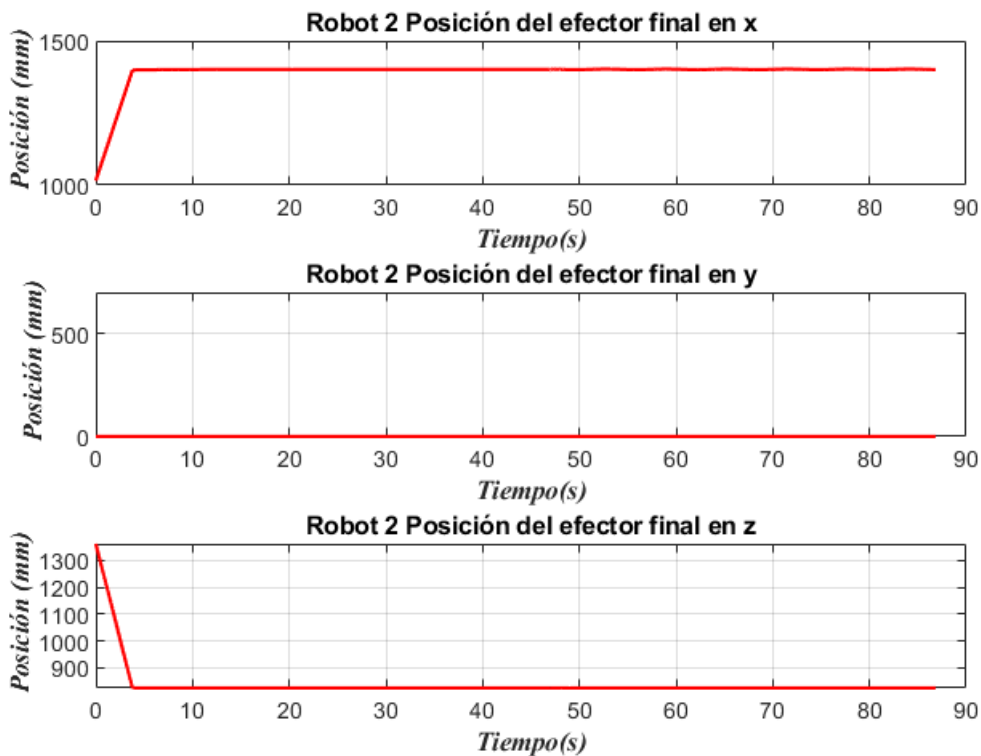


Figura 5.20: Posición  $x, y, z$  del robot 2 (derecho), experimento 3

Por otra parte, se muestra un acercamiento del desplazamiento en  $x$ , que se realiza en el segundo 47 cuando empieza el desplazamiento senoidal. En dicha gráfica lo que se observa es que el robot 2 está reaccionando al robot 1, por esta razón se activan las 2 gráficas senoidales al mismo tiempo.

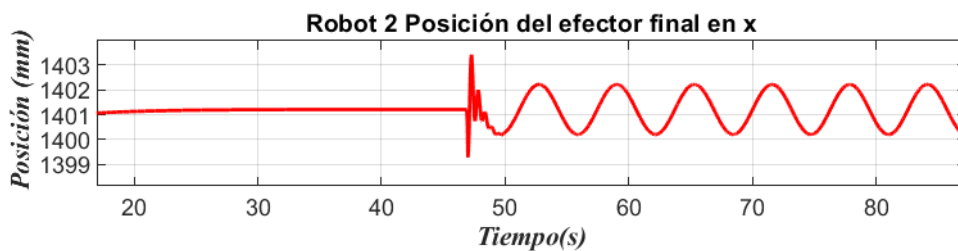


Figura 5.21: Posición del efector final robot 2 en  $x$  (acercamiento)

Como se explicó anteriormente, el robot 1 tiene la tarea de aplicar un diferencial de fuerza después de que ambos robots terminen la tarea de agarre. En la Figura 5.22 se observa que el manipulador hace contacto con el objeto a los 4 segundos, posteriormente aplica fuerza con una función suave descrita en la ecuación (5.3), de esta manera alcanza la fuerza deseada que, en este caso es de  $20N$ . Después de la etapa de agarre se aprecia una función senoidal en color azul que empieza a los 45 segundos de la simulación, en color rojo se tiene la señal del manipulador, la cual presenta oscilaciones antes de poder seguir correctamente a la trayectoria.

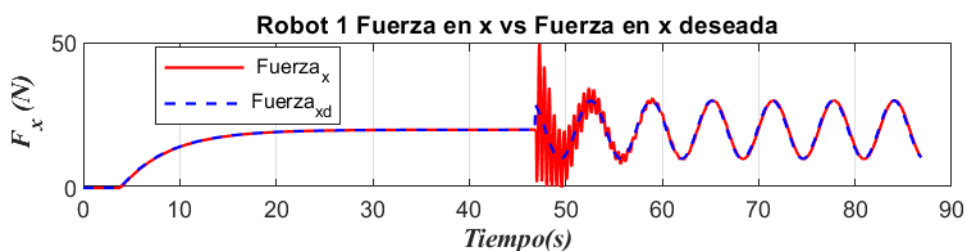


Figura 5.22: Seguimiento de fuerza robot 1 (izquierdo)

En la Figura 5.23 se muestra el error de fuerza del robot 1, el cual de los 40 a los 55 segundos el error es significativo, posterior a ese tiempo el manipulador presenta un error más pequeño.

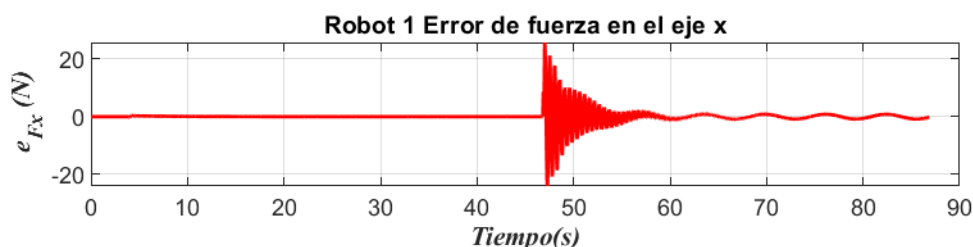


Figura 5.23: Error de fuerza Robot 1 (izquierdo)

De igual forma se tiene para el robot 2 en la figura 5.24 el seguimiento de fuerza deseada, este manipulador debe tratar de mantener un perfil de fuerza constante, en este caso de 20N.

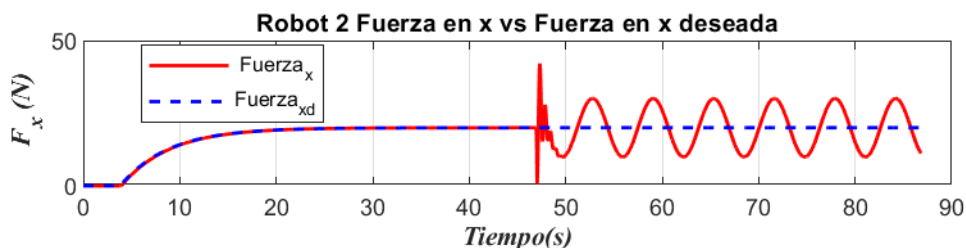


Figura 5.24: Seguimiento de fuerza Robot 2 (derecho)

En la figura 5.25 se tiene el error de fuerza, en el cual después de los 47 segundos presenta ligeras oscilaciones, y dicho manipulador reacciona al diferencial de fuerza que aplica el robot 1.

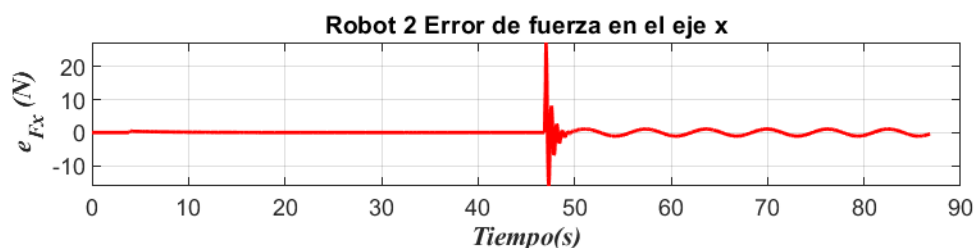


Figura 5.25: Error de fuerza Robot 2 (derecho)

El error de fuerza mostrado en las figuras 5.23 y 5.25 son las señales que entran al bloque de control de fuerza, entregando este bloque a la salida una posición de referencia en  $x$  la cual es un movimiento horizontal en el sistema cooperativo.

#### 5.4. Experimento 4: Sistema cooperativo con interacción humana

En la presente sección se muestran los resultados del experimento realizado al implementar en el sistema cooperativo los diagramas de control descritos en el Capítulo 3 y agregar la interacción de un tercer robot, como ya se mencionó en el Capítulo 4 resulta imposible incorporar la interacción del humano en RobotStudio, y que éste interactúe con el sistema cooperativo. De tal forma que, para poder simular el sistema con interacción humana, se requiere un tercer robot con las características mencionadas en el capítulo 4.

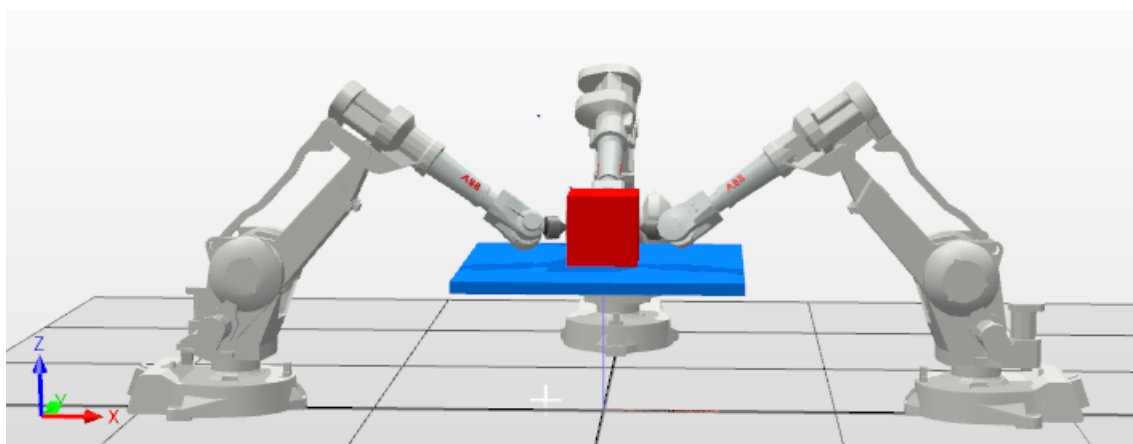


Figura 5.26: Sistema cooperativo con interacción de un tercer robot

En la Figura 5.26 se observa la distribución del sistema cooperativo con la interacción del tercer robot, de igual forma se presenta el objeto y la mesa que son modelados en el entorno de RobotStudio con las siguientes características.

- Comportamiento dinámico del objeto para que el movimiento del objeto se controle mediante la simulación.



- Se utilizó caucho para la presente simulación, las propiedades que ofrece RobotStudio se muestran en la Figura 5.27, dichas propiedades se pueden modificar con total libertad.
- Definir un origen local para el objeto.
- Comportamiento anclado de la mesa, por lo cual la mesa interactúa con otros objetos simulados, pero sigue en una posición fija.
- Definir origen local de la mesa.

Rubber	
▲ <b>Propiedades de superficie</b>	
Rugosidad (fricción)	1
▲ <b>Propiedades masivas</b>	
Densidad (Kg/m <sup>3</sup> )	1500
Módulo de Young (MPa)	50
Coefficiente de Poisson	0.5
Coefficiente de restitución	0.9
Amortiguaciones de contacto	0.2

Figura 5.27: Propiedades del objeto

La tarea de agarre del presente experimento se realiza con la interacción de los tres robots, para que esto sea posible se requiere que el robot 1 (izquierdo) y robot 2 (derecho) hagan contacto con el objeto con la orientación como se muestra en la Tabla 5.10 y la Tabla 5.13 respectivamente, de esta forma los tres manipuladores quedan configurados como se muestra en la Figura 5.28.

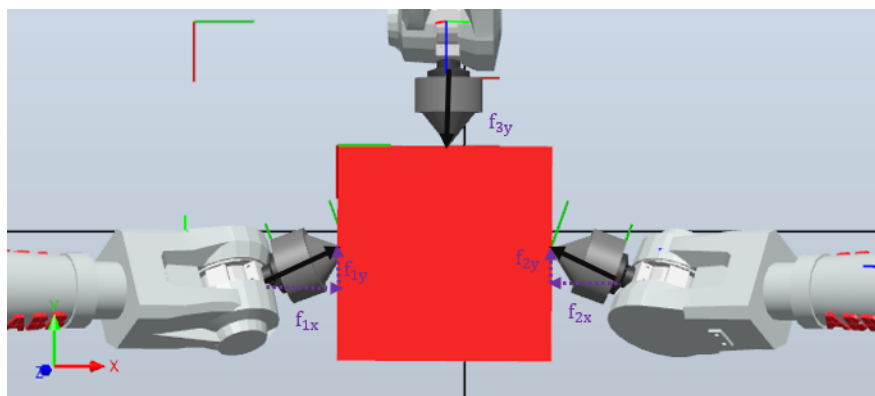


Figura 5.28: Diagrama del agarre experimento 4

Las tareas para el desarrollo del experimento consisten en los siguientes pasos para cada manipulador:

**Robot 1 (Izquierdo)**

1. Rutina de inicio donde se limpian variables y se envía al manipulador a la posición denominada HOME, véase la Tabla 5.8.

Posición $[x, y, z]$	[1015.2146, 0, 1362.5]
Orientación $[q_1, q_2, q_3, q_4]$	[0.5, 0, 0.8660, 0]

Tabla 5.8: Posición del efector final experimento 4 (home 1)

2. Mover el efector final del manipulador a una posición cercana al objeto, una vez posicionado se envía una señal al controlador del robot 2 y del robot 3 para coordinar el siguiente movimiento, véase la Tabla 5.9. (*Tiempo de ejecución 3 seg.*)

Posición $[x, y, z]$	[1217.86, 0, 830.63]
Orientación $[q_1, q_2, q_3, q_4]$	[0.7, 0, 0.71, 0]

Tabla 5.9: Posición del efector final experimento 4 (acercamiento 1)

3. Se recibe la señal del controlador 2 para realizar el movimiento que llevará al manipulador al primer contacto con el objeto, una vez que el manipulador se posicione en la etapa de contacto se envía nuevamente una señal para el controlador 2 y controlador 3 para de esta forma realizar la etapa de agarre con los tres robots, véase la Tabla 5.10. (*Tiempo de ejecución de 3 a 4.5 segundos.*)

Posición $[x, y, z]$	[1450, 0, 800]
Orientación $[q_1, q_2, q_3, q_4]$	[0.69, -0.12, 0.69, 0.12]

Tabla 5.10: Posición del efector final experimento 4 (contacto 1)

4. Se recibe señal del controlador 2 para iniciar la tarea de agarre, la cual se realiza con la ley de control híbrido posición-fuerza descrita en el Capítulo 3. El valor de referencia que se tiene para la presente simulación es de  $20N$ , pero con la posición y orientación del efector final como se muestra en la Tabla 5.10, ya que al trabajar con tres robots manipuladores se necesita una configuración distinta a las mostradas anteriormente, de esta manera cada manipulador ejerce fuerza en sus dos componentes  $(x, y)$ . Por otra parte, la trayectoria de fuerza para dicha tarea es como en la ecuación (5.4) para la componente en  $x$  y la ecuación (5.5) para la componente en  $y$ .

$$f_{dx} = 18.79(1 - e^{-0.3(t-t_i)}) \quad (5.4)$$

$$f_{dy} = 6.84(1 - e^{-0.3(t-t_i)}) \quad (5.5)$$

De tal manera que la fuerza resultante es de  $20N$  para cada manipulador.

- Por último se realiza la tarea de manipulación, la cual debe mover el objeto en el eje  $y$  del marco global pero este movimiento se genera por medio del tercer robot quedando en reacción el robot 1.

#### Resultados obtenidos robot 1:

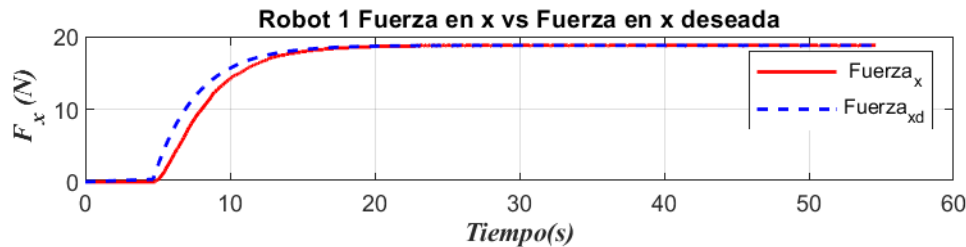


Figura 5.29: Seguimiento de fuerza en  $x$ , robot 1

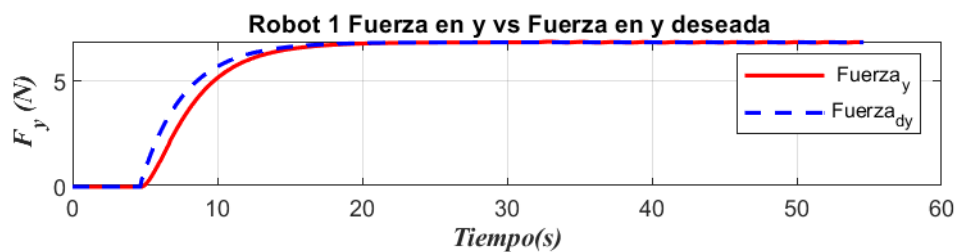


Figura 5.30: Seguimiento de fuerza en  $y$ , robot 1

Se puede observar en la Figura 5.29 y la Figura 5.30 el seguimiento de fuerza del robot manipulador y su comportamiento en función de la fuerza aplicada al objeto tanto en el eje  $x$  como en el eje  $y$ . Como se mencionó con anterioridad, el agarre en esta última etapa se realiza de acuerdo al diagrama que se muestra en la Figura 5.28, de esta forma, el primer manipulador debe ejercer una fuerza resultante de  $20N$ . Tomando en cuenta las ecuaciones (5.4) y (5.5) se tiene la fuerza resultante como en la ecuación (5.6).

$$F_{R1} = \sqrt{F_{dx}^2 + F_{dy}^2} \quad (5.6)$$

De esta forma se tiene que, el manipulador hace contacto con el objeto hasta los  $4.5$  segundos, posterior al contacto se sigue la trayectoria de fuerza de agarre correspondiente a la ecuación (5.6), hasta que se alcanza el valor de la referencia que para el eje  $x$  son  $18.79N$  y para el eje  $y$  son  $6.84N$

A continuación se muestra un acercamiento de las Figuras 5.29 y 5.30, para observar el comportamiento del manipulador en ambos ejes coordenados.

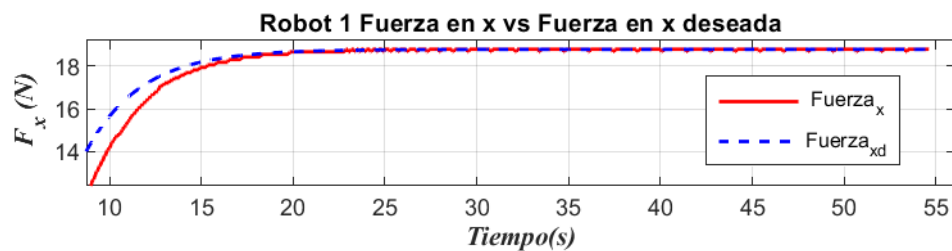


Figura 5.31: Seguimiento de fuerza en  $x$ , robot 1 (acercamiento)

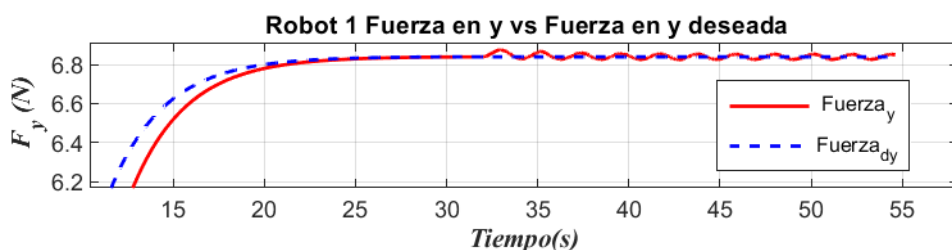


Figura 5.32: Seguimiento de fuerza en  $y$ , robot 1 (acercamiento)

En las Figuras 5.31 y 5.32 se puede observar que el valor de referencia de fuerza en color azul es fijo para ambos casos, mientras que en la Figura 5.32 la señal del robot en color rojo presenta oscilaciones, esto es debido a que el sistema cooperativo mueve el objeto en un solo eje ( $y$ ), de esta forma, cuando el tercer manipulador se empieza a mover, el primer robot (izquierdo) presentan ligeras alteraciones en sus valores de referencia como se tiene en las figuras antes mencionadas.

Por otra parte se tiene la posición del efector final que se observa en la Figura 5.34, el manipulador no presenta alteraciones en el eje  $x$  y en el eje  $z$ , pero en el eje  $y$  se tienen oscilaciones las cuales no se aprecian en dicha figura, por lo cual se presenta un acercamiento de la posición del efector en  $y$  en la Figura 5.33

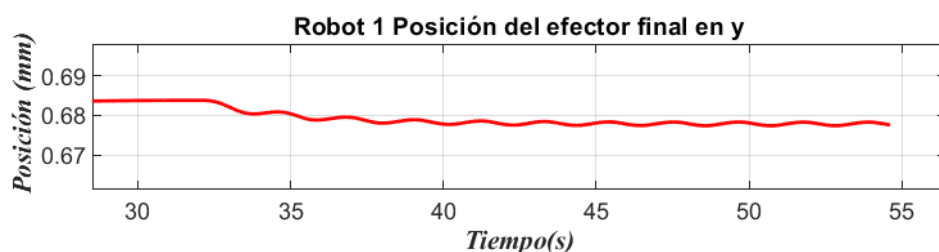


Figura 5.33: Posición del efector final en  $y$ , robot 1 (acercamiento)

El acercamiento en la Figura 5.33, permite observar que después de los 33 segundos el manipulador presenta variaciones en la posición en el eje  $y$ , ya que en él se desarrolla la tarea.

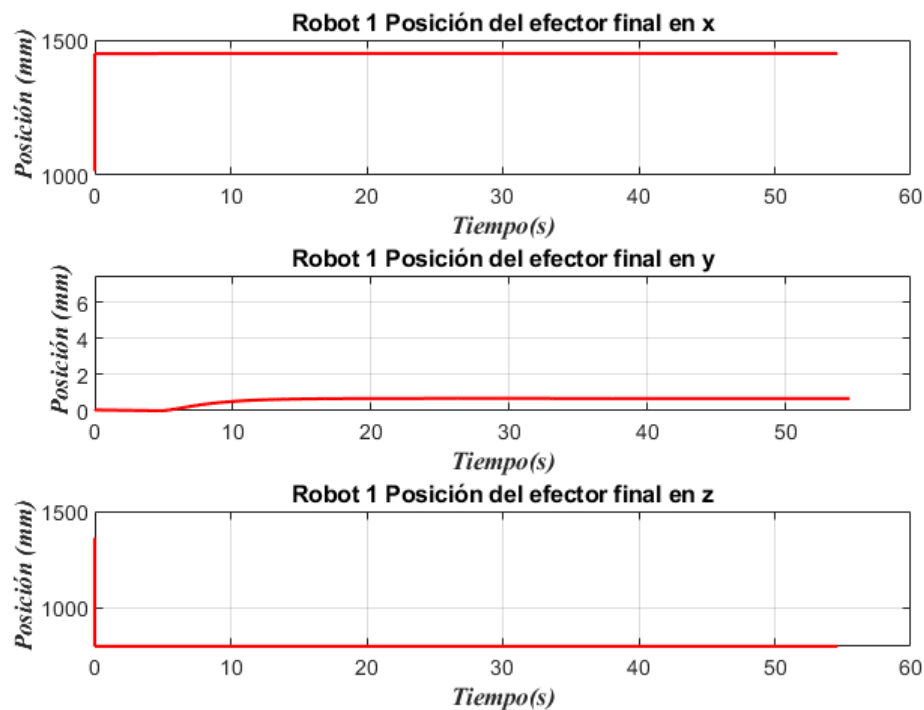
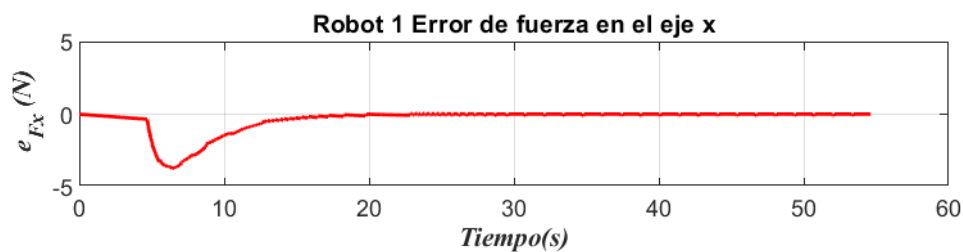


Figura 5.34: Posición del efector final en los 3 ejes, robot 1

De igual manera se presentan los errores de fuerza en cada eje, con lo cual se puede confirmar el buen funcionamiento del control de fuerza.

Figura 5.35: Error de fuerza en  $x$ , robot 1

Se observan oscilaciones pequeñas en la Figura 5.36, que representa el eje donde se realiza el movimiento, mientras que para el eje  $x$  que se observa en la Figura 5.35 no presenta mayor perturbación.

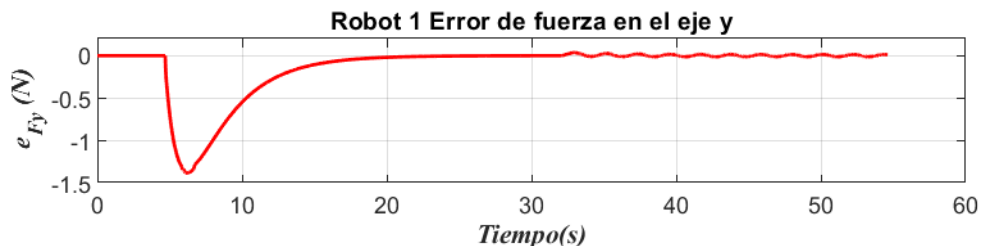
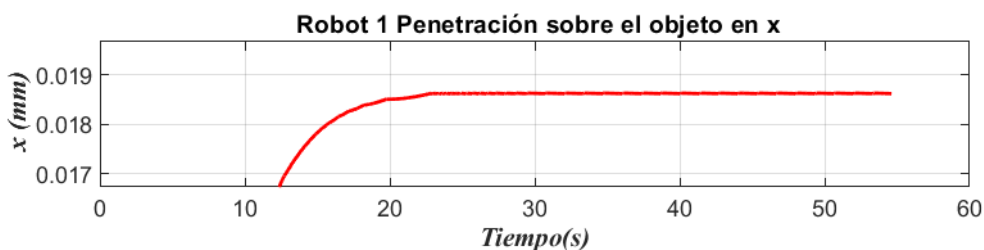
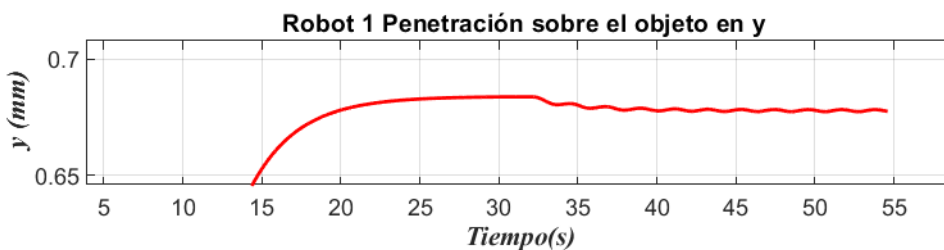


Figura 5.36: Error de fuerza en y, robot 1

Por último se presenta la penetración sobre el objeto en el eje  $x$  y en el eje  $y$ , obsérvese la Figura 5.37 para la penetración en  $x$ , en ella no se presentan variaciones en la penetración, sin embargo en la Figura 5.38 se presentan oscilaciones cuando el tercer robot empieza a empujar al objeto sobre el eje  $y$ , lo cual hace que la penetración sobre el objeto presente cambios.

Figura 5.37: Penetración sobre el objeto en el eje  $x$ , robot 1Figura 5.38: Penetración sobre el objeto en el eje  $y$ , robot 1

**Robot 2 (derecho)**

1. Rutina de inicio donde se limpian variables y se envía al manipulador a la posición denominada HOME, véase la Tabla 5.11.

Posición $[x, y, z]$	[1015.2146, 0, 1362.5]
Orientación $[q_1, q_2, q_3, q_4]$	[0.5, 0, 0.8660, 0]

Tabla 5.11: Posición del efector final experimento 4 (home 2)

2. Mover el efector final del manipulador a una posición cercana al objeto, una vez posicionado se envía una señal al controlador del robot 1 para coordinar el siguiente movimiento, véase la Tabla 5.12. (*Tiempo de ejecución 3 seg.*)

Posición $[x, y, z]$	[1217.86, 0, 830.63]
Orientación $[q_1, q_2, q_3, q_4]$	[0.7, 0, 0.71, 0]

Tabla 5.12: Posición del efector final experimento 4 (acercamiento 2)

3. Se recibe la señal del controlador 1 para realizar el movimiento que llevará al manipulador al primer contacto con el objeto, una vez que el manipulador se posicione en la etapa de contacto se envía nuevamente una señal para el controlador 1 para de esta forma realizar la etapa de agarre con los tres robots, véase la Tabla 5.13. (*Tiempo de ejecución de 3 a 4.5 segundos.*)

Posición $[x, y, z]$	[1450, 0, 800]
Orientación $[q_1, q_2, q_3, q_4]$	[0.69, -0.12, 0.69, 0.12]

Tabla 5.13: Posición del efector final experimento 4 (contacto 2)

4. Se recibe la señal del controlador 1 para iniciar la tarea de agarre, la cual se realiza con la ley de control híbrido posición-fuerza descrita en el Capítulo 3. El valor de referencia que se tiene para la presente simulación es de  $20N$ , pero con la posición y orientación del efector final como se muestra en la Tabla 5.13, como se mencionó anteriormente, de esta manera cada manipulador ejerce fuerza en sus dos componentes  $(x, y)$ . Por otra parte la trayectoria de fuerza para dicha tarea es como en la ecuación (5.7) para la componente en  $x$  y la ecuación (5.8) para la componente en  $y$ . (*Tiempo de ejecución de 4.5 segundos a 20 seg.*)

$$f_{dx} = 18.79(1 - e^{-0.3(t-t_i)}) \quad (5.7)$$

$$f_{dy} = 6.84(1 - e^{-0.3(t-t_i)}) \quad (5.8)$$

De tal manera que la fuerza resultante es de  $20N$  para cada manipulador siguiendo la ecuación (5.6).

5. Por último se realiza la tarea de manipulación la cual debe mover el objeto en el eje  $y$  del marco global, este movimiento se genera por medio del tercer robot quedando en reacción el robot 2. (*Tiempo de ejecución de 23 a 54 seg.*)

### Resultados obtenidos robot 2:

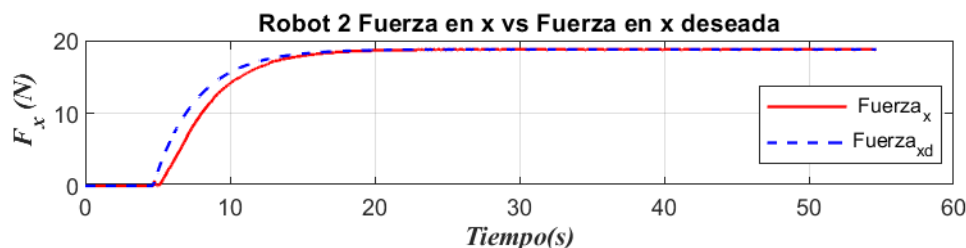


Figura 5.39: Seguimiento de fuerza en  $x$ , robot 2

Las gráficas correspondientes al robot 2 se muestran en las Figuras 5.39 y 5.40 para el seguimiento de fuerza en  $x$  y el seguimiento de fuerza en  $y$  respectivamente. Si se realiza una comparativa de las gráficas del robot 1 y del robot 2 se llega a la conclusión de que ambos manipuladores deben tener respuestas similares, ya que estos quedan en reacción al tercer robot, por lo que al coordinar sus movimientos y la etapa de agarre se esperan resultados similares.

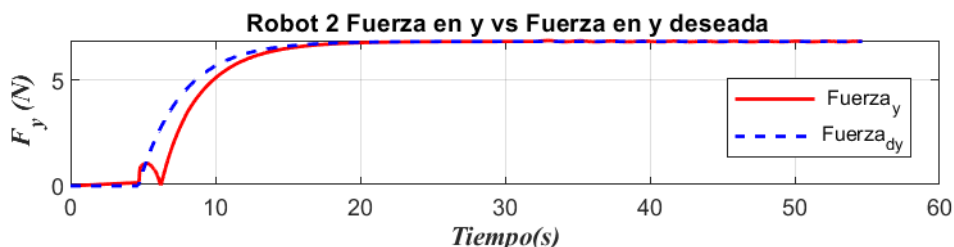


Figura 5.40: Seguimiento de fuerza en  $y$ , robot 2

De igual forma se realiza un acercamiento donde se pueda observar el comportamiento de los manipuladores en los dos ejes.



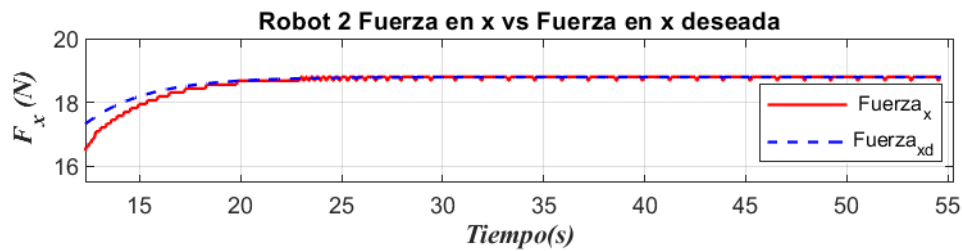


Figura 5.41: Seguimiento de fuerza en  $x$ , robot 2 (acercamiento)

Por un lado se tiene en la Figura 5.41 el seguimiento de fuerza en el eje  $x$  en el cual no se presentan mayores perturbaciones y se tiene un buen seguimiento de la fuerza deseada, en la Figura 5.42 se aprecian oscilaciones, éstas se deben a que el robot 2 (derecho) ha comenzado a desplazarse sobre el eje  $y$  ya que el tercer manipulador ha comenzado a empujar el objeto.

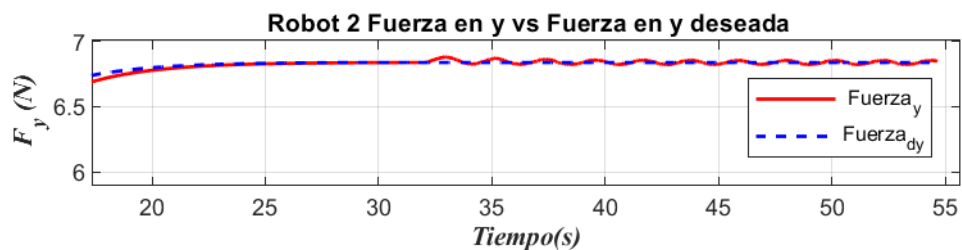


Figura 5.42: Seguimiento de fuerza en  $y$ , robot 2 (acercamiento)

La fuerza resultante se obtiene como en la ecuación (5.6) que aplica exactamente igual para ambos robots (sistema cooperativo).

La posición del efector final del robot 2 se presenta en la Figura 5.44, en ella se observa que no hay alteraciones en el eje  $x$ , así como en el eje  $z$ , sin embargo en el eje  $y$  se presenta un desplazamiento que corresponde al movimiento que realiza el tercer robot, esto se realiza a los 33 segundos de simulación, obsérvese la Figura 5.43.

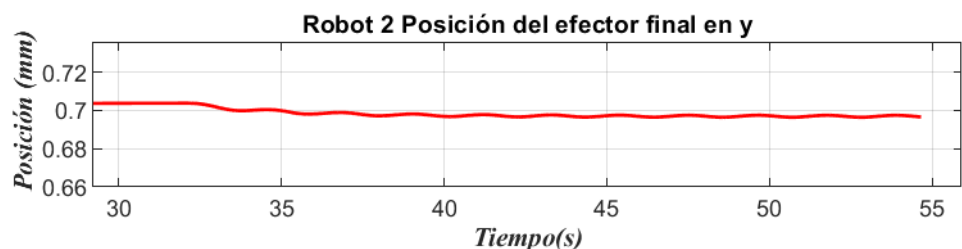


Figura 5.43: Posición del efector final en  $y$ , robot 2 (acercamiento)

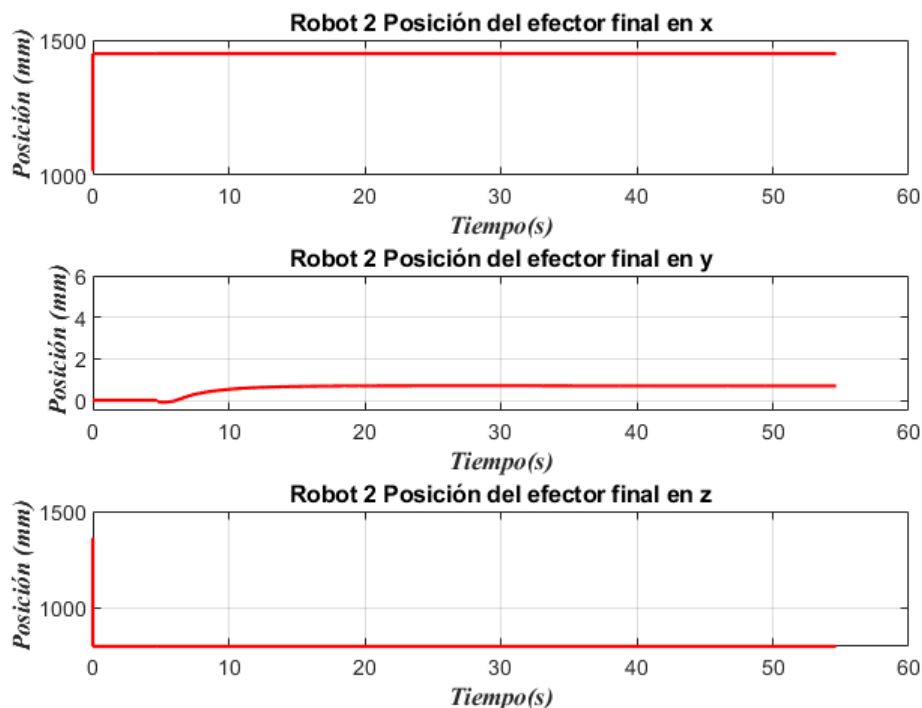


Figura 5.44: Posición del efector final en los 3 ejes, robot 2

Si se observan los errores de fuerza del robot 2 se puede confirmar el correcto funcionamiento de los controladores, obsérvese las Figuras 5.45 y 5.46, donde el error tiende a cero conforme el tiempo avanza.

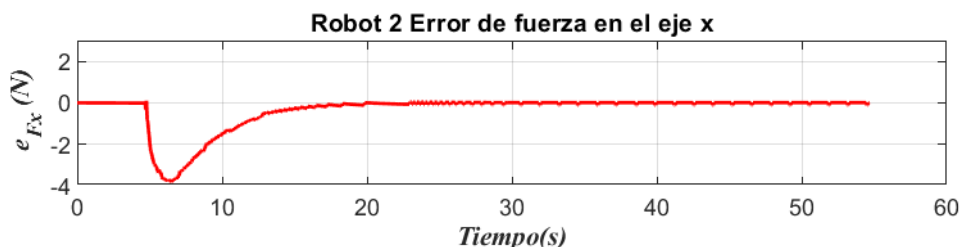


Figura 5.45: Error de fuerza en  $x$ , robot 2

Las últimas gráficas del robot 2, Figura 5.47 y Figura 5.48 se presenta la penetración en la superficie en el eje  $x$  y en el eje  $y$  respectivamente, al analizar las gráficas se comprueba que el manipulador siente la fuerza del tercer robot solamente sobre el eje  $y$ , no obstante en el eje  $x$  se tienen variaciones de menor proporción ya que en éste último no se realiza la traslación del objeto, sino en el eje  $y$ .

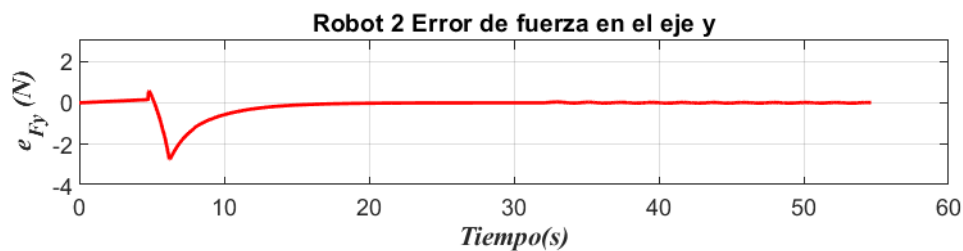
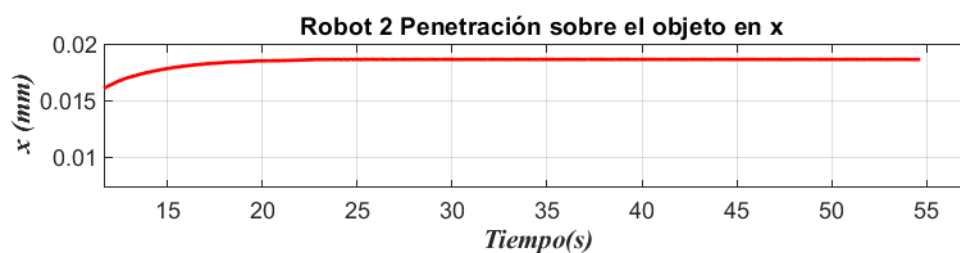
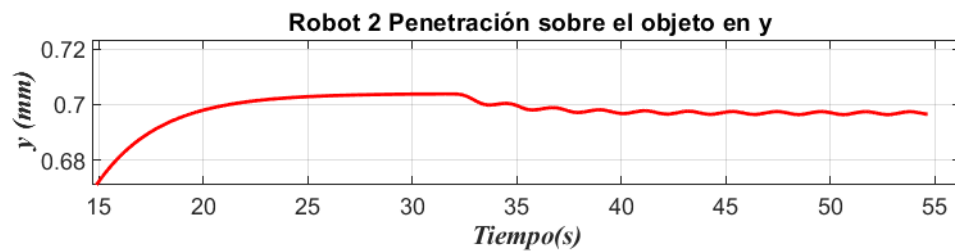


Figura 5.46: Error de fuerza en y, robot 2

La penetración en la superficie en  $x$  del robot 2 se presenta en la Figura 5.47, mientras que para el eje  $y$  se presenta en la Figura 5.48, en las que solo se tienen alteraciones en la penetración  $y$ .

Figura 5.47: Penetración sobre el objeto en el eje  $x$ , robot 2

Si se observa la penetración en el eje  $x$  se tiene que, permanece en un valor constante, lo cual indica que sobre dicho eje no realiza traslaciones, por otro lado al mover el objeto sobre el eje  $y$  éste presenta variaciones sobre dicho eje como se observa en la Figura 5.48

Figura 5.48: Penetración sobre el objeto en el eje  $y$ , robot 2

**Robot 3**

1. Rutina de inicio donde se limpian variables y se envía al manipulador a la posición denominada HOME, véase la Tabla 5.14.

Posición $[x, y, z]$	[1015.2146, 0, 1362.5]
Orientación $[q_1, q_2, q_3, q_4]$	[0.5, 0, 0.8660, 0]

Tabla 5.14: Posición del efector final experimento 4 (home 3)

2. Mover el efector final del manipulador a una posición cercana al objeto, véase la Tabla 5.15. (*Tiempo de ejecución 3 seg.*)

Posición $[x, y, z]$	[1217.86, 0, 830.63]
Orientación $[q_1, q_2, q_3, q_4]$	[0.7, 0, 0.71, 0]

Tabla 5.15: Posición del efector final experimento 4 (acercamiento 3)

3. Se recibe la señal del controlador 1 para realizar el movimiento que llevará al manipulador al primer contacto con el objeto, de esta manera el tercer robot espera la señal del controlador 1 para realizar la etapa de agarre con los tres robots, véase la Tabla 5.16. (*Tiempo de ejecución de 3 a 4.5 segundos.*)

Posición $[x, y, z]$	[1450, 0, 800]
Orientación $[q_1, q_2, q_3, q_4]$	[0.69, -0.12, 0.69, 0.12]

Tabla 5.16: Posición del efector final experimento 4 (contacto 3)

4. Se recibe la señal del controlador 1 para iniciar la tarea de agarre, la cual se realiza con la ley de control híbrido posición-fuerza descrita en el Capítulo 3. El valor de referencia que se tiene para la presente simulación en el tercer manipulador es de 40N, de esta forma se tiene que mantener el equilibrio en el objeto (agarre). (*Tiempo de ejecución de 4.5 segundos a 20 seg.*)

$$f_{dx} = 40(1 - e^{-0.3(t-t_i)}) \quad (5.9)$$

Así se logra la etapa de agarre, ya que el robot 3 aplica fuerza de tal manera que mantiene el equilibrio con la fuerza resultante del sistema cooperativo, el cual ejerce 20N por cada manipulador.

5. Por último se realiza la tarea de manipulación en la cual, el tercer manipulador debe mover el objeto en el eje  $y$  del marco global, de tal forma que al ejercer una fuerza mayor a los 40N se rompe con el estado estático del sistema. (*Tiempo de ejecución de 23 a 54 seg.*)

### Resultados obtenidos robot 3:

Recordando que la idea central del presente tema de tesis es que, el sistema cooperativo, compuesto por el robot 1 y robot 2 como se aprecia en la Figura 4.2 reaccionen a una tercer fuerza impuesta por el humano, tomando lo anterior en consideración, sabemos que, de la interacción humana no se pueden recibir o enviar señales analógicas o digitales provenientes de los manipuladores, por lo que solo se tendrían las gráficas correspondientes a los robots 1 y 2.

Al incluir un tercer manipulador en el sistema cooperativo, se necesita una programación para éste último, de esta forma se utilizan los esquemas de control mencionados en el Capítulo 3.

Cabe mencionar que el tercer manipulador debe estar en comunicación con los dos robots principales, ya que la tarea de agarre debe estar bien coordinada para garantizar que el objeto no se deslice o se mueva de lugar. Siendo así que, el tercer robot tendrá una interacción directa con el objeto en todo momento, y según el esquema descrito en la Figura 5.28 cuando el tercer manipulador ejerce una fuerza con la misma magnitud que la suma de la fuerza resultante de los robots 1 y 2, el objeto quedará en equilibrio estático, esto quiere decir que, en ese momento la suma de las fuerzas de los tres manipuladores quedarán en un valor constante igual a cero.

Caso contrario, cuando el tercer manipulador, después de la etapa de agarre ejerce una fuerza mayor a la suma de las fuerzas resultantes de los dos manipuladores, el objeto tendrá un desplazamiento sobre el eje  $y$ , al observar las gráficas se debe tener cuidado que los manipuladores tengan en todo momento contacto con el objeto, ya que de no ser así, el objeto caerá.

Por esta razón se omitirán algunas gráficas del tercer robot y se muestran solo las gráficas más representativas. Como primer gráfica se tiene el seguimiento de fuerza como se observa en la Figura 5.49

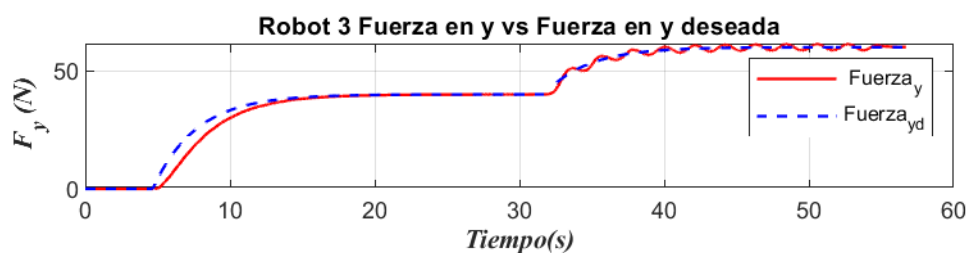


Figura 5.49: Seguimiento de fuerza en  $y$ , robot 3

En dicha figura podemos observar dos movimientos suaves de fuerza, el primero de ellos ocurre a los 4.5 segundos, en donde para garantizar la etapa de agarre el tercer manipulador debe ejercer una fuerza igual a la sumatoria de los 2 robots que constituyen el sistema cooperativo, esto se logra con un perfil de fuerza de 40N para el tercer manipulador, ya que cada robot ejerce una fuerza resultante de 20N.

El segundo movimiento suave que se observa en la Figura 5.49 ocurre a los *31.5 segundos*, esto con la intención de, después de realizada la etapa de agarre el tercer manipulador realice un diferencial de fuerza que rompa con el equilibrio de fuerzas con el sistema cooperativo, de esta forma el objeto y los dos robots se desplazarán sobre el eje  $y$ , ésto se ve reflejado en las gráficas anteriores correspondientes al robot 1 (izquierdo) y robot 2 (derecho), donde se puede observar el comportamiento del sistema cooperativo y la respuesta de los controladores, parte central del presente tema de tesis.

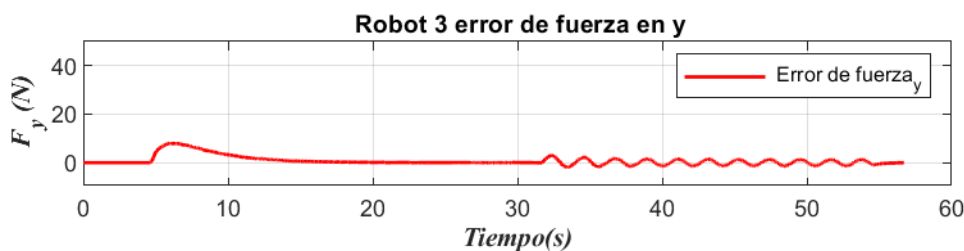


Figura 5.50: Error de fuerza en  $y$ , robot 3

Ahora bien, al analizar el error de fuerza que se tiene en el tercer manipulador en la Figura 5.50 se presentan dos momentos nuevamente, el primero que inicia a los *4.5 seg* y se logra que el error sea cero prácticamente a los *16 segundos*, dicho de otra manera el agarre en ese tiempo se ha realizado con éxito. Cabe mencionar que se agregó un tiempo extra después de la etapa de agarre, éste dura hasta los *31.5 segundos*, siendo así que, en ese instante de tiempo el tercer robot realiza un incremento en su fuerza, el cual tiene un valor de referencia de  $60N$ , no obstante, en la gráfica antes mencionada se presentan oscilaciones, las cuales se interpretan como el movimiento que realiza el sistema cooperativo, y que dicho manipulador siente la oposición de dichos robots, de la misma forma el controlador muestra que el error de fuerza es cada vez menor, por lo que se puede concluir que el tercer manipulador realiza la tarea exitosamente.

Por último se muestra en la Figura 5.51 la penetración sobre el objeto, en la cual se observa nuevamente la etapa de agarre y posterior a ello, el movimiento o la variación en la penetración sobre el objeto al sentir el tercer manipulador la oposición del sistema cooperativo.

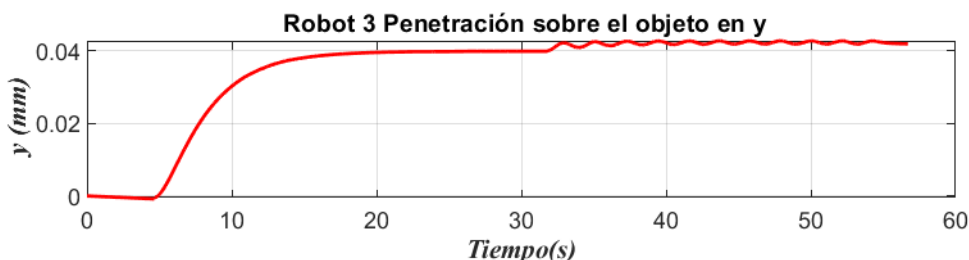


Figura 5.51: Penetración sobre el objeto en  $y$ , robot 3

Como se mencionó anteriormente, al trabajar con RobotStudio no se tiene acceso a los sensores de fuerza que ofrece ABB, por esta razón, las presentes simulaciones se realizan con el modelo de impedancia mostrado en el Capítulo 2, siendo así que, cada manipulador

tiene un modelo de impedancia dentro de su programación, por esta razón se envían los datos de la penetración a cada manipulador para que de esta forma los manipuladores se mantengan comunicados con la información de que el robot ha penetrado o ejercido más fuerza sobre el objeto, y así los robots puedan reaccionar gracias al control de fuerza.

Por ello los movimientos que se observan en la mayoría de las gráficas son relativamente pequeñas, ya que las penetraciones que se tienen en las simulaciones son milimétricas, y en consecuencia los desplazamientos de los manipuladores son pequeños también.

# Capítulo 6

## Conclusiones y trabajo futuro

### 6.1. Conclusiones

Ahora que se han visto los resultados obtenidos en el Capítulo 5 usando la plataforma virtual RobotStudio se observa que la implementación del sistema cooperativo con interacción humana resulta complicada, por una parte los robots manipuladores funcionan a la perfección en la plataforma, sin embargo se encontraron diferentes limitantes, como es el uso de los sensores de fuerza, ya que como se ha mencionado en capítulos anteriores, en la Sección de Mecatrónica y en trabajos como en [22], [23] y [24] se han usado los sensores ATI y JR3 descritos en el Capítulo 4. De esta manera los manipuladores pueden controlar sus movimientos gracias al arreglo de las galgas extensiométricas con las que cuentan los sensores, dicho de otra manera, cuando los robots tienen contacto con un objeto en común, cada uno siente la fuerza que aplica el otro manipulador, de esta forma, a través del objeto es como cada robot aplica la fuerza de referencia deseada. En el presente trabajo de tesis al no contar con los sensores antes mencionados y al no tener las librerías correspondientes a los sensores que ofrece ABB, y que estos solo se consiguen al adquirir los sensores propios de la empresa, se utiliza el modelo de impedancia que se muestra en el Capítulo 2, dicho modelo presenta varios inconvenientes al trabajar con un sistema cooperativo, ya que toma en cuenta una posición fija del objeto como si fuera una pared, para dar solución a este problema se recurrió a sumar en cada robot la penetración que tuviera el manipulador sobre la pared para tener el efecto real que se tendría en los laboratorios.

Se muestra que los lazos de control externos que se proponen funcionan correctamente en robots manipuladores de arquitectura cerrada en un ambiente virtual. Al implementar los esquemas de control, control híbrido posición-fuerza/par y el control híbrido posición-fuerza/par extendido, se observa que los errores de fuerza tienden a cero en ambos casos, siendo así que los manipuladores no pierden contacto en las tareas programadas.

ABB ofrece un kit especial de fuerza y sincronización de los robots ABB, parte central del presente trabajo es realizar la coordinación del sistema cooperativo sin dichos kits, en este caso se hace uso de banderas de sincronización utilizando entradas y salidas digitales virtuales de los controladores, los cuales no permiten que los robots continúen hasta que se dé la señal del resto de los robots, es importante mencionar que al ejecutar el programa sin las banderas de sincronización en RobotStudio los manipuladores empiezan en tiempos diferentes, esto es debido a que las instrucciones en el programa se ejecutan de manera secuencial, por ello la relevancia de desarrollar tareas bien coordinadas.



De esta manera en los dos primeros casos experimentales se muestran las gráficas correspondientes al control híbrido posición-fuerza/par, en el cual, la principal tarea es aplicar fuerza sobre una mesa, en este experimento lo relevante es mostrar que la herramienta que se instala en la brida del robot puede simular, debido al modelo de impedancia, al sensor de fuerza, y el control híbrido posición-fuerza/par extendido que tiene como tarea imponer trayectorias al manipulador sin dejar de aplicar fuerza en el objeto. Una consideración importante es la magnitud de la fuerza que se aplica sobre los objetos, esto se observa en el tercer experimento, donde se tiene que manipular un objeto horizontalmente, ya que, sino se aplica una fuerza para garantizar el agarre del objeto, éste podría presentar deslizamientos o pérdida de contacto.

Los resultados de simulación que se obtuvieron son satisfactorios, tanto para un manipulador como para el sistema cooperativo. Es importante mencionar que al programar robots virtuales y usando un modelo matemático para poder simular el sensor de fuerza, estos no presentan tiempos de muestreo que se tendrían al usar los sensores antes mencionados. Se menciona en [23] que los tiempos de muestreo para el sensor JR3 es de 50[ms], mientras que para el sensor ATI se tiene un tiempo de muestro de 75[ms].

Al ser uno de los primeros trabajos en la Sección de Mecatrónica en el Cinvestav, en realizar el sistema cooperativo en RobotStudio, se buscó obtener el mejor desempeño en los casos presentados en el Capítulo 5, sin embargo al no tener los sensores físicamente se vuelve complicado realizar las simulaciones, ya que al trabajar con la cinemática y dinámica de los manipuladores y con los gráficos que ofrece el software, las simulaciones se vuelven considerablemente lentas. Por otra parte, al programar diferentes secuencias o tareas en RAPID para varios robots manipuladores, éstos no se ejecutan al mismo tiempo, por lo que hacer uso de la lógica de estación, nos permite coordinar adecuadamente dichos manipuladores, y de esta forma, sin importar que alguno comience las trayectorias antes, éste no podrá continuar con las tareas siguientes hasta que reciba las señales necesarias provenientes de los otros manipuladores para continuar.

Existen diferentes aplicaciones para el sistema cooperativo con interacción humana, la más representativa es para tareas de ensamble, ya que un sistema cooperativo en la mayoría de los casos no tiene un conocimiento del entorno que lo rodea, y al guiar su movimiento por un humano, este resultaría factible.

## 6.2. Trabajo futuro

Al no tener acceso a los laboratorios a causa del virus SARS-CoV2(covid-19), los experimentos planeados al inicio de la tesis no se completaron debido al cierre de las instalaciones del CINVESTAV, debido a esto las pruebas faltantes se proponen como trabajos futuros, quedando como antecedente las simulaciones realizadas en RobotStudio.

- Finalizar los experimentos relacionados al sistema cooperativo con interacción humana, así como expandir las tareas en el espacio tridimensional, ya que en el presente

trabajo solo se tienen tareas en un espacio bidimensional por las limitaciones antes mencionadas.

- Aumentar el número de robots en el sistema.
- Implementar el sistema cooperativo con interacción humana para tareas de ensamblaje.
- Desarrollar un sistema que aporte seguridad para el humano que interactúa de manera directa con los manipuladores.
- Realizar el análisis de estabilidad de todo el sistema en lazo cerrado.
- Agregar un tercer robot teleoperado para imponer trayectorias en un espacio bidimensional.

# Bibliografía

- [1] ABB. *Go far with your new helping hand*. Accedido en 09/06/2021. <https://new.abb.com/products/robotics/collaborative-robots/crb-15000>
- [2] NASA. Rover Perseverance de la NASA aterriza con éxito en Marte. Accedido en 10/07/2021. URL: <https://www.libertaddigital.com/ciencia-tecnologia/ciencia/2021-02-18/perseverance-marte-tecnologia-espanola-mision-nasa-rover-vehiculo-espacial-sener-csic-inta-cab-6711252/>
- [3] Reyes, Fernando. *Robótica-control de robots manipuladores*, Primera edición. Alfaomega grupo editor, 2011.
- [4] John J. Craig. *Robótica*. Tercera edición. Pearson, Prentice Hall, 2006.
- [5] ABB. *MANual del operador RobotStudio*. Accedido en 4/06/2021. URL: [https://library.e.abb.com/public/6aeb483836740e11c1257b4b0052375b/3HAC032104-005\\_revE\\_es.pdf](https://library.e.abb.com/public/6aeb483836740e11c1257b4b0052375b/3HAC032104-005_revE_es.pdf)
- [6] ABB. *MANual de referencia técnica, instrucciones, funciones y tipos de datos de RAPID*. Accedido en 4/07/2021. URL: <http://personal.biada.org/~jhorriillo/INSTRUCCIONES%20RAPID.pdf>.
- [7] SONY. AIBO el perro robot de SONY. Accedido en 19/07/2021. URL: <https://es.digitaltrends.com/tendencias/aibo-robot-perro-mascota-sony/>
- [8] Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani y Giuseppe Oriolo. *Robotics: modelling, planning and control*. Springer, 2009.
- [9] Liste, V., Anigstein, M. (2007). Influencia de la rigidez del sistema y la frecuencia de correcciones en el control de fuerzas de robots industriales. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 4(4), 39-50.
- [10] Pires, J. N., Godinho, T. (2008). Control difuso de fuerza en robótica industrial. *Ingenierías*, 11(41), 55-62.
- [11] Mark W. Spong, Seth Hutchinson y M. Vidyasagar. *Robot modelling and control*. First edition. John Wiley y Sons, Inc., 1989.
- [12] Hu, Y. R., Goldenberg, A. A. (1989, May). An adaptive approach to motion and force control of multiple coordinated robot arms. In *Proceedings, 1989 International Conference on Robotics and Automation* (pp. 1091-1096). IEEE.
- [13] Sánchez-Sánchez, P., Arteaga-Pérez, M. A. (2020). Control de posición y fuerza con estimación de masa para sistemas cooperativos. *Revista Iberoamericana de Automática e Informática industrial*, 17(4), 368-379.

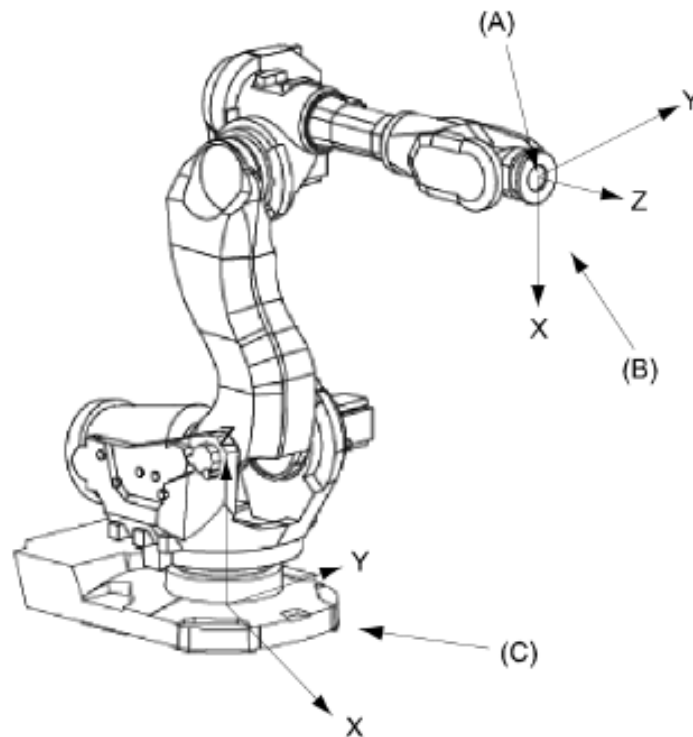
- [14] Díaz Baca, C.S., Puente Méndez, S.T., Torres, F.(2006). Interacción hombre-robot trabajando cooperativamente en una celda de desensamblado automático. Universidad de Alicante. Accedido en 6/08/2021. <https://rua.ua.es/dspace/bitstream/10045/2584/1/0216.pdf>.
- [15] Jara, C. A. (2007). Diseño de herramientas para la interacción remota on-line de robótica industrial. (Tesis Doctoral). Universidad de Alicante. Accedido en 9/08/2021. <http://rua.ua.es/dspace/handle/10045/2150>
- [16] Medina, J. (2017). Artículo. Avances en la interacción física segura humano-robot y el desarrollo de sistemas elásticos articulares Universidad Carlos III de Madrid.
- [17] Garcia, J. G., Robertsson, A., Ortega, J. G., Johansson, R. (2007). Estimación de la fuerza de contacto para el control de robots manipuladores con movimientos restringidos. Revista Iberoamericana de Automática e Informática Industrial RIAI, 4(1), 70-82.
- [18] Popov, V. L., Martín-Martínez, J. M., Moreno Flores, S. (2020). Principios y aplicaciones de la mecánica de contacto en tribología, fricción y adherencia. Revista de Universidad de Alicante.
- [19] Matos Agudo, B. (2017). Modelado, programación y simulación del Robot IRB 120 de ABB con RobotStudio. Tesis de maestría. Universidad de Sevilla.
- [20] Hanafusa, M., Ishikawa, J. (2020, January). A Practical Method for Acquiring Inverse Dynamics Model and its Application to Mechanical Impedance Control of Human-Cooperative Robot. In 2020 IEEE/SICE International Symposium on System Integration (SII) (pp. 579-584). IEEE.
- [21] Lu, W. S., Meng, Q. H. (1991). Impedance control with adaptation for robotic manipulations. IEEE Transactions on Robotics and Automation, 7(3), 408-415.
- [22] Luis Uriel Hernández. (2019). *Control para sistemas robóticos cooperativos en tareas de manipulación en interacción con el ambiente*. Tesis de maestría. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV-IPN)
- [23] Michelle Acevedo Velázquez. (2020). *Teleoperación háptica de sistemas de robots industriales cooperativos* Tesis de maestría. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV-IPN).
- [24] Gabriel Cubas Perfecto. (2020). *Interacción humano-robot para manipulación guiada*. Tesis de maestría. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV-IPN).
- [25] Conde Canaviri, M. M. (2008). Generaciones de la robótica. Revista de Información, Tecnología y Sociedad, 129. Recuperado de <https://docplayer.es/43450743-Generaciones-de-la-robotica.html>
- [26] Rivera-Duenas, J. C., Arteaga-Pérez, M. A. (2013). Robot force control without dynamic model: theory and experiments. Robotica, 31(1), 149-171.
- [27] Raibert, M. H., Craig, J. J. (1981). Hybrid position/force control of manipulators. Jet Propulsion Laboratory, California Institute of Technology, Pasadena, Calif. 91103.

- [28] Jesús A. López Silvano. *Diseño e implementación de sistema cooperativo de robots industriales para tareas de manipulación*. Tesis de maestría. Centro de Investigación y de Estudios Avanzados del Instituto Politecnico Nacional (CINVESTAV-IPN). 2018
- [29] Brumovsky, A. E., Liste, V. M., Anigstein, M. (2006). Implementación de control de fuerzas en robots industriales: un caso. IV Jornadas Argentinas de Robótica, JAR08. Córdoba. Accedido en 10/08/2021. <https://www.frc.utn.edu.ar/jar2006/docs/papers/008-jar06.pdf>
- [30] ATI INDUSTRIAL AUTOMATION. F/T Sensor: Mini40. Accedido en 11/10/2021. [https://www.ati-ia.com/productsft/ft\\_models.aspx?id=Mini40](https://www.ati-ia.com/productsft/ft_models.aspx?id=Mini40).
- [31] *F/T Transducer, Six-Axis Force/Torque Sensor System, Installation and Operation Manual*. Document 9620-05-Transducer Section-28. ATI INDUSTRIAL AUTOMATION.
- [32] *JR3 Multi-Axis Force-Torque Sensor Technical specification*. JR3 INC.
- [33] Craig, J., Hsu, P., Sastry, S. (1986, April). Adaptive control of mechanical manipulators. In Proceedings. 1986 IEEE international conference on robotics and automation (Vol. 3, pp. 190-195). IEEE.
- [34] R. Kelly, V. Santibañez, and A. Loria. *Control of Robot Manipulator in Joint Space.*, Springer-Verlag, 2005.
- [35] Caccavale, F., Chiacchio, P., Marino, A., Villani, L. (2008). Six-dof impedance control of dual-arm cooperative manipulators. *IEEE/ASME Transactions On Mechatronics*, 13(5), 576-586.
- [36] Bogdan, R. (2012). Implementación de un sensor de fuerza par en un robot ABB IRB 1600.(II) Tesis de licenciatura. Universidad Carlos III de Madrid
- [37] *Especificaciones del producto: IRB 2400*. ID: 3HAC042195-005, Revisión: P(2021- 09-16). ABB.
- [38] Neville Hogan. Impedance control: An approach to manipulation: Part I - theory. En: *Journal of Dynamic Systems, Measurement, and Control* Vol. 107 (mar. de 1985), págs. 1-7. DOI: 10.1115/1.3140702.
- [39] ABB. Integrate Force Control Adaptive motion technology. Accedido en 15/01/2022. <https://search.abb.com/library/Download.aspx?DocumentID=9AKK10103A6006&LanguageCode=en&DocumentPartId=&Action=Launch>
- [40] Soheil Sarabandi y Federico Thomas. Accurate computation of quaternions from rotation matrices. En: *Advances in robot kinematics 2018*. Springer, 2018, págs. 39-46. DOI: 10.1007/978-3-319-93188-3\_5.

# Apéndice A

## Conceptos generales.

**Punto central de la herramienta (TCP):** Por sus siglas en inglés Tool center point, es el punto cuyas coordenadas se almacenan en el programa. Se puede definir un TCP por cada herramienta que posea el robot.



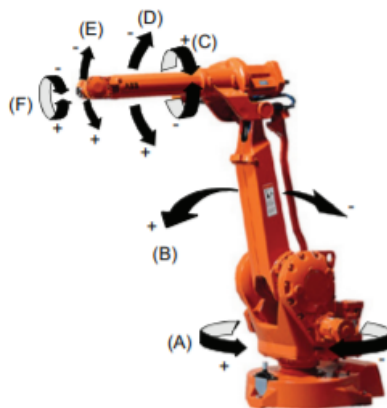
**Presición:** Distancia entre el punto programado y el punto realmente alcanzado (valor medio tras varios ciclos). Su valor está limitado por errores de calibración del robot, deformaciones, errores de redondeo en el cálculo (puntos singulares) o por errores entre las dimensiones teóricas y reales del robot.

**Repetibilidad:** Es el radio de la esfera que comprende los puntos alcanzados por el robot tras realizar suficientes movimientos, al ordenarle ir al mismo punto de destino programado, con condiciones iguales de carga, temperatura, etc. Su valor está limitado por problemas en el sistema mecánico de transmisión (rozamientos, histéresis, zonas muertas).



**Espacio de trabajo:** Está definido como el grupo de puntos que pueden ser alcanzados por su efector final, las especificaciones que se manejan mas adelante pueden incluir ciertas regiones del espacio de trabajo en las que el robot no pudiera alcanzar determinadas especificaciones de velocidad o carga.

**Grados de libertad:** A cada movimiento independiente que es capaz de realizar una articulación se le denomina GDL (Grado de Libertad), considerando un espacio tridimensional, el máximo de grados de libertad es de 6.



**Entorno de trabajo:** En muchas aplicaciones industriales los robots sustituyen a los operarios humanos, esto con la finalidad de cuidar la integridad del humano, ya que muchas tareas son en áreas potencialmente peligrosas, como ambientes contaminados, bajas o altas temperaturas. De esta manera los robots manipuladores son diseñados tomando en cuenta el entorno de trabajo.

**Puntos singulares:** Es aquella en la que un manipulador robótico pierde grados de libertad debido al alineamiento de dos o más ejes, ocasionando un comportamiento impredecible tal que pequeños movimientos del efector final del robot pueden suponer velocidades articulares de magnitud descontrolada [33].

Matemáticamente se puede decir que no tiene solución única, es decir, no existe matriz jacobiana inversa, el determinante es nulo.

**Velocidad:** El dato proporcionado en los catálogos corresponde a la velocidad en régimen permanente (velocidad nominal), que es la que podría alcanzar el robot en largas distancias. En movimientos cortos son más significativos los tiempos de arranque y parada; es por ello que, en algunos robots, en lugar de la velocidad se indica el tiempo empleado en realizar un movimiento determinado.



## **Apéndice B**

### **Especificaciones técnicas del robot ABB IRB2400.**

# 1 Descripción

## 1.1 Estructura

### 1.1.1 Introducción a la estructura

---

#### Familia de robots

El IRB 2400 es un robot industrial de 6 ejes diseñado específicamente para industrias de fabricación que utilizan una automatización flexible basada en robots. El robot tiene una estructura abierta especialmente adaptada para un uso flexible y presenta unas grandes posibilidades de comunicación con sistemas externos.

---

#### Sistema operativo

El robot está equipado con el controlador IRC5 y el software de control de robots RobotWare. RobotWare admite todos los aspectos del sistema de robot, como el control del movimiento, el desarrollo y la ejecución de programas de aplicación, la comunicación, etc. Consulte las Especificaciones del producto - Controlador IRC5 con FlexPendant.

---

#### Seguridad

Normas de seguridad válidas para todo el robot, manipulador y controlador.

---

#### Funcionalidad adicional

Para una funcionalidad adicional, el robot puede equiparse con software opcional para compatibilidad con aplicaciones –como por ejemplo aplicación de adhesivo y soldadura–, características de comunicación –comunicación de red– y funciones avanzadas, como la multitarea, el control de sensores, etc. Para ver una descripción completa del software opcional, consulte las Especificaciones del producto - Software de controlador IRC5.

---

#### Foundry Plus

##### Foundry Plus

La opción Foundry Plus ha sido diseñada para entornos agresivos en los que el robot está expuesto a los sprays de refrigerantes, lubricantes y virutas metálicas típicos de las aplicaciones de fundición a presión u otras aplicaciones similares. Las aplicaciones típicas son la inserción con pulverización y la extracción de piezas de máquinas de fundición a presión, manipulación en la fundición con macho de arena y la fundición por gravedad, etc. Se debe prestar una atención especial con respecto a los requisitos de funcionamiento y mantenimiento para las aplicaciones de fundición, así como en otras áreas de aplicaciones. Póngase en contacto con la organización comercial de ABB Robotics en caso de duda acerca de la viabilidad de aplicaciones específicas para el robot Foundry Plus. El robot Foundry Plus está pintado con pintura epoxi bicomponente sobre una imprimación para la protección contra la corrosión. Para aumentar aún más la protección contra la corrosión, se aplica antioxidante a las áreas cruciales expuestas, es decir, la brida para herramientas tiene un tratamiento preventivo especial. No obstante, la salpicadura

*Continúa en la página siguiente*

## 1 Descripción

### 1.1.1 Introducción a la estructura

#### Continuación

continuada con agua u otros líquidos similares que dan lugar a oxidación puede dar lugar a ataques por óxido en las áreas no pintadas del robot, las articulaciones u otras superficies no protegidas. En estas circunstancias se recomienda añadir un inhibidor del óxido al líquido o tomar otras medidas para impedir la posible formación de óxido en las zonas mencionadas. Todo el robot presenta la protección IP67 de acuerdo con la norma CEI 60529 desde la base hasta la muñeca (excepto el IRB 2400L, IP67 sólo en la muñeca y los conectores), lo que significa que los compartimentos eléctricos están sellados frente al agua y contaminantes sólidos. Entre otras cosas, todos los componentes sensibles están mejor protegidos que en la versión estándar.

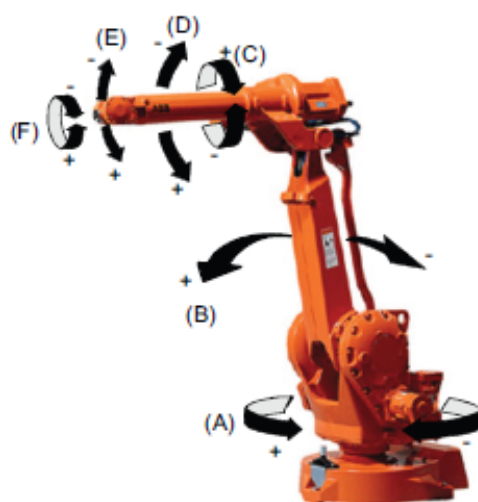
Características de Foundry Plus seleccionadas:

- Mejor sellado para prevenir la penetración en las cavidades para garantizar el grado de protección IP67
- Protección adicional del cableado y los componentes electrónicos
- Cubiertas especiales para proteger las cavidades
- Conectores contrastados
- Antioxidantes en los tornillos, arandelas y superficies no pintadas/mecanizadas
- Programa extendido de servicio y mantenimiento

El robot Foundry Plus puede limpiarse con equipos de lavado adecuados.

El robot cuenta con una etiqueta "Foundry" (IRB 2400F/L) o "Foundry Plus" (IRB 2400F/10 y F/16).

#### Ejes del manipulador



xxx110000569

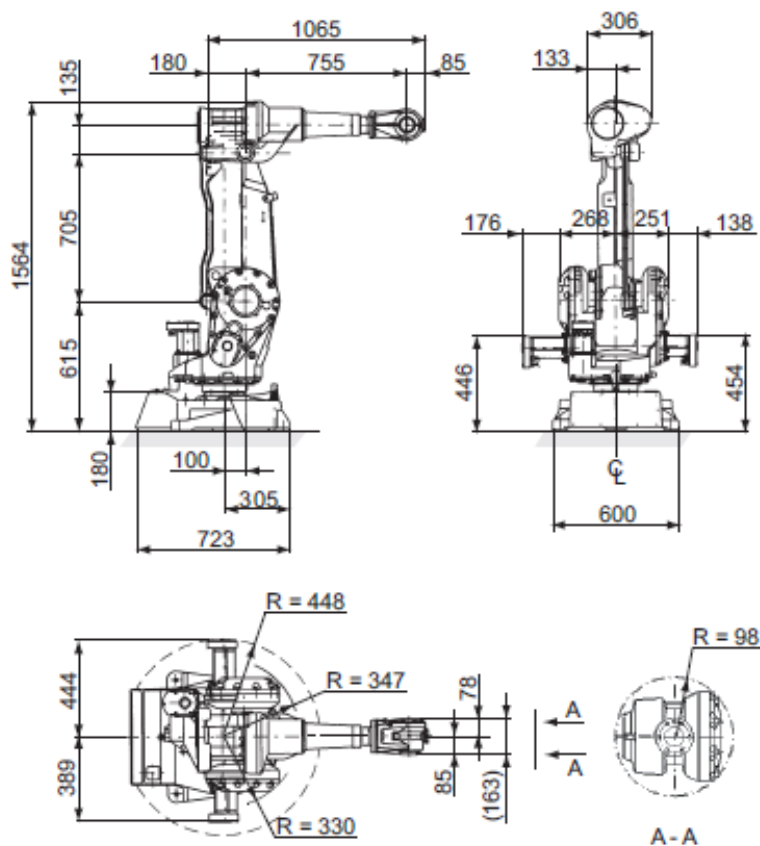
Pos	Descripción	Pos	Descripción
A	Eje 1	B	Eje 2
C	Eje 3	D	Eje 4
E	Eje 5	F	Eje 6

## 1 Descripción

### 1.1.2 Distintas versiones de robot

Continuación

Dimensiones IRB 2400/10 e IRB 2400/16



xx1100000546

### Información para la sustitución del manipulador

Los conectores R1.MP y R1.CP/CS en el manipulador en el tipo de protección *Standard* se modificaron en mayo de 2018, de conexiones de bloqueo de sujeción a conexiones de bloqueo roscadas. Por lo tanto, si un nuevo manipulador va a sustituir a otro más antiguo, se necesitan nuevos cables de suelo (cable de alimentación y CP/CS, igual que para el tipo de protección *Foundry Plus*).

Puede consultar más detalles en *Manual del producto - IRB 2400*.

## 1 Descripción

### 1.3.3 Montaje del manipulador

### 1.3.3 Montaje del manipulador

#### Carga máxima IRB 2400/10/16

Carga máxima en relación con el sistema de coordenadas de la base

Montado sobre el suelo

Fuerza	Carga de resistencia (en funcionamiento)	Carga máxima (paro de emergencia)
Fuerza xy	$\pm 2000$ N	$\pm 2600$ N
Fuerza z	$+4100 \pm 1400$ N	$+4100 \pm 1900$ N
Par xy	$\pm 3400$ Nm	$\pm 4000$ Nm
Par z	$\pm 550$ Nm	$\pm 900$ Nm

Suspendido

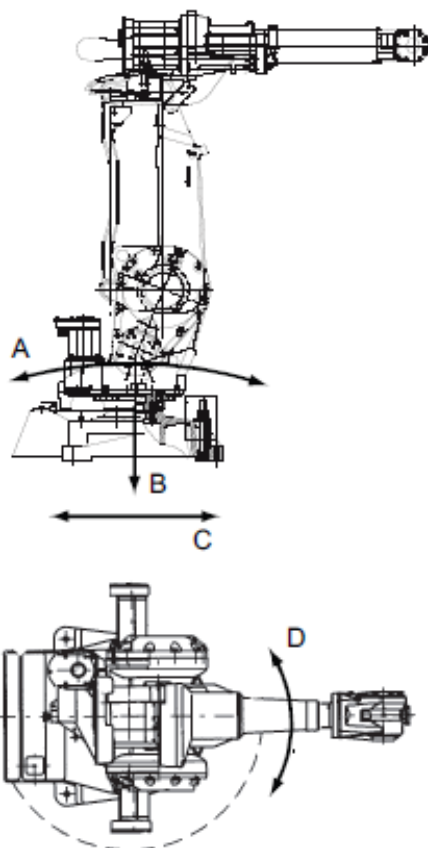
Fuerza	Carga de resistencia (en funcionamiento)	Carga máxima (paro de emergencia)
Fuerza xy	$\pm 2000$ N	$\pm 2600$ N
Fuerza z	$- 4100 \pm 1400$ N	$- 4100 \pm 1900$ N
Par xy	$\pm 3400$ Nm	$\pm 4000$ Nm
Par z	$\pm 550$ Nm	$\pm 900$ Nm

*Continúa en la página siguiente*

## 1 Descripción

### 1.3.3 Montaje del manipulador

Continuación



xx110000547

A	Par $_{xy}$ ( $T_{xy}$ )
B	Fuerza $_z$ ( $F_z$ )
C	Fuerza $_{xy}$ ( $F_{xy}$ )
D	Par $_z$ ( $T_z$ )

Nota acerca de  $M_{xy}$  y  $F_{xy}$

El par de doblado ( $M_{xy}$ ) puede realizarse en cualquier dirección en el plano XY del sistema de coordenadas de la base. Lo mismo se aplica a la fuerza transversal ( $F_{xy}$ ).

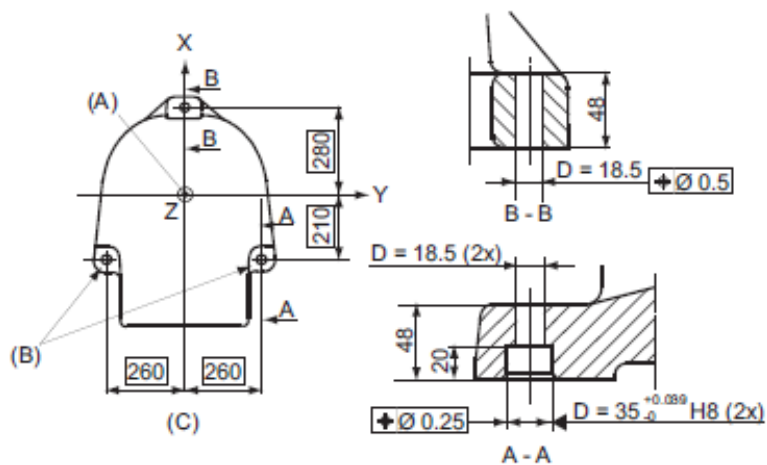
Continúa en la página siguiente

## 1 Descripción

### 1.3.3 Montaje del manipulador

Continuación

#### Orificios de fijación de la base del robot



xx1400002065

## **Apéndice C**

### **Especificaciones técnicas del controlador IRC5 y FlexPendant**



# 1 Descripción

## 1.1 Introducción

### 1.1.1 Introducción a la estructura

---

#### Generalidades

El controlador IRC5 contiene los elementos electrónicos necesarios para controlar el manipulador, los ejes adicionales y equipos periféricos.

---

#### Dual Cabinet Controller

El IRC5 se compone de los módulos siguientes:

- Módulo de accionamiento, que contiene el sistema de accionamiento
- Control Module, que contiene el ordenador principal (incluidas cuatro ranuras PCI para tarjetas de extensión), el panel del operador, el interruptor principal, las interfaces de comunicación, la conexión para FlexPendant, los puertos de servicio y cierto espacio para los equipos del cliente, por ejemplo tarjetas de E/S de ABB. El controlador también contiene el software de sistema, es decir RobotWare-OS, que incluye todas las funciones básicas de manejo y programación descritas más detalladamente en este capítulo. Sobre RobotWare-OS es posible instalar un número de opciones con funcionalidad adicional. Para obtener una descripción de estas opciones, consulte *Especificaciones del producto - Controller software IRC5*.

---

#### Controlador de armario sencillo

El contenido descrito anteriormente también puede montarse en un solo armario. El armario sencillo constituye una solución más compacta, adecuada para la mayoría de las aplicaciones que requieren menos equipamiento adicional en el interior.

---

#### Armarios

Datos	Peso
Controlador de armario sencillo	Máx. 150 kg
Dual Cabinet Controller	Máx. 180 kg
Control Module	50 kg
Drive Module	De 100 a 130 kg
Armario vacío pequeño	35 kg
Armario vacío grande	42 kg

Datos	Volumen (al. x an. x prof.)
Controlador de armario sencillo	970 x 725 x 710 mm
Armario vacío grande	
Dual Cabinet Controller	1.370 x 725 x 710 mm

*Continúa en la página siguiente*

Continuación

Datos	Volumen (al. x an. x prof.)
Drive Module Armario vacío pequeño	720 x 725 x 710 mm

#### Variantes adicionales del IRC5

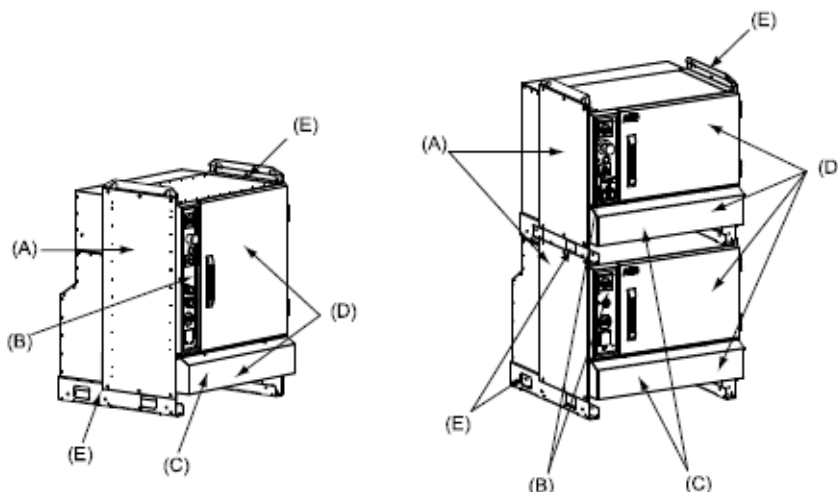
Existen dos variantes adicionales del IRC5

- IRC5 Panel Mounted Controller, en la que el integrador se encarga de la encapsulación.
- Controlador IRC5 Compact, un controlador con espacio reducido en planta disponible para los IRB pequeños

Para obtener más detalles de estas variantes, consulte [IRC5 Panel Mounted Controller en la página 63](#) y [Controlador IRC5 Compact en la página 72](#).

#### Nivel de ruido propagado por el aire

Nivel de ruido propagado por el aire	Descripción
Nivel de presión sonora exterior	< 70 dB (A) Leq (de acuerdo con la Directiva de máquinas 2006/42/CE para áreas de trabajo)



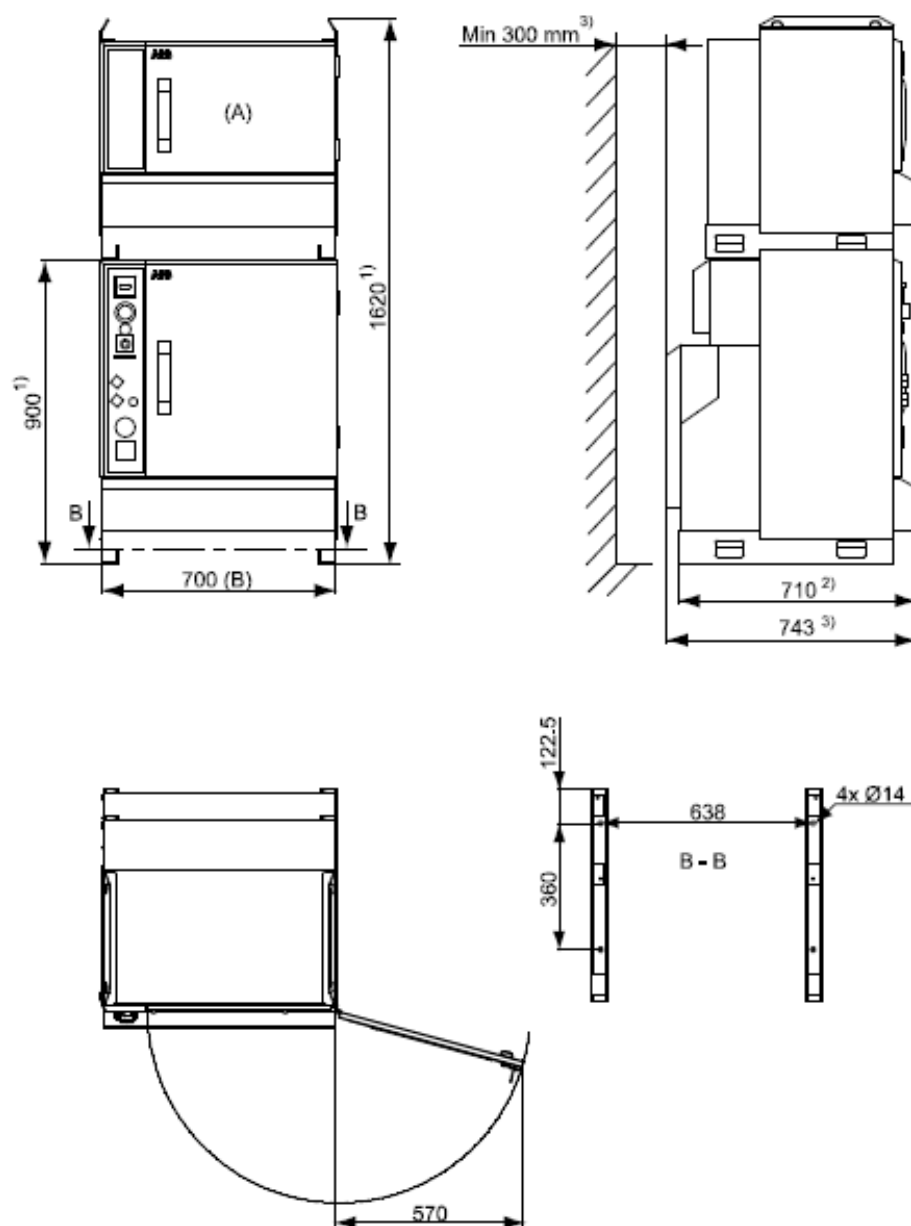
xx0900000927

Pos	Nombre	Descripción
A	Color del controlador	Aluzink
B	Panel de control	
C	cubiertas de conectores	Todas las conexiones para cables se encuentran en la parte delantera, opcionalmente cubiertas.
D	Color del controlador, puerta y cubiertas de conectores	NCS 2502 B (gris claro)
E	Color del controlador, pies y argollas de elevación	Negro

Continúa en la página siguiente

Continuación

## Armario sencillo - Distintas vistas



xx090000930

Pos	Descripción
A	Módulo de proceso opcional para controlador de armario sencillo
B	725 para integración

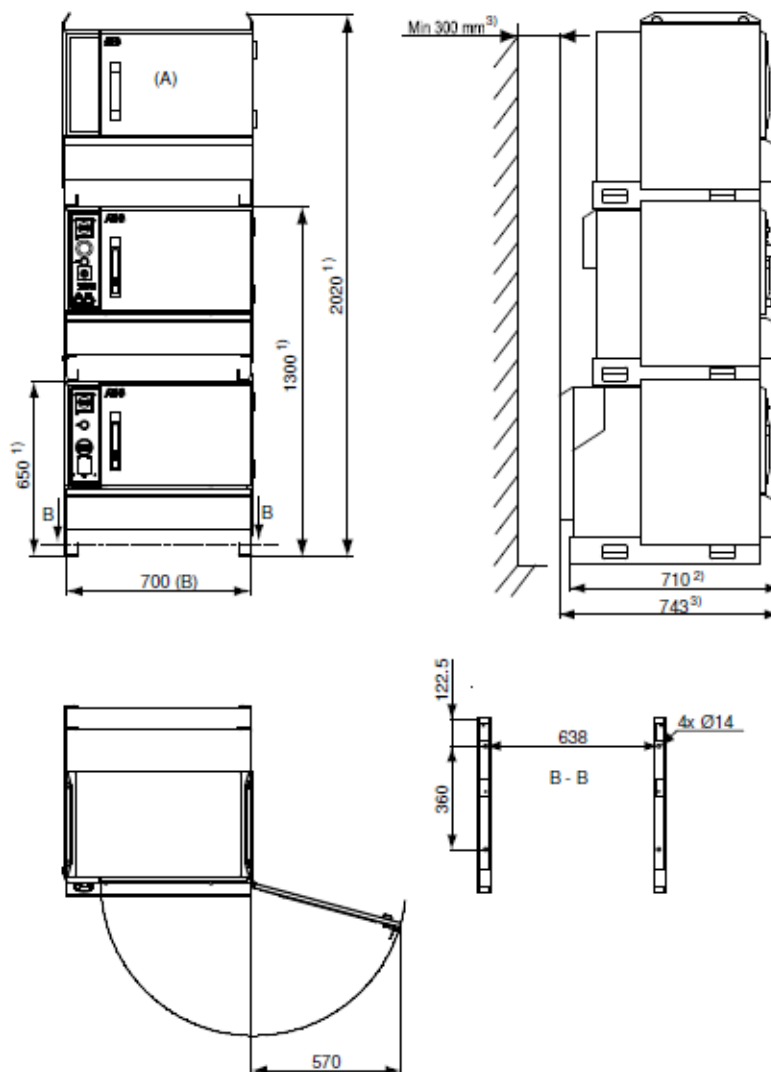
Pos	Descripción
1	Con la opción de ruedas, sume 10 mm a la altura.
2	Con el acceso de servicio en la parte posterior, sume 250 mm a la profundidad.

Continúa en la página siguiente

Continuación

Pos	Descripción
3	Filtro opcional contra polvo húmedo

## Doble armario - Distintas vistas



xx0900000929

Pos	Descripción
A	Módulo de proceso opcional
B	725 para integración

Pos	Descripción
1	Con la opción de ruedas, sume 10 mm a la altura.

Continúa en la página siguiente

*Continuación*

Pos	Descripción
2	Con el acceso de servicio en la parte posterior, sume 250 mm a la profundidad.
3	Filtro opcional contra polvo húmedo

### 1.3.3 FlexPendant

#### Generalidades

Todas las operaciones y tareas de programación pueden realizarse con el FlexPendant portátil (consulte la figura que aparece a continuación), el panel de operador y RobotStudio.



xx0900000936

Pos	Descripción
A	Pantalla
B	Teclas programables
C	Botón de paro de emergencia
D	Joystick
E	teclas de ejecución de programas
F	Conexión de memoria portátil USB
G	Alojamiento de puntero

La información se muestra en la pantalla de una forma intuitiva. No se requiere ninguna experiencia previa en programación ni ordenadores para aprender a usar el FlexPendant. Toda la información se muestra en inglés o, si se prefiere, en otro idioma (para saber qué idiomas están disponibles, consulte *Especificaciones del*

Continuación

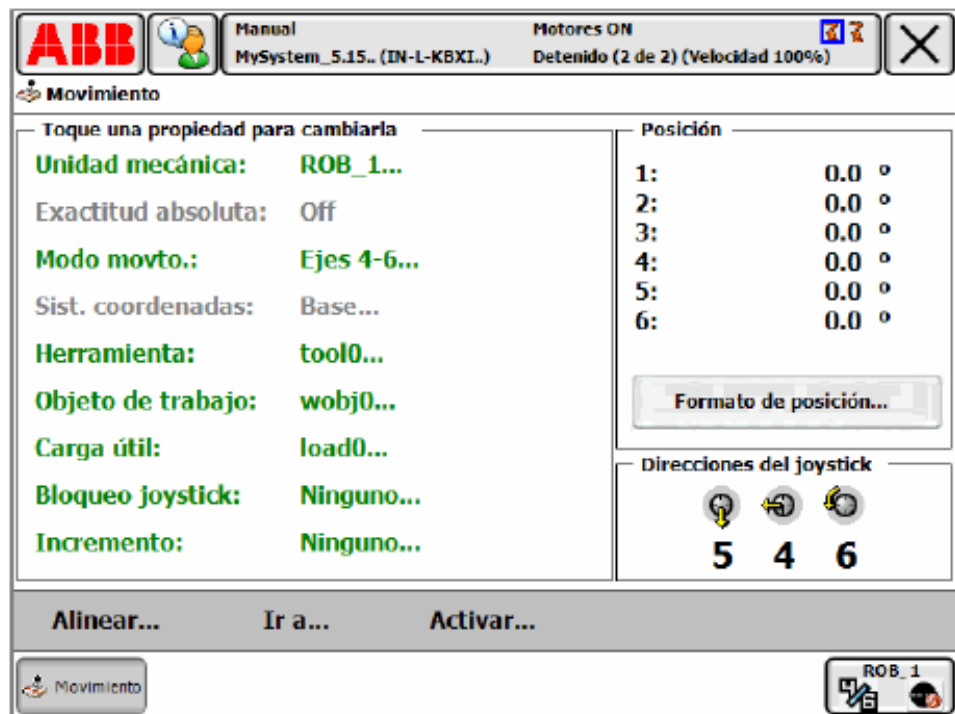
producto - Controller software IRC5). Se pueden instalar dos idiomas alternativos además del inglés, sin necesidad de recargar RobotWare.

Características	Descripción
Pantalla con pantalla táctil	<p>Una pantalla en color de 6,5 pulgadas que muestra textos e información gráfica. La selección e introducción de datos por parte del usuario se realiza presionando comandos y botones en la pantalla sólo con un dedo o con el puntero que se suministra. Es posible tener abiertas varias ventanas a la vez. En muchas vistas es posible ampliar y reducir la imagen.</p> <p>Muchas propiedades de la pantalla pueden ser configuradas por el usuario para conseguir un aspecto y una funcionalidad personalizados.</p> <p>Es posible invertir la pantalla y las direcciones del joystick para hacer que el FlexPendant sea adecuado para usuarios zurdos.</p> <p>El FlexPendant puede albergar potentes aplicaciones de usuario basadas en la tecnología .NET de Microsoft.</p>
teclas de ejecución de programas	Teclas para iniciar y detener programas y ejecutarlos paso a paso hacia delante o hacia atrás.
Hold-to-run	Una de las teclas de ejecución de programas debe mantenerse presionada al ejecutar el programa en el modo manual a máxima velocidad.
Teclas programables	Cuatro teclas definidas por el usuario que pueden configurarse para activar o restablecer una salida (por ejemplo para abrir o cerrar una pinza) o para activar una entrada del sistema.
Teclas de movimiento	Cuatro teclas de acción para el movimiento manual.
Dispositivo de habilitación	<p>Durante el modo manual, un pulsador que, si está presionado hasta la mitad de su recorrido, pone el sistema en el modo MOTORS ON.</p> <p>Cuando se libera el dispositivo de habilitación o si es pulsado totalmente, el robot pasará al estado MOTORS OFF.</p>
Joystick	<p>El joystick tridimensional se utiliza para mover el robot manualmente, por ejemplo durante los trabajos de programación. El usuario puede determinar la velocidad de este movimiento.</p> <p>Una inclinación pronunciada del joystick hace que el robot se mueva rápidamente. Una inclinación más leve hará que se mueva más lentamente.</p>
Botón de paro de emergencia	Cuando se presiona este botón, el robot se detiene inmediatamente.

Continúa en la página siguiente

Continuación

Ejemplo de ventana de FlexPendant



en040000654



## **Apéndice D**

### **Procedimientos iniciales con RobotStudio**

### 1.3.4 RobotStudio

---

#### Descripción general

RobotStudio es una aplicación para PC que permite trabajar eficientemente con datos del IRC5. RobotStudio puede considerarse como el compañero ideal del FlexPendant, usándolos de forma que se complementan y cada uno está optimizado para sus tareas concretas. Al explotar todas las ventajas de esta potente combinación, es posible disfrutar de una nueva forma de trabajar con más eficiencia.

El FlexPendant tiene como fines principales el movimiento del robot con el joystick, la programación de posiciones, el manejo y el ajuste de movimientos, mientras que RobotStudio resulta ideal para el manejo de datos de configuración, gestión de programas, documentación en línea y acceso remoto.

RobotStudio actúa directamente sobre los datos activos del controlador. La conexión al controlador puede hacerse localmente a través de la conexión para PC de servicio y, si el controlador cuenta con la opción de RobotWare PC Interface, a través de una conexión de red.

Un sistema de control maestro seguro garantiza que RobotStudio sólo pueda tomar el control de un robot si tal operación se autoriza desde el FlexPendant.

La entrada principal a la funcionalidad de RobotStudio es un explorador de vistas de robot. Desde él se selecciona el robot con el que se desea trabajar, si hay varios robots instalados, y las partes del sistema que se desea utilizar.

El paquete básico de RobotStudio contiene:

- System Builder para crear, instalar y mantener sistemas
- Un Editor de configuraciones, para editar los parámetros de sistema del sistema que se está ejecutando.
- Un Editor de programas para programación en línea.
- Una grabadora de eventos, para grabar y monitorizar los eventos del robot.
- Herramientas para realizar copias de seguridad y restauraciones de sistemas
- Una herramienta de administración para autorización de usuarios
- Otras herramientas para visualización y manejo de propiedades del controlador y del sistema

El acceso a todas las posibilidades de RobotStudio como una potente herramienta de programación y simulación fuera de línea se pide por separado.

## Continuación

## Detalles de las características Premium y Basic

En la tabla que aparece a continuación se enumeran las características ofrecidas por las *licencias* Basic y Premium.

Característica	Básico	Premium
Características necesarias para la puesta en servicio de un robot real o virtual <sup>1</sup> , tales como: <ul style="list-style-type: none"> <li>• System Builder</li> <li>• Visor del Registro de eventos</li> <li>• Editor de configuración</li> <li>• Editor de RAPID</li> <li>• Copia de seguridad/restauración</li> <li>• Ventana E/S</li> </ul>	Sí	Sí
Características de productividad, tales como: <ul style="list-style-type: none"> <li>• Editor de datos de <i>RAPID</i></li> <li>• RAPID Compare</li> <li>• Ajustar <i>Robtargets</i></li> <li>• Observación de RAPID</li> <li>• <i>Puntos de interrupción</i> de RAPID</li> <li>• Analizador de señales</li> <li>• Herramienta <i>MultiMove</i></li> <li>• ScreenMaker<sup>1,2</sup></li> <li>• Trabajos</li> </ul>		Sí
Características fuera de línea elementales, tales como: <ul style="list-style-type: none"> <li>• Abrir <i>estación</i></li> <li>• Unpack &amp; Work</li> <li>• Ejecutar simulación</li> <li>• Cambiar a <i>fuera de línea</i></li> <li>• Herramientas de movimiento de robots</li> <li>• Predicción de calor de caja reductora</li> <li>• <i>ABB Library</i> de robots</li> </ul>	Sí	Sí
Características fuera de línea avanzadas, tales como: <ul style="list-style-type: none"> <li>• Programación gráfica</li> <li>• Guardar estación</li> <li>• <i>Pack &amp; Go</i></li> <li>• Importar/exportar <i>geometría</i></li> <li>• Importar biblioteca</li> <li>• Crear Station Viewer y películas</li> <li>• Transferir</li> <li>• AutoPath</li> <li>• Operaciones en 3D</li> </ul>		Sí
Complementos		Sí
<p>1. Requiere la opción PC Interface RobotWare en el <i>controlador de robot</i> para permitir la comunicación LAN. Esta opción no es necesaria para la conexión a través del puerto de Servicio o para la comunicación del <i>controlador virtual</i>.</p> <p>2. Requiere la opción de RobotWare FlexPendant Interface en el sistema de controlador de robot.</p>		

## 1.2 Requisitos del sistema

### Descripción general

Antes de la instalación de RobotStudio, asegúrese de que el ordenador cumple los siguientes requisitos de hardware y software.

### Hardware

Estación de trabajo portátil o de sobremesa de altas prestaciones, con los siguientes requisitos:

Componente	Requisito
CPU	Procesador de 2.0 GHz o más rápido, se recomiendan varios núcleos
Memoria	8 GB mínimo. 16 GB o más para trabajar con modelos de CAD pesados.
Disco	Más de 10 GB de espacio disponible, unidad de estado sólido (SSD) recomendada.
Tarjeta gráfica	Alto rendimiento, compatible con DirectX 11, tarjeta gráfica para juegos de cualquier marca líder. Para el modo de iluminación avanzado, se requiere DirectX 3D feature level 10_1 o superior.
Configuración de pantalla	Se recomienda una resolución de 1920 x 1080 píxeles o superior.
Puntos por pulgada (dpi)	Sólo se admite tamaño Normal para Integrated Vision.
Ratón	Ratón de tres botones.
Ratón 3D [opcional]	Cualquier ratón 3D de 3DConnexion. Consulte <a href="http://www.3dconnexion.com">http://www.3dconnexion.com</a> .

RobotStudio es compatible con *RobotWare* versión 5.07 hasta la última versión publicada. Consulte las limitaciones de compatibilidad en las Notas de la versión de RobotStudio.

Puede conectar RobotStudio a un *controlador de robot* a través de su puerto de Servicio o a través de Ethernet.

- Para conectar RobotStudio a través de Ethernet, habilite la opción RobotWare en PC Interface.
- Para ejecutar las aplicaciones ScreenMaker o FlexPendant SDK en un controlador de robot, habilite la opción de RobotWare FlexPendant Interface.

### Software

Sistema operativo	Descripción
Windows 10 Anniversary Update o posterior	Edición de 64 bits

Se recomienda ejecutar actualizaciones de Windows para obtener las actualizaciones más recientes de Windows antes de instalar y ejecutar RobotStudio. El Firewall de Windows puede bloquear ciertas funciones que son necesarias para ejecutar RobotStudio, que deben estar desbloqueadas según se requiera. Puede

*Continúa en la página siguiente*

consultar y editar el estado de un programa en Inicio > Panel de control > Firewall de Windows. Para obtener más información sobre el Firewall de Windows, visite [www.microsoft.com](http://www.microsoft.com).

### Configuración de firewall

La configuración del firewall es aplicable a los controladores tanto de robot como virtuales. En la tabla siguiente se describen las configuraciones de firewall necesarias:

Estado	Nombre	Acción	Dirección	Protocolo	Remote Address	Local Service	Remote Service	Aplicación
	RobNetscanHost	Permitir	Salida	UDP/IP	Cualquiera	Cualquiera	5512,5514	robnetscanhost.exe
	IRS5Controller	Permitir	Entrada	UDP/IP	Cualquiera	5513	Cualquiera	robnetscanhost.exe
	RobComCtrlServer	Permitir	Salida	TCP/IP	Cualquiera	Cualquiera	5515	robcomctrlserver.exe
	RobotFTP	Permitir	Salida	TCP/IP	Cualquiera	Cualquiera	FTP(21)	Cualquiera
	RobotStudio	Permitir	Salida	HTTP	Cualquiera	Cualquiera	80	RobotStudio.exe
	RobotStudio	Permitir	Salida	HTTPS	Cualquiera	Cualquiera	443	RobotStudio.exe



#### Nota

El puerto 443 no es necesario para la comunicación con el controlador, pero sí que se requiere para la comunicación externa con ABB para aprovechar servicios tales como notificaciones de actualizaciones o servicios en línea.

La tabla siguiente describe las configuraciones de firewall necesarias para la opción RobotWare, Integrated Vision:

Estado	Nombre	Acción	Dirección	Protocolo	Remote Address	Local Service	Remote Service	Aplicación
	Telnet	Permitir	Salida	TCP/IP	Cualquiera	Cualquiera	23	RobotStudio.exe
	Protocolo In-Sight	Permitir	Salida	TCP/IP	Cualquiera	Cualquiera	1069	RobotStudio.exe
	Descubrimiento In-Sight	Permitir	Entrada/Salida	UDP/IP	Cualquiera	1069	1069	RobotStudio.exe
	Puerto de actualización (sólo PC)	Permitir	Salida	TCP/IP	Cualquiera	Cualquiera	1212	RobotStudio.exe
	DataChannel	Permitir	Salida	TCP/IP	Cualquiera	Cualquiera	50000	RobotStudio.exe

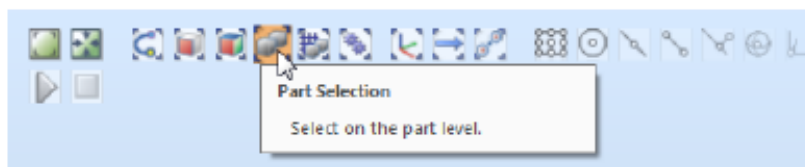
*Continúa en la página siguiente*

### Niveles de selección y modos de ajuste

Es posible mover cada elemento de una estación para conseguir el diseño necesario mediante el uso de los niveles de selección adecuados. RobotStudio proporciona un conjunto de niveles de selección que pueden usarse para seleccionar un objeto o una parte específica del mismo. Estos niveles de selección son curva, superficie, entidad, pieza, mecanismo, grupo, objetivo/base de coordenadas y trayectoria. La selección de tipo objetivo/base de coordenadas y la de trayectoria pueden combinarse con cualquiera de los demás niveles de selección.

Abrir una estación de demostración mientras se exploran niveles de selección.

- 1 En la ventana gráficos, haga clic en el icono Selección de pieza. Desplace el ratón sobre el icono para ver la información sobre la herramienta que contiene el nombre y el propósito del icono.



xx1900000134

- 2 En la ventana Gráficos, haga clic en el icono Ajustar a objetos. Este es un modo de ajuste múltiple, que ajusta al centro, borde o esquina más próximo.



xx1900000137

Haga clic en una parte en la vista Gráficos, el objeto completo aparecerá resaltado. Puede ver también el punto de elección como una estrella blanca que se ha ajustado al centro/borde/esquina más próximo.

- 3 En la pestaña Inicio, en el grupo Mano alzada, haga clic en el botón Mover.



xx1900000135

Una cruz con flechas en las direcciones X, Y y Z aparece ahora en la pieza. Arrastre las flechas para mover el objeto.

*Continúa en la página siguiente*

## 1 Procedimientos Iniciales

---

### 1.7 Administración de la interfaz de usuario con el ratón

#### Continuación

- En la pestaña Inicio, en el grupo Mano alzada, haga clic en el botón Mover eje y seleccione cualquier eje en el robot.



xx1900000136

Cuando se pulsa el botón izquierdo del ratón en el eje en la ventana Gráficos, puede moverse el robot en cualquier dirección.

---

#### Selección de un elemento en la ventana de gráficos

Para seleccionar elementos en la ventana Gráficos, realice las operaciones siguientes:

- En la parte superior de la ventana Gráficos, haga clic en el icono del nivel de selección deseado.
- Opcionalmente, haga clic en el modo de ajuste deseado para la parte del elemento que desee seleccionar.
- En la ventana Gráficos, haga clic en el elemento. El elemento seleccionado se resaltará.

---

#### Rectángulo de selección profundo

El rectángulo de selección profundo (modo de selección por defecto) se activa pulsando y manteniendo pulsada la tecla **SHIFT** y trazando un rectángulo en la vista en 3D mediante el ratón. Este modo selecciona los objetos cubiertos por el rectángulo de selección, independientemente de su visibilidad.

---

#### Rectángulo de selección profundo

Para activar el rectángulo de selección profundo (modo de selección por defecto), pulse y mantenga pulsada la tecla **SHIFT** y arrastre el ratón diagonalmente sobre los objetos que desee seleccionar. Este modo selecciona los objetos cubiertos por el rectángulo de selección, independientemente de su visibilidad. En este modo de selección, puede seleccionar varios elementos en la ventana Gráficos.

---

#### Rectángulo de selección poco profundo

Para activar la selección del rectángulo poco profundo, pulse las teclas **SHIFT + S** y trace un rectángulo en la vista en 3D mediante el ratón. En este modo puede seleccionar el objeto actualmente visible.

---

#### Selección de un elemento en los navegadores

Para seleccionar elementos en un navegador, haga lo siguiente:

- Haga clic en el elemento. El elemento seleccionado se resaltará en el navegador.

*Continúa en la página siguiente*

---

### Selección múltiple de elementos en los navegadores

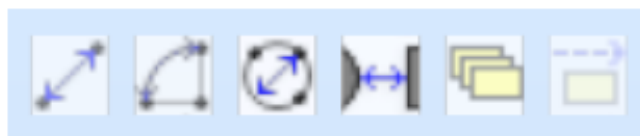
Para seleccionar varios elementos en un navegador, realice las operaciones siguientes:

- 1 Asegúrese de que todos los elementos a seleccionar sean del mismo tipo y estén situados en la misma rama de la estructura jerárquica. De lo contrario, los elementos no funcionarán.
- 2 Realice una de las operaciones siguientes:
  - Para seleccionar elementos adyacentes: En el navegador, mantenga presionada la tecla MAYÚS y haga clic en los elementos primero y último. La lista de elementos se resaltará.
  - Para seleccionar elementos separados: En el navegador, mantenga presionada la tecla CTRL y haga clic en los elementos que desee seleccionar. Los elementos se resaltarán.

---

### Herramientas de medición

RobotStudio proporciona las siguientes herramientas de medición para calibrar las distancias lineales y angulares entre varios objetos en la ventana de Gráficos. Pase el ratón por encima de estos iconos para ver el nombre y la finalidad de estas herramientas.



xx2000002521

- En la ventana Gráficos haga clic en el icono Punto a Punto para activar esta herramienta. Con el icono seleccionado, alinee dos objetos en la ventana Gráficos para ver la distancia entre estos objetos en milímetros.



xx2000002522

*Continúa en la página siguiente*



## 2.3 Creación de la estación

---

### Descripción general

En los siguientes pasos se define el flujo de trabajo de construcción de una nueva *estación*.

- 1 Crear una estación vacía.
- 2 Importar un modelo de robot.
- 3 Añadir *posicionadores* y Track Motion.
- 4 Crear un *controlador virtual*.
- 5 Importar una herramienta y fijarla al robot.
- 6 Crear un *objeto de trabajo*.
- 7 Definir trayectorias y objetivos para crear un programa de robot gráficamente.
- 8 *Sincronizar* con RAPID para crear el programa de RAPID.

---

### Creación de una solución con una estación vacía

- 1 Haga clic en la pestaña Archivo . Aparece la vista Backstage de RobotStudio; haga clic en Nuevo.
- 2 En Estaciones, haga clic en Solución con estación vacía.
- 3 Introduzca el nombre de la *solución* en el cuadro Nombre de solución y, a continuación, busque y seleccione la carpeta de destino en el cuadro Ubicación. La ubicación predeterminada de la solución es C:\User\La estación incluida recibe el nombre de la solución.



#### Nota

La carpeta de soluciones ayuda a mantener un sistema de carpetas bien estructurado por el que resulta fácil navegar.

- 4 Haga clic en Crear.  
Se crea la nueva solución. RobotStudio guarda esta solución de forma predeterminada.

## 2.5 Creación de un controlador virtual

---

Los robots que no están conectados a ningún controlador no pueden ser programados; por lo tanto, configure un *controlador virtual* para el robot.

- 1 En la pestaña Inicio, haga clic en Controlador virtual.
- 2 Haga clic en Desde diseño para mostrar la primera página del asistente.
- 3 En el cuadro Nombre, introduzca el nombre del controlador virtual. La ubicación del *controlador virtual* se muestra en el cuadro Ubicación.
- 4 En la lista RobotWare, seleccione la versión de *RobotWare* que desea utilizar.
- 5 Haga clic en Next (Siguiente).
- 6 En el cuadro Mecanismos, seleccione los *mecanismos* que desee incluir en el *controlador virtual*. Añada mecanismos como tracks o *posicionadores* a su estación antes de crear un controlador virtual.
- 7 Haga clic en Next (Siguiente).

El asistente solicita ahora una correlación de los *mecanismos* a una *tarea* de movimiento específica, de acuerdo con las reglas siguientes:

- Sólo se permite un robot de *TCP* por tarea.
- Es posible añadir hasta seis tareas de movimiento, pero sólo pueden usarse cuatro robots de *TCP* y éstos deben estar asignados a las cuatro primeras tareas.
- El número de tareas no puede ser mayor que el número de mecanismos.
- Si el *sistema* contiene un robot de *TCP* y un eje externo, ambos se asignarán a la misma tarea. Sin embargo, es posible añadir una nueva tarea y asignarle el eje externo.
- Si el sistema contiene más de un robot de *TCP*, todos los ejes externos se asignarán a una tarea separada. Sin embargo, es posible trasladarlos a las otras tareas.
- El número de *ejes externos* de una tarea está limitado por el número de módulos de accionamiento disponibles en el armario (uno en el caso de los robots grandes, dos en los robots medianos, tres en los pequeños).

Si sólo se ha seleccionado un *mecanismo* en la página anterior, esta página no se muestra.

Las *tareas* pueden ser añadidas y eliminadas con los botones correspondientes. Los mecanismos pueden ser movidos hacia arriba o hacia abajo con las flechas correspondientes. Para correlacionar los mecanismos con las tareas, realice este paso:

- 8 Opcionalmente, haga los cambios en la correlación y a continuación haga clic en Siguiente.

Aparece la página Opciones del controlador.

---

*Continúa en la página siguiente*

Continuación

- 9 En la página Opciones del controlador, tiene la opción de alinear las bases de coordenadas de las tareas con sus correspondientes *bases de coordenadas de la base*.
- En el caso de un sistema de robot, active la casilla de verificación para alinear la base de coordenadas de la tarea con la base de coordenadas de la base.
  - En el caso de un sistema *MultiMove Independent*, active la casilla de verificación para alinear la base de coordenadas de la tarea con la base de coordenadas de la base.
  - En el caso de un sistema *MultiMove Coordinated*, seleccione el robot en la lista desplegable y active la casilla de verificación para alinear la base de coordenadas de la tarea con la base de coordenadas de la base para el robot seleccionado.
- 10 Compruebe el resumen y a continuación haga clic en Finalizar.
- Si el controlador virtual contiene más de un robot, el número de tareas y las posiciones de la base de coordenadas del mecanismo deben ser verificados en la ventana Configuración de movimientos.



Nota

Para crear un sistema a partir de un diseño, todos los *mecanismos*, tales como robots, *Track Motion* y *posicionadores*, deben estar guardados como bibliotecas.

## 2.6 Sincronización con controlador virtual para crear un programa de RAPID

Los movimientos del robot pueden programarse mediante el uso de *RAPID*. Un robot que no participe en una tarea en un controlador no puede ser programado; por lo tanto, configure un *controlador virtual* para el robot antes de la *sincronización* con RAPID.

- 1 Con la estación abierta, en la pestaña RAPID, haga clic en Sincronizar.
- 2 En las opciones, haga clic en Sincronizar con RAPID para relacionar objetos en la estación con el código de RAPID.

En el navegador Controlador, amplíe la vista de árbol para encontrar el nodo de RAPID. Haga clic en el nodo de RAPID para ver los archivos de RAPID.