



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

UNIDAD ZACATENCO
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA
SECCIÓN DE MECATRÓNICA

**“Teleoperación háptica de robots móviles con retardos,
utilizando IMPACT”**

Tesis que presenta:
Omar Espitia Pérez

Para obtener el grado de:
Maestro en Ciencias

En la especialidad de:
Ingeniería Eléctrica

Directores de la Tesis:
Dr. Alejandro Rodríguez Angeles
Dr. Jaime Álvarez Gallegos

Agradecimientos

A Dios: *Por nunca dejarme solo, prestándome ayuda y cuidándome en este trayecto vivido.*

A mi madre: *Por su amor y apoyo incondicional en mis estudios, aunque eso significara numerosos sacrificios. Por ser además un ejemplo de fuerza, voluntad y amor ejerciendo el papel de madre y padre.*

A mi tía: *Por su amor y apoyo incondicional, que aunque nunca tuviera la obligación, siempre eligió ayudarme.*

A mi hermana: *Por su apoyo y ánimos. Espero que esto represente un ejemplo a seguir para ella.*

A los Dres. Alejandro Rodríguez Angeles y Jaime Álvarez Gallagos: *Por el asesoramiento, apoyo y paciencia que me mostrarón en el proceso de la elaboración de este trabajo. También por la oportunidad de trabajar en su laboratorio y confiar en mi.*

A los miembros del jurado: *Por sus comentarios y las aportaciones que hicieron a este trabajo de tesis.*

A mis amigos y compañeros del CINVESTAV: *Por estar conmigo compartiendo buenos y malos momentos, darme apoyo y consejo en el transcurso de la maestría.*

A los doctores de la Sección de Mecatrónica: *Por sus consejos, enseñanzas y por compartir conocimientos conmigo. Por darme la oportunidad de realizar mis estudios de maestría en la Sección de Mecatrónica del Departamento de Ingeniería Eléctrica del CINVESTAV.*

Al CONACYT: *Por su apoyo económico a través del proyecto CB2015-254329.*

Al CINVESTAV y CONACYT: *Por su apoyo económico a través del proyecto CB2017-2018-A1-S-26123.*

Resumen

En este trabajo de tesis se presenta un esquema de teleoperación bilateral maestro-esclavo con robots móviles diferenciales. El sistema general muestra retardos en la comunicación del maestro al esclavo y viceversa.

El esquema se compone dos robots móviles 2.0, los cuales son el maestro y el esclavo (originalmente uno real y uno virtual), y también se cuenta con una interfaz háptica llamada Novint Falcon. Este dispositivo permite el control del robot maestro mediante la inyección de velocidades traslacional y angular; también recibe la retroalimentación de fuerzas para la evasión de obstáculos. El maestro es el móvil que debe estar mas cerca del usuario pues nos permite visualizar la trayectoria deseada; además este robot manda constantemente su posición al esclavo para que pueda seguirlo. Por su parte, este ultimo, mediante un algoritmo, ejerce cierta retroalimentación sobre la interfaz para evitar colisionar con objetos.

El móvil esclavo sigue su respectiva referencia mediante el Control por Modelo Interno (IMC por sus siglas en ingles), el cual necesita el modelo matemático de la planta. Por esta razón se utilizó el modelo cinemático del robot diferencial. Debido a que este contiene no linealidades, se le aplicó una extensión dinámica mediante la cual se linealizó. Un inconveniente de utilizar este modelo linealizado, es el hecho de que ignora diversas dinámicas del robot esclavo, por lo que este debe moverse siempre a bajas velocidades.

Para la evasión de obstáculos se aplicó un algoritmo llamado Método de Control Geométrico para Evasión de Obstáculos (GOACM por siglas en ingles), el cual permite ubicar un punto de seguridad al cual dirigirnos para evadir algún objeto con el cual exista peligro de colisión. Esta información es convertida en fuerzas y se utiliza en la interfaz háptica para que se pueda esquivar al obstáculo.

La plataforma de simulación manejada en este trabajo es Gazebo, de ROS. En este mismo ambiente se utilizaron dos robots diferenciales Turtlebot 3 Burguer como elementos del esquema de teleoperación. El único elemento físico fue la interfaz, pues de otro modo no se podría percibir la retroalimentación de fuerzas.

Abstract

In this thesis work a bilateral master-slave teleoperation scheme with differential mobile robots is presented. The general system shows delays in communication from master to slave and vice versa.

The scheme consists of two mobile robots 2.0, which are the master and the slave (originally one real and other virtual), and there is also a haptic interface called Novint Falcon. This device allows the control of the master robot through the injection of translational and angular velocities; it also receives feedback forces for obstacle avoidance. The master is the mobile that must be closer to the user as it allows us to visualize the desired trajectory. Furthermore, this robot constantly sends its position to the slave so that it can follow it. For its part, the slave, through an algorithm, exercises feedback over the interface to avoid colliding with objects.

The slave mobile follows its respective reference through Internal Model Control (IMC), which requires the mathematical model of the plant. For this reason, the kinematic model of the differential robot was used. Because it contains non-linearities, a dynamic extension was applied by which it was linearized. One disadvantage of using this linearized model is that it ignores various dynamics of the slave robot, so it must always move at low velocities.

For avoidance of obstacles, an algorithm called Geometric Obstacle Avoidance Control Method (GOACM) was applied, which allows to locate a safety point to move to avoid an object with which there is danger of collision. This information is converted into forces and used in the haptic interface so that the obstacle can be avoided.

The simulation platform managed in this work is Gazebo, from ROS. In this same environment, two Turtlebot 3 Burger differential robots were used as elements of the teleoperation scheme. The only physical element was the interface, otherwise the feedback forces could not be perceived.

Contenido

Lista de figuras	XI
1. Introducción	1
1.1. Motivación	3
1.2. Objetivos	3
1.3. Aportaciones de la tesis	4
1.4. Organización de la tesis	4
2. Marco Teórico	7
2.1. Obtención del modelo cinemático	7
2.2. Restricciones del modelo	8
2.3. Linealización del modelo	8
2.4. Obtención de la matriz de transferencia	10
2.5. Control por Modelo Interno	11
2.6. Retardos	13
2.7. Sistemas de teleoperación	15
3. Plataforma experimental	17
3.1. ROS	17
3.2. Novit Falcon	19
4. Esquema de control aplicado	23
4.1. Control por Modelo Interno	23
4.2. Evasión de obstáculos	25
4.3. Esquema de teleoperación	30
4.4. Análisis de estabilidad	32
5. Resultados	45
6. Conclusiones y trabajo a futuro	73

6.1. Conclusiones	73
6.2. Trabajo futuro	74
Bibliografía	74
A. Instalación de la interfaz	77

Lista de figuras

1.1. Primer robot móvil: Shakey	1
2.1. Diagrama del vehículo diferencial	7
2.2. Diagrama del IMC	12
2.3. Diagrama del Predictor de Smith	14
3.1. Logotipo de ROS	17
3.2. Ambiente de Gazebo.	18
3.3. Robot móvil utilizado	19
3.4. Interfaz utilizada	20
3.5. Ejes en la interfaz	21
4.1. Diagrama del IMC en el esclavo	25
4.2. Áreas de detección y protección	26
4.3. Construcción del punto de seguridad	27
4.4. Obtención del punto de seguridad cuando hay 2 objetos	28
4.5. Esquema de teleoperación sin retroalimentación	31
4.6. Diagrama de bloques del sistema general simplificado	31
5.1. Condiciones iniciales con el obstáculo estático	45
5.2. Trayectorias realizadas por los robots	48
5.3. Distancias al objeto	48
5.4. Fuerzas recibidas en la interfaz	49
5.5. Velocidades	49
5.6. Errores en x & y	50
5.7. Error angular	50
5.8. Trayectorias realizadas por los robots	51
5.9. Distancias al objeto	52
5.10. Fuerzas recibidas en la interfaz	52
5.11. Velocidades	53
5.12. Errores en x & y	53
5.13. Error angular	54
5.14. Trayectorias realizadas por los robots	55

5.15. Distancias al objeto	55
5.16. Fuerzas recibidas en la interfaz	56
5.17. Velocidades	56
5.18. Errores en x & y	57
5.19. Error angular	57
5.20. Trayectorias realizadas por los robots	58
5.21. Distancias al objeto	59
5.22. Fuerzas recibidas en la interfaz	59
5.23. Velocidades	60
5.24. Errores en x & y	60
5.25. Error angular	61
5.26. Trayectorias realizadas por los robots	62
5.27. Distancias al objeto	62
5.28. Fuerzas recibidas en la interfaz	63
5.29. Velocidades	63
5.30. Errores en x & y	64
5.31. Error angular	64
5.32. Condiciones iniciales con el obstáculo dinámico	65
5.33. Trayectorias realizadas por los robots	66
5.34. Distancias al objeto	67
5.35. Fuerzas recibidas en la interfaz	67
5.36. Velocidades	68
5.37. Errores en x & y	68
5.38. Error angular	69
5.39. Trayectorias realizadas por los robots	70
5.40. Distancias al objeto	70
5.41. Fuerzas recibidas en la interfaz	71
5.42. Velocidades	71
5.43. Errores en x & y	72
5.44. Error angular	72
A.1. Comando para comprobar la correcta instalación	78
A.2. Respuesta obtenida cuando la instalación se realizó con éxito	78

Capítulo 1

Introducción

En la actualidad el avance tecnológico es cada vez mas rápido, y eso se debe a que es un aspecto estratégico para cualquier país, ya que el poder de una nación es influenciado por la capacidad de innovar tecnología constantemente. Al aplicar los nuevos conocimientos, y con los avances constantes, la calidad de los procesos de producción aumenta y por lo tanto también su complejidad permitiendo la generación de nuevos empleos y remuneraciones mas altas, todo esto incidiendo en la calidad de vida de las personas [1].

Debido a la necesidad de máquinas mas sofisticadas y especializadas, se desarrollaron diversos tipos de robots, como los brazos manipuladores y los robots móviles (el interés de esta tesis). El primer robot móvil del que se tiene registro fue desarrollado de 1966 a 1972 por el SRI (Stanford Research Institute) Internacional y fue llamada Shakey [2]. Este robot era una plataforma móvil independiente con sensor táctil y se controlaba mediante la visión de una cámara (Figura 1.1).

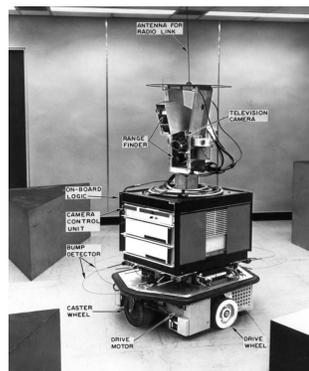


Figura 1.1: Primer robot móvil: Shakey

La utilización de los robots en la producción se debe a que estos pueden trabajar con mayor eficiencia en labores repetitivas y difíciles, además de que evitan que en muchos casos el ser humano se exponga a situaciones sumamente peligrosas. Sin embargo aún se tienen límites en la tecnología actual y por ello en algunas circunstancias es necesario que las personas aún tengan cierta participación en las acciones de los robots.

La teleoperación responde a la necesidad de controlar las acciones de un dispositivo pero a distancia, lo que en muchos casos facilita y protege al usuario. Estos robots suelen ser utilizados en trabajos de localización remota, en tareas que son difíciles de automatizar y en entornos no estructurados, como en construcciones.

Trabajos recientes sobre teleoperación son bastantes y muy variados. Tenemos ejemplos en [3] con su vehículo de 6 ruedas en suelo suave y en [4] donde se busca un control adaptativo con parámetros inciertos. Mas adelante se explicarán mas a fondo algunos aspectos relacionados con la teleoperación.

Un aspecto importante que influye mucho en la comunicación entre el operador y el robot es el retardo. De hecho si este es demasiado grande puede provocar problemas de estabilidad y por ende hacer incontrolable un sistema que originalmente no lo era. Existen muchos trabajos que se enfocan en esto, como [5] donde se busca predecir la posición de un robot cilíndrico, o [6] donde se utiliza un control con un observador para compensar retardos. Una solución que es comúnmente utilizada en retardos es el Predictor de Smith.

En [7] se utiliza una estrategia de control basada en modelos internos, llamada IMPACT (Internal Model Principle and Control Together) para un trabajo de teleoperación en un sistema bilateral maestro-esclavo con manipuladores. Y en [8] se utiliza la misma estructura de control IMPACT junto con el Predictor de Smith para un sistema de teleoperación bilateral maestro-esclavo con retroalimentación de posición en robots móviles.

Algo que busca añadir este trabajo es la adición de una interfaz háptica. Tenemos ejemplos de su uso en [9] donde se busca manejar un puente grúa con retroalimentación de fuerza, donde esta fuerza tendrá que ser un reflejo de la inercia del sistema y de su movilidad. También en [3] se utiliza la interfaz Novit Falcon para validar el método de teleoperación utilizado.

En [10] se utiliza una estrategia para evadir obstáculos, llamado Método de Control Geométrico para Evasión de Obstáculos (GOACM) el cual consiste en la obtención, mediante distancias (líneas), de un punto al cual dirigirse con la finalidad de esquivar el objeto. En esta tesis se planea utilizar este método para generar una retroalimentación de fuerza en la interfaz usada y con ello facilitarle al usuario la evasión de objetos.

1.1. Motivación

Es de conocimiento común que la robótica tenga una gran importancia en la investigación y en el desarrollo de un país, pues ayuda en la calidad de los procesos productivos.

Como se mencionó anteriormente un robot puede proteger a una persona evitando que se exponga a situaciones peligrosas, pero, ¿y si la situación es demasiado compleja como para que un robot autónomo pueda trabajar solo?. Una solución a esto es la teleoperación, pues esto garantiza que el usuario pueda interferir o manejar el robot manteniendo distancia, como en el caso de [11] y así evitando los peligros del área. Por ello buscamos añadir a un esquema de control de teleoperación, una interfaz que facilite el manejo de los robots.

Sin embargo, para que se obtengan mejores resultados en la teleoperación es necesario obtener información del ambiente en el que se mueva el vehículo. En nuestro caso se utilizara un sensor lidar que revisará el área frente al robot, para evitar colisiones mediante un algoritmo que utiliza el GOACM, para generar una retroalimentación de fuerzas.

1.2. Objetivos

El objetivo principal de esta tesis es implementar un sistema de teleoperación bilateral maestro-esclavo con robots móviles, el cual contará con una interfaz háptica mediante la cual se conducirá al maestro y se recibirá retroalimentación de fuerza por parte del esclavo, para evitar colisiones con obstáculos.

Para alcanzar el objetivo principal, se tienen que alcanzar los siguientes objetivos secundarios:

- Integración de la interfaz háptica Falcon Novint al sistema, mediante ROS y simulaciones.
- Aplicación de un algoritmo para evitar colisiones y generar la retroalimentación de fuerza en la interfaz háptica. También se harán simulaciones finales.
- Comprobar la robuztes ante retardos en dos direcciones (maestro a esclavo y viceversa)

1.3. Aportaciones de la tesis

En [7] se utilizó el IMC (Control por Modelo Interno) en teleoperación bilateral con robots manipuladores, obteniendo resultados de alto desempeño, y en [8] se utilizó la misma arquitectura pero con robots móviles, junto con el IMC adaptado; en este último también existieron retardos en las comunicaciones y los resultados fueron muy buenos.

En esta tesis se busca añadir cosas nuevas al trabajo con móviles de [8], siendo las contribuciones de esta tesis las siguientes:

- Adición de una interfaz háptica (Novit Falcon) para manejar al robot maestro. Esto garantiza que su trayectoria sea mas realizable para el esclavo.
- Adaptación del esquema de control IMC para los retardos teniendo la interfaz añadida.
- Aplicación del algoritmo GOACM.
- Obtención de la retroalimentación háptica en la interfaz para lograr evadir colisiones con objetos.

Todo esto para obtener un sistema de teleoperación bilateral con cierta robustez a los retardos en la comunicación, y con una interfaz háptica que permita al usuario tener sugerencias de acción para la evasión de obstáculos.

1.4. Organización de la tesis

La presente tesis se encuentra organizada en seis capítulos, de los cuales este pertenece al Capítulo 1. El Capítulo 2 nos muestra el marco teórico, donde se explica la obtención del modelo cinemático, sus restricciones, además de la linealización del modelo. También se explica en que consiste el IMC, el Predictor de Smith, y los esquemas de teleoperación.

El Capítulo 3 se centra en la aplicación particular del control. Se habla nuevamente de IMC de forma rápida para fundamentar un filtro utilizado. También se explora el algoritmo de GOACM, el cual nos ayuda a evitar choques. Finalmente se trata con el esquema de teleoperación que se utilizó y las modificaciones que recibió.

En el Capítulo 4 se presenta y explica sobre la plataforma en la que se realizaron las simulaciones experimentales. Se habla de forma rápida del framework utilizado y de sus características. Y también se dan detalles de la interfaz háptica que se añadió al sistema.

El Capitulo 5 muestra los resultados obtenidos de las simulaciones.

Por ultimo el Capitulo 6 da las conclusiones obtenidas de este trabajo, junto con ideas para un trabajo futuro, que de continuidad a esta tesis.

Capítulo 2

Marco Teórico

2.1. Obtención del modelo cinemático

En la Figura 2.1 se puede apreciar un esquema del robot móvil diferencial

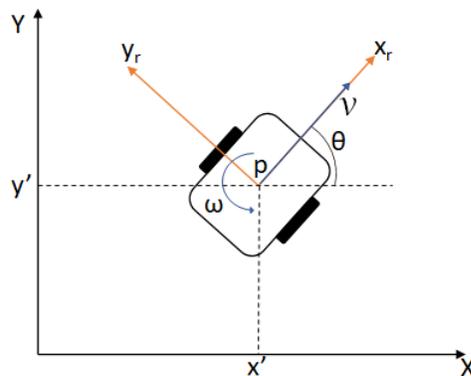


Figura 2.1: Diagrama del vehículo diferencial

Para la obtención de las ecuaciones del modelo es necesario un sistema inercial global XY , dentro del cual se ubica el robot diferencial. Las coordenadas y' & x' designan la posición del punto p , el cual es el origen del marco de referencia móvil (x_r, y_r) del vehículo. La orientación del vehículo con respecto al marco global está representado por θ .

La velocidad de traslación está representado por v y está en dirección perpendicular al eje de las ruedas, coincidiendo con el eje x del marco de referencia del robot. La velocidad rotacional es ω y está sobre el eje z del marco del vehículo.

Su respectiva ecuación matricial es la siguiente (2.1):

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (2.1)$$

Los valores \dot{x} y \dot{y} , son los componentes cartesianos de la velocidad traslacional y la velocidad rotacional se relaciona con $\dot{\theta}$.

Este modelo ofrece muchas ventajas debido a su sencillez. Sin embargo, como se ignoran muchas dinámicas presentes en la realidad, el control pudiera no funcionar adecuadamente bajo ciertas circunstancias. Por esta razón se necesita aplicar velocidades bajas, donde dichas dinámicas ignoradas no tengan un efecto considerable y se vuelvan despreciables.

2.2. Restricciones del modelo

Se tomán del modelo las proyecciones de v y se despeja el elemento común.

$$\begin{aligned} \dot{x} &= v \cos \theta \\ v &= \frac{\dot{x}}{\cos \theta} \\ \dot{y} &= v \sin \theta \\ v &= \frac{\dot{y}}{\sin \theta} \end{aligned} \quad (2.2)$$

De lo anterior se obtiene:

$$\frac{\dot{x}}{\cos \theta} = \frac{\dot{y}}{\sin \theta} \quad (2.3)$$

Y con esto finalmente se obtiene la restricción no holónoma:

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0 \quad (2.4)$$

Esta restricción nos da el conocimiento de que no es posible realizar cualquier tipo de trayectorias para que el vehículo se mueva de un lugar a otro, por el hecho de poseer dos ruedas fijas. En resumen el robot no puede moverse en perpendicular al sentido de giro de las llantas fijas.

2.3. Linealización del modelo

Observando la ecuación (2.1) se puede notar que se tiene un sistema subactuado y con no linealidades. Como se busca utilizar el IMC de forma sencilla, se busca obtener la matriz de transferencia, y debido a esto es necesario linealizar el sistema. Primero se busca linealizar el modelo por retroalimentación entrada salida.

Vemos que la ecuación (2.1) tiene la siguiente forma:

$$\dot{\psi} = f(\psi) + g_1(\psi)u_1 + g_2(\psi)u_2 \quad (2.5)$$

Con la siguientes salidas:

$$\begin{aligned} y_1 &= h_1(\psi) \\ y_2 &= h_2(\psi) \end{aligned} \quad (2.6)$$

Se tiene que $f(\psi) = 0$, $g_1(\psi)$ y $g_2(\psi)$ son el primer y segundo vector de la ecuación (2.1), $u_1 = v$ y $u_2 = \omega$. Las salidas corresponden a $h_1(\psi) = x$ y $h_2(\psi) = y$.

Nuestro sistema tiene mas de una entrada y mas de una salida, por lo que es MIMO. Si se desea linealizarlo, su vector de grado relativo debe estar bien definido. La matriz de desacoplamiento es:

$$A(\psi) = \begin{bmatrix} L_{g_1}h_1 & L_{g_2}h_1 \\ L_{g_1}h_2 & L_{g_2}h_2 \end{bmatrix} = \begin{bmatrix} \frac{\partial h_1}{\partial \psi} g_1(\psi) & \frac{\partial h_1}{\partial \psi} g_2(\psi) \\ \frac{\partial h_2}{\partial \psi} g_1(\psi) & \frac{\partial h_2}{\partial \psi} g_2(\psi) \end{bmatrix} \quad (2.7)$$

Con lo cual se obtiene:

$$A(\psi) = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \end{bmatrix} \quad (2.8)$$

La matriz (2.8) es singular y por lo tanto el vector de grado relativo no esta definido. La linealización exacta no es posible.

Volviendo a la ecuación (2.1), se puede observar que la velocidad traslacional y angular actúan como entradas del sistema. Esta vez se busca aplicar una extensión dinámica para poder linealizar el modelo y volver al sistema de 2 entradas con 2 salidas. La nueva expresión del sistema, extendido, es la siguiente:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\rho} \end{bmatrix} = \begin{bmatrix} \rho \cos \theta \\ \rho \sin \theta \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \check{u}_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} u_2 \quad (2.9)$$

Donde :

- $\rho = u_1 = v$
- $\check{u}_1 = \dot{\rho}$
- $u_2 = \omega$

Y las salidas quedan definidas como sigue:

$$\begin{bmatrix} h_1 \\ h_2 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} \quad (2.10)$$

Por lo tanto, con los valores adecuados de \check{u}_1 y u_2 se debería poder controlar la posición del vehículo. Se reacomoda el sistema con los siguientes cambios de variables:

- $z_1 = x$
- $z_2 = y$
- $z_3 = \dot{x} = \rho \cos \theta$
- $z_4 = \dot{y} = \rho \sin \theta$

Con lo cual se puede expresar el sistema (ec. (2.11)) con sus salidas (ec. (2.12)) de la siguiente forma:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{z_3}{\sqrt{z_3^2+z_4^2}} \\ \frac{z_4}{\sqrt{z_3^2+z_4^2}} \end{bmatrix} \check{u}_1 + \begin{bmatrix} 0 \\ 0 \\ -z_4 \\ z_3 \end{bmatrix} u_2 \quad (2.11)$$

$$\begin{bmatrix} h_1 \\ h_2 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \quad (2.12)$$

Debido a que el sistema aún tiene no linealidades, se aplica el siguiente linealizador, añadiendo nuevas entradas de control:

$$\begin{bmatrix} \check{u}_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \frac{z_3}{\sqrt{z_3^2+z_4^2}} & \frac{z_4}{\sqrt{z_3^2+z_4^2}} \\ \frac{-z_4}{z_3+z_4^2} & \frac{z_3}{z_3+z_4^2} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} \quad (2.13)$$

La matriz linealizadora de la ecuación (2.13) tiene singularidad cuando z_3 y z_4 suman cero. La expresión del sistema ya linealizado es la siguiente:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \phi_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \phi_2 \quad (2.14)$$

2.4. Obtención de la matriz de transferencia

Del sistema linealizado de la ecuación (2.14) obtenemos lo siguiente

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad D = 0 \quad (2.15)$$

Ahora se puede obtener la matriz de transferencia del sistema linealizado. Se utiliza la siguiente expresión:

$$Y(s) = H(s)U(s) = [C(sI - A)^{-1}B + D] U(s) \quad (2.16)$$

Con la ecuación (2.14) es posible obtener la matriz de transferencia $H(s)$, la cual es:

$$H(s) = \begin{bmatrix} \frac{1}{s^2} & 0 \\ 0 & \frac{1}{s^2} \end{bmatrix} = \begin{bmatrix} G' & 0 \\ 0 & G' \end{bmatrix} \quad (2.17)$$

El sistema es desacoplado con dos subsistemas de una entrada y una salida cada uno. Las funciones de transferencia contenidas en la matriz anterior son dobles integradores. Con fines de simplificación, en los diagramas y en las explicaciones, se mostrará como si únicamente se aplicara el control a un subsistema, aunque en realidad se les aplique a los dos que componen la matriz.

2.5. Control por Modelo Interno

La estrategia de IMPACT implica al IMC, el cual ya tiene algunos años existiendo, pues fue presentado por primera vez en 1989 por Manfred Morari [12]. Este control a su vez está fundamentado en el Principio de Modelo Interno, el cual dice que si se desea que la salida de un sistema sea igual a la referencia, la función de transferencia de dicho sistema debe ser igual a la unidad.

Para que se cumpla la condición mencionada, esta estrategia de control maneja la idea de utilizar el inverso del modelo de la planta, para que la dinámica interna de todo el sistema en conjunto sea de valor unitario.

Esto presenta algunas dificultades pues la robustez y la calidad del resultado depende mucho de que tan parecido sea el modelo matemático utilizado al sistema real. Además, aunque el sistema sea realizable su inverso no suele serlo, por lo que se debe aplicar un filtro que corrija este problema. Esto último puede aumentar la complejidad de las ecuaciones y dificultar su programación.

El diagrama del IMC se muestra en la Figura 2.2 y se utilizará para explicar mas a fondo este control.

El bloque $G(s)$ representa al sistema real, y el $G'(s)$ es el modelo matemático del sistema. $R(s)$ y $Y(s)$ son la entrada y salida del sistema respectivamente. La salida obtenida del modelo $G'(s)$ es $Y'(s)$. La diferencia entre estas dos salidas da una aproximación de las perturbaciones ($D'(s)$), y se resta este valor a la referencia para tener

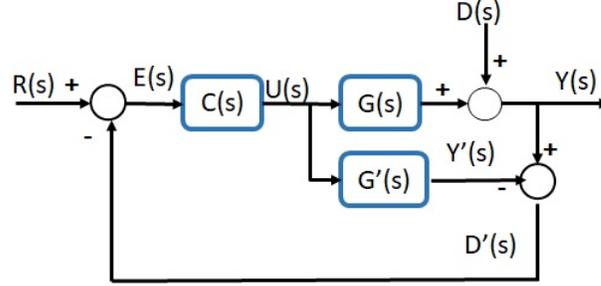


Figura 2.2: Diagrama del IMC

el error que será manejado por el control $C(s)$.

Las ecuaciones (2.18), (2.19) y (2.20) representan las relaciones del diagrama del IMC.

$$Y(s) = G(s)U(s) + D(s) = G(s)C(s)E(s) + D(s) = G(s)C(s)[R(s) - D'(s)] + D(s) \quad (2.18)$$

$$Y'(s) = G'(s)U(s) = G'(s)C(s)E(s) = G'(s)C(s)[R(s) - D'(s)] \quad (2.19)$$

Donde

$$D'(s) = Y(s) - Y'(s) \quad (2.20)$$

Como $Y(s)$ y $Y'(s)$ son iguales en las condiciones ideales, tenemos que

$$Y(s) = G'(s)C(s)R(s) + D(s) \quad (2.21)$$

$$Y'(s) = G'(s)C(s)R(s) \quad (2.22)$$

En el caso de la ecuación (2.22), si el modelo y el control juntos dan la unidad entonces la salida $Y'(s)$ será igual a la entrada $R(s)$:

$$Y'(s) = G'(s)C(s)R(s) = G'(s)G'^{-1}(s)R(s) = R(s) \quad (2.23)$$

Ahora usando la ecuación (2.21), y si es el caso en el que el modelo es idéntico al sistema real, entonces también la salida $Y(s)$ será igual a la referencia. En la siguiente ecuación se sustituye el control por su equivalente (y suponiendo que $D(s) = 0$):

$$Y(s) = G(s)G'^{-1}(s)E(s) = R(s) \quad (2.24)$$

Suponiendo el caso en el que $D(s)$ no es igual a cero y por lo tanto existen perturbaciones (y partiendo de la ecuación (2.18)), obtenemos lo siguiente:

$$Y(s) = G(s)G'^{-1}(s)R(s) + D(s) \quad (2.25)$$

Lo que da como consecuencia que la perturbación estimada es igual a la real, como se indica en la ecuación (2.26)

$$D'(s) = Y(s) - Y'(s) = D(s) \quad (2.26)$$

En este caso tenemos que

$$E(s) = R(s) - D(s) \quad (2.27)$$

Lo que indica que el IMC no tiene buena robustez ante perturbaciones fuertes, pues el error no llega a ser igual a la referencia, si no que mantiene las perturbaciones añadidas.

En resumen, si las perturbaciones $D(s)$ son nulas y el error es igual a la referencia, entonces deberíamos tener una salida igual a la referencia, sin embargo tenemos que comúnmente $G'(s)$ no es igual a $G(s)$, pues no siempre se pueden modelar perfectamente todos los sistemas. Esta es una de las razones por la cual se compara la salida del modelo ($Y'(s)$) con la salida del sistema real ($Y(s)$).

2.6. Retardos

En todo sistema de teleoperación los retardos son una cuestión importante. Esto es debido a que las señales ocupan un tiempo de traslado y de procesamiento, por ello estos valores pueden llegar a ser muy significativos en algunos casos.

Esto no puede simplemente ignorarse pues los retardos pueden causar vibraciones o ruido, e incluso volver inestable a un sistema estable. Existen muchos trabajos que buscan el tratamiento del retardo, algunos pocos de los cuales fueron citados anteriormente. De hecho las mismas estructuras de teleoperación tienen cierta robustez a lo retardos cuando se logra tener retroalimentación. En [8] se utiliza un control con el famoso Predictor de Smith.

Este estimador de atraso fue desarrollado en 1957 por Otto Smith y fue el inicio del estudio de los compensadores de tiempo muerto (DTC). En la Figura 2.3 se muestra el diagrama de este predictor.

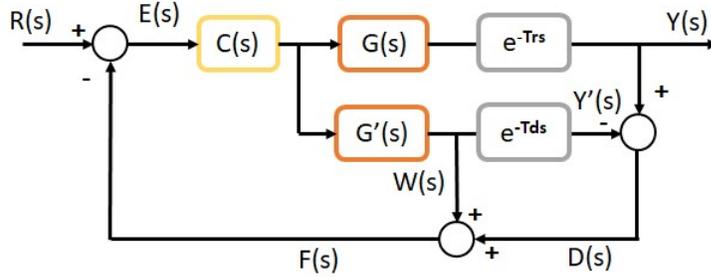


Figura 2.3: Diagrama del Predictor de Smith

Es posible observar cierta semejanza con el IMC, pues también se usa un observador. El Predictor de Smith tiene tres características fundamentales:

- Elimina el retardo de la ecuación característica del sistema, es decir, el control C puede diseñarse sin tener en cuenta el retardo en las dinámicas (este control puede ser el que sea).
- Anticipa la salida del proceso para el control (sin retardo).
- Divide el proceso en dos partes, esto es, en el observador se separa el modelo matemático $G'(s)$ del retardo estimado. Este retardo está representado por e^{-Tds} en el diagrama de bloques de la figura 2.3.

El retardo real está dado por e^{-Trs} y el sistema real está representado con $G(s)$, ambos en la figura 2.3. Primero tenemos que

$$Y(s) = G(s)e^{-Trs}C(s)E(s) \quad (2.28)$$

Donde $G(s)$ es el sistema real y $Y(s)$ es la salida. En nuestro caso, la salida es la posición obtenida mediante la odometría.

La salida del modelo con el retardo estimado es:

$$Y'(s) = G'(s)e^{-Tds}C(s)E(s) \quad (2.29)$$

Para nosotros es la posición del móvil estimada con el retardo nominal.

Y la salida sin el retardo e^{-Tds} es:

$$W(s) = G'(s)C(s)E(s) \quad (2.30)$$

En la primera resta tenemos que:

$$D(s) = Y(s) - Y'(s) \quad (2.31)$$

Con lo cual podemos calcular una aproximación del error del sistema real con el retardo real, con respecto al modelo y el retardo estimado. Finalmente un último

valor de comparación se obtiene de la siguiente forma:

$$F(s) = W(s) + D(s) \quad (2.32)$$

El control elegido se debe encargar de este valor $F(s)$.

Cuando se aplicó este predictor al esquema de teleoperación PERR, únicamente se añadió a la sección del maestro, pues solo se buscó predecir las posiciones del esclavo con los retardos en acción. Sin embargo, como se verá mas adelante, se elimino y solo sirvió para entender el trabajo que antecedió a este.

2.7. Sistemas de teleoperación

Ya se ha hablado un poco de la teleoperación y de una característica inherente a estos sistemas el cual es el retardo. La importancia de los esquemas de teleoperación radica en que permiten controlar algo a distancia, ya sea por seguridad o por que resulte imposible al usuario estar en esa misma localización.

En un sistema de teleoperación existen dos elementos: el maestro y el esclavo. El primero es el que genera la información ya sea por un usuario o por ecuaciones que le den una referencia a seguir, además de que casi siempre debe estar en una localización cercana al usuario. El segundo es el robot esclavo, el cual se suele encontrar a mayor distancia y sigue las instrucciones dadas por el maestro.

Estos sistemas obviamente tienen como característica importante la transmisión de información. De acuerdo a este criterio se pueden clasificar los sistemas teleoperados de la siguiente manera:

- Sistemas unilaterales. Estos son relativamente sencillos y cuentan con poca robustez. Aquí el maestro manda información al esclavo, pero este no regresa ningún dato.
- Sistemas bilaterales. Aquí el maestro no es el único que manda datos, también el esclavo lo hace y puede ser en el mismo canal que el maestro o en otro. Estos sistemas son mas complejos debido a que se busca acoplar a ambos sistemas mediante un algoritmo.

Otra forma de clasificar la arquitectura de teleoperación es de acuerdo al tipo de información que se transmite.

- PERR (Error de posición). Este tipo maneja la información del error de las posiciones del maestro y del esclavo. El maestro manda mediante su ubicación la referencia que el esclavo debe seguir y este ultimo manda su posición al maestro como retroalimentación. Como es obvio no se requiere una gran variedad de sensores para este caso. En [8] tenemos un ejemplo.

- De retroalimentación de fuerzas kinestésicas. En este tipo no se mandan las posiciones de ningún móvil, en cambio se utiliza la información mandada del esclavo que refleja la interacción de este robot con su ambiente. Esta información se expresa mediante actuadores como fuerzas. En [9] se utiliza este tipo de teleoperación.
- Arquitectura de control de cuatro canales. Este es una especie de unión de los dos tipos anteriores, pues ambos robots mandan información de su posición, y el esclavo también retroalimenta datos de su desenvolvimiento en el ambiente en forma de fuerzas. Obviamente aquí se tiene un mejor panorama de como se desempeña el esclavo en su sitio remoto, aunque es mas complicada su aplicación.

Capítulo 3

Plataforma experimental

3.1. ROS

La parte física de un robot está compuesta principalmente por sus actuadores y sensores, pero lo que permite que el robot pueda realizar tareas es la programación de este. Existen muchas plataformas para la programación, pero la de nuestro interés en esta tesis es ROS (Robot Operating System).

ROS es un middleware robótico, es decir, un framework gratuito y de código abierto usado como herramienta para la programación de robots, ya sea para aplicaciones comerciales o de investigación [13]. Aunque no es un sistema operativo, es muy parecido, pues puede proveer los servicios estándar de uno, como la abstracción del hardware, el control de dispositivos de bajo nivel, la implementación de funcionalidad de uso común, el paso de mensajes entre procesos y el mantenimiento de paquetes.



Figura 3.1: Logotipo de ROS

Este se desarrolló en 2007 bajo el nombre de switchyard por el Laboratorio de Inteligencia Artificial de Stanford para dar soporte al proyecto del Robot con Inteligencia Artificial de Stanford (STAIR2). En el 2010 la versión ROS C turtle fue lanzado, en el 2013 el ROS Hydro Medusa y en el 2016 el Kinetic. Este ultimo es el que se utilizó en esta tesis. Existen otras versiones pero no son de interés en este trabajo.

La librería de ROS está orientada para un sistema UNIX (Ubuntu), aunque también hay adaptaciones experimentales para otros sistemas como Windows. El hecho de que principalmente exista para sistemas Unix es lo que le facilita que sea de código abierto.

ROS ofrece varias capacidades como las que se mencionan a continuación:

- Interfaz de paso de mensajes entre procesos. De esta forma se pueden comunicar sensores y actuadores.
- Soporte y herramientas para lenguaje de programación de alto nivel. Este soporte abarca a diversos lenguajes populares como C++ o Python y existe soporte experimental para otros como Java. También se proveen herramientas para la elaboración de aplicaciones y paqueterías.
- Disponibilidad de librerías de terceros. El framework de ROS está integrado por diversas librerías, como OpenCV para visión artificial y PCL para percepción 3D, lo cual facilita la elaboración de programas robustos.
- Soporte comunitario. Esta es la principal razón de su popularidad y desarrollo. Existen diversas plataformas para la interacción entre usuario como ROS Answers.
- Variedad de herramientas y simuladores. ROS contiene muchas líneas de comando y herramientas para debug, visualizar y simular aplicaciones robóticas, por ejemplo Rviz o Gazebo.

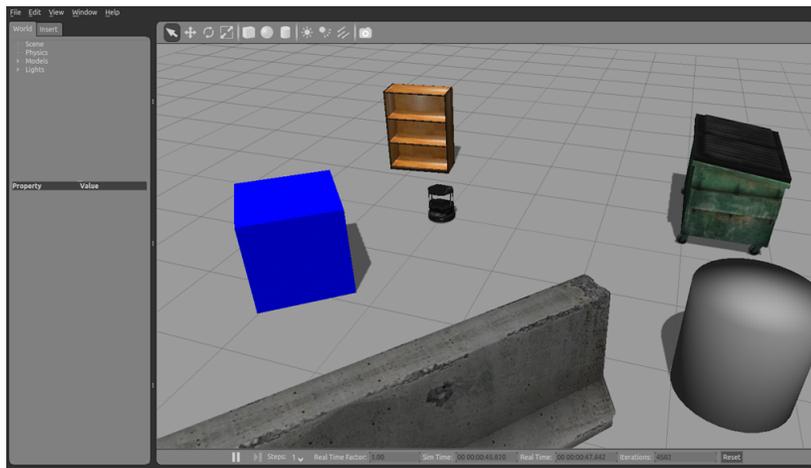


Figura 3.2: Ambiente de Gazebo.

Gazebo es un simulador de entornos 3D (Fig 3.2) que nos brinda la posibilidad de evaluar el comportamiento de un robot en un mundo virtual (se simulan ruidos en las señales y otras características que buscan realismo). Nos permite diseñar robots y personalizarlos, crear mundos virtuales con herramientas sencillas de CAD e importar modelos ya creados.

Además, es posible sincronizar los robots con ROS de forma que estos dispositivos emulados publiquen la información de sus sensores en nodos, así como implementar una lógica y un control que dé órdenes al robot.

En nuestro caso particular se simuló el robot móvil turtlebot 3 modelo burger, el cual se muestra en la figura 3.3. De este se utilizaron dos para el esquema de teleoperación, pues es necesario un maestro y un esclavo.

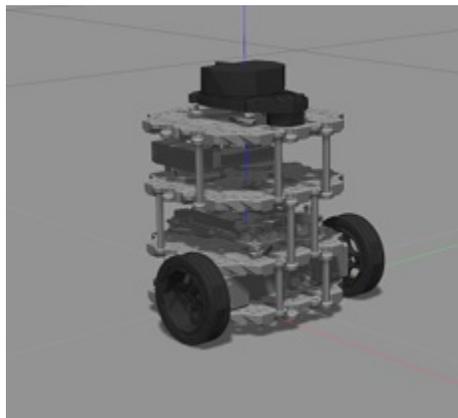


Figura 3.3: Robot móvil utilizado

Algunas especificaciones importantes de este robot que tuvieron presentes en la simulación son las siguientes:

- Velocidad traslacional máxima: 0.22 m/s
- Velocidad rotacional máxima: 2.84 rad/s
- Tamaño (L x W x H): 138mm x 178mm x 192mm

3.2. Novit Falcon

Para manejar al maestro y cumplir con los objetivos de esta tesis se utilizó una interfaz llamada Novit Falcon (fig. 3.4), la cual es un sofisticado dispositivo háptico 3D que le permite al usuario moverse con 3 grados de libertad y obtener retroalimentación háptica mediante actuadores que generan fuerzas.

El Falcon tiene 3 brazos que permiten movimiento en el espacio (x, y & z), y que están conectados cada uno a un motor, los cuales generan a 100 Hz sensaciones realistas al tacto. Hay que tener en cuenta que aunque puede usarse en investigación, fue ideado originalmente para videojuegos. La comunicación con la computadora es mediante el puerto USB 2.0.



Figura 3.4: Interfaz utilizada

Las especificaciones son las siguientes:

- Espacio de movimiento: 10 cm x 10 cm x 10 cm.
- Fuerza máxima: 9 N.
- Resolución de posición: 160 incrementos por cm.
- Grados de libertad: 3 (x,y,z).
- Tamaño: 22 cm x 28 cm x 24 cm.
- Peso: 2.8 kg.
- Potencia: 30 W.
- Voltaje: 100V-240V, 50Hz-60Hz.

Para añadir la interfaz Falcon se utilizó la librería libnifalcon, la cual nos permite que este hardware pueda tener comunicación con la PC en el sistema UNIX. También se utilizó la paquetería rosfalcon para poder establecer el intercambio de datos entre ROS y la interfaz.

Con la interfaz añadida al esquema general, se programaron las ecuaciones con las que se establecería el control al maestro.

En la Figura 3.5 se tiene en una imagen la distribución de los ejes en la interfaz. Usando este medio, junto con las ecuaciones (3.1) y (3.2), se le inyectó al robot maestro velocidades lineales y angulares.

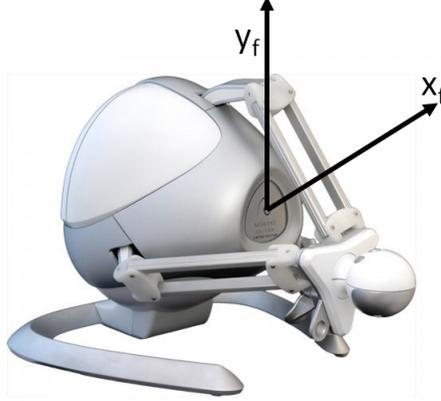


Figura 3.5: Ejes en la interfaz

$$\nu_m = v_{mx} \frac{d_{fy}}{60} \quad (3.1)$$

$$\omega_m = -\omega_{mx} \frac{d_{fx}}{55} \quad (3.2)$$

Donde:

- $\nu_m, \omega_m \rightarrow$ son las velocidades lineal y angular del maestro respectivamente.
- $v_{mx}, \omega_{mx} \rightarrow$ son las velocidades lineal y angular máximas (0.16 m/s y 0.12 rad/s), respectivamente, que puede alcanzar el maestro.
- $d_{fx}, d_{fy} \rightarrow$ son el número de unidades de distancia recorridas en el eje x , y y de la interfaz (más abajo se explica qué son estas unidades de distancia).

Como es posible observar, cuando se mueve la interfaz hacia el eje y positivo se genera una velocidad lineal. Cuando se mueve en el eje x se introduce una velocidad angular negativa si es hacia la derecha y positiva si es hacia la izquierda.

La interfaz tiene un espacio de movimiento de 10 cm x 10 cm x 10 cm, el cual ya se mencionó. Y dentro de su espacio, la esfera que funciona como agarradera se desplaza 60 unidades en el lado positivo del eje y_f y 60 en el lado negativo de este mismo eje. En cuanto al eje x_f , la esfera se desplaza 55 unidades hacia el lado positivo y 55 hacia el negativo. Los cocientes en las ecuaciones (3.1) y (3.2) dividen las unidades desplazadas de la esfera de la interfaz sobre sus desplazamientos máximos posibles; de esta manera las velocidades cambian gradualmente dependiente de que tanto se

mueva la interfaz.

Se debe aclarar que para facilitar el manejo del Falcon se añadieron zonas muertas en el espacio de movimiento de la interfaz. En y_f se declaró como zona muerta las primeras 7 unidades, y en el x_f las primeras 20 hacia la derecha y hacia la izquierda. Con esto podemos saber que la mínima velocidad lineal que manifiesta el maestro es 0.018 m/s y la mínima velocidad angular es 0.04 rad/s.

Capítulo 4

Esquema de control aplicado

En este capítulo se retomará parte de lo explicado en el Capítulo 2, pero con la diferencia de que se hará énfasis en lo propio realizado. Se mostrará el control utilizado y los filtros añadidos.

También se hará descripción del esquema de teleoperación utilizado, y del algoritmo para evadir obstáculos. Algo que se debe destacar es que la retroalimentación de posición del esclavo al maestro fue sustituida por la retroalimentación de fuerzas, lo cual causó que el Predictor de Smith no fuera utilizado.

4.1. Control por Modelo Interno

En la Sección 2.5 se habló de forma general del IMC explicando el principio que lo fundamenta y un poco de su teoría usando de apoyo un diagrama.

Recordando la matriz de la ecuación (2.17), tenemos que el modelo se puede tomar como un doble integrador, por lo tanto el control es un doble derivador (el subíndice indica que pertenece al esclavo) .

$$C(s)_S = G'^{-1}(s) = \left(\frac{1}{s^2}\right)^{-1} = s^2 \quad (4.1)$$

Aunque un doble integrador es realizable, su recíproco no lo es, y este problema al obtener el inverso de una función de transferencia con este tipo de control suele ocurrir con frecuencia, por lo que es necesario aplicar un filtro que tenga un efecto muy pequeño, pero que vuelva realizable la función del control. El filtro utilizado es el siguiente:

$$f(s) = \frac{n\lambda s + 1}{(\lambda s + 1)^n} \quad (4.2)$$

Donde

- n es el grado del filtro, en este caso 4.
- λ es el parámetro de ajuste de respuesta del sistema, en nuestro caso $\frac{1}{5}$.

El filtro utilizado con los parámetros asignados ya sustituidos es el siguiente:

$$f(s) = \frac{\frac{4}{5}s + 1}{(\frac{1}{5}s + 1)^4} \quad (4.3)$$

El control por lo tanto tiene la siguiente expresión:

$$C(s)_S = f(s)s^2 = \frac{\frac{4}{5}s^3 + s^2}{(\frac{1}{5}s + 1)^4} \quad (4.4)$$

Como se puede observar el control paso de ser irrealizable (doble derivador), a una expresión realizable (ec. 4.4).

Se debe recordar que Gazebo produce ruido para hacer una simulación mas realista (ruido gaussiano), y además se presentan algunas dinámicas que no fueron contempladas en el modelo matemático del sistema. Sin embargo este se amplifica mucho en el esclavo, por lo que se debió añadir un filtro ($F_p(s)$) que redujera el efecto del ruido antes de calcularse el error.

El filtro $F_p(s)$ esta expresado en la ecuación (4.5)

$$F_p(s) = \frac{A(s)}{Q(s)P(s)} \quad (4.5)$$

Donde $A(s)/P(s)$ representa un modelo interno de perturbaciones, y $Q(s)$ una función polinomial. En nuestro caso $Q(s)$ y $P(s)$ deben ser funciones con **raíces estables** para garantizar la estabilidad del filtro. Para ello se utilizan las siguientes funciones:

$$\begin{aligned} Q(s) &= 1 \\ P(s) &= (T_0s + 1)^r \end{aligned} \quad (4.6)$$

Donde T_0 es un tiempo constante, y r es el orden del filtro. Mientras mas pequeños sean los valores asignados a estos dos parámetros, mas rápido es la absorción de perturbaciones según [8]. En nuestro caso se eligieron los siguientes valores: $T_0 = 0.5$ y $r = 3$.

Para terminar el filtro se asignó $A(s) = (\epsilon s + 1)^2$, donde se dio a ϵ el valor de 1. Estos valores fueron encontrados mediante prueba y error, donde también se verificó que la ecuación (4.5) convergía más rápido a su fase estable. El filtro para ruido, con los valores sustituidos queda de la siguiente forma:

$$F_p(s) = \frac{(s + 1)^2}{(0.5s + 1)^3} \quad (4.7)$$

En la Figura 4.1 se puede observar el diagrama del IMC que se aplicó al esclavo, con el filtro recién mencionado. Esta figura es análoga a la de la Figura 2.2, por lo que no se volverán a explicar los bloques ni sus relaciones.

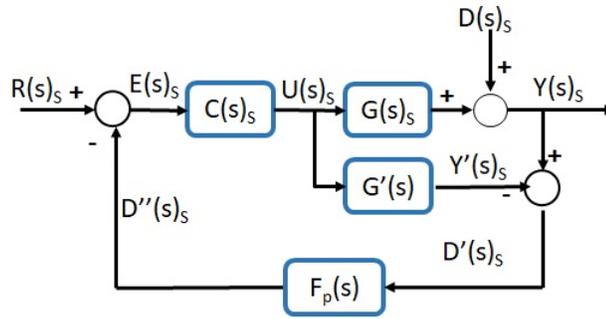


Figura 4.1: Diagrama del IMC en el esclavo

Se debe recordar que el sistema es desacoplado con 2 entradas y 2 salidas, por lo que se forman dos ecuaciones y a ambas se les aplica el control y los filtros ya explicados.

4.2. Evasión de obstáculos

Después de usar el IMC y añadir la interfaz se procedió a trabajar en el GOACM (ver [10] para más información), el cual consiste en un algoritmo capaz de calcular un punto de seguridad, localizado a cierta distancia de un obstáculo, dando inicialmente una sugerencia para la evasión.

En nuestro caso particular se tiene que el robot utilizado cuenta con un sensor que gira mapeando constantemente los alrededores y permite obtener las distancias a los objetos cercanos al móvil de forma rápida. Las mediciones son obtenidas en vectores de 360 elementos (en cada grado en el giro del sensor se hace una medición), los cuales contienen las distancias a las que se detectan presencias.

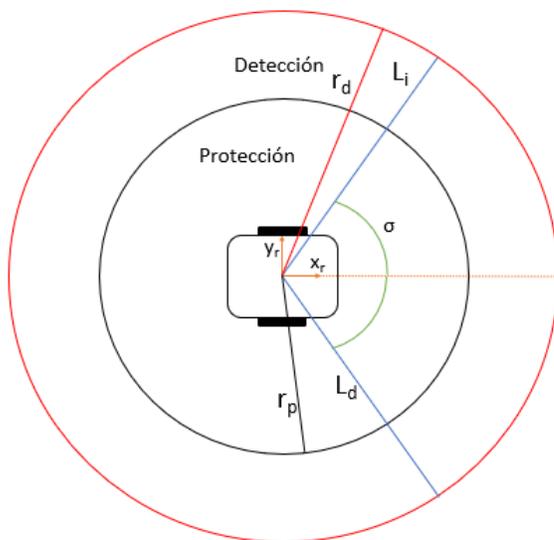


Figura 4.2: Áreas de detección y protección

La figura 4.2 muestra de forma gráfica como se distribuyen las áreas utilizadas para la detección y evasión de obstáculos. El área de detección tiene un radio r_d . El círculo de protección tiene un radio de r_p , el cual corresponde a la mínima distancia considerada para lograr evadir completamente un objeto.

Como el robot siempre se mueve hacia adelante, por decisión de restringirse a solo velocidades lineales positivas, se decidió manejar una porción del área circular limitada por L_i y L_d , y dada por el ángulo σ (de un valor de $\frac{\pi}{2}$ radianes); la mitad de este ángulo está definido con respecto al eje x_r . Los datos obtenidos en el resto del círculo son ignorados pues no habría manera de colisionar en esas direcciones.

A continuación, con ayuda de la figura 4.3, se explica como se obtiene el punto de seguridad para la evasión de obstáculos. Todo esto teniendo en cuenta que únicamente se utilizan valores que están dentro de la sección del círculo mencionada.

El objeto con el cual puede haber colisión esta representado por el arco inferior de color negro de la figura 4.3. Los puntos limite del objeto están dados por P_1 y P_2 . Sus respectivos vectores de distancia al móvil son d_1 y d_2 . Para facilitar las cosas se genera un vector d_3 (entre los puntos P_1 y P_2) el cual servirá para los siguientes cálculos. Se calcula la magnitud de d_3 con la siguiente ecuación:

$$d_3 = \sqrt{(P_{1x} - P_{2x})^2 + (P_{1y} - P_{2y})^2} \quad (4.8)$$

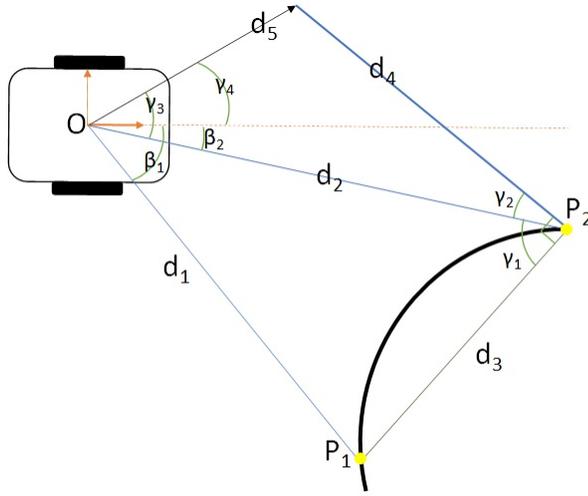


Figura 4.3: Construcción del punto de seguridad

A partir de este punto se usaran las siguientes asignaciones de variables para poder explicar de forma general el procedimiento, pues algunas cosas cambian dependiendo de la localización del obstáculo (si mayormente esta a la derecha o izquierda con respecto al eje x_r del robot).

Lo siguiente que se muestra, sirve para explicar de forma general el procedimiento que realiza el algoritmo. Dependiendo de la localización del obstáculo (si mayormente esta a la derecha o izquierda con respecto al eje x_r del robot), se elige el primer cambio de variables o el segundo.

<i>Derecha</i>	<i>Izquierda</i>
$L_1 = d_1$	$L_1 = d_2$
$L_2 = d_2$	$L_2 = d_1$
$\beta_s = \beta_1 - \beta_2$	$\beta_s = \beta_2 - \beta_1$

Se obtiene el valor del ángulo γ_1 :

$$\gamma_1 = \text{acos} \left(\frac{L_2 - L_1 \cos(\beta_s)}{d_3} \right) \quad (4.9)$$

Como el vector d_4 debe estar a $\frac{\pi}{2}$ de d_3 , se puede calcular el ángulo γ_2 de la siguiente forma:

Si $\gamma_1 \neq \frac{\pi}{2}$:

$$\gamma_2 = \frac{\pi}{2} - \gamma_1 \quad (4.10)$$

Y si $\gamma_1 = \frac{\pi}{2}$:

$$\gamma_2 = \frac{\pi}{2} - \gamma + \epsilon \quad (4.11)$$

El valor de ϵ debe ser pequeño, y debe usarse cuando alguno de los vectores d_1 o d_2 estén a $\frac{\pi}{2}$ de d_3 .

La distancia de d_4 es asignada por elección. La siguiente distancia a buscar es d_5 , la cual se puede obtener mediante la ecuación (4.12) obtenida a partir de la ley de los cosenos:

$$d_5 = L_2^2 + d_4^2 - 2L_2d_4\cos(\gamma_2) \quad (4.12)$$

Se busca el valor de γ_3 para utilizarse después:

$$\gamma_3 = \arccos\left(\frac{L_2 - d_4\cos(\gamma_2)}{d_5}\right) \quad (4.13)$$

Finalmente se calcula γ_4 , el cual es medido desde el eje x del sistema del móvil:

$$\gamma_4 = \gamma_3 + \beta_2 \quad (4.14)$$

Si el objeto esta a la izquierda, o la mayor parte de este se localiza ahí, se utiliza la siguiente ecuación:

$$\gamma_4 = -\gamma_3 + \beta_1 \quad (4.15)$$

Para el caso de que se encuentren dos objetos cerca como obstáculos, con la ayuda de la figura 4.4 se muestra la obtención del punto de seguridad (punto negro de la recta b_{12}).

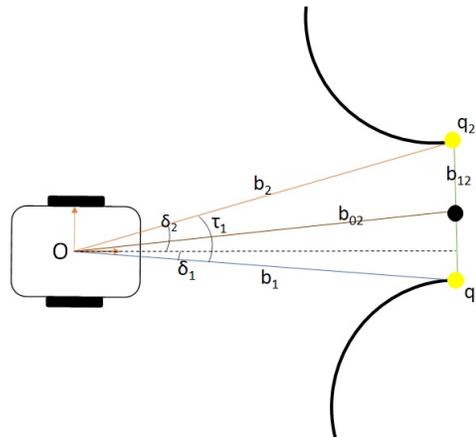


Figura 4.4: Obtención del punto de seguridad cuando hay 2 objetos

Los obstáculos están representados con los arcos de color negro localizados en el lado derecho. Los punto q_1 y q_2 corresponden a los puntos mas cercanos entre si que crean una abertura. Los vectores b_1 y b_2 representan las distancias del móvil a los limites de la abertura.

Primero se obtienen los ángulos δ_1 y δ_2 con los componentes de los vectores b_1 y b_2 :

$$\delta_1 = \text{atan} \left(\frac{q_{1y}}{q_{1x}} \right) \quad (4.16)$$

$$\delta_2 = \text{atan} \left(\frac{q_{2y}}{q_{2x}} \right) \quad (4.17)$$

Se calcula el ángulo de abertura de estos vectores, el cual es τ_1 :

$$\tau_1 = |\delta_1| + |\delta_2| \quad (4.18)$$

La magnitud de la recta b_{12} se obtiene mediante las componentes utilizadas en las ecuaciones (4.16) y (4.17):

$$b_{12} = \sqrt{(q_{1x} - q_{2x})^2 + (q_{1y} - q_{2y})^2} \quad (4.19)$$

Con la medida de esta abertura se decide si es viable que el vehículo pase a través de el. Para ello se compara este valor con una distancia mínima que debe existir para que el robot se desplace sin problemas.

Si se da el caso de que el móvil no pueda moverse entre estos dos obstáculos, se toman los dos objetos en conjunto como uno solo y de esa forma se evaden siguiendo los pasos ya explicados. En el caso de que las condiciones sean favorables, se procede a obtener el punto de seguridad entre los dos objetos.

Este punto de seguridad es representado por el punto negro en medio de la recta b_{12} , y con sus coordenadas se forma el vector b_{02} , las cuales se obtienen de la siguiente forma:

$$b_{02x} = \frac{q_{1x} + q_{2x}}{2} \quad (4.20)$$

$$b_{02y} = \frac{q_{1y} + q_{2y}}{2} \quad (4.21)$$

Con esto ya se tiene la trayectoria de seguridad que el móvil debe realizar para cruzar entre los 2 objetos.

En resumen, si se encuentra un objeto con el que exista peligro de colisión se decide la dirección hacia donde girar para asegurar la evasión. Si se encuentran dos objetos

cerca en la región frente al móvil, se busca la posibilidad de pasar entre estos dos; y si esto es inviable se toma como un solo objeto y se evade girando.

Una vez decidido el punto de seguridad para la evasión, se utiliza el ángulo γ_4 para dividir en componentes la fuerza asignada con la finalidad de poder expresar dicha fuerza en la interfaz háptica (ver Figura 3.5). También se realiza una multiplicación de la componente por un cociente que relaciona el radio r_p y la distancia al obstáculo d_o . Esto último hace que la fuerza en el Falcon cambie gradualmente dependiendo de que tan cerca este el esclavo del objeto. Todo esto esta expresado en las ecuaciones (4.22) y (4.23).

$$f_x = -F_a \sin(\gamma_4) \frac{r_p}{d_o} \quad (4.22)$$

$$f_y = F_a \cos(\gamma_4) \frac{r_p}{d_o} \quad (4.23)$$

Donde :

- $f_x, f_y \rightarrow$ Componentes de la fuerzas aplicada.
- $F_a \rightarrow$ Fuerza nominal asignada al Falcon.
- $\gamma_4 \rightarrow$ Ángulo de orientación en la linea de evasión.
- $r_p \rightarrow$ Radio de protección.
- $d_o \rightarrow$ Distancia al obstáculo.

4.3. Esquema de teleoperación

Hasta este punto ya se conocen los tipos de arquitectura en la teleoperación. En [8] se utilizó un sistema de teleoperación bilateral PERR. En nuestro caso, como se buscó añadir cosas nuevas tuvimos que hacer cambios gradualmente.

Primero, al añadirse la interfaz Novit Falcon el sistema se volvió unilateral. Esto es debido a que ya no se podía realizar la retroalimentación de la posición del esclavo, pues el maestro ya no seguía una referencia expresada en ecuaciones, sino las instrucciones del usuario. En la Figura 4.5 se muestra la configuración de esta nueva arquitectura temporal.

El bloque $S(s)_S$ representa a la Figura 4.1, pues es la parte correspondiente al esclavo y su IMC. El bloque $H(s)$ representa la relación entre el movimiento de la interfaz y la generación de velocidades lineal y angular (ec. 3.1 y 3.2), mientras que el bloque $M(s)$ representa al maestro (en este caso $M(s) \neq G(s)$ aunque ambos sean robots

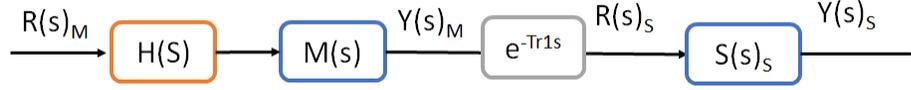


Figura 4.5: Esquema de teleoperación sin retroalimentación

móviles diferenciales). El retardo de la transmisión de la información del maestro al esclavo esta representado por e^{-Tr1S} .

Después de haber adaptado la interfaz al sistema, se procedió a añadir el algoritmo para la evasión de obstáculos por lo que la configuración del sistema de teleoperación cambio y este se volvió un sistema bilateral con retroalimentación de fuerzas kinestésicas.

Con el algoritmo que implementa el GOACM, se pudo obtener la dirección de seguridad, que se expresa como fuerzas proporcionales en la interfaz, para ayudar al usuario primeramente sugiriéndole modificaciones a la trayectoria que se busca realizar en el esclavo. La arquitectura del sistema en conjunto cambia al mostrado en la figura 4.6.

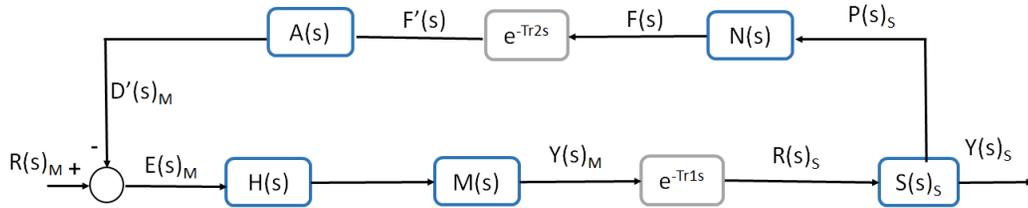


Figura 4.6: Diagrama de bloques del sistema general simplificado

El bloque e^{-Tr2S} representa el retardo de la retroalimentación del esclavo al maestro. El bloque $N(s)$ hace referencia a las ecuaciones (4.22) y (4.23), pues en esta parte se genera el valor de las fuerzas a partir de la distancia a la que se encuentra el esclavo del objeto; y el bloque $A(s)$ representa la acción de las fuerzas en la interfaz, las cuales causan un desplazamiento que influye en el movimiento realizado por el usuario para introducir la referencia al sistema. Este ultimo bloque hace referencia a las ecuaciones (4.24 y 4.25).

$$d_{fx} = \frac{f'_x}{k_{1x}} \quad (4.24)$$

$$d_{fy} = \frac{f'_y}{k_{1y}} \quad (4.25)$$

Donde:

- d_{fx} & d_{fy} : son los desplazamientos en x y y ocasionados por las fuerzas.
- f'_x & f'_y : son las fuerzas reflejadas en la interfaz, en sus ejes x y y .
- k_{1x} & k_{1y} : son las constantes que relacionan los elementos antes mencionados.

El bloque $S(s)_S$ representa a la Figura 4.1, pues es la parte correspondiente al esclavo y su IMC.

4.4. Análisis de estabilidad

Partiendo del diagrama de la Figura 4.1, se puede obtener la siguiente ecuación del error (4.26):

$$E(s)_S = R(s)_S - D''(s)_S = R(s)_S - F_P(s)D'(s)_S \quad (4.26)$$

El valor de $D'(s)_S$ es el siguiente:

$$D'(s)_S = C(s)_S [G(s)_S - G'(s)] E(s)_S \quad (4.27)$$

Se sustituye la ecuación (4.27) en la (4.26) y se despeja el error obteniendo:

$$E(s)_S = \frac{R(s)_S}{1 + F_p(s)C(s)_S [G(s)_S - G'(s)]} \quad (4.28)$$

Ahora se expresa la función de transferencia $\frac{E(s)_S}{R(s)_S}$ a partir de la ecuación anterior.

$$\frac{E(s)_S}{R(s)_S} = \frac{1}{1 + F_p(s)C(s)_S [G(s)_S - G'(s)]} \quad (4.29)$$

Para poder obtener su valor, se utilizan las ecuaciones del control (4.4) y del filtro de ruido (4.7). Para $G(s)_S$ se utiliza la linealización $\frac{1}{s^2}$ y para $G'(s)$ se utiliza la expresión $\frac{1}{s^2 + a_1s + b_1}$ donde a_1 y b_1 son constantes de incertidumbre del modelo.

Sustituyendo todo lo necesario en la ecuación (4.29) se tiene lo siguiente:

$$\frac{E(s)_S}{R(s)_S} = \frac{A_{10}s^{10} + A_9s^9 + A_8s^8 + A_7s^7 + A_6s^6 + A_5s^5 + B_4s^4 + B_3s^3 + B_2s^2 + B_1s + B_0}{A_{10}s^{10} + A_9s^9 + A_8s^8 + A_7s^7 + A_6s^6 + A_5s^5 + A_4s^4 + A_3s^3 + A_2s^2 + A_1s + A_0} \quad (4.30)$$

Donde:

- $A_{10} = 0.0625$

- $A_9 = 0.0625a_1 + 1.75$
- $A_8 = 0.0625b_1 + 1.75a_1 + 20.875$
- $A_7 = 1.75b_1 + 20.875a_1 + 138.25$
- $A_6 = 20.875b_1 + 138.25a_1 + 555.0625$
- $A_5 = 555.0625a_1 + 138.25b_1 + 1382.5$
- $A_4 = 1882.5a_1 + 555.0625b_1 + 2087.5$
- $A_3 = 1750 + 3712.5a_1 + 1882.5b_1$
- $A_2 = 625 + 3500a_1 + 3712.5b_1$
- $A_1 = 3500b_1 + 1250a_1$
- $A_0 = 1250b_1$
- $B_4 = 2087.5 + 1382.5a_1 + 555.0625b_1$
- $B_3 = 1750 + 2087.5a_1 + 1382.5b_1$
- $B_2 = 625 + 1750a_1 + 2087.5b_1$
- $B_1 = 625a_1 + 1750b_1$
- $B_0 = 625b_1$

Para obtener la ecuación en espacio de estados, a la función de transferencia de la ecuación (4.30) se le multiplica por $\frac{s^{-10}}{s^{-10}}$, dando lo siguiente:

$$\frac{E(s)_S}{R(s)_S} = \frac{A_{10} + A_9s^{-1} + A_8s^{-2} + A_7s^{-3} + A_6s^{-4} + A_5s^{-5} + B_4s^{-6} + B_3s^{-7} + B_2s^{-8} + B_1s^{-9} + B_0s^{-10}}{A_{10} + A_9s^{-1} + A_8s^{-2} + A_7s^{-3} + A_6s^{-4} + A_5s^{-5} + A_4s^{-6} + A_3s^{-7} + A_2s^{-8} + A_1s^{-9} + A_0s^{-10}} \quad (4.31)$$

La función de transferencia utilizada se puede separa en un producto, lo cual se muestra a continuación.

$$\frac{E(s)_S}{R(s)_S} = \frac{E(s)_S}{E(s)} \frac{E(s)}{R(s)_S} \quad (4.32)$$

Donde $E(s)$ es un error provisional necesario por el método utilizado para obtener la ecuación en espacio de estados.

De este producto podemos obtener las siguientes ecuaciones:

$$\frac{E(s)}{E(s)}_S = A_{10} + A_9 s^{-1} + A_8 s^{-2} + A_7 s^{-3} + A_6 s^{-4} + A_5 s^{-5} + B_4 s^{-6} + B_3 s^{-7} + B_2 s^{-8} + B_1 s^{-9} + B_0 s^{-10} \quad (4.33)$$

$$\frac{E(s)}{R(s)}_S = \frac{1}{A_{10} + A_9 s^{-1} + A_8 s^{-2} + A_7 s^{-3} + A_6 s^{-4} + A_5 s^{-5} + A_4 s^{-6} + A_3 s^{-7} + A_2 s^{-8} + A_1 s^{-9} + A_0 s^{-10}} \quad (4.34)$$

Tomando la ecuación (4.34), se realiza el siguiente cambio de variables:

- $\hat{x}_1 = s^{-10} E(s)$
- $\dot{\hat{x}}_1 = \hat{x}_2 = s^{-9} E(s)$
- $\dot{\hat{x}}_2 = \hat{x}_3 = s^{-8} E(s)$
- $\dot{\hat{x}}_3 = \hat{x}_4 = s^{-7} E(s)$
- $\dot{\hat{x}}_4 = \hat{x}_5 = s^{-6} E(s)$
- $\dot{\hat{x}}_5 = \hat{x}_6 = s^{-5} E(s)$
- $\dot{\hat{x}}_6 = \hat{x}_7 = s^{-4} E(s)$
- $\dot{\hat{x}}_7 = \hat{x}_8 = s^{-3} E(s)$
- $\dot{\hat{x}}_8 = \hat{x}_9 = s^{-2} E(s)$
- $\dot{\hat{x}}_9 = \hat{x}_{10} = s^{-1} E(s)$
- $\dot{\hat{x}}_{10} = A'_{10} R_S - A'_9 \hat{x}_{10} - A'_8 \hat{x}_9 - A'_7 \hat{x}_8 - A'_6 \hat{x}_7 - A'_5 \hat{x}_6 - A'_4 \hat{x}_5 - A'_3 \hat{x}_4 - A'_2 \hat{x}_3 - A'_1 \hat{x}_2 - A'_0 \hat{x}_1$

Donde:

- $A'_{10} = \frac{1}{A_{10}} = 16$
- $A'_9 = \frac{A_9}{A_{10}} = 28 + a_1$
- $A'_8 = \frac{A_8}{A_{10}} = 334 + b_1 + 28a_1$
- $A'_7 = \frac{A_7}{A_{10}} = 2212 + 28b_1 + 334a_1$
- $A'_6 = \frac{A_6}{A_{10}} = 8881 + 334b_1 + 2212a_1$
- $A'_5 = \frac{A_5}{A_{10}} = 22120 + 2212b_1 + 8881a_1$
- $A'_4 = \frac{A_4}{A_{10}} = 33400 + 8881b_1 + 30120a_1$

- $A'_3 = \frac{A_3}{A_{10}} = 28000 + 30120b_1 + 59400a_1$
- $A'_2 = \frac{A_2}{A_{10}} = 10000 + 59400b_1 + 56000a_1$
- $A'_1 = \frac{A_1}{A_{10}} = 56000b_1 + 20000a_1$
- $A'_0 = \frac{A_0}{A_{10}} = 20000b_1$

Por lo tanto la ecuación en espacio de estados resulta en

$$\begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \\ \dot{\hat{x}}_3 \\ \dot{\hat{x}}_4 \\ \dot{\hat{x}}_5 \\ \dot{\hat{x}}_6 \\ \dot{\hat{x}}_7 \\ \dot{\hat{x}}_8 \\ \dot{\hat{x}}_9 \\ \dot{\hat{x}}_{10} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -A'_0 & -A'_1 & -A'_2 & -A'_3 & -A'_4 & -A'_5 & -A'_6 & -A'_7 & -A'_8 & -A'_9 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \\ \hat{x}_5 \\ \hat{x}_6 \\ \hat{x}_7 \\ \hat{x}_8 \\ \hat{x}_9 \\ \hat{x}_{10} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ A'_{10} \end{bmatrix} R_S \quad (4.35)$$

Donde R_S es la referencia del esclavo, es decir, la salida del maestro después de cierto retardo (se debe recordar que la ecuación recién obtenida aun no expresa el retardo pues solo es el puro esclavo, por eso el retardo se añade mas adelante).

En cuanto a los retardos, recordemos que son e^{-Tr1s} y e^{-Tr2s} en la Figura 4.6. Las aproximaciones en Laplace utilizadas fueron tomadas de [14] con la finalidad de simplificar las cosas. Estas son las siguientes:

$$e^{-Tr1s} = \frac{1 - T_1s}{1 + T_1s} \quad (4.36)$$

$$e^{-Tr2s} = \frac{1 - T_2s}{1 + T_2s} \quad (4.37)$$

Donde:

$$T1 = \frac{Tr1}{2} \quad (4.38)$$

$$T2 = \frac{Tr2}{2} \quad (4.39)$$

Por lo tanto los retardos deben ser pequeños. No se eligieron aproximaciones de orden superior para no aumentar la complejidad del sistema tratado.

La ecuación en espacio de estados de las ecuaciones (4.36) y (4.37) son (4.40) y (4.41) respectivamente

$$\dot{\check{x}}_1 = \left[-\frac{1}{T_1} \right] \check{x}_1 + R_{r1} \quad (4.40)$$

$$\dot{\check{x}}_2 = \left[-\frac{1}{T_2} \right] \check{x}_2 + R_{r2} \quad (4.41)$$

Donde R_{r1} y R_{r2} son las entradas de los retardos. Las salidas que corresponden a las ecuaciones (4.40) y (4.41) son las (4.42) y (4.43) respectivamente

$$\check{y}_1 = \left[\frac{2}{T_1} \right] \check{x}_1 - R_{r1} \quad (4.42)$$

$$\check{y}_2 = \left[\frac{2}{T_2} \right] \check{x}_2 - R_{r2} \quad (4.43)$$

Se debe recordar que el esquema general de teleoperación utilizado puede separarse en 2 canales, en el de x y en el de y . En el caso de los retardos, para diferenciar en cuál canal están, se añade el subíndice de su eje correspondiente; por lo tanto, R_S del espacio de estados de la ecuación (4.35) puede ser R_{Sx} o R_{Sy} . Las variables recién mencionadas junto con los retardos son los siguientes

$$R_{Sx} = \check{y}_{x1} = \frac{2}{T_1} \check{x}_{x1} - R_{r1x} = \frac{2}{T_1} \check{x}_{x1} - x_m \quad (4.44)$$

$$R_{Sy} = \check{y}_{y1} = \frac{2}{T_2} \check{x}_{y1} - R_{r1y} = \frac{2}{T_2} \check{x}_{y1} - y_m \quad (4.45)$$

Para el maestro, se toma el modelo cinemático, el cual es el siguiente:

$$\begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \text{sen}(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_m \\ \omega_m \end{bmatrix} \quad (4.46)$$

Donde

- \dot{x}_m, \dot{y}_m son las velocidades en x y y del maestro.
- v_m, ω_m son las velocidades lineal y angular que entran al maestro.

Se sustituyen las ecuaciones (3.1) y (3.2) dando lo siguiente

$$\begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \cos(\theta_m) & 0 \\ \text{sen}(\theta_m) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{v_{mx}}{60} D_{fy} \\ \frac{-\omega_{mx}}{55} D_{fx} \end{bmatrix} \quad (4.47)$$

De la ecuación 4.47 se obtiene

$$\dot{x}_m = \cos(\theta_m) \frac{v_{mx}}{60} D_{fy} \quad (4.48)$$

$$\dot{y}_m = \text{sen}(\theta_m) \frac{v_{mx}}{60} D_{fy} \quad (4.49)$$

$$\dot{\theta}_m = \frac{-\omega_{mx}}{55} D_{fx} \quad (4.50)$$

De acuerdo al diagrama de la Figura 4.6, lo que entra en las ecuaciones (4.48) , (4.49) y (4.50) es $E(s)_M$, el cual tiene un valor en cada canal, siendo en x el valor de E_{Mx} y en el de y el valor de E_{My} . Por lo tanto las ecuaciones resultan en

$$\dot{x}_m = \cos(\theta) \frac{v_{mx}}{60} D_{fy} = \cos(\theta) \frac{v_{mx}}{60} E_{My} \quad (4.51)$$

$$\dot{y}_m = \text{sen}(\theta) \frac{v_{mx}}{60} D_{fy} = \text{sen}(\theta) \frac{v_{mx}}{60} E_{My} \quad (4.52)$$

$$\dot{\theta}_m = \frac{-\omega_{mx}}{55} D_{fx} = \frac{-\omega_{mx}}{55} E_{Mx} \quad (4.53)$$

Después se sustituyen los valores del error en su respectivo canal.

$$\dot{x}_m = \cos(\theta) \frac{v_{mx}}{60} E_{My} = \cos(\theta) \frac{v_{mx}}{60} [R_{My} - D'_{My}] \quad (4.54)$$

$$\dot{y}_m = \text{sen}(\theta) \frac{v_{mx}}{60} E_{My} = \text{sen}(\theta) \frac{v_{mx}}{60} [R_{My} - D'_{My}] \quad (4.55)$$

$$\dot{\theta}_m = \frac{-\omega_{mx}}{55} E_{Mx} = \frac{-\omega_{mx}}{55} [R_{Mx} - D'_{Mx}] \quad (4.56)$$

Si solo tratamos el lazo directo del esquema utilizado hasta ahora, tenemos que $D'_{Mx} = 0$ y $D'_{My} = 0$, por lo que las ecuaciones (4.54), (4.55) y (4.56) resultan en

$$\dot{x}_m = \cos(\theta) \frac{v_{mx}}{60} R_{My} \quad (4.57)$$

$$\dot{y}_m = \text{sen}(\theta) \frac{v_{mx}}{60} R_{My} \quad (4.58)$$

$$\dot{\theta}_m = \frac{-\omega_{mx}}{55} R_{Mx} \quad (4.59)$$

El análisis de estabilidad puede hacerse por separado en los 2 canales (x y y). Para x , únicamente tomamos las ecuaciones (4.54) y (4.56).

El sistema que hasta ahora se maneja tiene la siguiente forma

$$\dot{x} = f(t, x, u) \quad (4.60)$$

Según Khalil, en [15], es posible ver la función recién mencionada (4.60) como una forma perturbada de la (ec. 4.61), es decir, se puede tomar la entrada como una perturbación.

$$\dot{x} = f(t, x, 0) \quad (4.61)$$

Y para la ecuación (4.61) se puede aplicar diversas técnicas para analizar la estabilidad entrada-estado de la ecuación (4.60); en otras palabras, si se puede demostrar que un sistema sin perturbaciones es uniformemente asintóticamente estable, el mismo sistema pero con perturbaciones acotadas también lo será. Por esta razón, en el caso del sistema que se utiliza ahora, se puede tomar que R_{Mx} y R_{My} son iguales a cero, con lo cual se manejaría un sistema sin "perturbaciones". Con esto las velocidades \dot{x}_m, \dot{y}_m y $\dot{\theta}_m$ también valen cero.

Tomando las ecuaciones del primer retardo (ec. 4.40 pero en el canal x), se debe añadir un estado más a la ecuación (4.35), pues el análisis por medio de Lyapunov exige que se manejen todos los estados del sistema. El resultado (linealizado) es

$$\begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \\ \dot{\hat{x}}_3 \\ \dot{\hat{x}}_4 \\ \dot{\hat{x}}_5 \\ \dot{\hat{x}}_6 \\ \dot{\hat{x}}_7 \\ \dot{\hat{x}}_8 \\ \dot{\hat{x}}_9 \\ \dot{\hat{x}}_{10} \\ \dot{\check{x}}_{x1} \end{bmatrix} = \begin{bmatrix} M_1 & M_2 & M_3 \\ M_4 & M_5 & M_6 \\ M_7 & M_8 & M_9 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \\ \hat{x}_5 \\ \hat{x}_6 \\ \hat{x}_7 \\ \hat{x}_8 \\ \hat{x}_9 \\ \hat{x}_{10} \\ \check{x}_{x1} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ A'_{10} \\ 1 \end{bmatrix} x_m \quad (4.62)$$

Donde

$$M_1 = 0_{9 \times 1} \quad (4.63)$$

$$M_2 = I_{9 \times 9} \quad (4.64)$$

$$M_3 = 0_{9 \times 1} \quad (4.65)$$

$$M_4 = [-A'_0] \quad (4.66)$$

$$M_5 = \begin{bmatrix} -A'_1 & -A'_2 & -A'_3 & -A'_4 & -A'_5 & -A'_6 & -A'_7 & -A'_8 & -A'_9 \end{bmatrix} \quad (4.67)$$

$$M_6 = \begin{bmatrix} 2A'_{10} \\ T_1 \end{bmatrix} \quad (4.68)$$

$$M_7 = [0] \quad (4.69)$$

$$M_8 = 0_{1 \times 9} \quad (4.70)$$

$$M_9 = \begin{bmatrix} -1 \\ T_1 \end{bmatrix} \quad (4.71)$$

La matriz de estados de la ecuación (4.62) es de valores constantes en función de las ganancias y parámetros del sistema.

Se debe señalar que para simplificar las cosas se elige probar la estabilidad mediante Lyapunov, pues el método de obtener los polos es algo complicado aquí debido a que no se obtienen soluciones analíticas que permitan concluir condiciones sobre las ganancias y parámetros. Además se tiene que tener en cuenta que este es un análisis de estabilidad muy conservador, y local pues se basa en la aproximación, y por lo tanto existen limitaciones en el método. Con los estados de la matriz de la ecuación (4.62) proponemos lo siguiente como función de Lyapunov.

$$V = \frac{1}{2}(\hat{x}_1^2 + \hat{x}_2^2 + \hat{x}_3^2 + \hat{x}_4^2 + \hat{x}_5^2 + \hat{x}_6^2 + \hat{x}_7^2 + \hat{x}_8^2 + \hat{x}_9^2 + \hat{x}_{10}^2 + \check{x}_{x1}^2) \quad (4.72)$$

Se obtiene la derivada de la ecuación (4.72)

$$\dot{V} = \hat{x}_1\dot{\hat{x}}_1 + \hat{x}_2\dot{\hat{x}}_2 + \hat{x}_3\dot{\hat{x}}_3 + \hat{x}_4\dot{\hat{x}}_4 + \hat{x}_5\dot{\hat{x}}_5 + \hat{x}_6\dot{\hat{x}}_6 + \hat{x}_7\dot{\hat{x}}_7 + \hat{x}_8\dot{\hat{x}}_8 + \hat{x}_9\dot{\hat{x}}_9 + \hat{x}_{10}\dot{\hat{x}}_{10} + \check{x}_{x1}\dot{\check{x}}_{x1} \quad (4.73)$$

Se sustituyen los valores de las derivadas

$$\begin{aligned} \dot{V} = & \hat{x}_1\hat{x}_2 + \hat{x}_2\hat{x}_3 + \hat{x}_3\hat{x}_4 + \hat{x}_4\hat{x}_5 + \hat{x}_5\hat{x}_6 + \hat{x}_6\hat{x}_7 + \hat{x}_7\hat{x}_8 + \hat{x}_8\hat{x}_9 + \hat{x}_9\hat{x}_{10} + \hat{x}_{10}(-A'_0\hat{x}_1 \\ & - A'_1\hat{x}_2 - A'_2\hat{x}_3 - A'_3\hat{x}_4 - A'_4\hat{x}_5 - A'_5\hat{x}_6 - A'_6\hat{x}_7 - A'_7\hat{x}_8 - A'_8\hat{x}_9 - A'_9\hat{x}_{10} \\ & - A'_{10}x_m) + \check{x}_{x1}\left(\frac{-\check{x}_{x1}}{T_1} + x_m\right) \end{aligned} \quad (4.74)$$

Se reorganizan los términos obteniendo

$$\begin{aligned} \dot{V} = & -A'_9\hat{x}_{10}^2 + \hat{x}_1(\hat{x}_2 - A'_0\hat{x}_{10}) + \hat{x}_2(\hat{x}_3 - A'_1\hat{x}_{10}) + \hat{x}_3(\hat{x}_4 - A'_2\hat{x}_{10}) + \hat{x}_4(\hat{x}_5 - A'_3\hat{x}_{10}) \\ & + \hat{x}_5(\hat{x}_6 - A'_4\hat{x}_{10}) + \hat{x}_6(\hat{x}_7 - A'_5\hat{x}_{10}) + \hat{x}_7(\hat{x}_8 - A'_6\hat{x}_{10}) + \hat{x}_8(\hat{x}_9 - A'_7\hat{x}_{10}) \\ & + \hat{x}_9\hat{x}_{10}(1 - A'_8) + \check{x}_{x1}\left(\frac{2A'_{10}}{T_1}\hat{x}_{10} - \frac{\check{x}_{x1}}{T_1}\right) + (\check{x}_{x1} - A'_{10}\hat{x}_{10})x_m \end{aligned} \quad (4.75)$$

Considerando el primer término de la ecuación (4.75), para que esta sea negativa se debe cumplir que

$$A'_9 > 0 \quad (4.76)$$

Recordando que $A'_9 = 28 + a_1$, sustituimos en la desigualdad anterior, obteniendo

$$28 + a_1 > 0 \quad (4.77)$$

Se despeja la ecuación (4.77) para obtener la condición de a_1 , lo que resulta en

$$a_1 > -28 \quad (4.78)$$

Como el resto de la ecuación no tiene signo definido es necesario acotar, lo cual se muestra a continuación

$$\hat{x}_1(\hat{x}_2 - A'_0\hat{x}_{10}) \leq |\hat{x}_1| |\hat{x}_2 - A'_0\hat{x}_{10}| \leq \|\hat{x}\| \sqrt{1 + A'^2_0} \|\hat{x}\| = \sqrt{1 + A'^2_0} \|\hat{x}\|^2 \quad (4.79)$$

$$\hat{x}_2(\hat{x}_3 - A'_1\hat{x}_{10}) \leq |\hat{x}_2| |\hat{x}_3 - A'_1\hat{x}_{10}| \leq \|\hat{x}\| \sqrt{1 + A'^2_1} \|\hat{x}\| = \sqrt{1 + A'^2_1} \|\hat{x}\|^2 \quad (4.80)$$

$$\hat{x}_3(\hat{x}_4 - A'_2\hat{x}_{10}) \leq |\hat{x}_3| |\hat{x}_4 - A'_2\hat{x}_{10}| \leq \|\hat{x}\| \sqrt{1 + A'^2_2} \|\hat{x}\| = \sqrt{1 + A'^2_2} \|\hat{x}\|^2 \quad (4.81)$$

$$\hat{x}_4(\hat{x}_5 - A'_3\hat{x}_{10}) \leq |\hat{x}_4| |\hat{x}_5 - A'_3\hat{x}_{10}| \leq \|\hat{x}\| \sqrt{1 + A'^2_3} \|\hat{x}\| = \sqrt{1 + A'^2_3} \|\hat{x}\|^2 \quad (4.82)$$

$$\hat{x}_5(\hat{x}_6 - A'_4\hat{x}_{10}) \leq |\hat{x}_5| |\hat{x}_6 - A'_4\hat{x}_{10}| \leq \|\hat{x}\| \sqrt{1 + A'^2_4} \|\hat{x}\| = \sqrt{1 + A'^2_4} \|\hat{x}\|^2 \quad (4.83)$$

$$\hat{x}_6(\hat{x}_7 - A'_5\hat{x}_{10}) \leq |\hat{x}_6| |\hat{x}_7 - A'_5\hat{x}_{10}| \leq \|\hat{x}\| \sqrt{1 + A'^2_5} \|\hat{x}\| = \sqrt{1 + A'^2_5} \|\hat{x}\|^2 \quad (4.84)$$

$$\hat{x}_7(\hat{x}_8 - A'_6\hat{x}_{10}) \leq |\hat{x}_7| |\hat{x}_8 - A'_6\hat{x}_{10}| \leq \|\hat{x}\| \sqrt{1 + A'^2_6} \|\hat{x}\| = \sqrt{1 + A'^2_6} \|\hat{x}\|^2 \quad (4.85)$$

$$\hat{x}_8(\hat{x}_9 - A'_7\hat{x}_{10}) \leq |\hat{x}_8| |\hat{x}_9 - A'_7\hat{x}_{10}| \leq \|\hat{x}\| \sqrt{1 + A'^2_7} \|\hat{x}\| = \sqrt{1 + A'^2_7} \|\hat{x}\|^2 \quad (4.86)$$

$$\hat{x}_9\hat{x}_{10}(1 - A'_8) \leq (1 - A'_8) |\hat{x}_9| |\hat{x}_{10}| \leq (1 - A'_8) \|\hat{x}\| \|\hat{x}\| = (1 - A'_8) \|\hat{x}\|^2 \quad (4.87)$$

$$\check{x}_{x1} \left(\frac{2A'_0}{T_1} \hat{x}_{10} - \frac{\check{x}_{x1}}{T_1} \right) \leq |\check{x}_{x1}| \left| \frac{2A'_0}{T_1} \hat{x}_{10} - \frac{\check{x}_{x1}}{T_1} \right| \leq \|\hat{x}\| \sqrt{\frac{4A'^2_{10} - 1}{T_1^2}} \|\hat{x}\| = \sqrt{\frac{4A'^2_{10} - 1}{T_1^2}} \|\hat{x}\|^2 \quad (4.88)$$

$$(\check{x}_{x1} - A'_{10}\hat{x}_{10})x_m \leq |\check{x}_{x1} - A'_{10}\hat{x}_{10}| \tilde{x}_m \leq \sqrt{1 + A'^2_{10}} \|\hat{x}\| \tilde{x}_m \quad (4.89)$$

Donde $\|\hat{x}\|$ es la norma 2 de los estados de la ecuación (4.62), y \tilde{x}_m es igual a $|x_m|$.

Con las desigualdades de la (4.79) a la (4.89), y la ecuación (4.75) se obtiene lo siguiente

$$\begin{aligned} \dot{V} \leq & -A'_9\hat{x}_{10}^2 + \sqrt{1 + A'^2_0} \|\hat{x}\|^2 + \sqrt{1 + A'^2_1} \|\hat{x}\|^2 + \sqrt{1 + A'^2_2} \|\hat{x}\|^2 + \sqrt{1 + A'^2_3} \|\hat{x}\|^2 \\ & + \sqrt{1 + A'^2_4} \|\hat{x}\|^2 + \sqrt{1 + A'^2_5} \|\hat{x}\|^2 + \sqrt{1 + A'^2_6} \|\hat{x}\|^2 + \sqrt{1 + A'^2_7} \|\hat{x}\|^2 \\ & + (1 - A'_8) \|\hat{x}\|^2 + \frac{\sqrt{4A'^2_{10} - 1}}{T_1} \|\hat{x}\|^2 + \sqrt{1 + A'^2_{10}} \tilde{x}_m \|\hat{x}\| \end{aligned} \quad (4.90)$$

Lo anterior (ec. 4.90) se puede reacomodar de la siguiente forma

$$\dot{V} \leq -A'_9 \hat{x}_{10}^2 + \Omega \|\hat{x}\|^2 + (1 - A'_8) \|\hat{x}\|^2 + \frac{\sqrt{4A'_{10}{}^2 - 1}}{T_1} \|\hat{x}\|^2 + \sqrt{1 + A'_{10}{}^2} \tilde{x}_m \|\hat{x}\| \quad (4.91)$$

Donde

$$\Omega = \sqrt{1 + A'_0{}^2} + \sqrt{1 + A'_1{}^2} + \sqrt{1 + A'_2{}^2} + \sqrt{1 + A'_3{}^2} + \sqrt{1 + A'_4{}^2} + \sqrt{1 + A'_5{}^2} + \sqrt{1 + A'_6{}^2} + \sqrt{1 + A'_7{}^2} \quad (4.92)$$

De la ecuación (4.91), solo el primer término es definido negativo, y es necesario que el resto de la ecuación sea negativa o igual a cero, es decir

$$\Omega \|\hat{x}\|^2 + (1 - A'_8) \|\hat{x}\|^2 + \frac{\sqrt{4A'_{10}{}^2 - 1}}{T_1} \|\hat{x}\|^2 + \sqrt{1 + A'_{10}{}^2} \tilde{x}_m \|\hat{x}\| \leq 0 \quad (4.93)$$

De lo anterior se despeja T_1 , obteniendo

$$T_1 \leq \frac{-\sqrt{4A'_{10}{}^2 - 1} \|\hat{x}\|}{\|\hat{x}\| (\Omega + 1 - A'_8) + \sqrt{1 + A'_{10}{}^2} \tilde{x}_m} = \frac{-31.9843 \|\hat{x}\|}{\|\hat{x}\| (\Omega + 1 - A'_8) + 16.0312 \tilde{x}_m} \quad (4.94)$$

Debido a que no se conoce $\|\hat{x}\|$ y deseamos saber si es el sistema es asintóticamente estable, se debe trabajar con los valores iniciales de forma que se puede sustituir la norma recién mencionada por la norma de las condiciones iniciales de los estados, es decir $\|\hat{x}_0\|$. El resultado obtenido de esta acción es local, ya que depende de una región de convergencia en función de las condiciones iniciales, de forma que estas delimitan la frontera de la región de estabilidad del sistema. A partir del cambio mencionado se tiene la siguiente cota para T_1

$$T_1 \leq \frac{-31.9843 \|\hat{x}_0\|}{\|\hat{x}_0\| (\Omega + 1 - A'_8) + 16.0312 \tilde{x}_m} \quad (4.95)$$

Donde el denominador debe ser diferente de cero, tal como se muestra a continuación

$$\|\hat{x}_0\| (\Omega + 1 - A'_8) + \sqrt{1 + A'_{10}{}^2} \tilde{x}_m > 0 \quad (4.96)$$

$$\|\hat{x}_0\| (\Omega + 1 - A'_8) + \sqrt{1 + A'_{10}{}^2} \tilde{x}_m < 0 \quad (4.97)$$

Para la ecuación (4.96) se despeja A'_8

$$A'_8 < \frac{\sqrt{1 + A'_{10}{}^2 \tilde{x}_m}}{\|\hat{x}_0\|} + \Omega + 1 \quad (4.98)$$

Se sustituye el valor de A'_8 , A'_{10} y se despeja b_1 obteniendo

$$334 + b_1 + 28a_1 < \frac{16.0312\tilde{x}_m}{\|\hat{x}_0\|} + \Omega + 1 \quad (4.99)$$

$$b_1 < \frac{16.0312\tilde{x}_m}{\|\hat{x}_0\|} + \Omega - 333 - 28a_1 \quad (4.100)$$

Para la ecuación (4.97) también se despeja A'_8

$$A'_8 > \frac{\sqrt{1 + A'_{10}{}^2 \tilde{x}_m}}{\|\hat{x}_0\|} + \Omega + 1 \quad (4.101)$$

Se sustituyen los valores de A'_8 y A'_{10} , y se despeja b_1 , resultando en

$$b_1 > \frac{16.0312\tilde{x}_m}{\|\hat{x}_0\|} + \Omega - 333 - 28a_1 \quad (4.102)$$

Por lo tanto se debe cumplir que

$$b_1 \neq \frac{16.0312\tilde{x}_m}{\|\hat{x}_0\|} + \Omega - 333 - 28a_1 \quad (4.103)$$

Debido a la complejidad de lo manejado hasta ahora, no se pudieron encontrar expresiones mas sencillas para las expresiones (4.95) y (4.103), por lo que no es posible observar o imaginar el área en el que se cumplen.

Observando lo anterior, se puede concluir que la función de Lyapunov (4.72) siempre es positiva definida, y su derivada (4.73) siempre es negativa definida si $a_1 > -28$, b_1 cumple con la desigualdad (4.103) y T_1 con la desigualdad (4.95). Es decir, bajo estas circunstancias el sistema es asintóticamente estable. Se puede notar que el mismo resultado se obtendría con el canal y , por lo que no es necesario repetir el procedimiento anterior. Debido a la complejidad de las expresiones previas no se puede obtener de forma mas concreta la frontera de estabilidad.

Capítulo 5

Resultados

Lo siguiente que se mostrará corresponde a las gráficas obtenidas mediante la simulación en ROS, junto con la interfaz háptica. Se tienen 2 situaciones generales: cuando el objeto a evadir es estático, y cuando es dinámico. Estas situaciones tienen diversas variaciones, dándonos 6 casos diferentes, los cuales se explicarán en su momento. En sí, de forma general se manejan dos retardos: el de ida, que va del maestro al esclavo y manda información de posición, y el de regreso, que va del esclavo a la interfaz, mandando información reflejada en fuerza. Aunque la interfaz puede generar 8 N de fuerza, se asignó un valor de 5 N para tener un margen de seguridad. Se programó un tiempo de espera en el envío de datos entre el maestro y el esclavo y viceversa, de manera que existieran retardos de comunicación.

Objeto estático

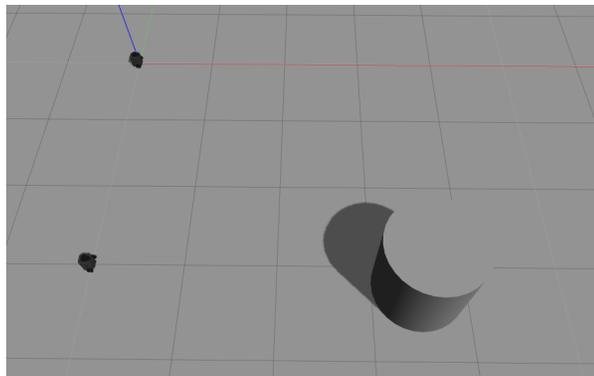


Figura 5.1: Condiciones iniciales con el obstáculo estático

La Figura 5.1 visualiza la forma en que inician siempre los 5 primeros casos que se exponen aquí. El robot maestro se encuentra en el origen $(0, 0)$ del plano de gazebo, mientras que el esclavo esta en la posición $(0, -3)$. Ambos robots se orientan hacia la derecha, por lo que a esa dirección se mueven inicialmente. El obstáculo es un cilindro de 1 metro de diámetro, localizado a 3 m del móvil esclavo; sus coordenadas son $(3.5, -3.2)$. La gráficas del maestro y del esclavo se realizarán con respecto a sus marcos de referencia inerciales particulares, es decir, ambos siempre partirán del $(0, 0)$ en sus gráficas. Por último, haciendo referencia a la Figura 4.2, se utilizó un radio de detección r_d de 1.3 m, mientras que el radio de protección r_p es de 0.4 m

Caso 1

En este primer caso se tiene que tanto los retardos de ida y de regreso son nulos, es decir valen cero.

En la Figura 5.2 se presenta la trayectoria que se buscará realizar siempre. Ambas gráficas están sobrepuestas, la del maestro y la del esclavo, mostrándonos el correcto seguimiento por parte esclavo. La tarea de evasión ocurre en la primera curva. Cuando se llega a la mitad de la curva (cerca de los 2.5 m desde el inicio) termina la tarea de evasión, quedando la tarea de regresar a la línea recta que se intentó hacer desde un principio. La Figura 5.3a da la distancia con peligro de colisión a la que se encuentra el esclavo en determinado momento, y nos dice que alrededor del segundo 12 se alcanza a detectar el objeto. También se puede observar que alrededor del segundo 26 ya no hay peligro de colisión, aunque obviamente aun se está cerca del cilindro. La curva que está entre el segundo 18 y 22 se debe a que en ese momento el movimiento angular tiene un valor mayor a cero, y es mayor que la velocidad lineal. Y en la parte final de la gráfica se muestra que la velocidad lineal vuelve a superar la angular.

Por su parte la Figura 5.3b nos muestra la posición angular a la que se encuentra el punto mas cercano del obstáculo con respecto al esclavo. Del segundo 12 al 18 se puede apreciar que, aunque se detectó el cilindro, hay poca acción de control en la velocidad angular, y por ello la variación del ángulo es baja. Después, del segundo 18 al 22 la acción de control en la misma velocidad ya mencionada aumenta provocando la variación angular que se ve. Y finalmente del segundo 22 al 24 se nota que la velocidad lineal sube, mientras que la angular baja. Algo interesante en estas dos ultimas gráficas es que parecen discretizadas. Esto se debe a que el sensor del que se obtienen los datos mapea cada cierto tiempo y por lo tanto mantiene valores cada cierto tiempo. Esto se repite en los siguientes casos.

La gráfica de la Figura 5.4 muestra la aplicación gradual de la fuerza en el Falcon, junto con sus componentes, mostrando una predominancia en girar a la izquierda.

El aumento de la fuerza es gradual, aunque como ya se dijo, el componente en x predomina mostrándonos que hasta el segundo 20 existe preferencia por girar para evadir el cilindro. Aproximadamente desde el segundo 20 hasta el 25 se empieza a manifestar la componente en y , dándonos a saber que ya se está lo suficientemente lejos del cilindro como para empezar a darle prioridad a la velocidad lineal.

En la Figura 5.5a se puede apreciar la comparación entre la velocidad lineal del maestro y la del esclavo. Se puede notar que existen similitudes a pesar de una diferencia importante la cual es la alta velocidad, relativamente, con la que inicia el esclavo. Es de notarse que la velocidad lineal del maestro pareciera no iniciar en cero, pues debido a problemas en la implementación la interfaz debía mandar una velocidad lineal diferente de cero desde el principio. En la Figura 5.5b se muestran las velocidades angulares del maestro y las del esclavo. Es fácil notar que el esclavo busca seguir al maestro, pues la velocidad del esclavo es parecida a la del maestro. En la zona entre los 12 y 25 segundos, en estas gráficas, se puede notar que la velocidad angular predomina sobre la lineal.

En la Figura 5.6a se puede observar el error en x del esclavo, el cual se mantiene acotado. La Figura 5.6b muestra el error en y del mismo robot, el cual inicia con valores bajos y variaciones más pronunciadas, sin embargo también se mantiene acotado siempre. Al final de la gráfica el error se reduce muchísimo, probablemente por la última línea recta de la trayectoria del robot. Finalmente, la Figura 5.7 tiene el error angular, el cual también cuenta con valores pequeños y con varias oscilaciones un tanto espaciadas. Las zonas correspondientes a los segundos 12 a 25 muestran el comportamiento de los errores en el momento de evasión. En x se notan ligeras disminuciones del error, si embargo en y se formó un pico, probablemente correspondiente al aumento de la acción de control. El error angular, en este aspecto también muestra un pico grande en sentido negativo ocasionado por el giro brusco que se realizó para esquivar el cilindro.

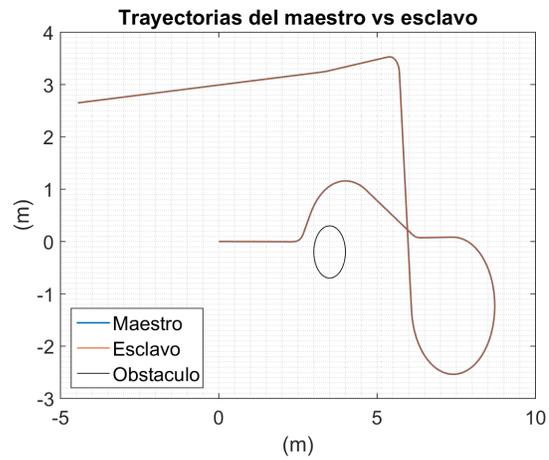


Figura 5.2: Trayectorias realizadas por los robots

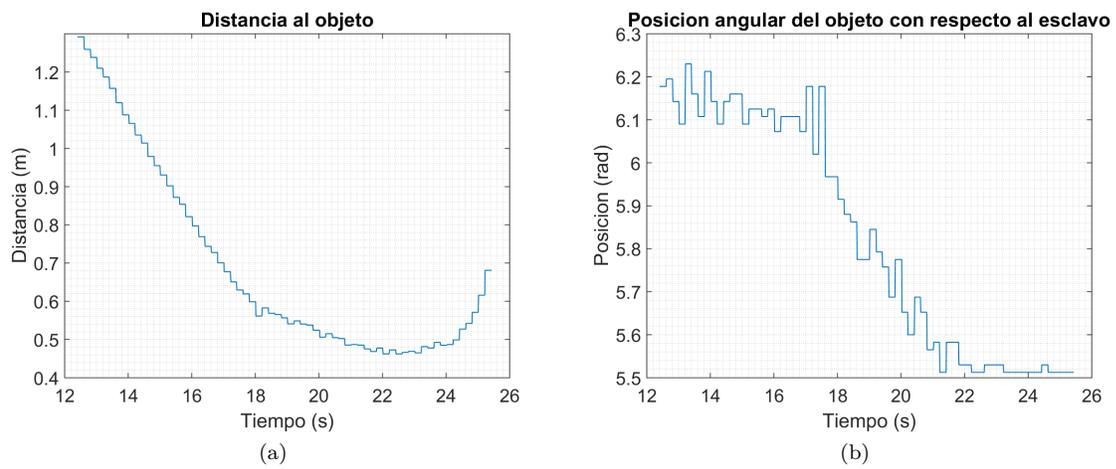


Figura 5.3: Distancias al objeto

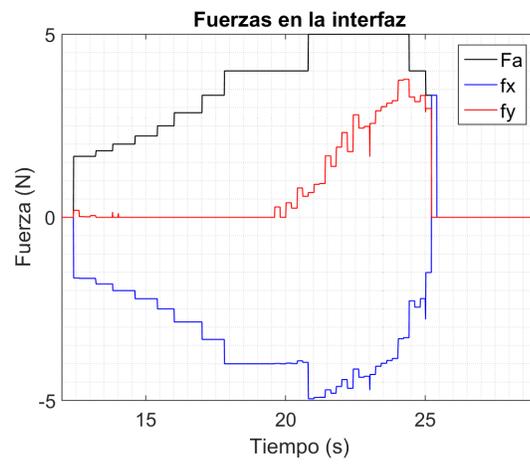


Figura 5.4: Fuerzas recibidas en la interfaz

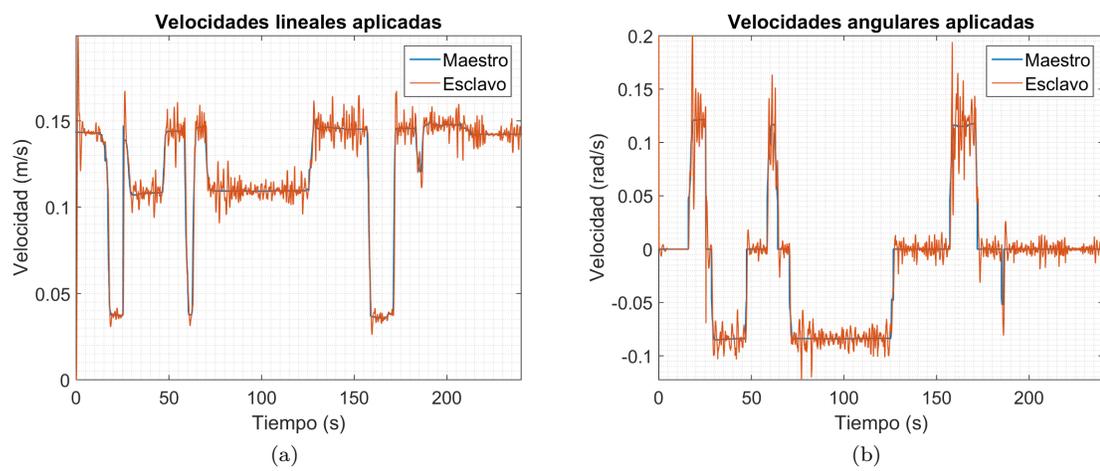


Figura 5.5: Velocidades

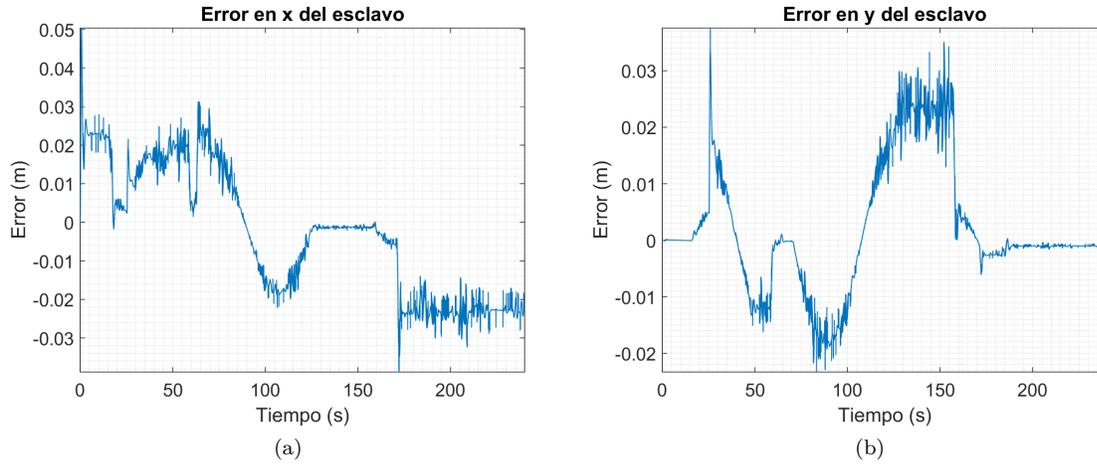


Figura 5.6: Errores en x & y

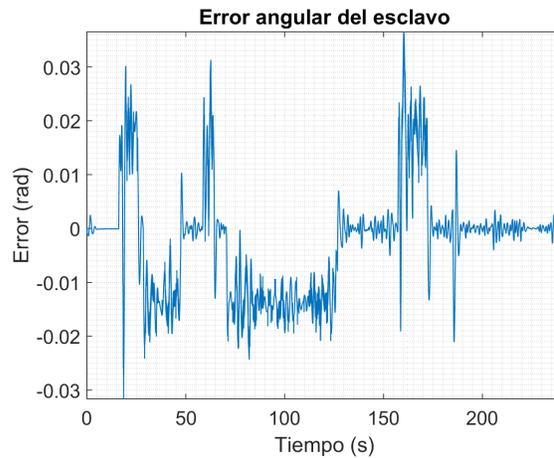


Figura 5.7: Error angular

Caso 2

En este caso inician los retardos con el obstáculo estático. Se asignó al retardo de ida y al de regreso un valor de 0.2 s a cada uno. En la Figura 5.8 se puede observar que se intentó realizar la misma trayectoria ya mostrada, y también que el esclavo la siguió correctamente. También es posible ver la presencia del objeto que funciona como obstáculo. Por su parte en la Figura 5.9a se puede ver como varió la distancia conforme el robot esclavo se acercaba al obstáculo y que cerca del segundo 23 ya no había peligro de colisión. Es interesante notar que esta gráfica es mas pequeña en el eje del tiempo que su antecesora, pues el carrito no se acercó tanto al cilindro como en el caso 1. Aproximadamente en el segundo 15, se dio un aumento en la velocidad

angular en los robots, por eso el esclavo se empezó a acercarse más lentamente al obstáculo. Del segundo 20 se nota un rápido alejamiento del objeto, pues en esta parte la velocidad lineal aumenta, aunque la velocidad angular disminuye. La gráfica de la figura 5.9b confirma lo anterior dicho, pues en los lapsos de tiempo indicados, se notan cambios en la variación de la distancia angular. Se puede saber que la interfaz inicio su acción un poco después de los 12 segundos de haber iniciado el experimento de este caso, tal como lo indica la Figura 5.10. Aquí también se puede observar que las fuerzas iniciaron más fuerte que en la gráfica análoga anterior, pues debido al retardo se necesitó esto. A pesar de este fuerte inicio, la fuerza neta no llegó a los 5 N, pues el elevado valor del inicio favoreció una rápida evasión.

En cuanto a las velocidades de ambos vehículos, podemos conocerlas en las Figuras 5.11a y 5.11b. Sabemos que la evasión se realizó entre los 12 y 24 segundos. Centrándonos en esta pequeña área en las gráficas de las velocidades, podemos observar que al momento de girar, los robots disminuían sus velocidades lineales, mientras que las rotacionales aumentaban. Fijándonos ahora en el esclavo, sus velocidades se muestran semejantes a las del maestro, solo que con ruido producido por Gazebo, además del ruido de los retardos. Los errores en x (Fig. 5.12a) y y (5.12b) tienen cierta similitud con los errores del Caso 1, sin embargo vemos más oscilaciones producidas por el retardo. Lo mismo sucede con el error angular expuesto en la Figura 5.13. En el resto de las gráficas de errores en los siguientes casos, se podrá notar más ruido que en las gráficas del Caso 1; esto es debido a los retardos.

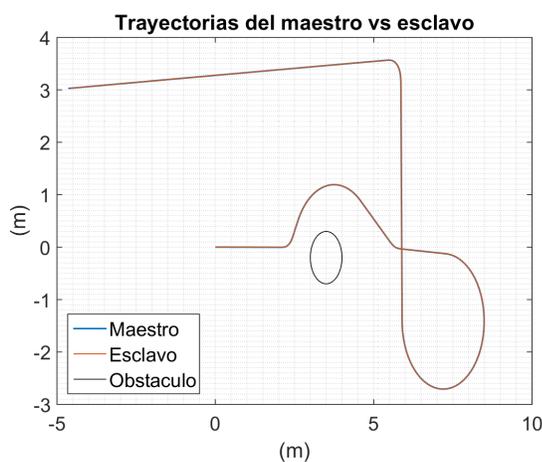


Figura 5.8: Trayectorias realizadas por los robots

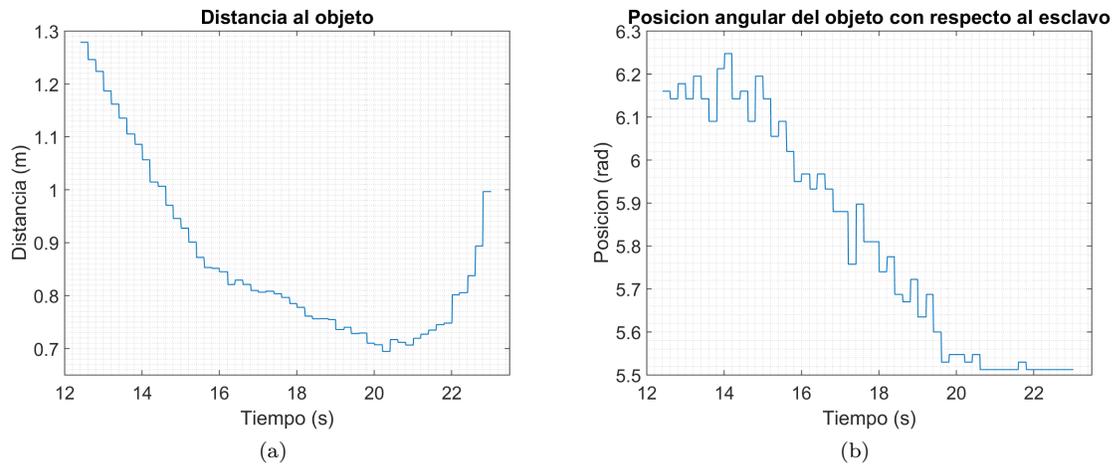


Figura 5.9: Distancias al objeto

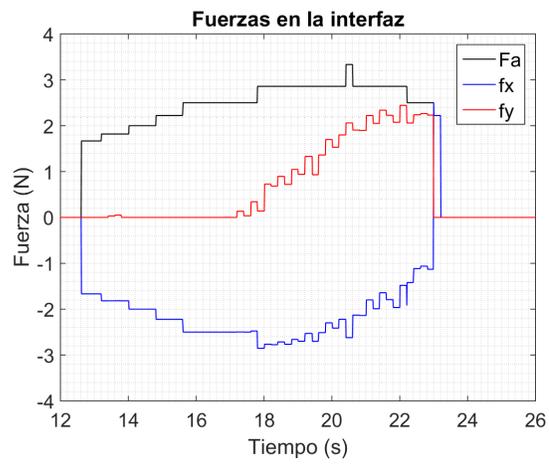


Figura 5.10: Fuerzas recibidas en la interfaz

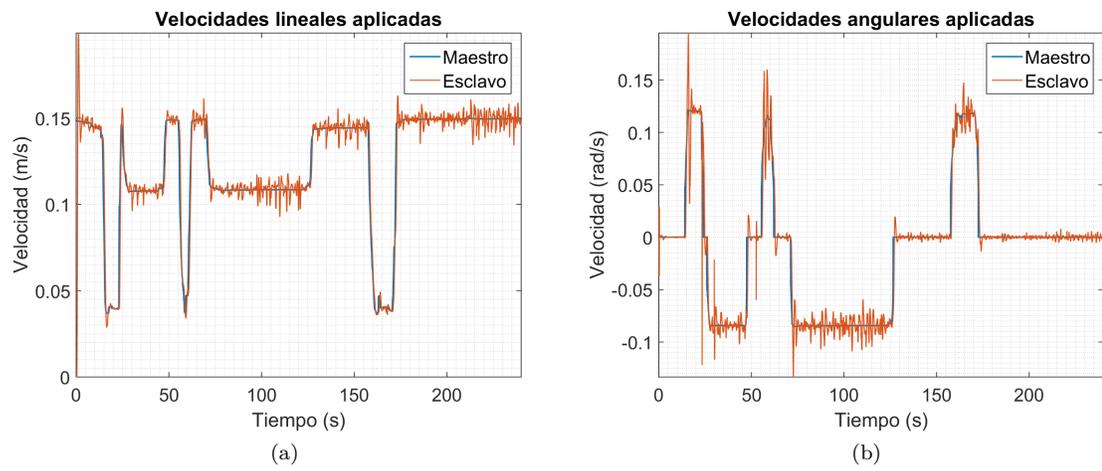


Figura 5.11: Velocidades

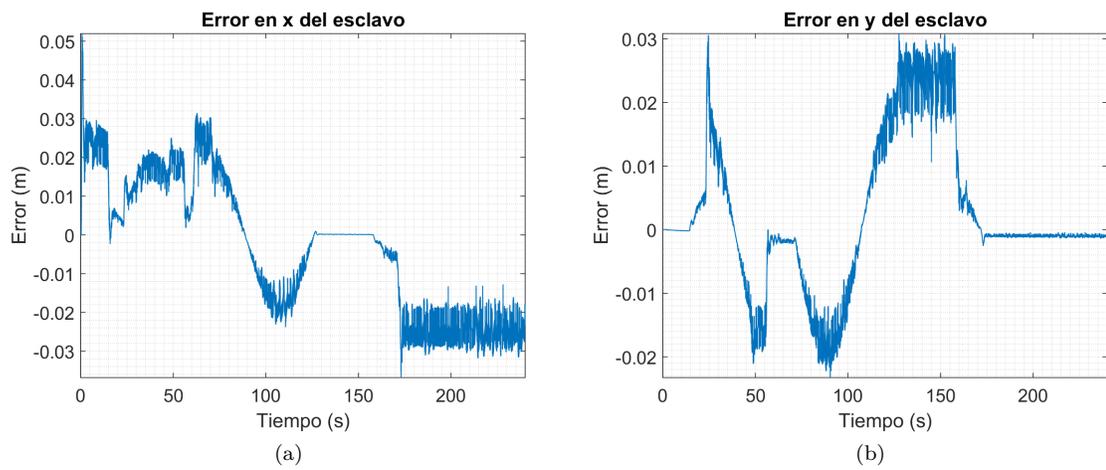


Figura 5.12: Errores en x & y

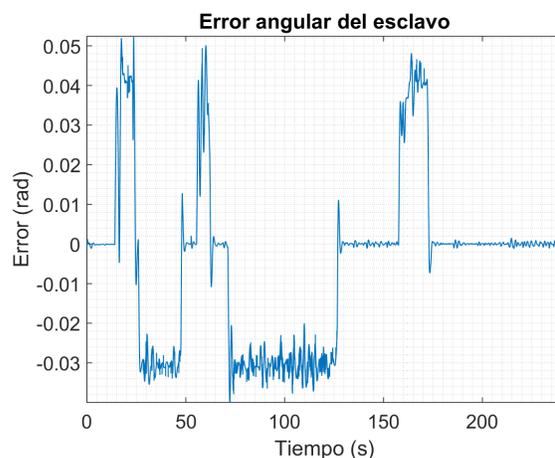


Figura 5.13: Error angular

Caso 3

En este caso los retardos, aunque son de igual valor entre si, aumentaron; se les dio un valor de 0.5 s a cada uno de ellos. Como se puede ver en la Figura 5.14 se buscó realizar la misma trayectoria de siempre, aunque el tramo final se inclinó un poco más. La Figura 5.15a nos da información de que esta vez, se tardó en detectar al obstáculo un poco más debido al retardo aumentado. Sin embargo podemos ver una forma semejante a las gráficas análogas anteriores, aunque se dio un acercamiento un poco más que en el caso dos. Alrededor del segundo 18 se nota una disminución ligera del acercamiento al objeto. Entre los segundos 18 y 22 se nota el efecto de un control que prioriza en esos momentos el girar para evadir. Después del segundo 22 ya se puede notar que ahora la velocidad lineal es la que aumenta, mientras que la angular mengua. La Figura 5.15b añade un poco más de información, pues hasta cerca del segundo 17 el giro ocurrido en ese momento fue el más lento dentro de esa gráfica, pues el cambio angular lo era también. La velocidad angular se muestra relativamente alta entre el segundo 17 y 22, y finalmente se reduce para que el ángulo se mantuviera constante. La Figura 5.16 nos muestra que por algunas ocasiones se alcanzo los 4 N en la interfaz. También se puede ver que existe un aumento en la variación de la fuerza usada en el Falcon debido al retardo. Esta variación es mayor que en el caso dos, pero menor que en el caso uno. Cerca del segundo 19 aparece la componente en y , pues el objeto se está evadiendo y ya no hace falta girar.

Las velocidades lineal y angular del maestro se muestran en las Figuras 5.17a y 5.17b respectivamente. También en estos se puede notar que al girar, los robots disminuyen su velocidad de traslación. Poniendo atención a las velocidades del esclavo, se puede apreciar que muestran un comportamiento parecido a las velocidades del maestro, pues se busca seguirlo. Los errores en x (5.18a) y y (5.18b) se muestran muy parecidos

a los anteriores, aunque en los picos máximos se tienen valores un poco mayores en comparación con las gráficas análogas del caso dos. El error angular (Fig. 5.19) muestra una forma parecida a sus antecesores, solo que ahora existen valores mas grandes, relativamente.

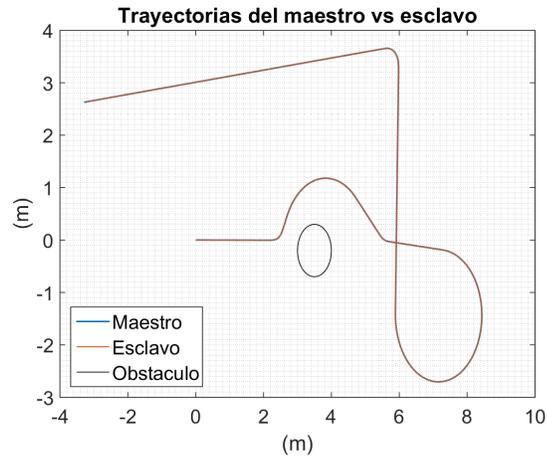


Figura 5.14: Trayectorias realizadas por los robots

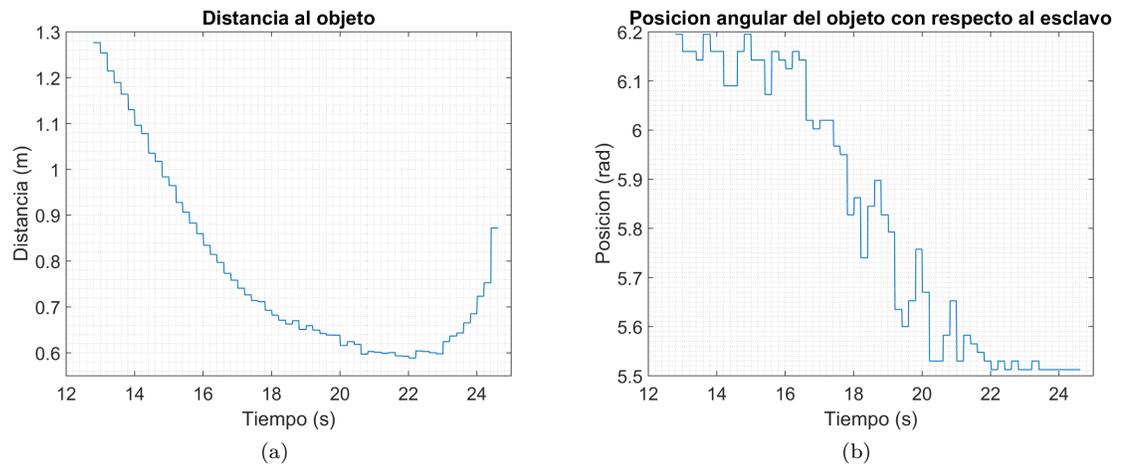


Figura 5.15: Distancias al objeto

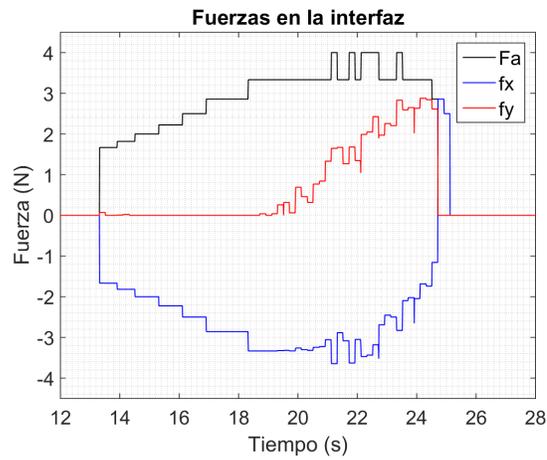


Figura 5.16: Fuerzas recibidas en la interfaz

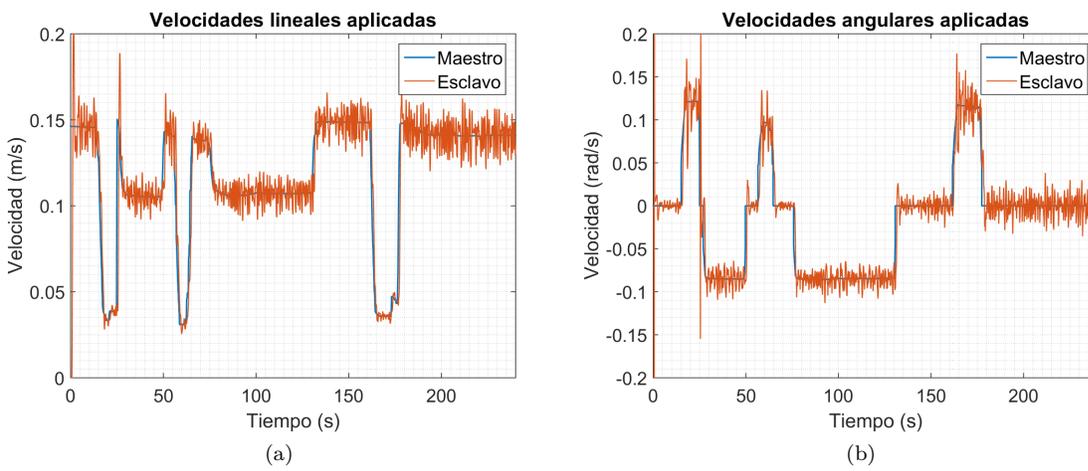


Figura 5.17: Velocidades

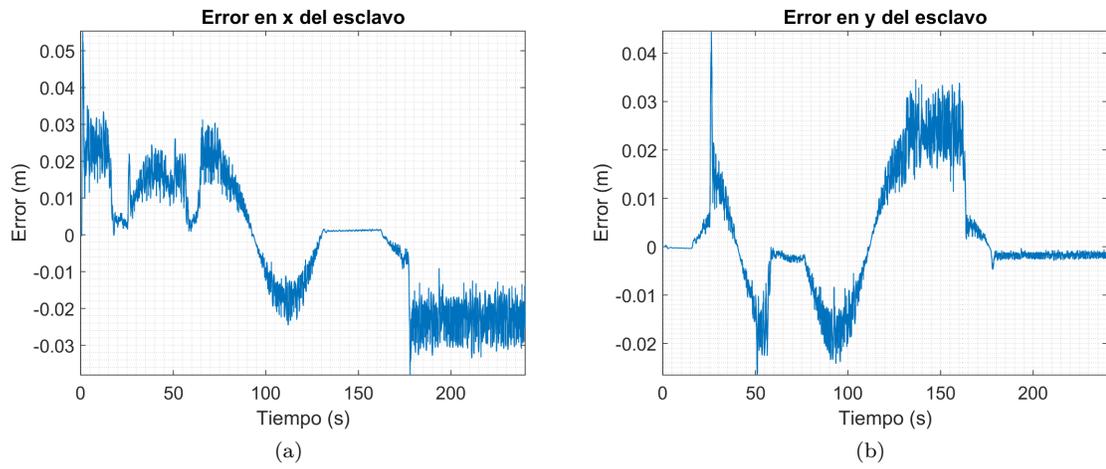


Figura 5.18: Errores en x & y

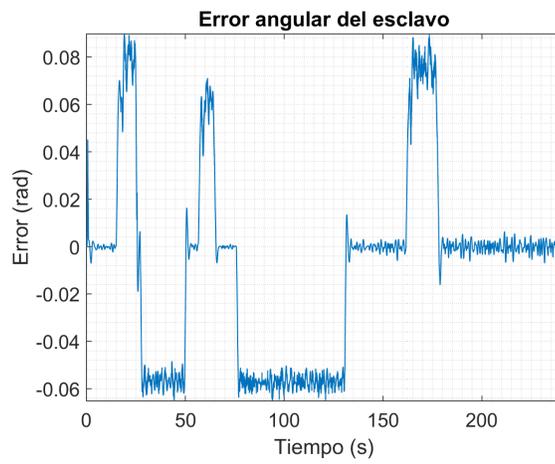


Figura 5.19: Error angular

Caso 4

En este caso se asignó al retardo de ida un valor de 0.5 s, y al de regreso 0.2 s. En la Figura 5.20 está la trayectoria realizada por ambos robots, junto con la figura del obstáculo a evadir. La evasión fue realizada con la primera curva. La distancia a la que se acercaba el esclavo al cilindro mientras había peligro de colisión está en la Figura 5.21a, donde se puede apreciar una forma similar a las que ha tomado en las gráficas anteriores. La Figura 5.21b también nos muestra un comportamiento semejante a lo anterior. Es de destacar que el parecido de estas gráficas con las del caso 3 es mayor, que con las del caso 2. Esto nos puede indicar que el retardo entre el maestro y el esclavo tiene un efecto mas predominante que el retardo de regreso. La Figura 5.22 nos expone que durante ciertos momentos se alcanzaron los 4 N, y que desde el segundo 19 apareció una proyección de fuerzas en y y por lo tanto la evasión ya se estaba logrando.

Las velocidades lineal y angular de ambos robots corresponden a las Figuras 5.23a y 5.23b. Comparando estas gráficas entre si es posible notar que si la velocidad angular aumenta, la lineal disminuye, y viceversa. Esto también esta presente entre los 12 y 24 segundo, que es el tiempo entre el que se realiza la tarea de evasión. Regresar el móvil a la linea recta que se intentaba hacer desde un principio ocurre cerca de los 60 segundos. En las Figuras 5.24a, 5.24b y 5.25 están los errores en x , y y angulares respectivamente, los cuales muestran un comportamiento similar con respecto a lo anterior mostrado, pero el máximo pico en y es menor que en el caso tres.

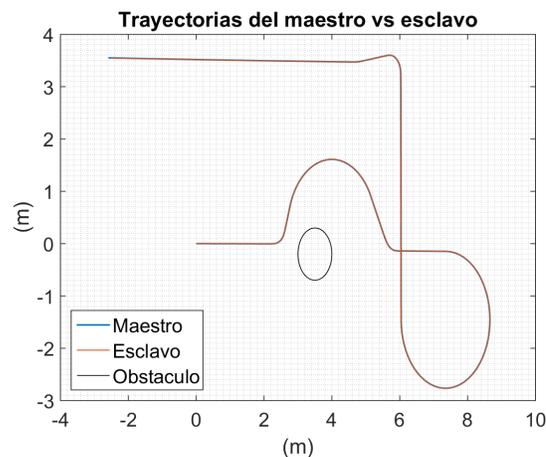


Figura 5.20: Trayectorias realizadas por los robots

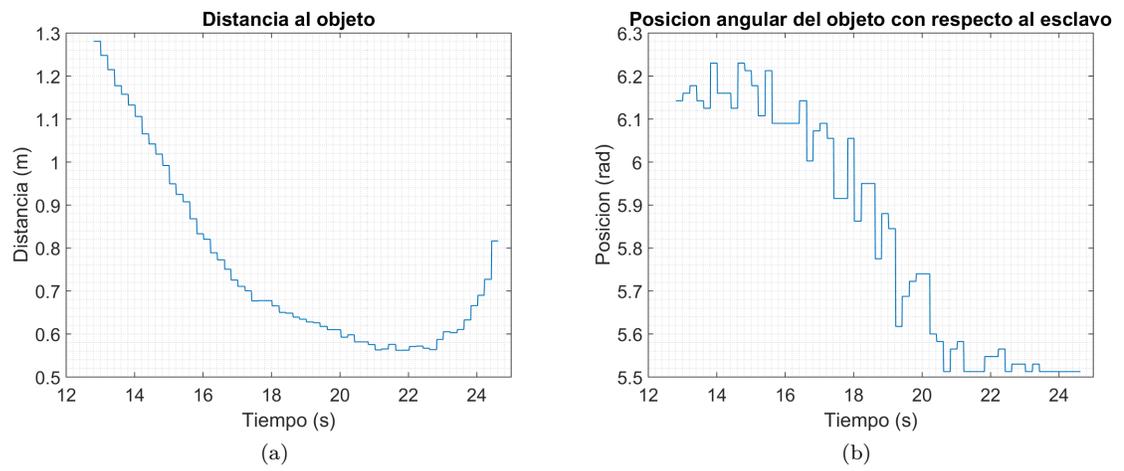


Figura 5.21: Distancias al objeto

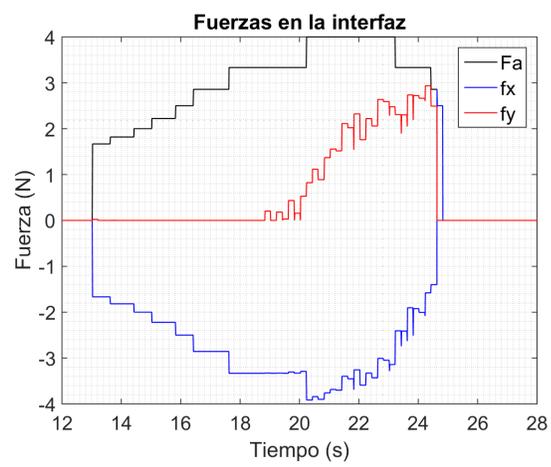


Figura 5.22: Fuerzas recibidas en la interfaz

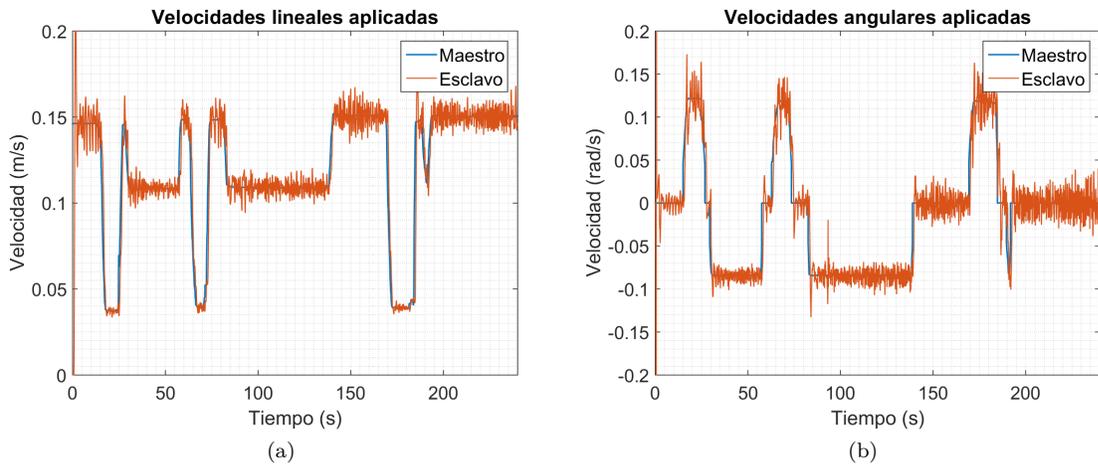


Figura 5.23: Velocidades

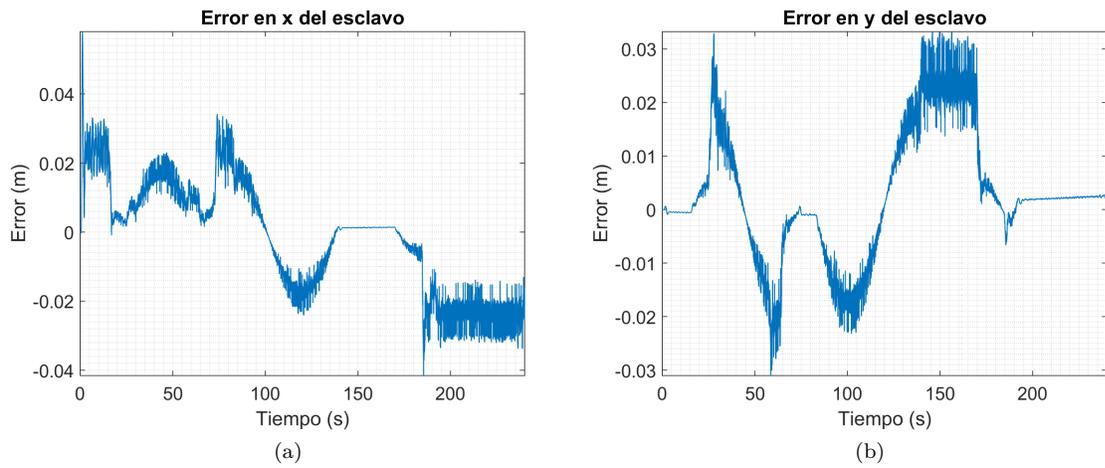


Figura 5.24: Errores en x & y

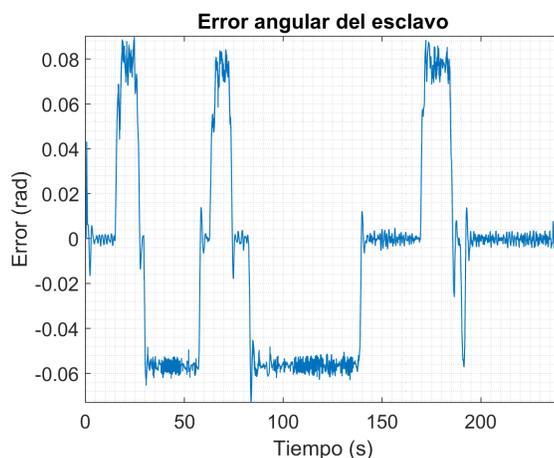


Figura 5.25: Error angular

Caso 5

En este caso se invirtieron el valor de los retardos, para el de ida se adoptó un valor de 0.2 s y para el de regreso un valor de 0.5 s. En la Figura 5.26 se expone la trayectoria de ambos robots, siendo que también en este caso hay un buen seguimiento del maestro por parte del esclavo. El gran tamaño de la curva donde se esquia al obstáculo se debe a que el algoritmo de evasión funciona hasta que se evita el peligro de colisión completamente. La Figura 5.27a nos muestra una gráfica bastante parecida a la del caso dos, pues desde cerca del segundo 17 hasta el segundo 23 hay una acercamiento mas lento que el ocurrido antes del primer tiempo mencionado. El último tramo de la gráfica corresponde a un alejamiento rápido del cilindro por parte del esclavo. La Figura 5.27b expone cambios variables en la posición angular que tienen relación con la Figura 5.27a, pues entre los segundos 17 y 23 se dan cambios rápidos en el giro del esclavo. En cuanto a las fuerzas mostradas en la Figura 5.28, se observa que se alcanzó un valor de 4 N durante mas tiempo que en los casos anteriores.

En las Figuras 5.29a y 5.29b están las velocidades lineal y angular tanto del maestro como la del esclavo. Los comportamientos son similares a los de los casos anteriores, lo cual es lógico pues se buscó realizar siempre las mismas trayectorias. Lo mismo sucede con los errores (Fig. 5.30a, 5.30b y 5.31), pues están acotados y poseen valores pequeños.

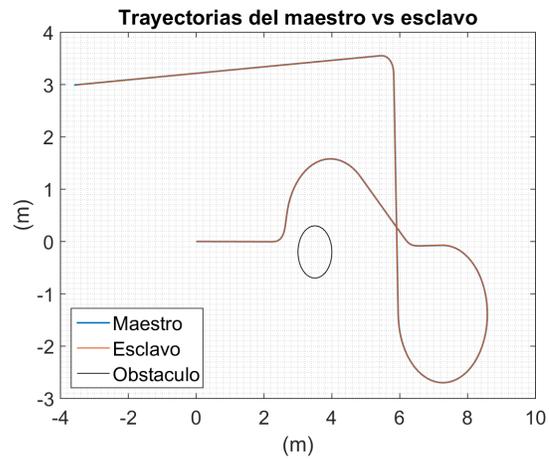


Figura 5.26: Trayectorias realizadas por los robots

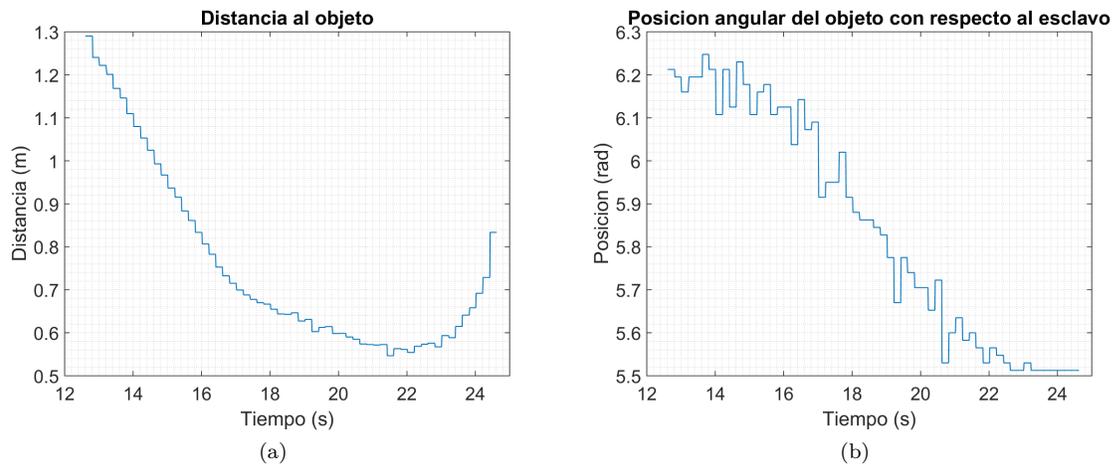


Figura 5.27: Distancias al objeto

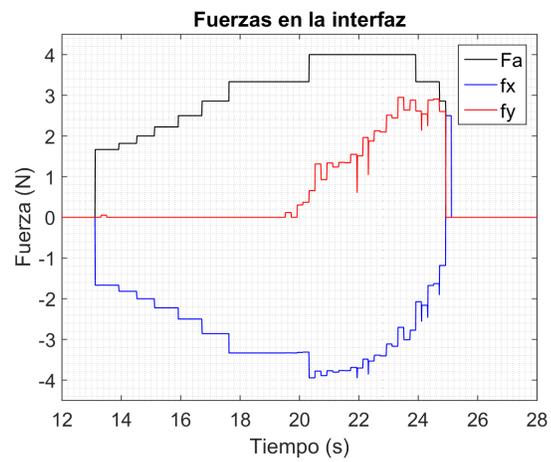


Figura 5.28: Fuerzas recibidas en la interfaz

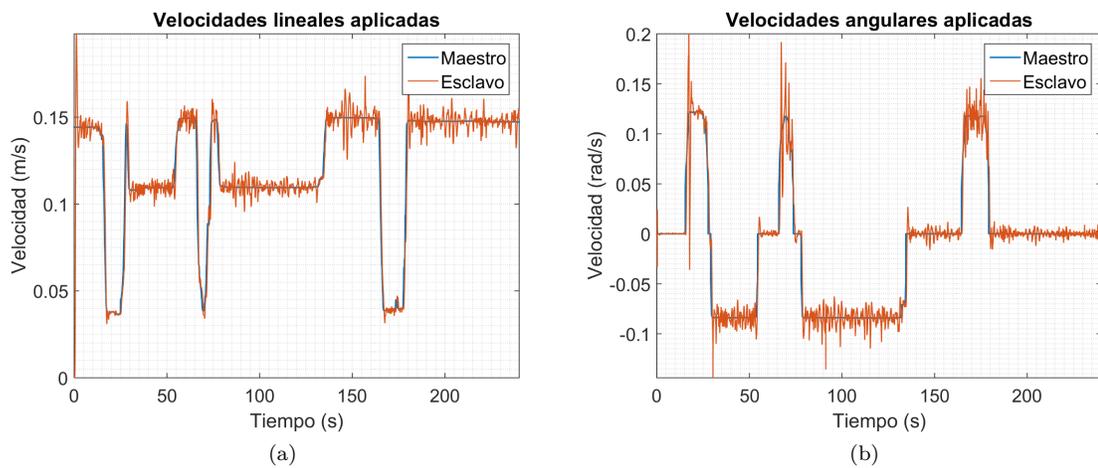


Figura 5.29: Velocidades

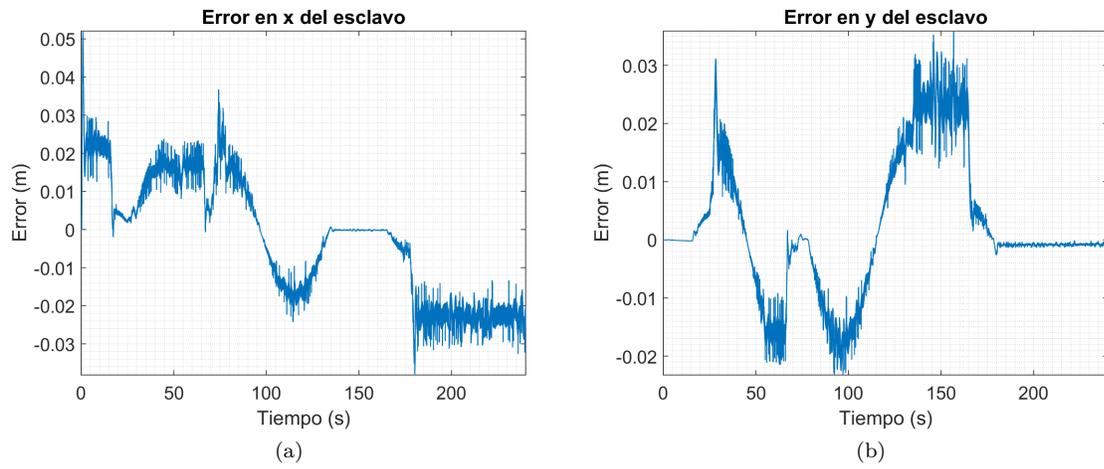


Figura 5.30: Errores en x & y

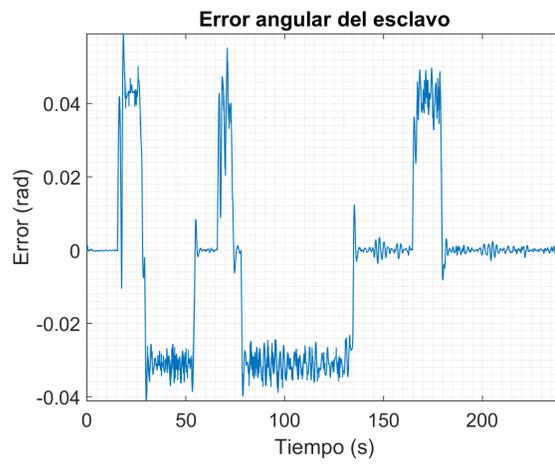


Figura 5.31: Error angular

Objeto dinámico

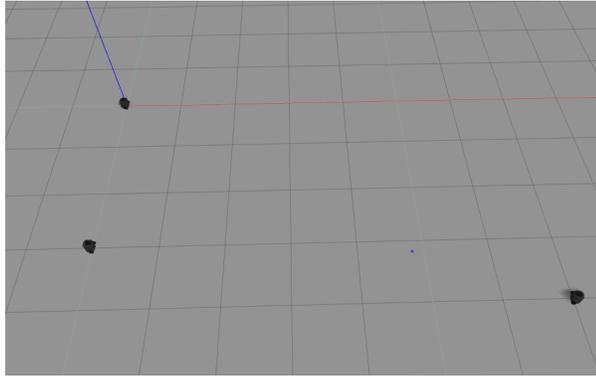


Figura 5.32: Condiciones iniciales con el obstáculo dinámico

Las condiciones iniciales para las simulaciones con un obstáculo dinámico están en la Figura 5.32. El móvil maestro inicia en las coordenadas de origen $(0, 0)$ del plano general del Gazebo. Y el esclavo esta en las coordenadas $(0, -3)$ del mismo plano, sin embargo las gráficas de ambos robots siempre inician en $(0, 0)$ y comienzan su movimiento en línea recta hacia la derecha. El obstáculo que se eligió aquí consiste en un tercer móvil con coordenadas iniciales de $(7, -4,)$ que se mueve hacia $(0, 1)$, ambas pertenecientes al plano de Gazebo, y su velocidad únicamente es lineal con un valor de 0.1 m/s. Sin intervención, la colisión entre el turtle esclavo y el obstáculo ocurriría cerca de $(4.23, -3.17)$ del plano de Gazebo y cerca de $(4.13, -0.17)$ en el plano del esclavo. Para terminar, y haciendo referencia a la Figura 4.2, se utilizó un radio de detección r_d de 2.1 m, mientras que el radio de protección r_p fue de 0.4 m.

Caso 6

Este es el primero caso con un obstáculo en movimiento, y como tal, los retardos aquí tienen un valor de cero.

En la Figura 5.33 están las trayectorias realizadas por los tres robots. Los desplazamientos del maestro y esclavo son similares a las gráficas de los casos anteriores; lo único nuevo es el desplazamiento del tercer turtle. La curva causada por la evasión es mas pequeña, pues el obstáculo se mueve y por eso se puede esquivar mas rápido. En la Figura 5.34a se puede observar que tan rápido se va acercando el esclavo al tercer robot. La gráfica nos muestra un acercamiento constante, con un ligero incremento de velocidad entre el segundo 26 y 27. De la Figura 5.34b se puede obtener un poco mas de información, pues desde que inicialmente se detecta al obstáculo hasta el segundo 26, la posición angular del objeto con relación al esclavo cambia de forma ligera.

Después de ese tiempo, se da un cambio mas rápido ocasionado por el giro del robot esclavo, para luego finalmente alejarse con una velocidad lineal predominante. La Figura 5.35 muestra un comportamiento bastante diferente a lo anterior mostrado, pues se puede notar el cambio gradual de las fuerzas hasta el momento en el que ya no hay peligro de colisión. Algo que se debe recalcar es el hecho de que la mayor parte del tiempo la componente en y de la fuerza es cero. Esto se debe a que los 2 robots se acercan entre si de forma rápida, por lo que se requiere un control en velocidad angular predominante. Esto también se ve reflejado en el valor máximo de la fuerza general de la interfaz, pues supera el valor nominal asignado por 1 N.

Las velocidades del maestro y del esclavo se presentan en las Figuras 5.36a y 5.36b. Entre los segundos 21 y 31, ocurre la evasión del obstáculo, y con ello se puede ver en las gráficas que la velocidad angular aumenta y la lineal disminuye, algo que ya había sucedido con anterioridad. Los errores en x (Fig. 5.37a) y y (5.37b) muestran resultados con oscilaciones bastantes espaciadas, lo cual se debe a la ausencia de retardos. En el Caso uno las gráficas también muestran oscilaciones espaciadas en comparación con los casos 2 al 5, sin embargo en el Caso seis se nota mucho mas. La Figura 5.38 muestra el error angular, el cual también tiene ruido.

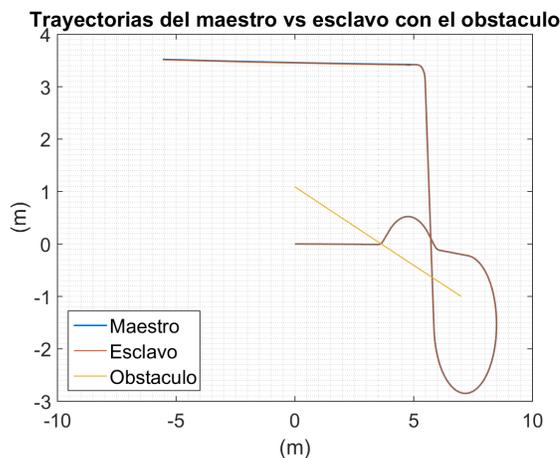


Figura 5.33: Trayectorias realizadas por los robots

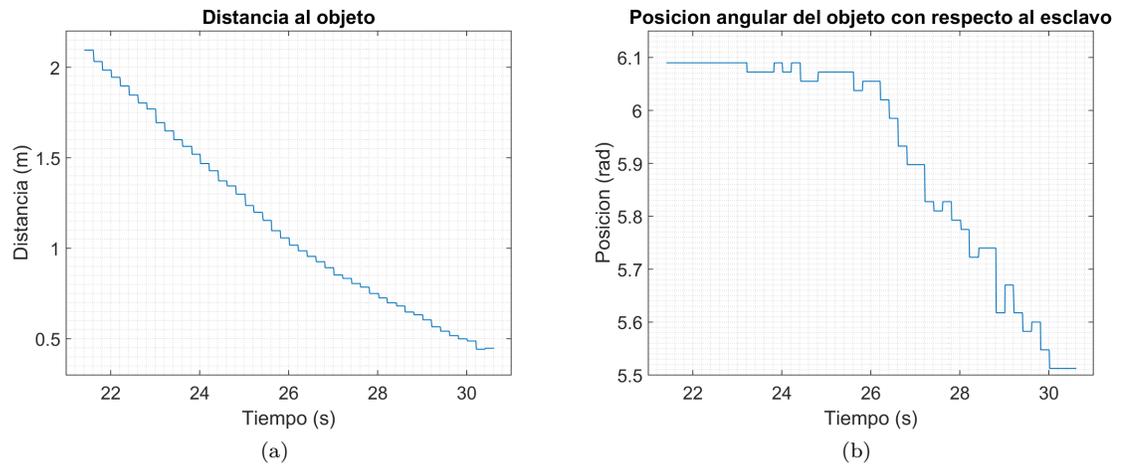


Figura 5.34: Distancias al objeto

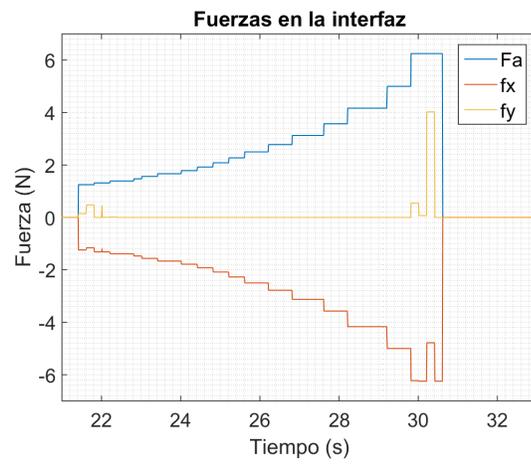


Figura 5.35: Fuerzas recibidas en la interfaz

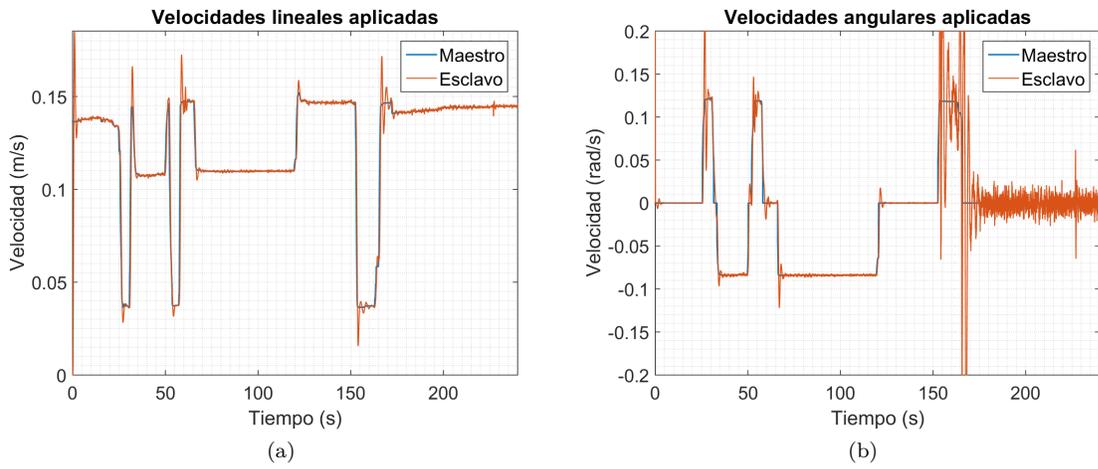


Figura 5.36: Velocidades

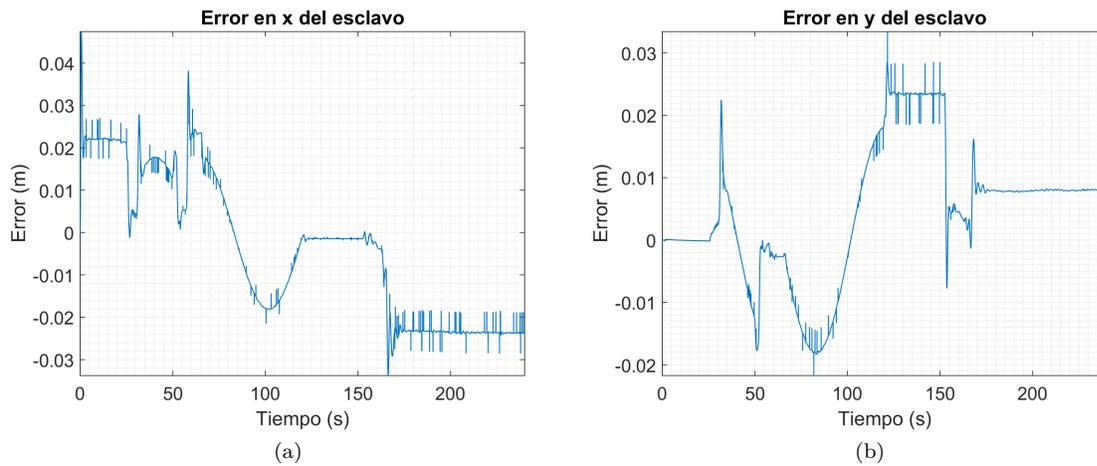


Figura 5.37: Errores en x & y

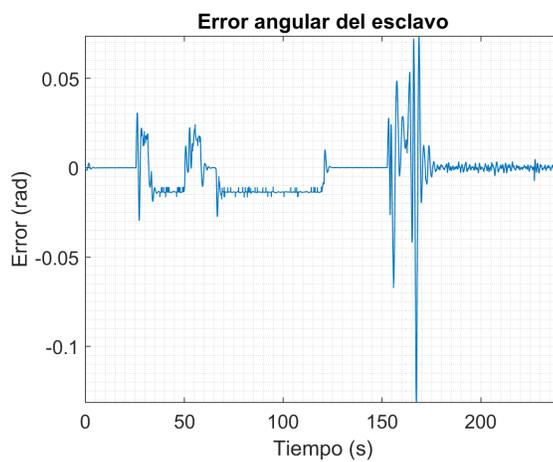


Figura 5.38: Error angular

Caso 7

En este caso los retardos, tanto de ida como de regreso tuvieron valores de 0.2 s. Este es el último caso, pues al aumentar el valor de los retardos aun mas, el sistema completo se desestabilizaba.

La Figura 5.39 muestra la trayectoria descrita por los tres robots, siendo bueno el seguimiento del esclavo. La Figura 5.40a contiene la distancia a la que el esclavo se va acercando al obstáculo. Se nota semejante a la del Caso seis. Por su parte la Figura 5.40b muestra la posición angular del objeto con relación al esclavo; en todos los casos se inicia la detección del cilindro cerca de los 6.1 radianes, y se termina cerca de los 5.5 radianes. Del segundo 21 hasta 26 se notan cambios pequeños, y después un aumento de velocidad angular del esclavo hasta que ya no hay peligro de colisión. La fuerza total manifestada en la interfaz supera los 6 N esta vez, lo que da a entender que por poco colisionan los robots.

Las velocidades lineales (Fig. 5.42a) del maestro y del esclavo manifiestan un comportamiento similar a lo anterior ya visto. Lo mismo sucede con las velocidades angulares (Fig. 5.42b). Lo que esta entre los segundos 21 y 31 muestra el proceso de evasión del obstáculo, exponiendo que la velocidad lineal disminuía mientras la angular aumentaba. En cuanto a los errores en x (Fig. 5.43a) y y (Fig. 5.43b), estos se muestran relativamente mas uniformes, pero con bastantes oscilaciones en algunas zonas; sin embargo el error angular (Fig. 5.44) pareciera mostrar menos oscilaciones que en el Caso seis.

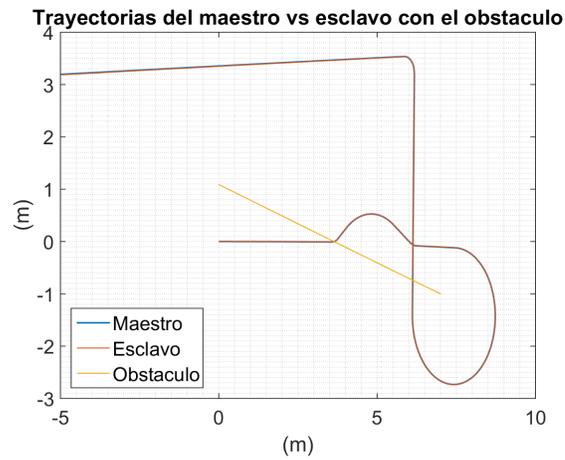


Figura 5.39: Trayectorias realizadas por los robots

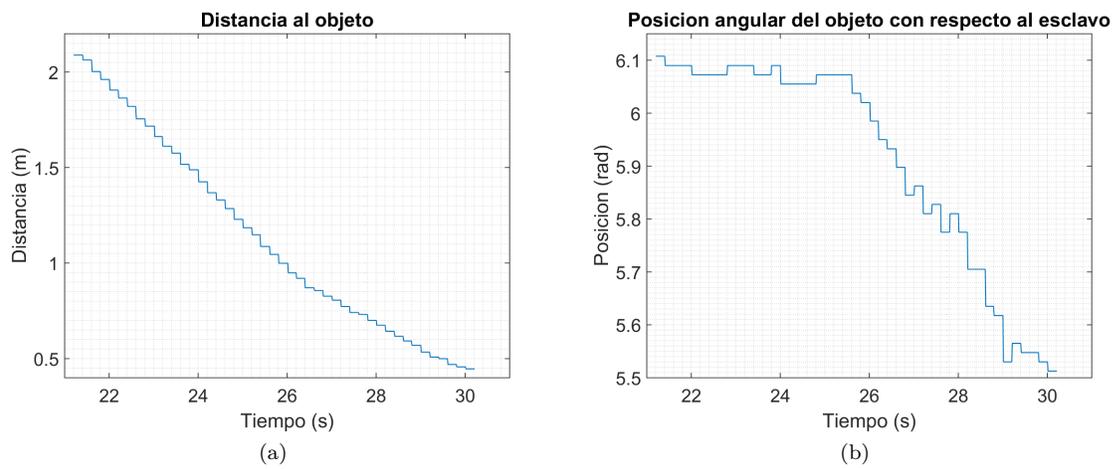


Figura 5.40: Distancias al objeto

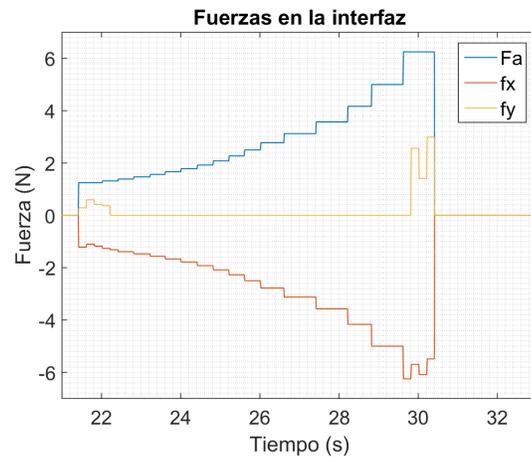


Figura 5.41: Fuerzas recibidas en la interfaz

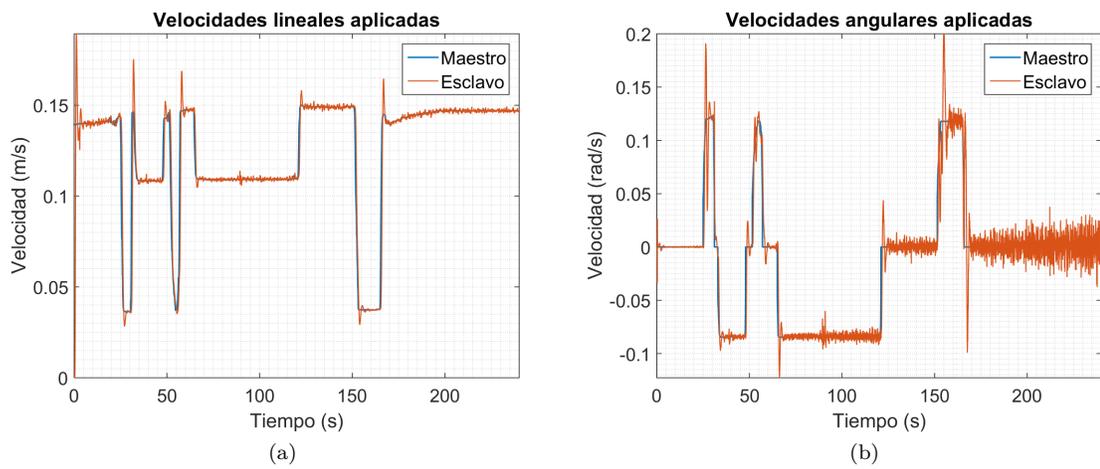


Figura 5.42: Velocidades

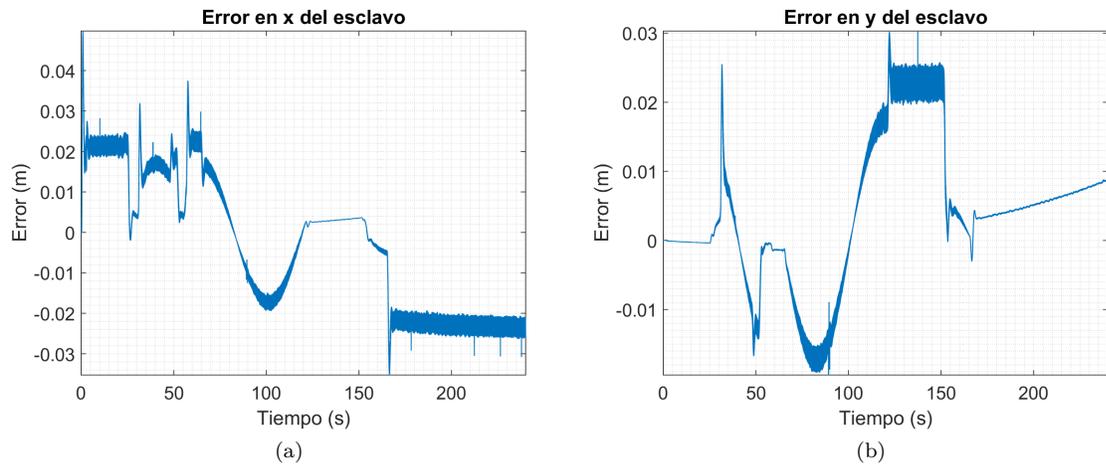


Figura 5.43: Errores en x & y

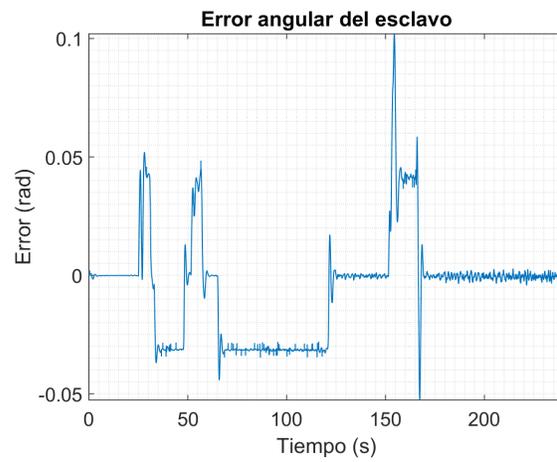


Figura 5.44: Error angular

Capítulo 6

Conclusiones y trabajo a futuro

6.1. Conclusiones

Se presenta a continuación una lista con algunas conclusiones importantes obtenidas mediante el desarrollo de este trabajo:

- La sencillez del IMC depende de la complejidad del modelado de la planta. En el caso de este trabajo, únicamente se utilizó el modelo cinemático, ignorando diversas dinámicas. Esto como resultado nos dio un control sencillo de calcular y aplicar, pero restringido a bajas velocidades donde las dinámicas ignoradas tengan presencia muy baja.
- El IMC presenta por sí mismo cierta robustez a los retardos, aunque estos deben ser iguales o menores a 0.2 segundos.
- La retroalimentación de fuerzas utilizada inicialmente es un apoyo para la evasión, sin embargo, con la sintonización adecuada el GOACM puede llegar a tomar por completo el control de la interfaz para garantizar que se esquite el obstáculo.
- El GOACM ofrece sencillez en su aplicación en comparación a otros métodos de evasión, como por los basados en campos vectoriales, en especial en nuestro caso, donde se maneja una retroalimentación de fuerzas.
- El esquema utilizado con todos sus elementos, mostró cierta robustez a los retardos, siendo el límite máximo 0.5 segundos tanto en retardo de ida como en el de regreso, y con evasión de un obstáculo estático. Para obstáculos dinámicos, los retardos máximos soportados fueron de 0.2 segundos en ambas direcciones.

6.2. Trabajo futuro

1. La experimentación del IMC con el GOACM en físico, pues debido a la pandemia del 2020 no se pudo realizar esto.
2. La implementación de retos variables en el tiempo para probar la robustez del sistema en este aspecto.
3. Implementación de un control que maneje el modelo dinámico de los robots móviles, lo cual permitirá desplazamientos a velocidades mas altas.
4. Experimentación a través del internet para observar el comportamiento del sistema de teleoperación con los retardos y el ruido ocasionados en este medio.
5. Añadir un control que de cierta robustez ante perturbaciones, pues el sistema actual es muy vulnerable a estos.

Bibliografía

- [1] Reyes, F. : *Robótica, Control de robots manipuladores*. Alfaomega, 2011.
- [2] Nilson N. J.: *Shakey the robot*. SRI Internationa, 1984.
- [3] Li, W., Li, Z., Liu, Y., Ding, L., Wang, J., Gao, H., y Deng, Z.: *Semi-autonomous bilateral teleoperation of six-wheeled mobile robot on soft terrains*. En *Robotics and Computer-Integrated Manufacturing*. volumen 133, 2019.
- [4] Aboutalebian, B., Talebi, H. A., Etedali, S. A. y Suratgar, A. A. : *Adaptive Control of Teleoperation System Based on Nonlinear Disturbance Observer*. En *European Journal of Control*, volumen 53, páginas 109–116, 2019.
- [5] Arriaga, S. : *SEMINARIO DE REDES DE COMPUTADORES: Safety for physical human-robot interaction*. En *Departamento de Electrónica, Universidad Técnica Federico Santa María*, 2014.
- [6] Sira, H., López, U. A. y Velasco, M. : *2010 7th International Conference on Electrical Engineering Computing Science and Automatic Control: Trajectory-tracking control of an input delayed omnidirectional mobile robot*. En *European Journal of Control*, páginas 470–475, 2010.
- [7] Denasi, A., Nijmeijer, H. y Kotic, D. : *Time Delay Compensation in Bilateral Teleoperations Using IMPACT*. En *IEEE Transactions on Control Systems Technology*, volumen 21, páginas 704–715, 2013.
- [8] Vences, J. : *Compensación de retardos para teleoperación bilateral de robots móviles diferenciales*. CINVESTAV, 2018.
- [9] Fernández, A., Barreiro, A. y Raimúndez, C. : *Teleoperación háptica de un puente grúa*. En *XXVII Jornadas de Automática*, páginas 136–143, 2006.
- [10] Legius, M. : *Control and Integration of a Flexible Transportation System Based on Autonomous Mobile Robots*. En *Eindhoven University of Technology*, 2015.
- [11] Caiza, G., Garcia, C., Garcia, M. y Naranjo, J. : *Flexible robotic teleoperation architecture for intelligent oil fields*. En *Heliyon*, volumen 6, 2020.
- [12] Morari, M. y Evangelhos, Z.: *Robust process control*. Prentice-Hall, 1989.

- [13] Lentin, J. : *Robot Operating System (ROS) for Absolute beginners*. Apress, 2018.
- [14] Xian, Yang., Hua, Changchun. y Yan, Jin. : *An Exact Stability Condition for Bilateral Teleoperation With Delayed Communication Channel*. En *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, volumen 46, páginas 434-439, 2016.
- [15] Khalil, H. : *Nonlinear Systems*. Prentice Hall, 1996.

Apéndice A

Instalación de la interfaz

A continuación se presentan los pasos para la instalación de la interfaz Novint Falcon en Ubuntu 16.4.

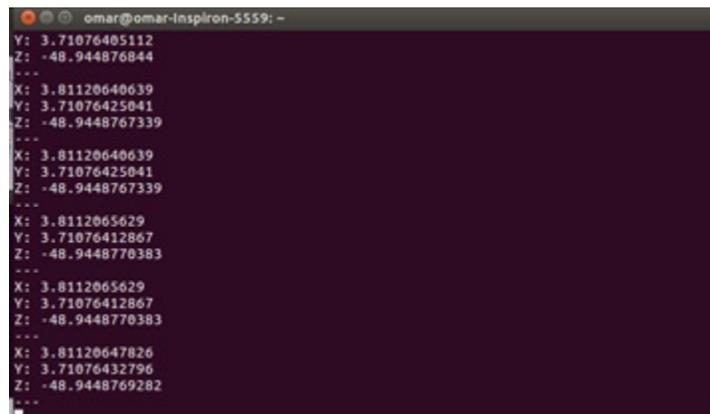
1. Primeramente se debe entrar a cmake.org y descargar el archivo .zip de cmake
2. Se debe seguir las instrucciones del archivo llamado READ ME que esta en el .zip recién descargado
3. Se instala libusb 1.0. Para esto se escribe `sudo apt-get install libusb-dev`
4. Se crea una carpeta llamada git con el siguiente comando: `mkdir ~/git`
5. Se escribe `cd git`
6. Se clona el repositorio
7. Se escribe `cd libnifalcon`
8. A continuación se escribe `mkdir build`
9. Después se escribe `cd build`
10. Se ejecuta `cmake -G "Unix Makefiles" ..`
11. Se escribe `sudo make`
12. Se ejecuta `sudo make install`
13. Se añade `sudo ldconfig`
14. Se conecta el Falcon y se prueba si se instalo con `findfalcons`. Debe aparecer un mensaje en el que se indica que se detecta la interfaz
15. Se instala ROS-falcon, para la conexión entre la interfaz y el sistema ROS. Para esto se debe clonar el paquete de ros-falcon corregido en `catkin_ws/src`. Este paquete se obtiene de la sitio web https://github.com/jcorredorc/ros_falcon

16. Se escribe `cd ..` para ir hacia `catkin_ws` y ahí se ejecuta `catkin_make`
17. Ahora, en otra terminal se escribe `roscd ros-falcon`
18. Se ejecuta `sudo cp udev_rules/99-udev-novint.rules /etc/udev/rules.d`
19. Se debe desconectar y conectar el falcon, y después escribir `sudo idconfig`
20. Se corre `roscore`
21. Se ejecuta `roslaunch ros-falcon falcon_node`. Después de poner esto, y si aparece un problema, solo basta con desconectar y conectar el usb de la interfaz y se debe volver a escribir el comando. Con esto ya se debería poder utilizar el falcon. Como comprobación se puede poner `rostopic echo /falconPos` y deben aparecer las coordenadas del falcon, como se muestra en la siguiente figura.



```
omar@omar-Inspiron-5559: ~$ rostopic echo /falconPos
```

Figura A.1: Comando para comprobar la correcta instalación



```
Y: 3.71076405112
Z: -48.944876844
---
X: 3.81120640639
Y: 3.71076425041
Z: -48.9448767339
---
X: 3.81120640639
Y: 3.71076425041
Z: -48.9448767339
---
X: 3.8112065629
Y: 3.71076412867
Z: -48.9448770383
---
X: 3.8112065629
Y: 3.71076412867
Z: -48.9448770383
---
X: 3.81120647826
Y: 3.71076432796
Z: -48.9448769282
---
```

Figura A.2: Respuesta obtenida cuando la instalación se realizó con éxito