



INSTITUTO POLITÉCNICO NACIONAL

CENTRO DE INVESTIGACIÓN EN
COMPUTACIÓN

Control visual difuso de un sistema no-lineal

Tesis que presenta el
Ing. César Aarón Pérez Olvera

para obtener el grado de
Maestro en Ciencias de la Computación

Directores de tesis:

Dr. Marco Antonio Moreno Armendáriz
Dra. Elsa Rubio Espino

México D.F., 7 de diciembre de 2009.



0272
SIP-14

INSTITUTO POLITECNICO NACIONAL
SECRETARIA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D.F. siendo las 12:00 horas del día 11 del mes de Junio de 2009 se reunieron los miembros de la Comisión Revisora de Tesis designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro de Investigación en Computación

para examinar la tesis de grado titulada:

“CONTROL VISUAL DIFUSO DE UN SISTEMA NO-LINEAL”

PÉREZ

(Apellido Paterno)

OLVERA

(Apellido Materno)

CÉSAR AARÓN

(Nombre)

Con registro:

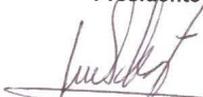
B	0	6	1	0	4	2
---	---	---	---	---	---	---

aspirante al grado de: **MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN**

Después de intercambiar opiniones los miembros de la Comisión manifestaron **SU APROBACIÓN DE LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Presidente


Dr. Luis Pastor Sánchez Fernández

Secretario


Dr. Juan Humberto Sossa Azuela

**Primer vocal
(Director)**


Dr. Marco Antonio Moreno Armendáriz

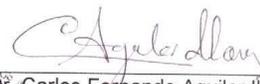
**Segundo vocal
(Directora)**


Dra. Elsa Rubio Espino

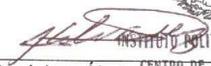
Tercer vocal


Dr. Wen Yu Liu

Suplente


Dr. Carlos Fernando Aguilar Báñez

EL PRESIDENTE DEL COLEGIO


Dr. Jaime Álvarez Callegos
INSTITUTO POLITECNICO NACIONAL
CENTRO DE INVESTIGACION
EN COMPUTACION



INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México, D.F. el día 12 del mes de Octubre del año 2009, el que suscribe: Pérez Olvera César Aarón, alumno del Programa de Maestría en Ciencias de la Computación, con número de registro B061042, adscrito al Centro de Investigación en Computación, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección del Dr. Marco Antonio Moreno Armendáriz y Dra. Elsa Rubio Espino y cede los derechos del trabajo titulado "Control visual difuso de un sistema no-lineal", al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección cperezob06@ipn.mx . Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

César Aarón Pérez Olvera

Nombre y firma

Resumen

La presente tesis trata del diseño, construcción e implementación del sistema Plano-Esfera (PE). El sistema PE es un sistema no-lineal multivariable. Está compuesto por una esfera de acero (balín) y un plano de vidrio, el plano puede rotar alrededor de los ejes perpendiculares X y Y, de esta forma el balín se mueve bajo la influencia de la fuerza de gravedad sobre la superficie del plano.

El objetivo del trabajo es mover a la esfera en equilibrio de un punto a otro del plano. La posición de la esfera es controlada por la inclinación del plano. Para ello, se diseñó un sistema de control basado en lógica difusa con retroalimentación visual. Nos permitimos dividir el sistema de control en dos subsistemas: el subsistema de tratamiento digital de imágenes (TDI) y el subsistema de control difuso tipo Mamdani.

Se logró implementar el sistema de control basado en lógica difusa en tiempo real. La señal de retroalimentación del sistema es adquirida con una cámara digital capaz de capturar 20 imágenes por segundo. Se optimizó el procesamiento de imágenes, ya que la imagen se procesa conforme es adquirida por el sistema.

El sistema diseñado y construido, es un excelente prototipo para poner a prueba diferentes estrategias de control, por lo que será una valiosa herramienta para los investigadores del área.

Abstract

This thesis deals with the design, construction and implementation of a Ball-Plate (BP) system. The BP is a non-linear multivariate system. It consists of a steel ball and a glass plate, the plate can rotate around the perpendicular axes X and Y. In this way, the ball move on the influence of gravity on the plane's surface.

The goal is to move the ball from an initial position toward another stabilized position in the plane. The position of the ball is controlled by the plane's tilt. Therefore was designed a system based on fuzzy logic control with visual feedback. The control system is composed for two subsystems: the subsystem of digital image processing and the subsystem of fuzzy control.

The control system was implemented in real-time. The system's feedback signal was acquired from a digital camera at 20 frames per second. System's image processing is optimized because is done while image is acquired.

This new system can be used like an excellent benchmark prototype for testing many others control schemes. It will be a valuable tool for the research workers in this area.

Agradecimientos

A mis padres, J. Ascensión Pérez Vega y María Isidora Olvera Govea, que la vida me han regalado, por su apoyo incondicional y el haber sido mi primer escuela.

A mi hermano, abuelitos y tíos por acompañarme en los momentos gratos y haberme alentado en las dificultades.

A mis asesores Dr. Marco Antonio Moreno Arnedáriz y Dra. Elsa Rubio Espino, por su dedicación y paciencia para llevar a buen término la elaboración de esta tesis.

Al CONACYT y a la SIP-IPN por el apoyo económico brindado, sin el cual el presente trabajo de tesis no sería posible.

A Altera Corporation y Víctor Maruri por la donación de equipo y licencias de Quartus II.

Índice general

Glosario	VIII
1. Introducción.	1
1.1. Arquitectura de control visual.	1
1.2. Control basado en logica difusa.	3
1.2.1. Tipos de fuzificadores.	3
1.2.2. Tipos de defuzificadores.	5
1.3. FPGA.	6
1.4. Motivación.	7
1.5. Objetivos.	7
1.5.1. Objetivo general.	7
1.5.2. Objetivos paticulares.	7
1.6. Estructura de la tesis.	8
2. Estado del Arte.	9
3. Modelo del sistema Plano-Esfera.	12
3.1. Introducción.	12
3.2. Modelo del sistema Plano-Esfera: Parte mecánica.	12
3.2.1. Ecuación de Euler-Lagrange.	13
3.3. Modelo del sistema de visión.	18
3.3.1. Proyección perspectiva.	19
3.3.2. Factor de escala de la imagen digitalizada.	21
3.3.3. Caso del sistema Plano-Esfera	22
3.4. Mólodo completo del sistema Plano-Esfera	23
4. Control del sistema Plano-Esfera con cámara fija.	26
4.1. Introducción.	26
4.2. Mecanismo Plano-Esfera.	27
4.2.1. Mecanismo de actuación.	28
4.2.2. Características de la captura de la imagen.	31
4.2.3. Implementación del tratamiento digital de imágenes y control en tiempo real.	32
4.3. Sistema de tratamiento digital de imágenes.	33
4.3.1. Función de umbral.	33
4.3.2. Operador morfológico de erosión.	34

4.3.3. Cálculo del centroide	37
4.4. Control del sistema Plano-Esfera.	38
4.4.1. Fuzificador	38
4.4.2. Base de conocimiento	42
4.4.3. Mecanismo de inferencias	43
4.4.4. Defuzificación	45
5. Experimentos y resultados.	50
5.1. Módulo de visión	50
5.2. Control difuso tipo Mamdani	51
5.2.1. Introducción	51
5.2.2. Experimento tipo A	52
5.2.3. Experimento tipo B	55
5.2.4. Experimento tipo C	57
5.2.5. Resultados	60
6. Conclusiones y trabajo futuro.	63
6.1. Conclusiones.	63
6.2. Trabajos Futuros.	64
Referencias	67
Lista de apéndices	68
A. CAD del Prototipo Plano-Esfera	68
B. Componentes del diseño en hardware	70
C. Programas	72
D. Gráficas del experimento A	73
E. Gráficas del experimento B	84
F. Gráficas del experimento C	111

Glosario

base de conocimiento

Contiene el conocimiento obtenido de la descripción lingüística del experto de como realizar buen control. Una forma clásica de representar el conocimiento es con un conjunto de reglas del tipo Si... Entonces. Una regla es una estructura condicional que relaciona lógicamente la información contenida en la parte del antecedente con otra información contenida en la parte del consecuente. [1.2](#), [4.1](#)

componente conexas

Es un conjunto de píxeles tal que para cualquier par de píxeles del conjunto, existe un camino digital que los une. [4.1](#), [4.3](#)

FPGA

Del inglés *Field Programmable Gate Array*, fue inventadas en el año 1984 por Ross Freeman y Bernard Vonderschmitt, co-fundadores de Xilinx, es un dispositivo semiconductor que contiene bloques de lógica cuya interconexión y funcionalidad se puede programar. La lógica programable puede reproducir desde funciones tan sencillas como las llevadas a cabo por una puerta lógica o un sistema combinacional hasta complejos sistemas en un chip. [1.0](#), [1.3–1.6](#), [4.3](#), [4.4](#), [5.1](#)

grado de libertad

Es el conjunto de movimientos independientes que describen completamente el desplazamiento y orientación de un robot. [1.0](#), [2.0](#), [3.2](#)

imagen digital

Es una retícula de puntos, a cada uno de los cuales se le demonina píxel. Éstos asumen un valor tonal y se disponen a modo de mosaico definiendo áreas cromáticas. [3.3](#)

LEs

Del ingles *Logic Elements*. En los FPGA de la compañía Altera, una LE es la unidad más pequeña para la implementación eficiente de funciones lógicas del usuario. [4.2](#), [5.2](#)

muestreo

Se habla de muestreo en un lazo de control, cuando se introduce uno o más elementos de control que operen ya no de manera continua, sino de manera discreta en el tiempo. En estos casos la variable controlada debe de ser registrada en intervalos de tiempo equidistantes. La frecuencia de muestreo de tal dispositivo por supuesto debe de ser alta con respecto a la velocidad de cambio de la variable controlada, para que el comportamiento de la variable controlada sea efectiva. [1.1](#), [4.4](#)

píxel

Acrónimo del ingles picture element, superficie homogénea más pequeña de las que componen una imagen, que se define por su brillo y color. [3.3](#), [4.1](#), [4.3](#), [4.4](#), [5.1](#)

robot

Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas solo a las personas. [1.1](#)

sensor digital de imágenes CMOS

Es un sensor que detecta la luz basado en tecnología CMOS (del inglés, *Complementary Metal Oxide Semiconductor*), Gracias a esta tecnología es posible integrar más funciones en un chip. Está formado por numerosos fotosensores, uno para cada píxel, que producen una corriente eléctrica que varía en función de la intensidad de luz recibida. En el CMOS, a diferencia del CCD (del inglés, *charge-coupled device*) se incorpora un amplificador de la señal eléctrica en cada fotosensor y es común incluir el convertidos digital en el propio chip. En un CCD se tiene que enviar la señal eléctrica producida por cada fotosensor al exterior y desde allí se amplifica. [3.3](#), [5.1](#)

sistema de control

Conjunto de componentes que pueden regular su propia conducta o la de otro sistema con el fin de lograr un funcionamiento predeterminado, de modo que se reduzcan las probabilidades de fallos y se obtengan los resultados buscados. [1.1](#), [1.2](#), [4.1](#)

variable lingüística

De la misma manera que una variable algebraica toma números como valores, para una variable lingüística sus valores son palabras o sentencias en un lenguaje natural o artificial. El conjunto de valores que esta puede tomar es llamado *conjunto de términos*. Cada valor del conjunto de términos es una *variable difusa* definida sobre una *variable base*. La variable base define el universo del discurso para todas la variables difusas en el conjunto de términos. El comportamiento es el siguiente: variable lingüística \rightarrow variable difusa \rightarrow variable base. Un ejemplo de esto seria tomar como etiqueta de la variable lingüística a la edad. Los términos de esa variable lingüística, que son los conjuntos difusos, podria ser “*joven*”, “*viejo*”, “*muy viejo*”. $T(\text{edad})=\text{viejo, muy viejo, no tan viejo, más o menos joven, bastante joven, muy joven}$ [4.1](#), [4.4](#)

vecindad

Conjunto de celdas que se encuentran alrededor de la celda de referencia en un elemento de estructura, definida mediante un circunferencia de radio r . [4.3](#)

VHDL

Del inglés *Very High Speed Integrated Circuit Hardware Description Language*, lenguaje creado a partir de un proyecto promovido por el gobierno de Estados Unidos en 1981 y fue homologado por el IEEE en 1987 en el estándar 1076-87. VHDL utiliza una metodología de diseño de arriba abajo (top-bottom) que permite describir el circuito de forma estructural (señales y su conexionado) y funcional (qué hace). Cada bloque se puede definir empleando subcircuitos definidos en el mismo proyecto o disponibles en librerías propias o de terceros. [4.2](#), [4.3](#)

Índice de figuras

1.1. Configuraciones básicas de sistemas de visión.	2
1.2. Sistemas de control con retroalimentación visual basados en imagen.	2
3.1. Diagrama del sistema plano-esfera.	13
3.2. Geometría de la cámara y proyección perspectiva.	19
4.1. Sistema de control de lazo cerrado	27
4.2. Diseño del mecanismo PE	28
4.3. Señal de control de los servomotores.	30
4.4. Lectura de la imagen.	31
4.5. Adquisición de una línea de la imagen.	31
4.6. Adquisición de una imagen.	32
4.7. Imagen $I(u, v)$, esfera sobre el plano.	33
4.8. Comparador de 10-bits	34
4.9. Imagen binaria resultado de la función de umbral	35
4.10. Circuito equivalente de la operación morfológica de erosión.	36
4.11. Simulación de la operación morfológica de erosión en un FPGA.	37
4.12. Diagrama del controlador difuso.	38
4.13. Distribución de los conjuntos difusos de entrada	39
4.14. Conjuntos difusos almacenados en memoria ROM	40
4.15. Determinación del grado de pertenencia al conjunto difuso PM.	41
4.16. Fuzificador empleando memorias ROM	41
4.17. Dinámica del sistema PE.	42
4.18. Mecanismo de inferencia MAX-MIN	43
4.19. Simulación del Mecanismo de inferencia MAX-MIN.	46
4.20. Distribución de los conjuntos difusos de salida	47
4.21. Defuzificador promedio de centros.	47
4.22. Simulación del componente Defuzificador.	49
5.1. Sistema plano-esfera	50
5.2. Imágenes del sistema plano-esfera procesadas por el módulo de visión	51
5.3. Clasificación de los experimentos.	52
5.4. Comparación del movimiento de la esfera entre la simulación y el modelo real.	53
5.5. Gráficas del Error, Velocidad y Señal de control.	54
5.6. Comparación entre el movimiento de la simulación y el experimento real.	55

5.7. Gráficas del Error, Velocidad y Señal de control.	56
5.8. Conjuntos modificados de la Variable lingüística <i>Velocidad</i>	58
5.9. Conjuntos modificados de la Variable lingüística <i>Inclinación</i>	58
5.10. Comparación entre el movimiento de la simulación y el experimento real . .	59
5.11. Gráficas del Error, Velocidad y Señal de control para la simulación del sistema PE.	59
A.1. CAD de la base del Prototipo Plano-Esfera	68
A.2. CAD de la base del area de trabajo	69
A.3. CAD del soporte de la cámara	69
B.1. Módulo completo del sistema Plano-Esfera.	71
D.1. Comparación entre el movimiento de la simulación y el modelo real, experimento A1.	73
D.2. Gráficas del Error, Velocidad y Señal de control, simulación A1.	74
D.3. Gráficas del Error, Velocidad y Señal de control, modelo real A1.	74
D.4. Comparación entre el movimiento de la simulación y el modelo real, experimento A2.	75
D.5. Gráficas del Error, Velocidad y Señal de control, simulación A2.	75
D.6. Gráficas del Error, Velocidad y Señal de control, modelo real A2.	76
D.7. Comparación entre el movimiento de la simulación y el modelo real, experimento A3.	76
D.8. Gráficas del Error, Velocidad y Señal de control, simulación A3.	77
D.9. Gráficas del Error, Velocidad y Señal de control, modelo real A3.	77
D.10. Comparación entre el movimiento de la simulación y el modelo real, experimento A4.	78
D.11. Gráficas del Error, Velocidad y Señal de control, simulación A4.	78
D.12. Gráficas del Error, Velocidad y Señal de control, modelo real A4.	79
D.13. Comparación entre el movimiento de la simulación y el modelo real, experimento A5.	79
D.14. Gráficas del Error, Velocidad y Señal de control, simulación A5.	80
D.15. Gráficas del Error, Velocidad y Señal de control, modelo real A5.	80
D.16. Comparación entre el movimiento de la simulación y el modelo real, experimento A6.	81
D.17. Gráficas del Error, Velocidad y Señal de control, simulación A6.	81
D.18. Gráficas del Error, Velocidad y Señal de control, modelo real A6.	82
D.19. Comparación entre el movimiento de la simulación y el modelo real, experimento A7.	82
D.20. Gráficas del Error, Velocidad y Señal de control, simulación A7.	83
D.21. Gráficas del Error, Velocidad y Señal de control, modelo real A7.	83
E.1. Comparación entre el movimiento de la simulación y el modelo real, experimento B1.	85
E.2. Gráficas del Error, Velocidad y Señal de control, simulación B1.	85
E.3. Gráficas del Error, Velocidad y Señal de control, modelo real B1.	86
E.4. Comparación entre el movimiento de la simulación y el modelo real, experimento B2.	86
E.5. Gráficas del Error, Velocidad y Señal de control, simulación B2.	87
E.6. Gráficas del Error, Velocidad y Señal de control, modelo real B2.	87
E.7. Comparación entre el movimiento de la simulación y el modelo real, experimento B3.	88
E.8. Gráficas del Error, Velocidad y Señal de control, simulación B3.	88
E.9. Gráficas del Error, Velocidad y Señal de control, modelo real B3.	89
E.10. Comparación entre el movimiento de la simulación y el modelo real, experimento B4.	89
E.11. Gráficas del Error, Velocidad y Señal de control, simulación B4.	90
E.12. Gráficas del Error, Velocidad y Señal de control, modelo real B4.	90

E.13. Comparación entre el movimiento de la simulación y el modelo real, experimento B5. 91

E.14. Gráficas del Error, Velocidad y Señal de control, simulación B5. 91

E.15. Gráficas del Error, Velocidad y Señal de control, modelo real B5. 92

E.16. Comparación entre el movimiento de la simulación y el modelo real, experimento B6. 92

E.17. Gráficas del Error, Velocidad y Señal de control, simulación B6. 93

E.18. Gráficas del Error, Velocidad y Señal de control, modelo real B6. 93

E.19. Comparación entre el movimiento de la simulación y el modelo real, experimento B7. 94

E.20. Gráficas del Error, Velocidad y Señal de control, simulación B7. 94

E.21. Gráficas del Error, Velocidad y Señal de control, modelo real B7. 95

E.22. Comparación entre el movimiento de la simulación y el modelo real, experimento B8. 95

E.23. Gráficas del Error, Velocidad y Señal de control, simulación B8. 96

E.24. Gráficas del Error, Velocidad y Señal de control, modelo real B8. 96

E.25. Comparación entre el movimiento de la simulación y el modelo real, experimento B9. 97

E.26. Gráficas del Error, Velocidad y Señal de control, simulación B9. 97

E.27. Gráficas del Error, Velocidad y Señal de control, modelo real B9. 98

E.28. Comparación entre el movimiento de la simulación y el modelo real, experimento B10. 98

E.29. Gráficas del Error, Velocidad y Señal de control, simulación B10. 99

E.30. Gráficas del Error, Velocidad y Señal de control, modelo real B10. 99

E.31. Comparación entre el movimiento de la simulación y el modelo real, experimento B11.100

E.32. Gráficas del Error, Velocidad y Señal de control, simulación B11. 100

E.33. Gráficas del Error, Velocidad y Señal de control, modelo real B11. 101

E.34. Comparación entre el movimiento de la simulación y el modelo real, experimento B12.101

E.35. Gráficas del Error, Velocidad y Señal de control, simulación B12. 102

E.36. Gráficas del Error, Velocidad y Señal de control, modelo real B12. 102

E.37. Comparación entre el movimiento de la simulación y el modelo real, experimento B13.103

E.38. Gráficas del Error, Velocidad y Señal de control, simulación B13. 103

E.39. Gráficas del Error, Velocidad y Señal de control, modelo real B13. 104

E.40. Comparación entre el movimiento de la simulación y el modelo real, experimento B14.104

E.41. Gráficas del Error, Velocidad y Señal de control, simulación B14. 105

E.42. Gráficas del Error, Velocidad y Señal de control, modelo real B14. 105

E.43. Comparación entre el movimiento de la simulación y el modelo real, experimento B15.106

E.44. Gráficas del Error, Velocidad y Señal de control, simulación B15. 106

E.45. Gráficas del Error, Velocidad y Señal de control, modelo real B15. 107

E.46. Comparación entre el movimiento de la simulación y el modelo real, experimento B16.107

E.47. Gráficas del Error, Velocidad y Señal de control, simulación B16. 108

E.48. Gráficas del Error, Velocidad y Señal de control, modelo real B16. 108

E.49. Comparación entre el movimiento de la simulación y el modelo real, experimento B17.109

E.50. Gráficas del Error, Velocidad y Señal de control, simulación B17. 109

E.51. Gráficas del Error, Velocidad y Señal de control, modelo real B17. 110

F.1. Comparación entre el movimiento de la simulación y el modelo real, experimento C1.112

F.2. Gráficas del Error, Velocidad y Señal de control, simulación C1. 112

F.3. Comparación entre el movimiento de la simulación y el modelo real, experimento C2.113

F.4. Gráficas del Error, Velocidad y Señal de control, simulación C2. 113

F.5. Comparación entre el movimiento de la simulación y el modelo real, experimento C3.114

F.6. Gráficas del Error, Velocidad y Señal de control, simulación C3.	114
F.7. Comparación entre el movimiento de la simulación y el modelo real, experimento C4.115	
F.8. Gráficas del Error, Velocidad y Señal de control, simulación C4.	115
F.9. Comparación entre el movimiento de la simulación y el modelo real, experimento C5.116	
F.10. Gráficas del Error, Velocidad y Señal de control, simulación C5.	116
F.11. Comparación entre el movimiento de la simulación y el modelo real, experimento C6.117	
F.12. Gráficas del Error, Velocidad y Señal de control, simulación C6.	117
F.13. Comparación entre el movimiento de la simulación y el modelo real, experimento C7.118	
F.14. Gráficas del Error, Velocidad y Señal de control, simulación C7.	118
F.15. Comparación entre el movimiento de la simulación y el modelo real, experimento C8.119	
F.16. Gráficas del Error, Velocidad y Señal de control, simulación C8.	119
F.17. Comparación entre el movimiento de la simulación y el modelo real, experimento C9.120	
F.18. Gráficas del Error, Velocidad y Señal de control, simulación C9.	120
F.19. Comparación entre el movimiento de la simulación y el modelo real, experimento C10.121	
F.20. Gráficas del Error, Velocidad y Señal de control, simulación C10.	121
F.21. Comparación entre el movimiento de la simulación y el modelo real, experimento C11.122	
F.22. Gráficas del Error, Velocidad y Señal de control, simulación C11.	122
F.23. Comparación entre el movimiento de la simulación y el modelo real, experimento C12.123	
F.24. Gráficas del Error, Velocidad y Señal de control, simulación C12.	123
F.25. Comparación entre el movimiento de la simulación y el modelo real, experimento C13.124	
F.26. Gráficas del Error, Velocidad y Señal de control, simulación C13.	124
F.27. Comparación entre el movimiento de la simulación y el modelo real, experimento C14.125	
F.28. Gráficas del Error, Velocidad y Señal de control, simulación C14.	125
F.29. Comparación entre el movimiento de la simulación y el modelo real, experimento C15.126	
F.30. Gráficas del Error, Velocidad y Señal de control, simulación C15.	126
F.31. Comparación entre el movimiento de la simulación y el modelo real, experimento C16.127	
F.32. Gráficas del Error, Velocidad y Señal de control, simulación C16.	127
F.33. Comparación entre el movimiento de la simulación y el modelo real, experimento C17.128	
F.34. Gráficas del Error, Velocidad y Señal de control, simulación C17.	128

Índice de tablas

3.1. Variables del sistema Plano-Esfera.	13
4.1. Características del servomotor Hitec HS-805BB.	30
4.2. Tabla de verdad de $Q(u, v) = f(S(u, v))$	36
4.3. Conjunto de reglas basadas en un experto.	43
4.4. Reglas evaluadas para “Error = -4” y “Velocidad = 5”.	45
5.1. Cálculo de centroide de Fig. 5.2	52
5.2. Conjunto de reglas del experimento A.	53
5.3. Conjunto de reglas del experimento B.	55
5.4. Conjunto de reglas del experimento C.	57
5.5. Tabla de uso de recurso en el FPGA.	60
5.6. Resumen de las gráficas del experimento A.	61
5.7. Resumen de las gráficas del experimento B.	61
5.8. Resumen de las gráficas del experimento C.	62

Capítulo 1

Introducción.

El presente trabajo de tesis, trata del modelado, análisis y control visual de un sistema Plano-Esfera (PE), que es una evolución del sistema Barra-Esfera, que solo tiene dos **grados de libertad**, ya que consta de una esfera que rueda sobre una barra. Mientras que el PE tiene cuatro **grados de libertad** y consta de una esfera que puede moverse libremente sobre el plano. Esto lo hace atractivo para poner a prueba múltiples estrategias de control. En nuestro caso, incorporamos un sistema de visión y un controlador basado en lógica difusa, ambos embebidos en un **FPGA**.

Debido a la naturaleza del objeto a manipular en algunos sistemas, como es el caso del sistema PE, es casi imposible utilizar sensores convencionales (ultrasónicos, infrarrojos, de presión, etc.) que permitan adaptarse a su entorno y proporcionar información adecuada para realizar una determinada tarea, ya que esta depende de un conocimiento a priori de su espacio de trabajo y de la localización del objeto a manipular. Una característica importante del sistema PE es la incorporación del sistema de visión, el cual tienen como ventaja principal mimetizar el sistema de visión humana que es capaz de obtener rasgos del objeto a manipular. Esta información será usada por el controlador.

1.1. Arquitectura de control visual.

Cuando se introduce un sistema de visión como lazo de retroalimentación, representa más que un simple sensor que solo es capaz de adquirir una señal. Al dotar un sistema de control con visión, se busca incrementar la flexibilidad y capacidad de interacción de este con el medio ambiente. Es común que en aplicaciones donde se involucra control visual, la complejidad o inexistencia de algoritmos acotados para el análisis de la imagen, incrementa el costo computacional de la tarea de visión. Por esta razón es común hacer fuertes simplificaciones ya sea de la tarea, del entorno o del mismo análisis.

El control visual ha sido incorporado a los **robots** para dotarlos de la capacidad de interactuar con su entorno, por lo que se dispone de dos configuraciones básicas cámara-robot [21]: cámara en mano y cámara fija, ver Figura 1.1. En la primera tiene la cámara

montada sobre el efecto final del **robot**, proporciona información solo del espacio de trabajo. En la segunda la cámara se encuentra fija en el espacio de trabajo, por lo que la cámara captura al robot y su espacio de trabajo. Existe una tercera configuración que sería la combinación de las dos anteriores.

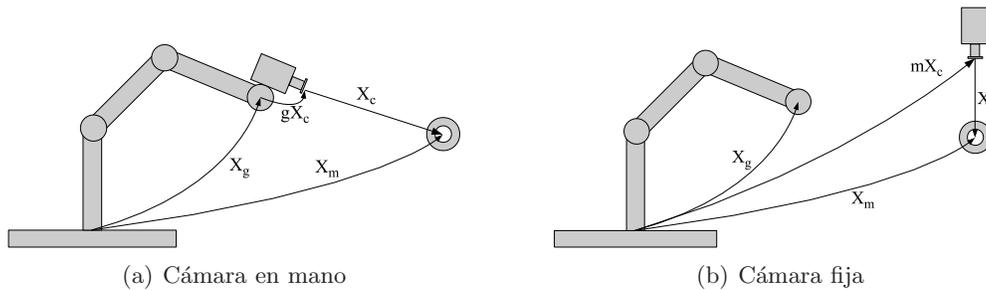


Figura 1.1: Configuraciones básicas de sistemas de visión.

El sistema de visión artificial resulta más útil cuando se incorpora al lazo de retroalimentación del esquema de control. Por lo antes mencionado se requieren de dos subsistemas: el sistema de visión y el **sistema de control**. Donde se tiene que diferenciar entre dos estrategias de sistemas con retroalimentación visual: “*Mirar y mover*” y “*Servo visión*”. La estrategia “*mirar y mover*” utiliza lazos internos de control con velocidad mayor de **muestreo** que el sistema de visión para lograr estabilizar al robot, esta estrategia se puede dividir en dos ramas: *estática* y *dinámica*. En el caso de la “*estrategia mirar y mover estática*”, se extraen los rasgos visuales de una imagen inicial del espacio de trabajo y la consecuencia es la acción de control del robot, el robot ejecuta la tarea considerando que el espacio de trabajo no se ha modificado. Ahora en el caso de la “*estrategia mirar y mover dinámica*”, la información visual es continuamente capturada y utilizada en el lazo de control, el objeto a manipular puede encontrarse en movimiento mientras la siguiente imagen es capturada, lo que incrementa la precisión en el posicionamiento. Con respecto a la “*estrategia servo visión*”, elimina lazos internos de control y lo reemplaza con un controlador servo visual, que sólo utiliza la información visual para estabilizar al robot, lo que incrementa la frecuencia de procesamiento de la imagen.

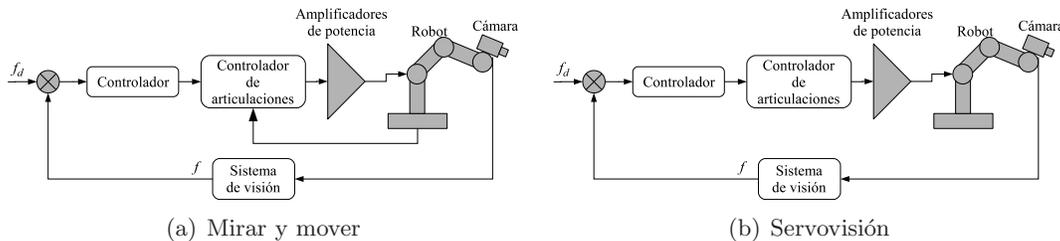


Figura 1.2: Sistemas de control con retroalimentación visual basados en imagen.

De acuerdo a las características extraídas de la imagen, los sistemas de control visual pueden estar basados en posición o basados en imagen. En los sistemas de control basados en posición, las características extraídas de la imagen en conjunto con el modelo geométrico del objeto y el modelo conocido de la cámara, se determina la posición del objeto con

respecto de la cámara. Para los sistemas de control visual basados en imagen, los valores del controlador son calculados directamente con los rasgos extraídos de la imagen, en esta última técnica se reduce el procesamiento computacional debido a que no es necesario interpretar los rasgos obtenidos de la imagen.

1.2. Control basado en logica difusa.

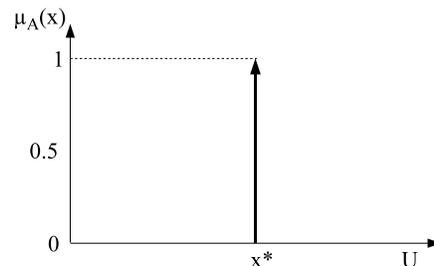
Es una técnica utilizada para implementar **sistemas de control** que toman decisiones similares al razonamiento humano. Un controlador basado en lógica difusa se puede diseñar de manera que imite el proceso deductivo humano. Es decir, el proceso que las personas utilizan para inferir conclusiones de las cosas que conocen.

Un controlador basado en lógica difusa tipo I consiste en cuatro componentes principales: una base de reglas, un mecanismo de inferencia difuso, una interfaz de fuzificación de las entradas y una interfaz defuzificadora de las salidas. La base de reglas almacena un conjunto de reglas SI-ENTONCES que cuantifican el conocimiento que los humanos expertos poseen en relación a la manera de resolver problemas particulares. Esta **base de conocimiento**, le sirve de recurso al mecanismo de inferencia difuso, el cual toma decisiones sucesivas de cuales reglas son las más relevantes a la situación corriente y aplica las acciones indicadas por esas reglas. El fuzificador toma los valores numéricos de las entradas y los convierte en forma difusa, necesaria para poder ser procesado por el mecanismo de inferencia. En la salida, el defuzificador combina las conclusiones del mecanismo de inferencia y las convierte en un valor numérico real para poder ser utilizado como salida del controlador.

1.2.1. Tipos de fuzificadores.

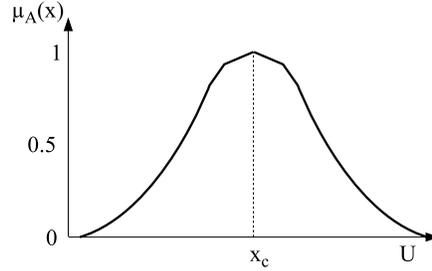
Fuzificador “singleton”- Transforma un valor numérico real $x^* \in U$ en un conjunto difuso “singleton” A en U, el cual tiene un grado de pertenencia de 1 en x^* y 0 en todos los demas puntos de U.

$$\mu_A(x) = \begin{cases} 1 & , x = x^* \\ 0 & , x \neq x^* \end{cases} \quad (1.1)$$



Fuzificador “gaussiano”- Transforma un valor numérico real $x^* \in U$ en un conjunto difuso A en U, el cual posee la siguiente función de pertenencia gaussiana:

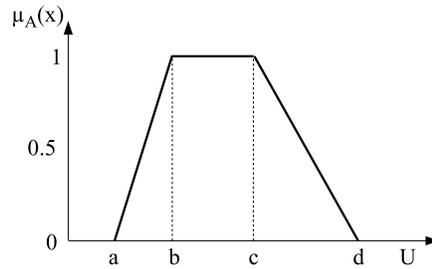
$$\mu_A(x) = e^{-\frac{(x^* - x_c)^2}{2\sigma^2}} \quad (1.2)$$



donde x_c es el valor de x donde la función gaussiana es máxima y σ es la desviación estandar.

Fuzificador “trapezoidal”- Transforma un valor numérico real $x^* \in U$ en un conjunto difuso A en U , el cual posee la siguiente función de pertenencia trapezoidal:

$$\mu_A(x) = \begin{cases} 1 & , \text{si } b \leq x^* \leq c \\ \left(1 - \frac{|b-x^*|}{b-a}\right) & , \text{si } a \leq x^* < b \\ \left(1 - \frac{|x^*-c|}{d-c}\right) & , \text{si } c \leq x^* < d \\ 0 & , \text{de otra manera.} \end{cases} \quad (1.3)$$

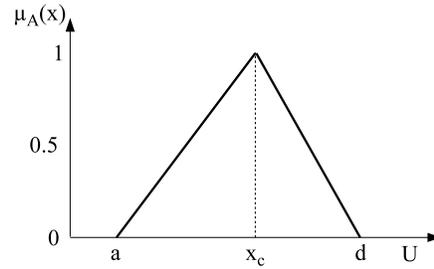


donde a , b , c y d son los parámetros de la función trapezoidal.

Fuzificador “triangular”- Transforma un valor numérico real $x^* \in U$ en un conjunto difuso A en U , el cual posee la siguiente función de pertenencia triangular:

$$\mu_A(x) = \begin{cases} 1 & , \text{si } x = x^* \\ \left(1 - \frac{|x_c - x^*|}{x_c - a}\right) & , \text{si } a \leq x^* < x_c \\ \left(1 - \frac{|x_c - x^*|}{b - x_c}\right) & , \text{si } x_c \leq x^* < b \\ 0 & , \text{de otra manera.} \end{cases} \quad (1.4)$$

donde a y b son los parámetros de la función triangular y x_c es el valor de x donde la función triangular es máxima. Según se puede observar, el fuzificador triangular es un caso especial del fuzificador trapezoidal.



1.2.2. Tipos de defuzificadores.

Defuzificador de criterio máximo - La salida es el valor numérico real del punto donde la función de pertenencia del conjunto de la salida asume su valor máximo.

$$z^* = \text{cualquier punto de la altura (B)} \quad (1.5)$$

donde z^* es la variable numérica de salida y B es el conjunto difuso de la salida.

Si la altura (B) contiene un solo punto, z^* esta definida en forma única. Si la altura (B) contiene más de un punto, se pueden especificar tres defuzificadores:

Defuzificador del valor más pequeño del valor máximo

$$z^* = \inf \{z \in \text{altura}(B)\} \quad (1.6)$$

donde $\inf \{ \}$ es la cota inferior.

Defuzificador del valor más grande del máximo

$$z^* = \sup \{z \in \text{altura}(B)\} \quad (1.7)$$

donde $\sup \{ \}$ es la cota superior.

Defuzificador del valor promedio del máximo

$$z^* = \frac{\int_{\text{altura}(B)} z \, dz}{\int_{\text{altura}(B)} dz} \quad (1.8)$$

Defuzificador de centro de gravedad - Especifica la variable de salida z^* como el centro del área cubierta por la función de pertenencia del conjunto difuso de salida B.

$$z^* = \frac{\int \mu_B(z) z \, dz}{\int \mu_B(z) \, dz} \quad (1.9)$$

Defuzificador del promedio de los centros - En este defuzificador, también conocido como *centro de sumas*, el valor escalar de la variable de salida z^* esta determinado por el promedio

de los centros de los M conjuntos difusos de salida con los pesos w , siendo igual a la altura de los conjuntos difusos correspondientes. A su vez, este metodo es el más utilizado para la realización en hardware.

$$z^* = \frac{\sum_{i=1}^M \mu_{B_i}(z_i) w_i}{\sum_{i=1}^M \mu_{B_i}(z_i)} \quad (1.10)$$

1.3. FPGA.

Las **FPGA** (Field Programmable Gate Array) son el resultado de la convergencia de dos tecnologías diferentes, los dispositivos lógicos programables (PLDs, Programmable Logic Devices) y los circuitos integrados de aplicación específica (ASIC, Application-Specific Integrated Circuit). La historia de los PLDs comenzó con los primeros dispositivos PROM (Programmable Read-Only Memory) y se les añadió versatilidad con los PAL (Programmable Array Logic) que permitieron un mayor número de entradas y la inclusión de registros. Esos dispositivos han continuado creciendo en tamaño y potencia. Mientras, los ASIC siempre han sido potentes dispositivos, pero su uso ha requerido tradicionalmente una considerable inversión tanto de tiempo como de dinero. Intentos de reducir esta carga han provenido de la modularización de los elementos de los circuitos, como el los ASIC basados en celdas, y de la estandarización de las máscaras, tal como Ferranti fue pionero con la ULA (Uncommitted Logic Array). El paso final era combinar las dos estrategias con un mecanismo de interconexión que pudiese programarse utilizando fusibles, antifusibles o celdas RAM, como los innovadores dispositivos Xilinx de mediados de los 80's. Los circuitos resultantes son similares en capacidad y aplicaciones a los PLDs más grandes, aunque hay diferencias puntuales que delatan antepasados diferentes. Además de en computación reconfigurable, las FPGAs se utilizan en controladores, codificadores/decodificadores y en el prototipado de circuitos VLSI (Very-Large-Scale Integration) y microprocesadores a medida.

El diseño de una FPGA se lleva a cabo especificando la función lógica a desarrollar, bien mediante un sistema de diseño asistido por computadora (CAD, Computer-aided design) de dibujo de esquemas, bien mediante un lenguaje de programación de alto nivel (HDL, Hardware Description Language). Una vez definida la función a realizar, el diseño se traslada a la FPGA. Este proceso programa los bloques lógicos configurables (CLB, Configurable Logic Blocks) para realizar una función específica (existen miles de bloques lógicos configurables en la FPGA). La configuración de estos bloques y la flexibilidad de sus interconexiones son las razones por las que se pueden conseguir diseños de gran complejidad. Las interconexiones permiten conectar los bloques lógicos (CLBs) entre sí.

Básicamente lo que conseguimos es poder programar un hardware a partir de un software, con lo cual nos simplificaría el proceso de creación de hardware evitando así todos los problemas que ello conlleva, además de permitirnos mucha más flexibilidad a la hora de diseñar. Una gran ventaja también es el poco espacio físico que ocupan, pues al tratarse de un solo circuito integrado siempre ocupará menos que cualquier hardware a base de circuitos integrados que hagamos.

El primer fabricante de estos dispositivos fue Xilinx y los dispositivos de Xilinx se mantienen como uno de los más populares en compañías y grupos de investigación. Otros vendedores en este mercado son Altera, Atmel, AMD y Motorola.

1.4. Motivación.

La no-linealidad del sistema PE y el movimiento de sus componentes en 3-D, son algunas de las características que ha atraído el interés de investigadores en el área de control. Además de la naturaleza no-lineal del sistema, es un excelente ejercicio de diseño la creación del mecanismo PE, lo que también se ha convertido en la principal desventaja y justificación de su poca popularidad con respecto a otros sistemas, como es el caso del sistema Barra-Esfera.

Dada la naturaleza del objeto a manipular en el sistema PE y el entorno en el que se desenvuelve, es necesario disponer de sensores que proporcionen información apropiada para llevar a cabo la tarea de control. Un sensor digital de imágenes es una buena opción, ya que no interfiere en el desarrollo de la tarea que este inspecciona. Un inconveniente es la gran cantidad de información que proporciona, además de ser un sensor con alto nivel de ruido en su medición, por lo que el proceso de interpretar la información suele tener un alto costo computacional, por ello el sistema debe ser capaz de tomar una decisión antes de que cambien lo menos posible las condiciones en las que se realizó dicha medición.

En un intento por disminuir el retardo de tiempo, introducido por el procesamiento de imágenes en tiempo real, ha resultado en arquitecturas de procesamiento en paralelo, una manera de cubrir esta necesidad es optar por hardware especializado. Las arquitecturas específicas pueden ser implementados en **FPGA**, cuando los microcontroladores o un procesador digital de señales (DSP, Digital signal processing), no ofrecen el paralelismo requerido.

Por lo anterior, el sistema PE requiere de la sinergia de esfuerzos en las áreas de control, mecánica, electrónica y computación, que lo hace un atractivo caso de estudio.

1.5. Objetivos.

1.5.1. Objetivo general.

Diseñar, construir e implementar el sistema PE, que sea capaz de llevar una esfera de la posición inicial a la final dentro de un área de trabajo. La posición final es propuesta previamente por el usuario, donde la esfera deberá permanecer equilibrada. La forma de manipular la esfera, es a través de la inclinación de los ejes del plano.

1.5.2. Objetivos particulares.

- Diseñar y construir el sistema PE.
- Utilizar una cámara como sensor de la posición de la esfera dentro del plano.

- Realizar un tratamiento digital y extracción de rasgos a las imágenes capturadas por la cámara, de modo que el resultado será la posición de la esfera dentro del área de trabajo.
- Diseñar un controlador basado en lógica difusa para lograr estabilizar la esfera en la posición propuesta por el usuario.
- Implementar el tratamiento digital de imágenes, extracción de rasgos y el controlador basado en lógica difusa en un dispositivo programable **FPGA**, con la finalidad de realizar la tarea en tiempo real.

1.6. Estructura de la tesis.

En este capítulo se han presentado: la introducción, motivación y el objetivo. El resto del documento está organizado de la siguiente forma.

En el capítulo 2 se presenta el estado del arte con relación al estudio del sistema PE. El cual abarca desde novedosos diseños de construcción del sistema PE, esquemas de control PID, hasta controladores difusos que aprenden en línea. Por su parte el capítulo 3 expone el modelo matemático del sistema plano esfera, el cual se divide en el estudio del sistema en su parte mecánica y el sistema de visión. Al término del capítulo se expresa el modelo completo del sistema PE así como las consideraciones para llegar a este.

El capítulo 4 es la parte más relevante de este documento, se explica lo correspondiente al diseño y construcción del sistema PE, así como la descripción del módulo de tratamiento digital de imágenes, extracción de características y el diseño del controlador difuso tipo Mamdani, todos implementados en **FPGA**. El contenido del capítulo también incluye simulaciones, imágenes y esquemas ilustrativos del diseño de los diferentes módulos. Por lo tanto, el capítulo 5 presenta los resultados derivados del módulo de tratamiento digital de imágenes y el comportamiento del controlador difuso, para el cual se basó en tres experimentos dada la posición final de la esfera.

En el capítulo 5 se presentan las conclusiones dados los objetivos alcanzados y posibles opciones de trabajo para futuros investigadores. Por último, en los apéndices **A**, **B** y **C**, se ilustran los planos de construcción de sistema PE, los esquemas de los módulos embebidos en el **FPGA** y los programas utilizados.

Capítulo 2

Estado del Arte.

El sistema PE está conformado por un balón que puede rodar libremente sobre un plano rígido, el sistema posee cuatro **grados de libertad** y sólo dos actuadores para la rotación del plano sobre el *eje x* y *eje y*. Dadas estas condiciones lo hacen interesante como punto de prueba para el desempeño de diferentes tipos de estrategias de control tales como PID, controladores basados en redes neuronales y lógica difusa, etc. El antecesor de este sistema es conocido como bola-balancín (ball and beam, en inglés), el cual sólo posee dos grados de libertad y un actuador.

Algunos ejemplos en la literatura del sistema PE son: [8] donde el prototipo del sistema PE fue de diseño propio, se utilizó un controlador PID discreto, así como modificaciones al implementar un filtro de primer orden en la parte derivativa del PID, con lo que se logró incrementar la distancia hacia los extremos del plano y el tiempo de posicionamiento de la esfera mejoró. Otra modificación fue el control con función de saturación en la parte proporcional para eliminar el sobretiro presentado en el controlador con filtro en la parte derivativa. Por último se probó el control PID con generación de trayectoria mejorando la rapidez con que la esfera llegaba a la referencia, además se logró posicionar la esfera en cualquier parte del plano.

En [2] después de linealizar el modelo matemático del sistema PE, concluyeron que el uso del controlador PID no era recomendable, por lo que utilizaron un controlador de adelanto, con el cual lograron estabilizar el sistema. La plataforma de experimentación en la que se basaron no fue de diseño propio, fue adquirida a la empresa HUMOSOFT [12][13] dedicada a la venta y asesoría de modelos educativos para ingeniería de sistemas de control entre otros.

En [16] también utilizaron dicha plataforma. Los autores realizaron un controlador para el seguimiento de trayectorias, este artículo es interesante porque el punto de referencia es móvil, lo que permite probar la eficiencia de un buen algoritmo. En este caso y tomando en consideración las simulaciones presentadas, el algoritmo sí resulta muy eficiente. El algoritmo fue puesto a prueba con una trayectoria circular y otra en forma de diamante. Se propone un control difuso del tipo Mamdani para la solución del problema. Las reglas de inferencia difusas son descritas a través de módulos *single input rule modules* (SIRMs), las cuales

permiten disminuir el número de reglas difusas que describen al problema. El problema Plano-Esfera es visto desde el punto de vista clásico, el cual claro, incluye la cámara para registrar el movimiento de la esfera. Se retomó de su sistema de visión el cálculo aproximado de Euler [14] para estimar la siguiente posición de la esfera, donde utilizan la posición y velocidad actual de la esfera para predecir la siguiente posición, considerando que la velocidad de la esfera es constante ya que el tiempo de captura de la imagen es muy corto. En la presente tesis se procesó la imagen completa que a diferencia de este artículo se utilizó la técnica de ventanas dinámicas, ya que la imagen es procesada en una computadora personal.

Un esquema de tres niveles de control de comportamiento difuso es mostrado en [27], el nivel más bajo del esquema está ocupado por un control de rastreo difuso tipo Takagi-Sugeno que controla el balón para seguir la trayectoria deseada. Los niveles superiores, supervisión y planeación son controladores difusos tipo Mamdani. Este artículo fue fundamental en la presente tesis para el estudio del modelo matemático del sistema PE.

El artículo [9] también contiene una descripción del modelo matemático del sistema PE, a diferencia del anterior artículo, este modelo sí considera la fricción entre la esfera y el plano. Como parte de la estrategia de control este artículo propone un observador no lineal de estados, que en este caso son las velocidades del movimiento de la esfera sobre los ejes XY. Además utiliza un regulador de posición diseñado con módulos SIRMs. Los autores utilizan un filtro de Kalman para el diseño del observador no-lineal y para la estimación de la velocidad de la esfera. El sistema trabaja con ocho estados derivados del modelo. Clásicamente como en este tipo de modelos, el sistema de visión cierra el lazo de control.

En [1] a partir de las señales de entrada y salida de un controlador PD previamente diseñado y con un razonable desempeño, fue obtenido un conjunto de reglas difusas con ayuda de un controlador difuso de aprendizaje en línea, donde el aprendizaje es realizado a través de un esquema de Wang de búsqueda en tablas [26]. Mientras que en [17], el sistema PE consiste en un mecanismo paralelo espacial de cinco enlaces donde los enlaces son conectados con cinco uniones de un grado de libertad, para esta plataforma es propuesto dos esquemas de control. El primero es basado en un sistema difuso supervisado y un controlador PID. El segundo está basado en un controlador por modo deslizante con un aproximador difuso. El método que utilizaron para detectar la posición de la esfera es similar al nuestro, en su trabajo, también extraen la posición de la esfera de una imagen obtenida por una cámara, la diferencia radica en que implementaron un método de ventana dinámica, al utilizar este método solo una porción de la imagen alrededor de la esfera es procesada y para ubicar la siguiente posición de la ventana estimaron la siguiente posición de la esfera.

En [15] se muestra una variante del sistema Plano-Esfera, los autores trabajan con el diseño de una superficie cóncava, los experimentos a realizar son; posicionar automáticamente la esfera sobre el centro de la superficie cóncava y seguimiento de trayectorias. Utilizan técnicas de control moderno como lo es el uso del regulador cuadrático lineal y el filtro de Kalman, el cual utilizan para estimar la posición de la esfera. Muestran claramente y paso a paso la obtención del modelo del sistema. Obtienen una respuesta muy eficiente según se muestra en las gráficas. Incluyen (aunque pocas) simulaciones de la respuesta al escalón y a una

entrada senoidal.

Los autores en [7] tratan el problema de la estabilización de un sistema no-holonómico de 5 dimensiones llamado: Plano-Esfera o rodamiento de una esfera. El sistema se trata y se resuelve a través de una ley de control híbrida, utilizando una máquina de estados finitos. Se asocia una ley de control para cada estado de la máquina, éste control toma en cuenta saturaciones en sus acciones. La ley de control global, se realiza a través de conmutaciones entre los estados. El control se considera híbrido debido a la representación discreta de la acción de control y a la representación continua de la ley de control. Se estudia la estabilidad por medio de una función de Lyapunov.

Este artículo [4] se tomo como referencia ya que discute el concepto y desarrollo de sistema Plano-Esfera basado en el principio de diseño mecatrónico. El mecanismo utilizado para mover el plano en los ejes XY, consiste de un pivote y dos pequeños brazos bajo el plano; uno de ellos está en el centro y los otros dos a una distancia muy cercana del centro, pero debajo de cada eje de movimiento. Consta de dos motores de CD para mover de manera acoplada con los brazos: (eje y, pivote) y (eje x, pivote). La posición de la esfera sobre el plano es registrada a través de la superficie del plano, debido a que cuenta con tecnología touchscreen. Finalmente, describe al sistema por medio de dos ecuaciones no-lineales de segundo orden.

Debido a que se decidió integrar en el presente trabajo de tesis el tratamiento digital de imágenes (TDI) y el controlador difuso en un dispositivo programable (tecnología FPGA), las siguientes referencias fueron básicas: [10] y [25]. El primer artículo sirvió como base para el TDI, ya que estudia diferentes tareas de TDI como lo son la detección de componentes conexos, detección de bordes, remoción de puntos, entre otros. Estas tareas son resueltas utilizando las salidas y entradas binarias de redes neuronales celulares en tiempo discreto (Discrete-Time Cellular Neural Networks, DTCNNs) que pueden ser realizadas usando funciones Booleanas, donde su correspondiente circuito lógico puede ser derivado e implementado por la tecnología FPGA. Por su parte en la tesis de maestría [25] trata sobre el diseño e implementación de controladores digitales basados en lógica difusa, este estudio permite obtener soluciones en arquitecturas de procesamiento serie y procesamiento paralelo codificadas en lenguajes de descripción de hardware (Hardware Description Language, HDL), que resulta ser una excelente opción para control en tiempo-real.

Capítulo 3

Modelo del sistema Plano-Esfera.

3.1. Introducción.

El sistema Plano-Esfera (PE) se muestra en la Figura 3.1, y está compuesto por una esfera de acero y un plano de vidrio, el plano puede rotar alrededor de los ejes perpendiculares X y Y, de esta forma la esfera se mueve bajo la influencia de la fuerza de gravedad. La posición de la esfera es controlada por la inclinación del plano, para ello se utilizaron dos servomotores (uno por cada eje) y un sensor digital de imágenes CMOS (cámara digital) para conocer la posición de la esfera sobre el plano.

El objetivo que se propuso es equilibrar la esfera en una posición sobre el plano a través de la inclinación de este, para llevar a cabo la tarea es necesarios el estudio del movimiento del sistema PE, con la finalidad de conocer más de su dinámica. El sistema de visión debe de ser incluido en el modelo, ya que existe una conversión del sistema de coordenadas del mundo real, al sistema de coordenadas de la imagen digitalizada y así poder realizar control basado en la imagen [20].

Para realizar el modelado, se plantea como hipótesis que: la esfera rueda con suavidad sobre el plano, por lo tanto la esfera no patina, lo que impone una restricción al movimiento de rotación del plano. Los movimientos del plano son pequeños y lentos, aproximadamente de 2.5° , por lo que la distancia entre la cámara y el plano se considera constante.

3.2. Modelo del sistema Plano-Esfera: Parte mecánica.

El sistema PE se puede considerar como un sistema compuesto por dos cuerpos rígidos. El plano tiene dos **grados de libertad** que son la rotación sobre el *eje x* y *eje y*. La esfera tiene dos grados de libertad para la traslación a lo largo del plano (x,y) . Por lo tanto el sistema completo tiene cuatro grados de libertad x, y, θ_x y θ_y , que son elegidos como coordenadas generalizadas de posición, los actuadores son 2, τ_x y τ_y relacionados con el movimiento del

plano.

Las variables del sistema PE son las siguientes:

Símbol	Descripción	Unidades
$\omega(t)$	Velocidad angular de la esfera alrededor de su centro de masa, con $\omega(t) = \frac{\dot{x}^2(t) + \dot{y}^2(t)}{R^2}$.	rad
$\theta_x(t), \theta_y(t)$	Posición angular del plano	rad
$\dot{\theta}(t)$	Velocidad angular del plano, con $\dot{\theta}^2(t) = \dot{\theta}_x^2(t) + \dot{\theta}_y^2(t)$.	rad/s
$x(t), y(t)$	Posición de la esfera sobre el plano	m
$\dot{r}(t)$	Velocidad lineal de la esfera, con $\dot{r}^2(t) = \dot{x}^2(t) + \dot{y}^2(t)$.	m/s
$\dot{s}(t)$	Velocidad lineal debido a la rotación (longitud del arco), con $\dot{s}^2(t) = \left(x(t) \dot{\theta}_x(t) + y(t) \dot{\theta}_y(t) \right)^2$.	m/s
I_{cdm}	Momento de inercia de la esfera es, $\frac{2}{5}MR^2$.	Kg m ²
M	Masa de la esfera.	Kg
R	Radio de la esfera.	m
G	Aceleración debido a la gravedad.	m/s ²
h	Altura. $h = x \sin(\theta_x(t)) + y \sin(\theta_y(t))$	m
$\tau_x(t)$	Torque aplicado al plano en el eje x	Kg m ² /s ²
$\tau_y(t)$	Torque aplicado al plano en el eje y	Kg m ² /s ²

Tabla 3.1: Variables del sistema Plano-Esfera.

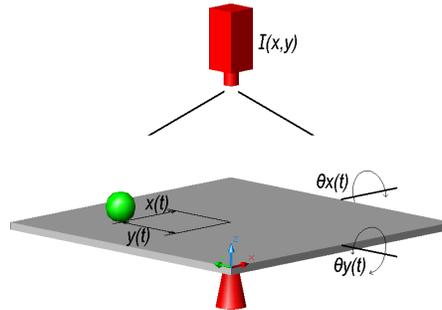


Figura 3.1: Diagrama del sistema plano-esfera.

3.2.1. Ecuación de Euler-Lagrange.

Para obtener el modelo matemático del sistema PE se utilizó el método de Euler-Lagrange por medio de las coordenadas generalizadas de posición $(x, y, \theta_x, \theta_y)$. El lagrangiano del sistema se obtuvo de la siguiente forma:

$$L(t) = E_C(t) - E_P(t) \quad (3.1)$$

Donde:

$$\begin{aligned} L(t) &= \text{Lagrangiano.} \\ E_c(t) &= \text{Energía Cinética.} \\ E_p(t) &= \text{Energía Potencial.} \end{aligned}$$

Dado que el sistema presenta movimientos de traslación y rotación tenemos que:

$$E_C(t) = E_{Tra}(t) + E_{Rot}(t) \quad (3.2)$$

$$E_{Tra}(t) = \frac{1}{2} M v_{cdm}^2(t) \quad (3.3)$$

$$E_{Rot}(t) = \frac{1}{2} I_{cdm} \omega^2(t) \quad (3.4)$$

Donde:

$$\begin{aligned} E_{Tra}(t) &= \text{Energía de traslación.} \\ E_{Rot}(t) &= \text{Energía de rotación.} \end{aligned}$$

Para el caso donde la esfera se comporta como un cuerpo rígido uniforme con movimiento de rodamiento suave, tenemos:

a) La $E_{Tra}(t)$ del centro de masa de la esfera es: $\frac{1}{2} M \dot{r}^2(t)$

b) La $E_{Rot}(t)$ de la esfera alrededor de su centro de masa es: $\frac{1}{2} I_{cdm} \left(\frac{\dot{r}(t)}{R} \right)^2$

Cuando la esfera se comporta como una partícula en movimiento que sigue una trayectoria circular.

a) La $E_{Tra}(t)$ del centro de masa de la esfera es: $\frac{1}{2} M \dot{s}^2(t)$

b) La $E_{Rot}(t)$ del centro de masa de la esfera es: $\frac{1}{2} I_{cdm} \dot{\theta}^2(t)$

Sustituyendo la energía de traslación y rotación en (3.3) y (3.4) tenemos:

$$E_{Tra}(t) = \frac{1}{2} M \dot{r}^2(t) + \frac{1}{2} M \dot{s}^2(t) \quad (3.5)$$

$$E_{Rot}(t) = \frac{1}{2} I_{cdm} \left(\frac{\dot{r}(t)}{R} \right)^2 + \frac{1}{2} I_{cdm} \dot{\theta}^2(t) \quad (3.6)$$

La Energía Potencial del sistema PE se obtiene de la siguiente forma:

$$E_P(t) = M G h \quad (3.7)$$

Sustituyendo la energía potencial y cinética en (3.1):

$$\begin{aligned} L(t) &= \frac{1}{2} M (\dot{x}^2(t) + \dot{y}^2(t)) + \frac{1}{2} M \left(x(t) \dot{\theta}_x(t) + y(t) \dot{\theta}_y(t) \right)^2 + \\ &+ \frac{1}{2} I_{cdm} \left(\frac{\dot{x}^2(t) + \dot{y}^2(t)}{R^2} \right) + \frac{1}{2} I_{cdm} (\dot{\theta}_x^2(t) + \dot{\theta}_y^2(t)) \\ &- M G (x \sin(\theta_x(t)) + y \sin(\theta_y(t))) \end{aligned} \quad (3.8)$$

Con ayuda de (3.8), se obtienen las ecuaciones que rigen el movimiento del sistema PE por medio de las ecuaciones de Euler-Lagrange, entonces:

$$\frac{d}{dt} \left[\frac{\partial L(t)}{\partial \dot{x}(t)} \right] - \frac{\partial L(t)}{\partial x(t)} = 0 \quad (3.9)$$

$$\frac{d}{dt} \left[\frac{\partial L(t)}{\partial \dot{\theta}_x(t)} \right] - \frac{\partial L(t)}{\partial \theta_x(t)} = \tau_x(t) \quad (3.10)$$

$$\frac{d}{dt} \left[\frac{\partial L(t)}{\partial \dot{y}(t)} \right] - \frac{\partial L(t)}{\partial y(t)} = 0 \quad (3.11)$$

$$\frac{d}{dt} \left[\frac{\partial L(t)}{\partial \dot{\theta}_y(t)} \right] - \frac{\partial L(t)}{\partial \theta_y(t)} = \tau_y(t) \quad (3.12)$$

Evaluando la derivada parcial en (3.9):

$$\begin{aligned} \frac{\partial}{\partial x(t)} [L(t)] &= My(t) \dot{\theta}_x(t) \dot{\theta}_y(t) + Mx(t) \dot{\theta}_x(t) \dot{\theta}_x(t) - MG \sin \theta_x(t) \\ \frac{\partial}{\partial \dot{x}(t)} [L(t)] &= \left(M + \frac{I_{cdm}}{R^2} \right) \dot{x}(t) \\ \frac{d}{dt} \left[\frac{\partial}{\partial \dot{x}(t)} [L(t)] \right] &= \left(M + \frac{I_{cdm}}{R^2} \right) (\ddot{x}(t)) \\ \frac{d}{dt} \left[\frac{\partial}{\partial \dot{x}(t)} [L(t)] \right] - \frac{\partial}{\partial x(t)} [L(t)] &= \left(M + \frac{I_{cdm}}{R^2} \right) \ddot{x}(t) - My \dot{\theta}_x(t) \dot{\theta}_y(t) \\ &\quad - Mx \dot{\theta}_x(t) \dot{\theta}_x(t) + MG \sin \theta_x(t) \\ \left(M + \frac{I_{cdm}}{R^2} \right) \ddot{x}(t) - Mx(t) \dot{\theta}_x^2(t) - My(t) \dot{\theta}_x(t) \dot{\theta}_y(t) + MG \sin \theta_x(t) &= 0 \end{aligned} \quad (3.13)$$

Evaluando la derivada parcial en (3.10):

$$\begin{aligned} \frac{\partial}{\partial \theta_x(t)} [L(t)] &= MGx(t) \cos \theta_x(t) \\ \frac{\partial}{\partial \dot{\theta}_x(t)} [L(t)] &= Mx(t) y(t) \dot{\theta}_y(t) + Mx^2(t) \dot{\theta}_x(t) + I_{cdm} \dot{\theta}_x(t) \\ \frac{d}{dt} \left[\frac{\partial}{\partial \dot{\theta}_x(t)} [L(t)] \right] &= M \left(xy \ddot{\theta}_y + x \dot{y} \dot{\theta}_y + \dot{x} y \dot{\theta}_y \right) + M \left(x^2 \ddot{\theta}_x + 2x \dot{x} \dot{\theta}_x \right) + I_{cdm} \ddot{\theta}_x \\ \frac{d}{dt} \left[\frac{\partial}{\partial \dot{\theta}_x(t)} [L(t)] \right] - \frac{\partial}{\partial \theta_x(t)} [L(t)] &= M \left(xy \ddot{\theta}_y + x \dot{y} \dot{\theta}_y + \dot{x} y \dot{\theta}_y \right) + M \left(x^2 \ddot{\theta}_x + 2x \dot{x} \dot{\theta}_x \right) \\ &\quad + I_{cdm} \ddot{\theta}_x + MGx(t) \cos \theta_x(t) \end{aligned}$$

$$(I_{cdm} + Mx^2) \ddot{\theta}_x + 2Mx\dot{x}\dot{\theta}_x + Mxy\ddot{\theta}_y + (Mxy\dot{\theta}_y + M\dot{x}y)\dot{\theta}_y + MGx(t) \cos \theta_x(t) = \tau_x(t) \quad (3.14)$$

Evaluando la derivada parcial en (3.11):

$$\begin{aligned} \frac{\partial}{\partial y(t)} [L(t)] &= M \left(x(t) \dot{\theta}_x(t) \dot{\theta}_y(t) + y(t) \dot{\theta}_y(t) \dot{\theta}_y(t) \right) - MG \sin \theta_y(t) \\ \frac{\partial}{\partial \dot{y}(t)} [L(t)] &= M\dot{y}(t) + \frac{I_{cdm}}{R^2} \dot{y}(t) = \left(M + \frac{I_{cdm}}{R^2} \right) \dot{y}(t) \\ \frac{d}{dt} \left[\frac{\partial}{\partial \dot{y}(t)} [L(t)] \right] &= \frac{d}{dt} \left[\left(M + \frac{I_{cdm}}{R^2} \right) \dot{y}(t) \right] = \left(M + \frac{I_{cdm}}{R^2} \right) \ddot{y}(t) \\ \frac{d}{dt} \left[\frac{\partial}{\partial \dot{x}(t)} [L(t)] \right] - \frac{\partial}{\partial x(t)} [L(t)] &= \left(M + \frac{I_{cdm}}{R^2} \right) \ddot{y}(t) - M \left(x(t) \dot{\theta}_x(t) \dot{\theta}_y(t) + y(t) \dot{\theta}_y(t) \dot{\theta}_y(t) \right) \\ &\quad + MG \sin \theta_y(t) \end{aligned}$$

$$\left(M + \frac{I_{cdm}}{R^2} \right) \ddot{y}(t) - Mx(t) \dot{\theta}_x(t) \dot{\theta}_y(t) - My(t) \dot{\theta}_y^2(t) + MG \sin \theta_y(t) = 0 \quad (3.15)$$

Evaluando la derivada parcial en (3.12):

$$\begin{aligned} \frac{\partial}{\partial \theta_y(t)} [L(t)] &= MGy(t) \cos \theta_y(t) \\ \frac{\partial}{\partial \dot{\theta}_y(t)} [L(t)] &= Mx(t)y(t)\dot{\theta}_x(t) + My(t)y(t)\dot{\theta}_y(t) + I_{cdm}\dot{\theta}_y(t) \\ \frac{d}{dt} \left[\frac{\partial}{\partial \dot{\theta}_y(t)} [L(t)] \right] &= M \left(xy\ddot{\theta}_x + \dot{x}y\dot{\theta}_x + x\dot{y}\dot{\theta}_x \right) + M \left(y^2\ddot{\theta}_y + 2y\dot{y}\dot{\theta}_y \right) + I_{cdm}\ddot{\theta}_y \\ \frac{d}{dt} \left[\frac{\partial}{\partial \dot{\theta}_y(t)} [L(t)] \right] - \frac{\partial}{\partial \theta_y(t)} [L(t)] &= M \left(xy\ddot{\theta}_x + \dot{x}y\dot{\theta}_x + x\dot{y}\dot{\theta}_x \right) + M \left(y^2\ddot{\theta}_y + 2y\dot{y}\dot{\theta}_y \right) \\ &\quad + I_{cdm}\ddot{\theta}_y + MGy(t) \cos \theta_y(t) \end{aligned}$$

$$(I_{cdm} + My^2) \ddot{\theta}_y + 2My\dot{y}\dot{\theta}_y + Mxy\ddot{\theta}_x + (M\dot{x}y + Mx\dot{y})\dot{\theta}_x + MGy(t) \cos \theta_y(t) = \tau_y(t) \quad (3.16)$$

Despejando la aceleración lineal de (3.13) y (3.15):

$$\ddot{x}(t) = \frac{M}{\left(M + \frac{I_{cdm}}{R^2} \right)} \left[x(t) \dot{\theta}_x^2(t) + y(t) \dot{\theta}_x(t) \dot{\theta}_y(t) - G \sin \theta_x(t) \right] \quad (3.17)$$

$$\ddot{y}(t) = \frac{M}{\left(M + \frac{I_{cdm}}{R^2} \right)} \left[y(t) \dot{\theta}_y^2(t) + x(t) \dot{\theta}_x(t) \dot{\theta}_y(t) - G \sin \theta_y(t) \right] \quad (3.18)$$

Despejando la aceleración angular de (3.14) y (3.16):

$$\ddot{\theta}_x = \frac{\tau_x(t)}{(I_{cdm} + Mx^2(t))} - \frac{2Mx(t)\dot{x}(t)\dot{\theta}_x(t)}{(I_{cdm} + Mx^2(t))} - \frac{Mx(t)y(t)\ddot{\theta}_y(t)}{(I_{cdm} + Mx^2(t))} - \frac{M(x(t)\dot{y}(t) + \dot{x}(t)y(t))\dot{\theta}_y(t)}{(I_{cdm} + Mx^2(t))} - \frac{MGx(t)\cos\theta_x(t)}{(I_{cdm} + Mx^2(t))} \quad (3.19)$$

$$\ddot{\theta}_y = \frac{\tau_y(t)}{(I_{cdm} + My^2(t))} - \frac{2My(t)\dot{y}(t)\dot{\theta}_y(t)}{(I_{cdm} + My^2(t))} - \frac{Mx(t)y(t)\ddot{\theta}_x(t)}{(I_{cdm} + My^2(t))} - \frac{M(\dot{x}(t)y(t) + x(t)\dot{y}(t))\dot{\theta}_x(t)}{(I_{cdm} + My^2(t))} - \frac{MGy(t)\cos\theta_y(t)}{(I_{cdm} + My^2(t))} \quad (3.20)$$

Ahora para poder describir completamente el comportamiento del sistema PE es necesario determinar las variables de estado:

$$x_1(t) = x(t) \quad (3.21)$$

$$\dot{x}_1(t) = \dot{x}(t) = x_2(t) \quad (3.22)$$

$$\dot{x}_2(t) = \ddot{x}(t)$$

$$x_3(t) = \theta_x(t) \quad (3.23)$$

$$\dot{x}_3(t) = \dot{\theta}_x(t) = x_4(t) \quad (3.24)$$

$$\dot{x}_4(t) = \ddot{\theta}_x(t)$$

$$x_5(t) = y(t) \quad (3.25)$$

$$\dot{x}_5(t) = \dot{y}(t) = x_6(t) \quad (3.26)$$

$$\dot{x}_6(t) = \ddot{y}(t)$$

$$x_7(t) = \theta_y(t) \quad (3.27)$$

$$\dot{x}_7(t) = \dot{\theta}_y(t) = x_8(t) \quad (3.28)$$

$$\dot{x}_8(t) = \ddot{\theta}_y(t)$$

Como se puede observar de (3.17), (3.18), (3.19) y (3.20), es un sistema no lineal ya que existe una dependencia entre variables de estado y la relación entre ellas es no lineal, además de que los movimientos del plano son acoplados.

$$\dot{x}(t) = f[X(t)] + G[\tau(t)]$$

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \\ \dot{x}_5(t) \\ \dot{x}_6(t) \\ \dot{x}_7(t) \\ \dot{x}_8(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ A \\ x_4(t) \\ B \\ x_6(t) \\ C \\ x_8(t) \\ D \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{I_{cdm} + Mx_1^2(t)} \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{I_{cdm} + Mx_5^2(t)} \end{bmatrix} \begin{bmatrix} \tau_x(t) \\ \tau_y(t) \end{bmatrix} \quad (3.29)$$

$$Y(t) = h[X(t)] := (x_1(t), x_5(t))^T \quad (3.30)$$

Donde:

$$\begin{aligned}
A &= \frac{M}{\left(M + \frac{I_{cdm}}{R^2}\right)} \left(x_1(t)x_4^2(t) + x_5(t)x_4(t)x_8(t) - G \sin x_3(t) \right) \\
B &= \frac{M}{\left(M + \frac{I_{cdm}}{R^2}\right)} \left(x_5(t)x_8^2(t) + x_1(t)x_4(t)x_8(t) - G \sin x_7(t) \right) \\
C &= -\frac{2Mx_1(t)x_2(t)x_4(t)}{I_{cdm} + Mx_1^2(t)} - \frac{Mx_1(t)x_5(t)\dot{x}_8(t)}{I_{cdm} + Mx_1^2(t)} - \frac{M(x_1(t)x_6(t) + x_2(t)x_5(t))x_8(t)}{I_{cdm} + Mx_1^2(t)} - \frac{MGx_1(t)\cos(x_3(t))}{I_{cdm} + Mx_1^2(t)} \\
D &= -\frac{2Mx_5(t)x_6(t)x_8(t)}{I_{cdm} + Mx_5^2(t)} - \frac{Mx_1(t)x_5(t)\dot{x}_4(t)}{I_{cdm} + Mx_5^2(t)} - \frac{M(x_2(t)x_5(t) + x_1(t)x_6(t))x_4(t)}{I_{cdm} + Mx_5^2(t)} - \frac{MGx_5(t)\cos(x_7(t))}{I_{cdm} + Mx_5^2(t)} \\
X(t) &= (x_1(t), x_2(t), x_3(t), x_4(t), x_5(t), x_6(t), x_7(t), x_8(t))^T \\
&= (x(t), \dot{x}(t), \theta_x(t), \dot{\theta}_x(t), y(t), \dot{y}(t), \theta_y(t), \dot{\theta}_y(t))^T
\end{aligned}$$

Se tiene como entrada del modelo el torque que aplica el motor para modificar la inclinación del plano, la ecuación de estado esta dada por (3.29) y la ecuación de salida (3.30), tenemos:

$$\begin{aligned}
f[X(t)] &= \text{Matriz de Estado} \\
G[\tau(t)] &= \text{Matriz de Entrada} \\
X(t) &= \text{Vector de Estado} \\
\tau(t) &= \text{Vector de Entrada} \\
Y(t) &= \text{Vector de Salida}
\end{aligned}$$

3.3. Modelo del sistema de visión.

Los rayos de luz que inciden en una cámara entran a través de una apertura muy pequeña (apertura óptica), la intersección de estos rayos con el fondo de la cámara forma una imagen en 2-D (plano imagen) de un objeto en 3-D (mundo real), a este mapeo se le llama proyección perspectiva.

Una cámara digital forma una representación discreta del plano imagen que incide sobre un arreglo de fotosensores, por lo regular el número de fotosensores en una línea del arreglo no coincide con el número de elementos en la **imagen digital** debido al procesamiento análogo y la digitalización, además de que las coordenadas del centro del plano imagen tampoco concuerdan con las coordenadas del centro de la **imagen digital**. Por todo esto, para encontrar el modelo geométrico de la cámara primero se tiene que hacer una transformación de un sistema de coordenadas 3-D en el que se encuentra el objeto, al sistema de coordenadas del plano imagen, considerando la orientación y posición de la cámara con respecto del sistema en 3-D, posteriormente una nueva transformación del plano imagen a las coordenadas de la **imagen digital**, tomando en cuenta las características intrínsecas de la cámara.

Por lo regular el sistema de coordenadas de la escena “ O_m ” no esta alineado con el sistema de

coordenadas de la cámara que tiene su origen en el centro de proyección “ O ”, lo que requiere transformar al punto $P(x_m, y_m, z_m)$ en el sistema de coordenadas de la escena a el mismo punto P en el sistema de coordenadas de la cámara $P(x, y, z)$ (parámetros extrínsecos).

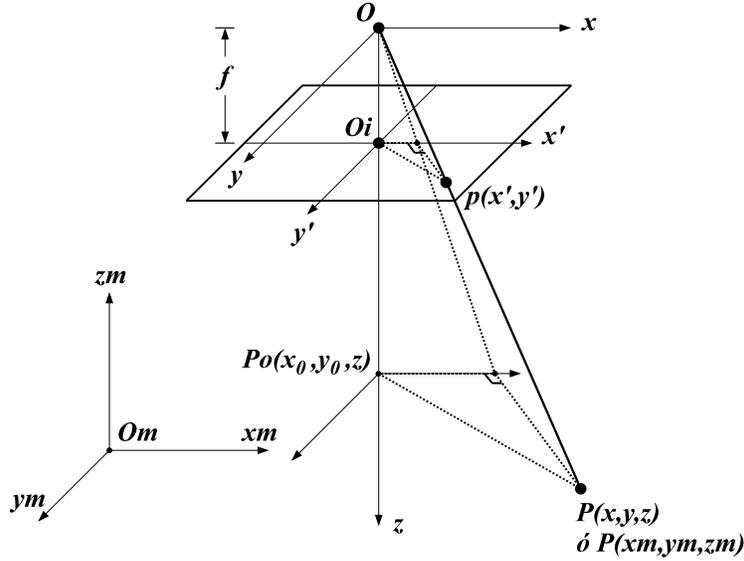


Figura 3.2: Geometría de la cámara y proyección perspectiva.

Si se considera al punto P como una matriz 3×1 se puede hacer una transformación lineal de dicho punto multiplicado por una matriz 3×3 y la adición de una matriz 3×1 que representa una transformación no lineal, estas transformaciones son respectivamente la orientación y posición del sistema de coordenadas de la cámara con respecto del sistema de coordenadas de la escena, así que.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (3.31)$$

3.3.1. Proyección perspectiva.

El modelo geométrico de la cámara, ver Figura 3.2, consiste en un plano imagen y el centro de proyección $O(x_0, y_0, z_0)$, f es la distancia entre el centro de proyección y el plano imagen (distancia focal), el eje óptico se encuentra a lo largo del eje z y es perpendicular al plano imagen, la intersección del eje óptico con el plano imagen es el centro de la imagen O_i .

La recta que es formada por los puntos OP , presenta un grado de inclinación con respecto al eje óptico, esta misma recta intercepta al plano imagen en el punto $p(x', y', f)$ por lo que el segmento de recta Op tiene la misma pendiente, considerando los planos (z, x) y (z, y) del sistema de coordenadas de la cámara. De esta forma se puede encontrar la proyección del punto P en el plano imagen dada la distancia focal, por lo que se tienen las siguientes ecuaciones:

$$x' = \frac{fx}{z}, y' = \frac{fy}{z}, z' = f \quad (3.32)$$

La ecuación (3.32) se puede expresar de la siguiente forma:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \frac{f}{z} & 0 & 0 \\ 0 & \frac{f}{z} & 0 \\ 0 & 0 & \frac{f}{z} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (3.33)$$

La ecuación (3.33) solo expresa la proyección del punto $P(x, y, z)$ en el plano imagen con respecto al sistema de coordenadas de la cámara, pero no la posición con respecto del sistema de coordenadas de la escena, por lo que al sustituir la matriz de rotación de (3.33) en (3.31) tenemos:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \frac{f}{z} & 0 & 0 \\ 0 & \frac{f}{z} & 0 \\ 0 & 0 & \frac{f}{z} \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (3.34)$$

Al agregar una cuarta coordenada “s” al punto $p(x', y', z')$ tenemos a $p(sx', sy', sz')$, donde “s” representa un factor de escala y (3.34) puede ser expresada de la siguiente forma:

$$\begin{bmatrix} sx' \\ sy' \\ sz' \\ s \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & \frac{1}{f} & 0 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} \quad (3.35)$$

Para simplificar la matriz de transformación homogénea (3.35), hacemos las siguientes suposiciones:

1. El centro de proyección de la cámara, coincide con el origen del sistema de coordenadas de la escena.
2. El eje óptico de la cámara coincide con el eje z_m del sistema de coordenadas de la escena.
3. Para evitar la imagen invertida, suponemos que el plano imagen se encuentra frente al centro de proyección.

Por lo antes mencionado, la matriz de transformación homogénea (3.36) indica que el sistema de coordenadas de la cámara está orientado y no está trasladado con respecto del sistema de coordenadas de la escena, pero si presenta una proyección perspectiva con respecto del eje óptico dada por la razón $\frac{f}{z}$. Para obtener las coordenadas de la proyección perspectiva $p(x', y', z', 1)$, hay que dividir las primeras tres coordenadas entre la cuarta, así.

$$\begin{bmatrix} sx' \\ sy' \\ sz' \\ s \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3.36)$$

3.3.2. Factor de escala de la imagen digitalizada.

Con el modelo geométrico de la cámara podemos mapear un punto 3-D de la escena al plano imagen, para el caso del sistema PE se utilizó una cámara digital, el elemento captor de la cámara es un arreglo de fotosensores que forman una representación discreta del plano imagen, cada elemento del arreglo es representado como un elemento en la **imagen digitalizada** llamado **píxel**, debido al procesamiento análogo y el digitalizador en la cámara la mayoría de las veces el número de elementos sensores en una fila del arreglo no es el mismo en la **imagen digitalizada**, aunado al aspecto del elemento sensor (dimensiones), se le conoce a la relación como factor de escala horizontal, para el caso del factor de escala vertical la relación de elementos sensores y **píxeles** en la imagen es biunívoca.

$$\begin{aligned}\alpha_x &= \frac{1}{\frac{N_c}{N_d} p_x} \\ \alpha_y &= \frac{1}{p_y}\end{aligned}\quad (3.37)$$

Donde:

- N_c Número de elementos fotosensores en la dirección X de la cámara.
- N_d Número de píxeles por fila en la imagen digitalizada.
- p_x, p_y Dimensiones de un elemento sensor en la cámara dadas en micrómetros.

El plano imagen y la **imagen digitalizada** están orientadas, pero las coordenadas del centro del plano imagen no coinciden con el centro de la **imagen digitalizada**, por lo que se tiene que hacer una transformación afín, que es:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} sx' \\ sy' \\ sz' \\ s \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \\ w_0 \end{bmatrix}\quad (3.38)$$

$$\begin{bmatrix} u \\ v \\ w \\ k \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & 0 & u_0 \\ 0 & \alpha_y & 0 & v_0 \\ 0 & 0 & 0 & w_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} sx' \\ sy' \\ sz' \\ s \end{bmatrix}\quad (3.39)$$

Debido a que en el sistema de coordenadas de la **imagen digitalizada** se pierde la profundidad, la coordenada w_0 es nula y al sustituir (3.36) en (3.39) tenemos:

$$\begin{bmatrix} u \\ v \\ w \\ k \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & 0 & u_0 \\ 0 & \alpha_y & 0 & v_0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}\quad (3.40)$$

La ecuación (3.40) representa la transformación del sistema de coordenadas de la escena al sistema de coordenadas de la **imagen digitalizada**, la cual se reduce a lo siguiente.

$$\begin{bmatrix} u \\ v \\ w \\ k \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & \frac{u_0}{f} & 0 \\ 0 & \alpha_y & \frac{v_0}{f} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}\quad (3.41)$$

Si multiplicamos los coeficientes de la matriz (3.41) por f , podemos expresarla de las siguientes formas:

$$\begin{bmatrix} u \\ v \\ w \\ k \end{bmatrix} = \begin{bmatrix} f\alpha_x & 0 & u_0 & 0 \\ 0 & f\alpha_y & v_0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3.42)$$

$$\begin{bmatrix} u \\ v \\ w \\ k \end{bmatrix} = \begin{bmatrix} x(f\alpha_x) & 0 & z(u_0) & 0 \\ 0 & y(f\alpha_y) & z(v_0) & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & z & 0 \end{bmatrix} \quad (3.43)$$

Ya que (3.42) esta compuesta por una matriz de transformación del sistema de coordenadas de la escena $[x, y, z, 1]^T$, al sistema de coordenadas de la **imagen digital** $[u, v, z, k]^T$, se puede expresar en forma de (3.43). Para encontrar las tres primeras coordenadas de la **imagen digital**, se tiene que dividir por la cuarta que es k y representa el escalado global.

$$\begin{bmatrix} u \\ v \\ w \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f}{z}\alpha_x & 0 & u_0 & 0 \\ 0 & \frac{f}{z}\alpha_y & v_0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3.44)$$

La ecuación (3.44) también puede ser expresada como se muestra en (3.45), dada la ausencia de profundidad en la **imagen digital**.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{f}{z} \begin{bmatrix} \alpha_x & 0 \\ 0 & \alpha_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (3.45)$$

3.3.3. Caso del sistema Plano-Esfera

En el caso del sistema PE, para conocer la posición de la esfera sobre el plano de vidrio, se consideró como lazo de retroalimentación una cámara digital, por lo que el centro de la **imagen digital** coincide con el centro del plano y se colocó la cámara a una distancia fija sobre este.

La cámara que se utilizó para el sistema de visión es un **sensor digital de imágenes CMOS**, modelo MT9M011 [23]. La resolución máxima del sensor es de 1280H x 1024V **píxeles** activos, las dimensiones de cada **píxel** es de $3.6\mu m$ H x $3.6\mu m$ V y una distancia f entre el lente y el APS (*Active-pixel sensor*) de $4.8mm$. El tamaño del plano de vidrio es de $0.5m$ x $0.5m$ y la distancia entre el plano y la cámara es aproximadamente de $0.651m$. Una de las funciones de este sensor, es poder especificar el tamaño de la imagen menor ó igual al formato SXGA, por lo que la cámara fue configurada con un tamaño de imagen igual a 1024H x 1024V **píxeles** activos para que la forma del plano coincida con la forma de la imagen, de esta manera se cubre por completo el área del plano.

Debido al filtro de color Bayer en el sensor, es necesario interpolar para formar los planos de color verde, rojo y azul, con los que se forma una imagen de color, para evitar esto se utilizó

el fondo negro y un balón de color blanco, ya que el color blanco es la superposición de todos los colores, por lo que no fue necesaria la interpolación para el caso particular del sistema PE.

Dadas las características intrínsecas de la cámara, para obtener el factor de escala de la **imagen digital**, consideramos que un elemento sensor del MT9M011, corresponde a un **píxel** de la **imagen digital** y además su forma es cuadrada, por lo que (3.37) queda de la siguiente forma:

$$\alpha = \frac{1}{p_x} = \frac{1}{p_y}, \frac{pixel}{m} \quad (3.46)$$

Como el factor de escala horizontal y vertical es el mismo podemos escribir (3.45) de la siguiente forma:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{f\alpha}{z} \begin{bmatrix} x \\ y \end{bmatrix} = \frac{(4,8e - 3mm) \left(\frac{1}{3,6\mu m}\right)}{0,651m} \begin{bmatrix} x \\ y \end{bmatrix} \quad (3.47)$$

Por las características intrínsecas y extrínsecas de la cámara antes mencionadas f , α y z , se puede considerar esta relación como una transformación K del sistema de coordenadas de la escena al sistema de coordenadas de la **imagen digital**.

$$\begin{bmatrix} u \\ v \end{bmatrix} = K \begin{bmatrix} x \\ y \end{bmatrix} = 2048 \begin{bmatrix} x \\ y \end{bmatrix} \quad (3.48)$$

De esta manera se tiene que para un punto en la superficie plana móvil existe un punto en el plano imagen que es multiplicado por un factor de conversión aproximadamente constante. Este modelo es el más sencillo para relacionar las coordenadas del espacio de trabajo a las coordenadas de la imagen.

3.4. Modelo completo del sistema Plano-Esfera

Dada la no-linealidad el modelo matemático del sistema PE (3.29), se toman las siguientes consideraciones:

- Los movimientos de la esfera sobre el plano son lentos.
- Los ángulos de inclinación de la esfera son pequeños.

Ya que los movimientos de la esfera son lentos, no presenta tendencia a patinar (rodamiento suave), lo que impone una restricción a la aceleración de rotación del plano. Debido a la baja velocidad y aceleración de rotación del plano, suponemos la interacción de los movimientos del plano desacoplados. Definimos dos nuevas entradas $c_x(t)$ y $c_y(t)$ para (3.14) y (3.16):

$$\begin{aligned} \tau_x(t) &= (I_{cdm} + Mx^2) c_x + 2Mx\dot{x}\dot{\theta}_x + Mxy\ddot{\theta}_y + (Mxy + M\dot{x}y)\dot{\theta}_y + MGx(t) \cos \theta_x(t) \\ \tau_y(t) &= (I_{cdm} + My^2) c_y + 2My\dot{y}\dot{\theta}_y + Mxy\ddot{\theta}_x + (M\dot{x}y + Mxy)\dot{\theta}_x + MGy(t) \cos \theta_y(t) \end{aligned}$$

Por lo que el modelo del sistema PE puede ser aproximado de la siguiente forma:

$$\dot{x}(t) = f[X(t)] + G[\tau(t)]$$

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \\ \dot{x}_5(t) \\ \dot{x}_6(t) \\ \dot{x}_7(t) \\ \dot{x}_8(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ \frac{M}{\left(M + \frac{I_{cdm}}{R^2}\right)} (x_1(t)x_4^2(t) - G \sin x_3(t)) \\ x_4(t) \\ 0 \\ x_6(t) \\ \frac{M}{\left(M + \frac{I_{cdm}}{R^2}\right)} [x_5(t)x_8^2(t) - G \sin x_7(t)] \\ x_8(t) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ c_x(t) \\ 0 \\ 0 \\ 0 \\ c_y(t) \end{bmatrix} \quad (3.49)$$

$$Y(t) = h[X(t)] := (x_1(t), x_5(t))^T \quad (3.50)$$

Donde:

$$\begin{aligned} X(t) &= (x_1(t), x_2(t), x_3(t), x_4(t), x_5(t), x_6(t), x_7(t), x_8(t))^T \\ &= (x(t), \dot{x}(t), \theta_x(t), \dot{\theta}_x(t), y(t), \dot{y}(t), \theta_y(t), \dot{\theta}_y(t))^T \end{aligned}$$

Apartir de la relación (3.48), se escogen la siguientes variables de estado:

$$\begin{aligned} u_1 &= u = Kx_1(t) \\ \dot{u}_1 &= \dot{u} = K\dot{x}_1(t) = Kx_2(t) \\ \dot{u}_2 &= \ddot{u} = K\dot{x}_2(t) \end{aligned} \quad (3.51)$$

$$\begin{aligned} v_1 &= v = Kx_5(t) \\ \dot{v}_1 &= \dot{v} = K\dot{x}_5(t) = Kx_6(t) \\ \dot{v}_2 &= \ddot{v} = K\dot{x}_6(t) \end{aligned} \quad (3.52)$$

Sustituyendo \dot{u}_2 y \dot{v}_2 en (3.49) tenemos el modelo completo del sistema PE.

$$\begin{bmatrix} \dot{u}_1(t) \\ \dot{u}_2(t) \\ \dot{u}_3(t) \\ \dot{u}_4(t) \\ \dot{v}_1(t) \\ \dot{v}_2(t) \\ \dot{v}_3(t) \\ \dot{v}_4(t) \end{bmatrix} = \begin{bmatrix} u_2(t) \\ E (u_1(t)u_4^2(t) - KG \sin u_3(t)) \\ u_4(t) \\ 0 \\ v_2(t) \\ E (v_1(t)v_4^2(t) - KG \sin v_3(t)) \\ v_4(t) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ c_u(t) \\ 0 \\ 0 \\ 0 \\ c_v(t) \end{bmatrix} \quad (3.53)$$

$$Y(t) = h[X(t)] := (u_1(t), v_1(t))^T \quad (3.54)$$

Donde:

$$\begin{aligned} X(t) &= (u_1(t), u_2(t), u_3(t), u_4(t), v_1(t), v_2(t), v_3(t), v_4(t))^T \\ &= (u(t), \dot{u}(t), \theta_u(t), \dot{\theta}_u(t), v(t), \dot{v}(t), \theta_v(t), \dot{\theta}_v(t))^T \end{aligned} \quad (3.55)$$

$$E = \frac{M}{\left(M + \frac{I_{cm}}{R^2}\right)} = \frac{M}{\left(M + \frac{\frac{2}{5}MR^2}{R^2}\right)} = \frac{M}{\left(M + \frac{2}{5}M\right)} = \frac{M}{\frac{7}{5}M} = \frac{5}{7} \quad (3.56)$$

Como se observa en (3.56) el tamaño y masa de la esfera son variables requeridas en el modelo matemático, pero al sustituir la inercia de la esfera, el valor de la masa M y radio R son divididos por sí mismos, como resultado obtenemos que el modelo matemático del sistema PE es invariante ante el tamaño y masa de la esfera.

Capítulo 4

Control del sistema Plano-Esfera con cámara fija.

4.1. Introducción.

El presente capítulo trata el diseño, construcción e implementación del sistema mecánico PE, selección de sensores y actuadores, así como el diseño del **sistema de control** en lazo cerrado, ver Figura 4.1. Nos permitimos dividirlo en dos sub-sistemas, el sub-sistema de tratamiento digital de imágenes (TDI) que lo forman los módulos de pre-procesamiento y extracción de características. El sub-sistema de control se dividió en dos módulos, un controlador difuso tipo Mamdani con interfaz de motores.

El módulo de pre-procesamiento del sub-sistema de TDI es el encargado de configurar la cámara, adquirir la imagen $I(u,v)$, aplicar una función de umbral a los datos obtenidos de la cámara como método de segmentación y utilizar la transformación morfológica de erosión para eliminar **píxeles** que no pertenecen al objeto de interés (esfera). Una vez que la imagen $q(u,v)$ solo contenga al objeto de interés representado por la región de **componentes conexos** $q(u,v) = 1$ y el fondo de la imagen $q(u,v) = 0$, podemos obtener las características cinemáticas del objeto. En este caso, posición y velocidad.

Por su parte, el módulo de control difuso, cuenta con una interfaz fuzificadora, que toma las características cinemáticas del objeto y las expresa en forma de **variables lingüísticas** [29] (error estimado de posición y velocidad), la **base de conocimiento**, contiene la información y experiencia de un experto en el manejo del prototipo PE, estructurada mediante inferencias (*reglas difusas*) del tipo *SI... ENTONCES*, por lo que el mecanismo de inferencia [MAX-MIN] produce una salida basada en el conocimiento del experto, esta salida todavía es difusa, por lo tanto, el defuzificador toma las conclusiones del mecanismo de inferencia y las convierte en un valor escalar, que la interfaz de motores interpreta para generar la señal apropiada y realizar la acción de control.

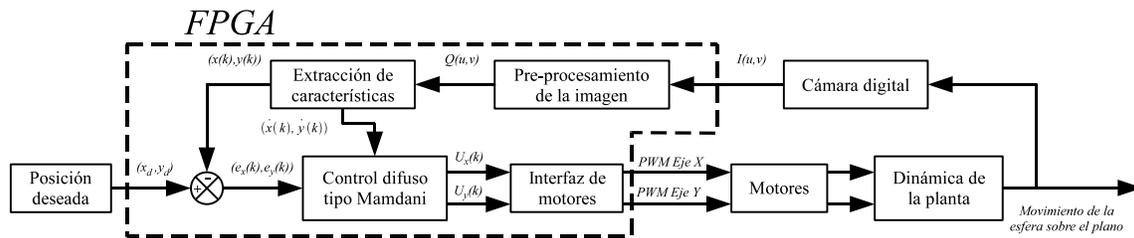


Figura 4.1: Sistema de control de lazo cerrado

4.2. Mecanismo Plano-Esfera.

Este trabajo de tesis requirió la construcción del mecanismo PE, ver Figura 4.2(a). El propósito de su construcción es poseer una plataforma de pruebas, en donde se pueda desarrollar tanto esquemas de control como algoritmos de rastreo, planificación de trayectorias, procesamiento de imágenes en tiempo real, por dar algunos ejemplos.

El mayor y principal reto al que se enfrentó como parte del diseño de construcción del mecanismo PE, fue la detección de la posición de la esfera sobre el plano, la selección de un sensor que proporcione la mayor cantidad de información, fácil de adecuar al mecanismo y además pueda ser explotado su potencial. Por lo que se valoraron dos opciones:

1. Utilizar un pantalla táctil, la implementación puede ser difícil de realizar y la experimentación se limita a entornos estáticos.
2. Utilizar una cámara digital, tiene como principal ventaja, la detección de objetos en entornos dinámicos. Por desventaja encontramos el incremento del costo computacional para el procesamiento de imágenes en tiempo real.

Presentadas las bondades y carencias de las opciones que consideramos, declinamos por utilizar una cámara digital, ya que este sensor nos permite explorar el campo del procesamiento digital de imágenes en tiempo real, que en los últimos años a tenido un gran auge y no solo limitar la investigación al desarrollo de esquemas de control.

El siguiente paso fue la elección del mecanismo de actuación, que hace posible el movimiento del plano sobre sus ejes de rotación, para ello se acoplo en el centro del plano un pivote. Debido a este diseño, se consideraron las siguientes opciones:

1. Dos actuadores lineales colocados en los bordes del plano, proporcionan los grados necesarios de movimiento. Estos son demasiado caros.
2. Mecanismo de brazos ligados al rotor de dos motores eléctricos (motor DC, motor a pasos, servomotor). La inclinación del plano es igual al ángulo de rotación del rotor del motor.
3. Arreglo de cables y poleas usando dos motores eléctricos (motor DC, motor a pasos, servomotor). La inclinación del plano esta relacionada con el tamaño de la polea.

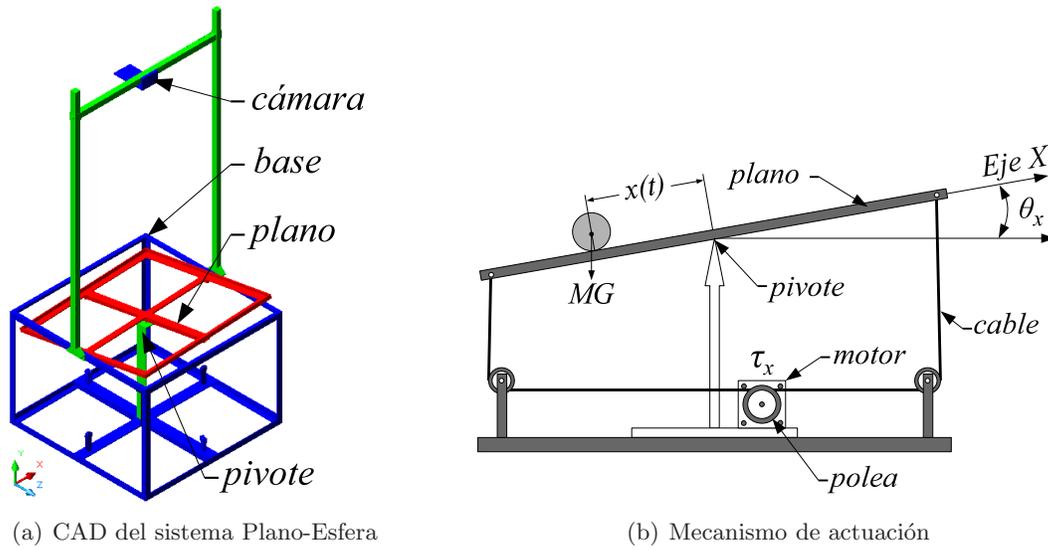


Figura 4.2: Diseño del mecanismo PE

La tercera opción fue la que se eligió; en la Figura 4.2(b) ilustra el mecanismo de actuación para el eje X, un mecanismo idéntico se utiliza en el eje Y. El apéndice A contiene los planos de construcción de la estructura metálica del sistema PE. En las siguientes subsecciones se puntualizan detalles del mecanismo de actuación, así como características de la captura de la imagen, del dispositivo donde se implementó el tratamiento digital de imágenes y el control difuso.

4.2.1. Mecanismo de actuación.

Al haber elegido el arreglo de cables y poleas como mecanismo de actuación, se tiene que considerar la relación existente entre el ángulo de giro de la polea y el ángulo de inclinación del plano, si se toma en cuenta que la longitud del arco “s”, descrita por el movimiento del plano, es igual a la longitud del arco al girar la polea, podemos expresar dicha relación de la siguiente forma:

$$\theta_1 R_1 = \theta_2 R_2 \quad (4.1)$$

Donde:

Símbolo	Descripción	Valor/Unidades
θ_1	Ángulo de inclinación del plano	rad
R_1	Radio de giro del plano	0.25m
θ_2	Posición angular de la polea	rad
R_2	Radio de la polea	0.0325m

Ya que R_1 , R_2 son conocidos y el ángulo de inclinación del plano será proporcionado por el controlador, despejando la posición angular de la polea de (4.1), se tiene que:

$$\theta_2 = \frac{\theta_1 R_1}{R_2} \quad (4.2)$$

Para la rotación de la polea se dispone de un motor de corriente directa, este dispositivo es capaz de transformar la energía eléctrica en energía mecánica. Podemos clasificarlos en tres categorías:

- **Motor de corriente directa:** Se compone principalmente de dos partes: rotor y estátor. El rotor es una pieza giratoria compuesta por una serie de espiras de cobre por las cuales circula una corriente eléctrica. El estátor está compuesto por imanes permanentes para crear un campo magnético en el cual está sumergido el rotor. La fuerza con la que gira el rotor (toque) está determinada por la corriente que circula a través de las espiras. Por ello, a mayor tensión, mayor corriente y mayor torque. La principal característica es la capacidad de regular la velocidad desde el vacío a plena carga. Tienen la desventaja de ser un tanto imprecisos para mantener el rotor en una posición.
- **Motor a pasos:** Está conformado por una serie de bobinas en el estátor y sobre el rotor van aplicados varios imanes permanentes. El torque del rotor (paso) es resultado de la atracción entre el polo de la bobina excitada en el estátor y el imán permanente en el rotor. Para que el rotor gire es necesario seguir una secuencia (pulsos) al energizar las bobinas en el estátor. Como el voltaje de excitación de las bobinas es constante, el toque del rotor es fijo. La característica de estos motores es el hecho de poder moverlos un paso por cada pulso que se les aplique. Estos son adecuados generalmente en sistemas que operan a baja aceleración y con carga estática. Una desventaja se presenta al sobrecargar el motor en un sistema de lazo abierto, el conocimiento de la posición del rotor se pierde y el sistema debe de ser reiniciado.
- **Servomotor:** Es un tipo especial de motor de corriente directa, tiene la capacidad de ubicar la posición angular del rotor en cualquier parte dentro de un rango de operación y mantenerla estable. Está integrado por un motor de corriente directa, un reductor de velocidad que adapta la velocidad y potencia mecánica del rotor, por lo general se utiliza una resistencia variable para conocer la posición angular del eje de salida y un circuito de control en lazo cerrado. El margen de operación de estos dispositivos suele estar limitado a 180° aproximadamente. Para mantener la posición angular del rotor es necesario suministrarle al motor una señal modulada por anchura de pulso (PWM, Pulse Width Modulation), donde el ancho del pulso en alto es proporcional a la posición angular del rotor.

Tras haber conocido el funcionamiento de los tipos de motores de corriente directa, se seleccionó el servomotor como actuador para el mecanismo PE por su gran torque y estabilidad. El servomotor seleccionado es el **HS-805BB+ MEGA 1/4 SCALE SERVO** [19] de la empresa **Hitec RCD**, sus características eléctricas y mecánicas se muestran en la Tabla 4.1.

Todos los servomotores disponen de tres cables de entrada: el cable *rojo* es conectado a la fuente de alimentación, el *negro* va conectado a tierra y el *amarillo* suministra la señal

Sistema de control	Modulación por anchura de pulso, 1.5ms posición neutral.	
Voltaje de operación	4.8 V	6 V
Velocidad de operación	0.19s/60° sin carga	0.14s/60° sin carga
Torque	19.8kg.cm	24.7kg.cm
Ángulo de operación	Desplazamiento de 45° con un ancho del pulso de 0.4ms	
Dirección de desplazamiento	Giro en el sentido de las manecillas del reloj, el pulso incrementa de 1.5ms a 1.9ms	
Dimensiones	66x30x57.6mm	
Peso	152g	

Tabla 4.1: Características del servomotor Hitec HS-805BB.

de control al servomotor. Esta última es una señal cuadrada pulsante directa que se repite cada 20ms, ver Figura 4.3. La duración del pulso en nivel alto es proporcional a la posición angular del eje de salida del servomotor, por lo que existe una relación biunívoca entre la posición angular y la duración del pulso en alto. En nuestro caso el fabricante asegura que si los pulsos de la señal de control tiene una duración de 1.9ms, el eje del servomotor girara 45° con respecto a la posición neutral en el sentido de las manecillas del reloj. Con ello la relación de la inclinación del plano y la duración del pulso de la señal de control es la siguiente:

$$\text{Ancho del pulso de la señal de control} = \text{Posición neutral} + \left(\frac{\theta_1 R_1}{R_2} * \frac{0,4ms}{0,7854} \right) \quad (4.3)$$

la constante “0.7854” es la conversión a radianes de 45°, la posición neutral se encuentra en 1.5ms y el dominio de θ_1 es $[-0,04364rad, 0,04364rad]$ o $[-2,5^\circ, 2,5^\circ]$.

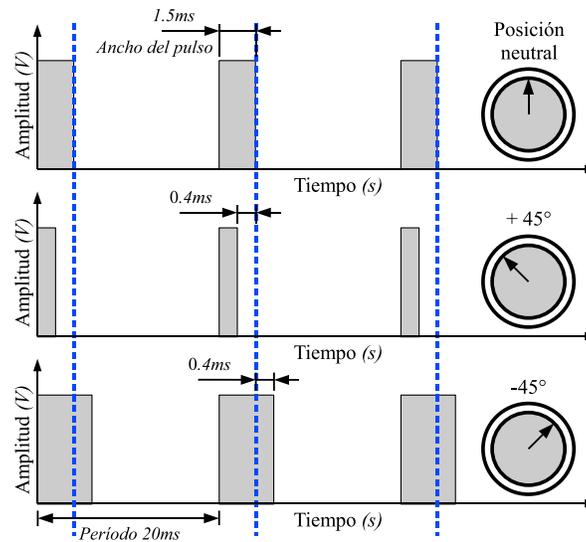


Figura 4.3: Señal de control de los servomotores.

4.2.2. Características de la captura de la imagen.

La cámara como sensor de la posición de la esfera, es una excelente opción ya que no es un sensor invasivo, esto quiere decir, que la medición de la posición no afecta la dinámica de la esfera, por ello se utilizó el sensor de imágenes de la tarjeta TRDB_DC2 [24] de la empresa Terasic. El sensor de la tarjeta, lee los datos de la imagen en un barrido progresivo de izquierda a derecha y de arriba hacia abajo, ver Figura 4.4, estos datos son sincronizados con ayuda de una señal de reloj de 25MHz (*PIXCLK*). Al ser un sensor de tipo CMOS tiene integrado convertidores analógico-digital, por lo que la salida de los datos son digitales.

$P_{0,0}$ $P_{0,1}$ $P_{0,2}$ $P_{0,3}$ $P_{0,n-1}$ $P_{0,n}$	00000 00000
$P_{1,0}$ $P_{1,1}$ $P_{1,2}$ $P_{1,3}$ $P_{1,n-1}$ $P_{1,n}$	00000 00000
Imagen válida	Borrado de imagen horizontal
$P_{m-1,0}$ $P_{m-1,1}$ $P_{m-1,n-1}$ $P_{m-1,n}$	00000 00000
$P_{m,0}$ $P_{m,1}$ $P_{m,n-1}$ $P_{m,n}$	00000 00000
000000000000000 000000000000000	00000 00000
000000000000000 000000000000000	00000 00000
Borrado de imagen vertical	
000000000000000 000000000000000	00000 00000
000000000000000 000000000000000	00000 00000

Figura 4.4: Lectura de la imagen.

A la imagen son agregados datos en negro (borrado de imagen horizontal y vertical), con el objetivo de ajustar el número de imágenes que leerá la cámara en un segundo. Por ello, la tarjeta proporciona las señales *Line_valid*, *Frame_valid* y el dato de la imagen puede ser leído a la salida del buffer *Dout* de 10Bits, el cual actualiza su valor cada periodo de la señal *PIXCLK*. La lectura de los dato de la imagen debe efectuarse en cada flanco de subida del *PIXCLK* y durante el nivel lógico en alto de *Line_Valid* y *Frame_valid* que es cuando ocurre una línea válida e imagen válida, ver Figura 4.5. En la Figura 4.6 se ilustra el comportamiento de las señales *Line_Valid* y *Frame_Valid*, donde el intervalo de tiempo *L* corresponde a la duración de una línea válida, *B* son datos en negro y *P* es tiempo muerto con duración de 6 ciclos de *PIXCLK* para indicar el inicio y fin de una imagen válida.

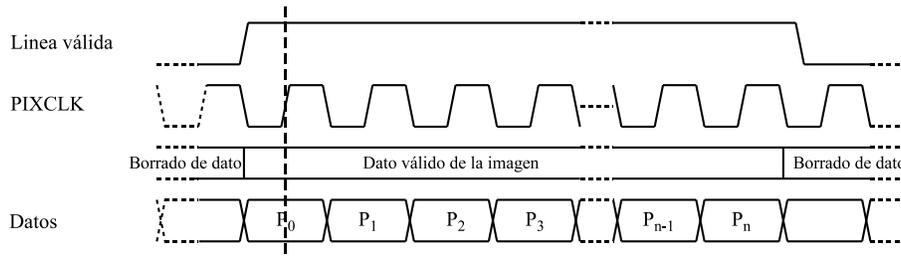


Figura 4.5: Adquisición de una línea de la imagen.

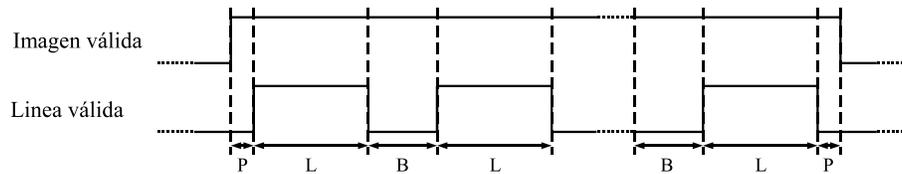


Figura 4.6: Adquisición de una imagen.

4.2.3. Implementación del tratamiento digital de imágenes y control en tiempo real.

Para el presente trabajo de tesis se contemplo la posibilidad de insertar tanto el TDI y el controlador difuso del sistema PE en un chip (FPGA), dadas sus bondades como lo son su inherente flexibilidad, desempeño y alto nivel de integración. A diferencia de integrar todos los componentes de una computadora en un chip, en donde aún persiste el problemas del procesamiento de forma secuencial, es posible gracias al lenguaje de descripción de hardware (en nuestro caso **VHDL**), mimetizar el comportamiento de cada sistema y modelarlo para que un circuito digital lo realice. Esta opción tiene la gran ventaja de combinar tanto la capacidad de procesamiento en serie y paralelo en un dispositivo dedicado y con ello disminuir el tiempo de procesamiento.

Es por ello que se utilizó la tarjeta de desarrollo Altera DE2 [6], la cual permite implementar desde un simple circuito hasta un proyectos multimedia, ya que cuenta con diferentes periféricos como lo son: decodificador de TV (NTSC/PAL), convertidos digital-análogo VGA, conector PS/2 teclado/ratón, modulo LCD de 16x2 líneas, toggle switches, push-button switches, leds, modulos de memoria SDRAM y Flash, controladores de Ethernet y USB, IrDA transceiver, headers de expansión de 40-pin y el elemento más importantes, un FPGA Cyclone II 2C35. Además de los 33216 **LEs** con los que cuenta el FPGA, encontramos elementos que son de nuestro interés como:

- Bloques de RAM. Estos bloques nos permiten instanciar hasta 483,840 bits de memoria, útiles para el TDI y almacenar las funciones de pertenencia del fuzificador.
- Multiplicadores. Se cuenta con 35 multiplicadores con entradas de 18bits, de los cuales disponemos para el cálculo de la posición de la esfera y en la defuzificación.
- Manejador de reloj digital (PLLs, phase-locked loops). 4 PLLs permiten cada uno multiplicar, dividir, desplazar la fase y programar el ciclo útil del reloj del sistema. La tarjeta DE2 cuenta con dos relojes de sistema, uno de 50MHz y otro de 27MHz, para poder sincronizar los módulos del sistema PE insertados en el FPGA, con el sensor digital de imágenes, se dividió la señal de reloj de 50MHz para obtener una de 25MHz, que es la frecuencia máxima en que se transmiten los datos de la imagen.
- Bloques de Entrada/Salida. Permiten el flujo bidireccional de datos entre las funciones lógicas programadas por el usuario en el FPGA y los dispositivos externos. De los 475-pins disponibles para el usuario en el FPGA Cyclone II 2C35, 72-pins son de propósito general en la tarjeta DE2. En este trabajo de tesis, se utilizaron 2-pins para envían

las señales de control a los servomotores y 16-pins con la tarjeta TRDB_DC2, de los cuales 13-pins son para la adquisición de la imagen, 2-pins para configurar el sensor de imágenes y 1-pin para el reloj maestro.

4.3. Sistema de tratamiento digital de imágenes.

Debido a que el mayor tiempo de procesamiento lo consume la tarea de TDI en aplicaciones en tiempo-real, resulta difícil la ejecución de éstos en equipos con procesadores de uso general [3] y de los cuáles sólo se tiene acceso a través de un sistema operativo. Una opción para reducir el tiempo de ejecución de los procesos de TDI, es formular un procesamiento concurrente y secuencial utilizando herramientas de diseño de hardware. Además de que se cuenta con la posibilidad de probar físicamente el funcionamiento mediante su equivalente circuito lógico embebido, en un circuito integrado que resulta de la tecnología **FPGA**.

4.3.1. Función de umbral.

Es una de las transformaciones más sencillas que se puede aplicar a una imagen en niveles de grises $I(u,v)$ para obtener una imagen $TH(u,v)$ en dos tonos (blanco y negro). También es una técnica de segmentación, ya que divide la imagen en subregiones. El valor de umbral B es el que decide cuales puntos de la imagen pasan al negro o al blanco.

La elección del valor de umbral B depende de la complejidad de la imagen, ya que puede haber uno o varios puntos de umbral dependiendo de la segmentación que se quiera conseguir.

La expresión matemática de la transformación para este sistema es la siguiente:

$$TH(u,v) = f(I(u,v)) = \begin{cases} 0, & 0 \leq I(u,v) \leq B \\ 1, & B < I(u,v) \leq 1023 \end{cases} \quad (4.4)$$

Con el fin de facilitar el procesamiento de la imagen, se consideró la esfera de color blanco y el plano de color negro para resaltar el objeto de interés del fondo, ver Figura 4.7, por lo que sólo se necesita un valor de umbral B para poder segmentar la imagen y representar los **componentes conexos** de la esfera con $th(u,v) = 1$ y el fondo con $th(u,v) = 0$.

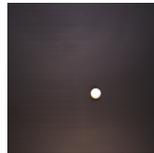


Figura 4.7: Imagen $I(u,v)$, esfera sobre el plano.

La función de umbral fue implementada en el **VHDL** [22], que es un lenguaje mediante el cual es posible describir un circuito digital. Con ayuda del lenguaje, se tiene la posibilidad de abordar la descripción de una idea o diseño de forma estructural o comportamental [28]. La descripción estructural consiste en enumerar los componentes de un circuito y su interconexión. Con la descripción comportamental, es factible diseñar un circuito electr

ónico a partir de su comportamiento, en vez de los elementos de los que está conformado.

Para el caso de la función de umbral, se optó por la descripción comportamental mostrada en la Figura 4.8. Los píxeles leídos de la cámara mayores que el valor de umbral B , corresponden al objeto y por debajo del umbral al fondo de la escena, lo que resultó en un circuito comparador combinatorio de 10-bits.

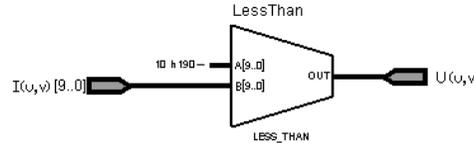


Figura 4.8: Comparador de 10-bits

4.3.2. Operador morfológico de erosión.

La morfología se refiere al estudio de las formas y estructuras [18]. La morfología matemática descansa en la teoría de conjuntos, en el campo del procesamiento de imágenes, es empleada para la extracción de componentes, a su vez, es útil en la representación y descripción de la forma de una región.

Algunas de las operaciones básicas en teoría de conjuntos son la traslación, reflexión y complemento. Por lo que nos permitimos definir la región TH como un subconjunto de Z^2 y $th = th(u, v)$ como el conjunto de elementos de dicha región, por lo tanto:

La *traslación* de TH por un elemento $a = a(m_1, n_1)$ se define como:

$$(TH)_a = \{g \mid g = th + a, \forall th \in TH\} \quad (4.5)$$

La *reflexión* de TH se define como:

$$\hat{TH} = \{g \mid g = -th, \forall th \in TH\} \quad (4.6)$$

El *complemento* de TH se define como:

$$(TH)^c = \{g \in Z^2 \mid g \notin TH\} \quad (4.7)$$

La morfología matemática fundamenta todas sus operaciones en dos operadores: la erosión y la dilatación, estos operadores están asociados con un conjunto llamado elemento de estructura. Para el caso del procesamiento de imágenes binarias, el objetivo de la erosión es decrecer la región de interés a través de un proceso de eliminación de elementos, el tamaño y forma del conjunto erosionado dependerá fuertemente del tamaño y forma del elemento de estructura que se seleccione.

Para formalizar lo dicho anteriormente, consideramos a TH y W subconjuntos de Z^2 . La erosión de TH con respecto del elemento de estructura W se define como:

$$TH \ominus W = \{th \mid W_{th} \subseteq TH\} \quad (4.8)$$

La operación morfológica de erosión, es el resultado de comprobar sí el elemento de estructura W , pertenece completamente al conjunto TH , donde su origen es colocado en th .

Si consideramos la imagen resultante de la función de umbral mostrada en la Figura 4.9, esta presentan píxeles que no pertenecen a la esfera, dichos píxeles podrían ser objetos en la escena sí no conociéramos el origen de la imagen, pero en realidad son producto del reflejo de la iluminación en el plano. La imagen para ser procesada (cálculo del centroide de la esfera), requiere de eliminar todos los píxeles que no pertenecen al objeto de interés.



Figura 4.9: Imagen binaria resultado de la función de umbral

Descripción del operador erosión [10]: Es una operación que elimina los píxeles conexos con alguno que son parte del fondo, lo que supone eliminar una línea del contorno del objeto, esto en su forma más simple. El resultado de la erosión puede ser expresado en la siguiente ecuación, usando los índices “ m, n ” como la posición de la componente conexas en la r -vecindad de la celda ‘ $th(u, v)$ ’.

$$S(u, v) = \sum_{-r \leq m \leq r} \sum_{-r \leq n \leq r} w(m, n) th(u + m, v + n) \quad (4.9)$$

$$Q(u, v) = f(S(u, v)) = \begin{cases} 0, & S(u, v) \neq i \\ 1, & S(u, v) = i \end{cases} \quad (4.10)$$

Donde ‘ $S(u, v)$ ’ y ‘ $Q(u, v)$ ’ es el estado y salida resultante de la celda $th(u, v)$. ‘ $th(u, v)$ ’ es una componente de la imagen binaria proveniente de la función de umbral (4.4), para $0 \leq u \leq C$ y $0 \leq v \leq F$, donde ‘ C ’ es el número de columnas y ‘ F ’ el número de filas de la imagen. ‘ i ’ es igual al número de pesos $w(m, n)$ con valor diferente de cero. ‘ r ’ es la máxima distancia entre la celda de referencia $th(u, v)$ y sus vecinos $th(u + m, v + n)$. Si ‘ $r = 1$ ’, los índices (m, n) forman el siguiente conjunto $\{(-1, -1), (1, -1), (-1, 1), (1, 1), (0, -1), (0, 1), (-1, 0), (1, 0), (0, 0)\}$ y el elemento estructurante W , es definido con respecto a $w(m, n)$ como sigue:

$$W = \begin{bmatrix} w(-1, -1) & w(-1, 0) & w(-1, 1) \\ w(0, -1) & w(0, 0) & w(0, 1) \\ w(1, -1) & w(1, 0) & w(1, 1) \end{bmatrix} \quad (4.11)$$

$w(m, n)$ son los pesos de control en la ecuación (4.9) que dan forma al elemento estructurante. El elemento estructurante (4.11) toma la forma y pesos de la ecuación (4.12) para realizar la tarea de erosión con $i = 8$ y peso de referencia en $w(0, 0)$.

$$W = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (4.12)$$

Comprendida la transformación de erosión expresada en (4.9) y (4.10), ahora puede ser descrita de forma estructural en VHDL. Para iniciar se define la siguiente notación:

$$\begin{aligned}
 A &= th(u - 1, v - 1) & B &= th(u - 1, v) & C &= th(u - 1, v + 1), \\
 D &= th(u + 0, v - 1) & E &= th(u, v) & F &= th(u + 0, v + 1) \\
 G &= th(u + 1, v - 1) & H &= th(u + 1, v) & I &= th(u + 1, v + 1), \\
 Q &= q(u, v)
 \end{aligned}$$

La Tabla de verdad 4.2, es la interpretación de la ecuación (4.10), para todos los posibles casos en que el elemento de estructura W recorre la imagen TH y de esta manera poder identificar la función Booleana correspondiente al operador morfológico de erosión, el cual es el operador lógico AND (4.13).

Entradas									Q
A	B	C	D	E	F	G	H	I	Q
1	1	1	1	1	1	1	1	1	1
*	*	*	*	*	*	*	*	*	0

* No importa

Tabla 4.2: Tabla de verdad de $Q(u, v) = f(S(u, v))$

$$Q = ABCDEFGHI \tag{4.13}$$

A partir de (4.13), se puede implementar el circuito digital equivalente mostrado en la Figura 4.10 en un FPGA, lo que resulta en disminuir el costo computacional. Por otro lado un pre-procesamiento de imágenes binarias de este tipo, requiere de la convolución y un comparador en un equipo convencional (PC's), además de no contar con el ambiente concurrente que tiene VHDL.

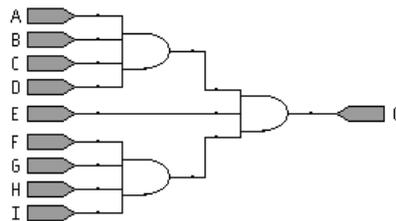


Figura 4.10: Circuito equivalente de la operación morfológica de erosión.

Lo antes mencionado se puede ejemplificar con la simulación de esta operación morfológica, ver Figura 4.11, la cual esta basada en el software Quartus II 7.2 que proporciona el fabricante de la tarjeta con la que se trabajó. La simulación se realizó con una imagen de muestra de 8x8 píxeles y un píxel de contorno en la imagen con valor igual a '0' ilustrado por la región de color gris, el cual es agregado por software.

Debido a que el elemento captor de la imagen es leído en un barrido progresivo, para poder iniciar con la erosión de la imagen, es necesario tener la capacidad de almacenar 3 filas de esta en un registro de corrimiento, pero para poder operar el elemento de estructura W al primer elemento de la imagen $th(0,0)$, solo se requiere tener almacenadas las dos primeras filas y los dos primeros elementos de la tercer fila. Como se observa en la simulación de la Figura 4.11, las señales de la A a la I corresponden al elemento de estructura W , enmarcadas en un rectángulo negro, donde las 9 están almacenadas al mismo tiempo en un ciclo de reloj, por lo que en el siguiente ciclo de reloj se obtiene el resultado Q de la operación lógica AND , lo que hace de esto un proceso sincrónico debido a la naturaleza del elemento captor.

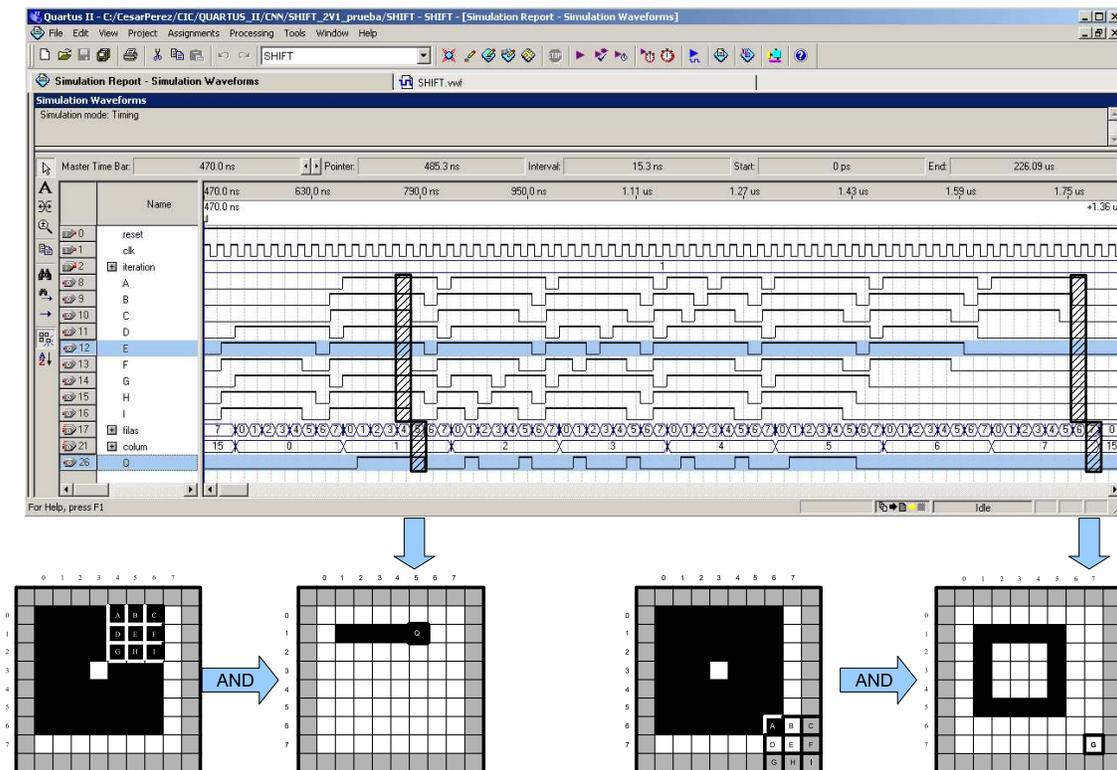


Figura 4.11: Simulación de la operación morfológica de erosión en un FPGA.

4.3.3. Cálculo del centroide

Este es el último módulo del sub-sistema de TDI, encargado de calcular el centroide de la esfera que se encuentra en el plano. Una vez que la imagen está libre de objetos que no pertenezcan a la esfera, es posible calcular su centroide. Los datos de la imagen que provienen del módulo de erosión se leen en un barrido progresivo de izquierda a derecha y de arriba hacia abajo, además se utilizó un contador para las columnas y uno para las filas, por lo tanto, nos permitimos tomado estos dos como ejes de referencia (x, y) . Las coordenadas del centroide de la esfera representada por una región R de **componentes conexos** $q(u, v) = 1$, pueden ser calculadas mediante los momentos geométricos estándar $m_{0,0}$, $m_{0,1}$, $m_{1,0}$ [11] de la forma siguiente:

$$Eje\ x(k) = \frac{m_{1,0}}{m_{0,0}} = \frac{\sum_{(u,v) \in R} uq(u,v)}{\sum_{(u,v) \in R} q(u,v)} \quad (4.14)$$

$$Eje\ y(k) = \frac{m_{0,1}}{m_{0,0}} = \frac{\sum_{(u,v) \in R} vq(u,v)}{\sum_{(u,v) \in R} q(u,v)} \quad (4.15)$$

El centroide de la esfera será proporcionado al sub-sistema de control, que calculará la inclinación necesaria para que la esfera ruede en la dirección apropiada y se estabilice en la posición deseada.

4.4. Control del sistema Plano-Esfera.

La posición de la esfera sobre el plano en la dirección de control X y la dirección de control Y es proporcionada por el sub-sistema de TDI. Como se propuso en las ecuaciones 3.53, suponemos la interacción entre las direcciones de control desacopladas, por lo tanto, solo se explicara el diseño del controlador difuso en la dirección X , ver Figura 4.12.

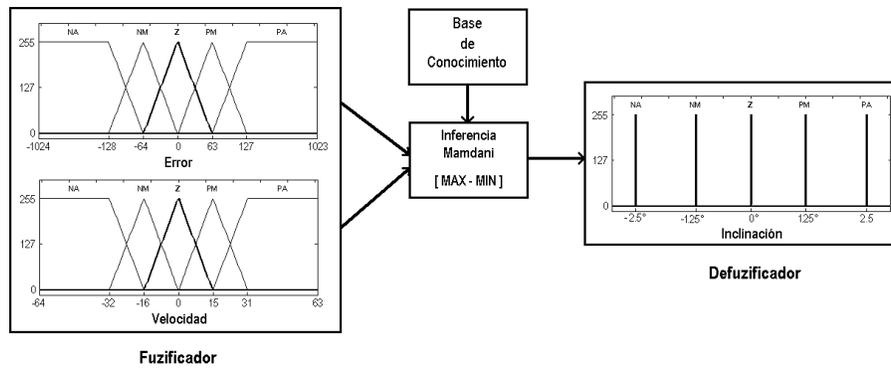


Figura 4.12: Diagrama del controlador difuso.

4.4.1. Fuzificador

Si tomamos a T como el tiempo de **muestreo** (20 imágenes por segundo o $50mseg$), refiriéndonos al tiempo en que el sub-sistema de TDI tarda en capturar y procesar una imagen, la posición de la esfera $x(k)$, será la posición en el tiempo de **muestreo** k . Por lo tanto, se puede aproximar el cálculo de la velocidad $v(k)$ y la aceleración $a(k)$ a partir de (4.16) y (4.17) respectivamente. Si suponemos que la velocidad y aceleración es constante para un tiempo de **muestreo** T pequeño, de esta forma podemos estimar la siguiente posición de la esfera $x_e(k+1)$ con (4.18). Llamamos a la posición deseada x_d y al error estimado $e(k+1)$, que es expresado mediante (4.19). El error estimado y la velocidad de la esfera

son considerados como entradas para el fuzificador.

$$v(k) = \frac{(x(k) - x(k-1))}{T} \quad (4.16)$$

$$a(k) = v(k) - v(k-1) \quad (4.17)$$

$$x_e(k+1) = x(k) + \frac{v(k)}{T} + \frac{a(k)T^2}{2} \quad (4.18)$$

$$e(k+1) = x_d - x_e(k+1) \quad (4.19)$$

Para fuzificar las **variables lingüísticas** de entrada, error estimado $e(k+1)$ y velocidad $v(k)$, se utilizaron cinco conjuntos difusos por entrada, los cuales son: negativa alta (NA), negativa media (NM), cero (Z), positiva media (PM) y positiva alta (PA). NA y PA son funciones de pertenencia trapezoidales (1.3). NM , Z y PM son funciones de pertenencia triangular (1.4), que están distribuidas como se muestran en las Figura 4.13.

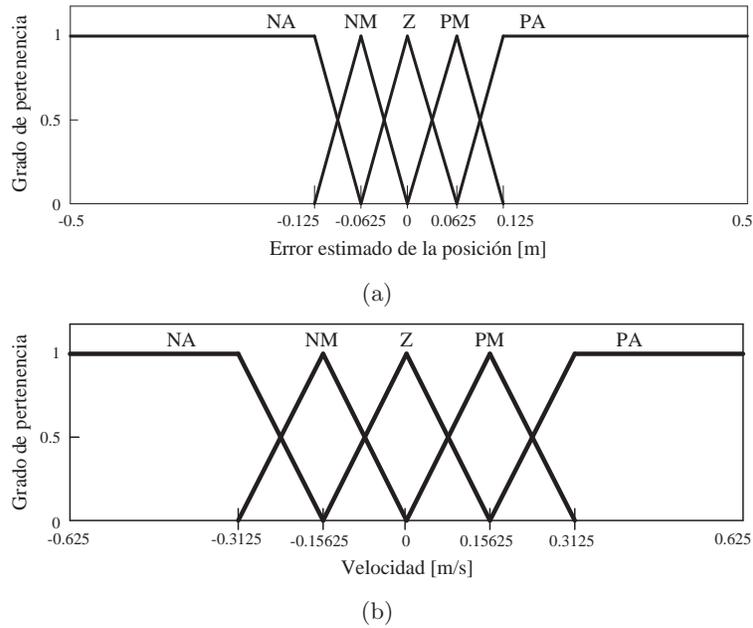
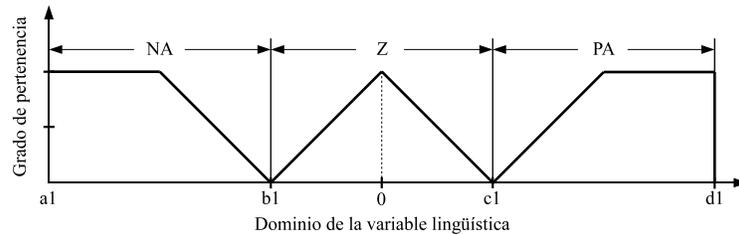


Figura 4.13: Distribución de los conjuntos difusos de entrada

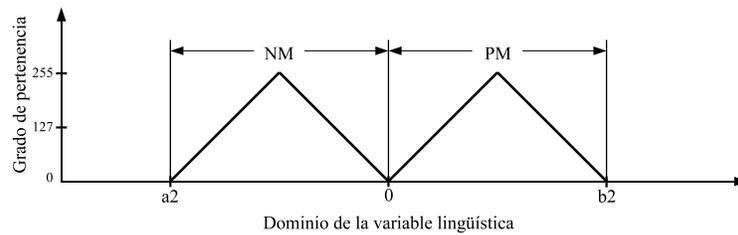
Para poder mapear las mediciones de error estimado y velocidad de la esfera a su correspondiente grado de pertenencia de los conjuntos difusos e iniciar el proceso de inferencia, las funciones de pertenencia fueron discretizadas y almacenadas en memorias ROM embebidas en el **FPGA**.

La forma en que se almacenó en las memorias ROM cada función de pertenencia de los conjuntos difusos, ver Figura 4.14, es la siguiente: ya que la correspondencia entre una localidad de memoria con una dirección de memoria es biunívoca, en una memoria se almacenaron los conjuntos disjuntos NA , Z y PA , por la misma razón, en una segunda memoria se almacenaron NM y PM , esto para cada variable lingüística. Con ayuda de

comparadores determinamos el grado de pertenencia correspondiente a cada conjunto difuso. Donde la dirección de memoria representa el valor de la variable lingüística de entrada y la localidad o bloque de memoria representa el grado de pertenencia al correspondiente conjunto difuso.



(a) Memoria uno



(b) Memoria dos

Figura 4.14: Conjuntos difusos almacenados en memoria ROM

Ya que trabajamos con las coordenadas de la imagen, el universo de discurso de $(e(k+1), v(k))$ queda de la siguiente forma: el error máximo que puede ocurrir dada la posición inicial y final de la esfera es de $[-0.5, 0.5\text{m}]$, su equivalente en **píxeles** es de $[-1024, 1023]$ o 2^{11} píxeles signados. Por su parte, el desplazamiento máximo de la esfera en una muestra de tiempo $v(k)$, considerada la inclinación máxima del plano de 2.5° , fue de 36 **píxeles**, partiendo de esta medición experimental se decidió considerar el universo de discurso de $v(k)$ el intervalo $[-64, 63\text{píxeles}]$ o 2^6 **píxeles** signados. La magnitud del grado de pertenencia de los conjunto difusos es el intervalo $[0, 255]$, donde “0” indica no pertenencia y “255” pertenencia.

Determinación del grado de pertenencia.

La memoria ROM mapea una dirección con un bloque de memoria, en nuestro caso es imprescindible determinar a qué conjunto difuso corresponde el grado de pertenencia que se acceso. Por lo anterior es necesario utilizar un circuito comparador, el cual determina, dado el valor de la variable lingüística, el grado de pertenencia a cada conjunto difuso. Esto se ilustra en la Figura 4.15, donde solo se ejemplifica el conjunto PM, pero lo mismo ocurre para los demás conjuntos.

El rango de valores al cual pertenece el conjunto difuso NM está entre $[a2, 0)$ y el correspondiente para el conjunto PM es $[0, b2]$, ver Figura 4.14(b). Para determinar el

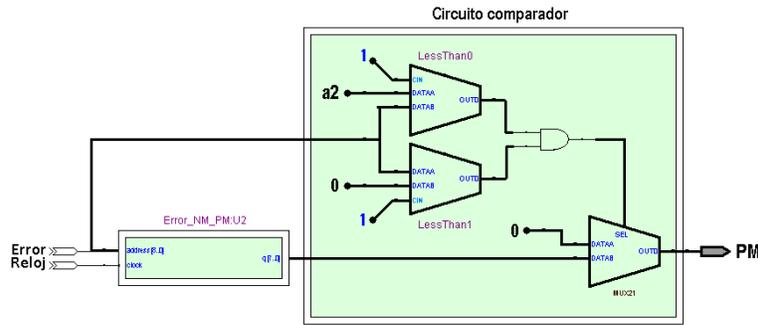


Figura 4.15: Determinación del grado de pertenencia al conjunto difuso PM.

grado de pertenencia al conjunto difuso PM , la medición del error estimado corresponde a una dirección de la memoria ROM, la memoria accesa a dicha localidad que arroja un grado de pertenencia, pero aun no se a determinado a que conjunto difuso corresponde, por lo que se utilizan dos circuitos comparadores combinatorios, si el valor de error estimado se encuentra entre el intervalo $0 \leq error \leq b2$, se indica por medio de una bandera lógica la igualdad o desigualdad, donde un multiplexor determina, si el error se encuentra entre el intervalo del conjunto difuso PM , su grado de pertenencia será el valor a la salida de la memoria ROM, de lo contrario el valor de PM será cero. Esto se hace para todos los terminos de las variables lingüísticas, ver Figura 4.16.

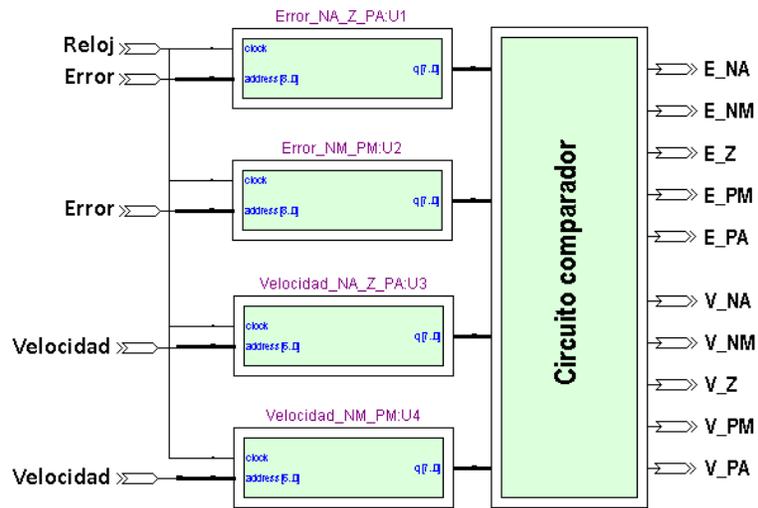


Figura 4.16: Fuzificador empleando memorias ROM

Algunas alternativas para el diseño del fuzificador es almacenar cada conjunto difuso en una memoria ROM, con lo que se evitan los circuitos comparadores pero se incrementan los recursos utilizados en el **FPGA**, otra opción es emplear multiplicadores y divisores embebidos en el **FPGA** para construir las funciones de pertenencia, una desventaja de esta última opción es que se limita a utilizar funciones de pertenencia triangulares y trapezoidales, además de que el tiempo de procesamiento incrementa, ya que acceder a una localidad de memoria consume menor tiempo de procesamiento que una multiplicación

o división.

4.4.2. Base de conocimiento

Para el control del movimiento de la esfera se usó el cálculo del error estimado y velocidad como características cinemáticas, ya que el error estimado por si solo no aporta suficiente información, en cambio, la velocidad indica el sentido y magnitud del movimiento de la esfera. Estas variables son suficientes para conocer el estado de la esfera sobre el plano, por lo tanto basándonos en el comportamiento obtenido de la experimentación con el sistema PE en forma manual, ver Figura 4.17, encontramos un conjunto de reglas básicas que se deben de respetar para que el sistema opere de forma correcta.

1. Si el **Error** de posición es negativo (N) y la **Velocidad** es positiva (P), entonces la **Inclinación** del plano será positiva (P).
2. Si el **Error** de posición es positivo (P) y la **Velocidad** es negativa (N), entonces la **Inclinación** del plano será negativa (N).
3. Si el **Error** de posición es cero (Z) y la **Velocidad** es cero (Z), entonces la **Inclinación** del plano será cero (Z).

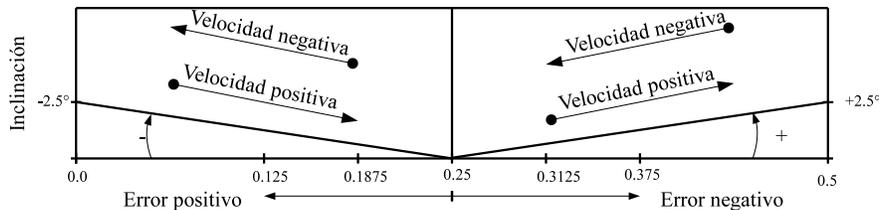


Figura 4.17: Dinámica del sistema PE.

Pero las reglas no son tan claras en los casos: **Error** positivo (P) y **Velocidad** positiva (P) o **Error** negativo (N) y **Velocidad** negativa (N), ya que la **Inclinación** del plano dependerá de la velocidad y la cercanía entre la posición de la esfera y la posición deseada. Algunos ejemplos serían cuando la esfera se encuentre *cerca* de la posición deseada y la velocidad sea *alta*, en esta situación el plano debe inclinarse para reducir la velocidad de la esfera, por el contrario, cuando la esfera se encuentra *alejada* de la posición deseada y la velocidad es *muy baja*, entonces el plano se debe de inclinar para incrementar la velocidad de la esfera. Como podemos observar, hemos introducido términos tales como *alta*, *muy baja*, *cerca*, *lejos*, para calificar las variables de entrada y apoyarnos en estos términos hizo posible determinar correctamente el valor de la inclinación del plano. Planteadas estas consideraciones se ideó la base de conocimiento, donde se utilizaron los conjuntos difusos *NA*, *NM*, *Z*, *PM* y *PA* como términos de las variables lingüísticas *Error estimado* y *Velocidad*, ver Tabla 4.3. Como observación el conjunto de reglas presenta una división simétrica para los conjuntos difusos de salida, en la esquina inferior izquierda se agrupan los conjuntos

difusos con valores negativos, en la esquina superior derecha los conjuntos difusos con valor positivo y en la parte central se encuentra una transición suave entre ellos, ya que va de valores máximos negativos, medios negativos, cero, medios positivos y máximos positivos, con ello se buscan movimientos sutiles para la inclinación del plano.

<i>Inclinación</i>		Velocidad				
		NA	NM	Z	PM	PA
Error estimado	NA	<i>NA</i>	<i>PM</i>	<i>PM</i>	<i>PA</i>	<i>PA</i>
	NM	<i>NA</i>	<i>NM</i>	<i>PM</i>	<i>PA</i>	<i>PA</i>
	Z	<i>NA</i>	<i>NA</i>	<i>Z</i>	<i>PA</i>	<i>PA</i>
	PM	<i>NA</i>	<i>NA</i>	<i>NM</i>	<i>PM</i>	<i>PA</i>
	PA	<i>NA</i>	<i>NA</i>	<i>NM</i>	<i>NM</i>	<i>PA</i>

Tabla 4.3: Conjunto de reglas basadas en un experto.

4.4.3. Mecanismo de inferencias

Obtenido el conocimiento lingüístico del comportamiento del controlador (base de conocimiento), debe de ser procesado para generar una salida difusa, por ello se utilizó el mecanismo de inferencias MAX-MIN (Mamdani), ya que este método se adecua a la lógica interna del FPGA. Para obtener el consecuente de cada regla difusa, se determina el valor mínimo de los antecedentes de la regla (grado de pertenencia de las variables difusas de entrada), posteriormente se obtiene el valor máximo de todas aquellas reglas que tienen como consecuente el mismo conjunto difuso de salida, ver Figura 4.18. Para la implementación en el FPGA del mecanismo de inferencia es posible elegir entre una arquitectura en paralelo, donde se reduce el tiempo de procesamiento de está tarea, teniendo como secuela un

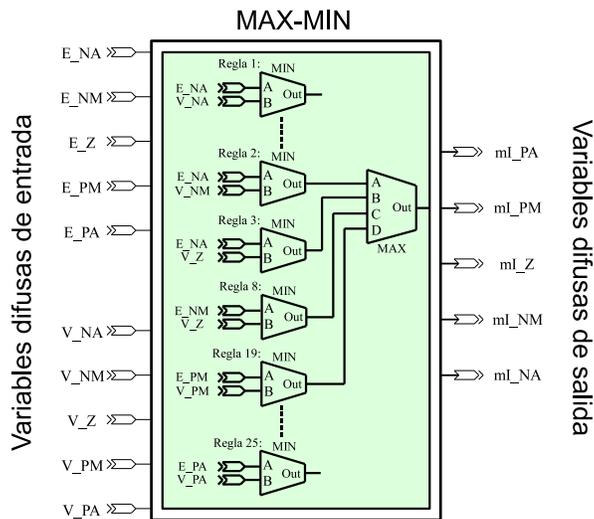


Figura 4.18: Mecanismo de inferencia MAX-MIN

incremento en el uso de recursos del FPGA o una arquitectura en serie, donde se incrementa el tiempo de procesamiento y es posible reducir el uso de recursos. En nuestro caso, existe una delgada línea en la elección de la arquitectura adecuada para resolver esta tarea, ya que el número de reglas que se evalúan no es grande (25 reglas) y no requiere de operaciones aritméticas complejas, pero un punto importante es el resolver esta tarea en tiempo real y es por ello que se decidió utilizar la arquitectura en paralelo, ya que existe un retardo de tiempo considerable introducido por el sistema de tratamiento digital de imágenes.

Con la intención de ilustrar el funcionamiento del mecanismo de inferencia, se muestra en la Fig. 4.19 su simulación en Quartus II. Las unidades de los datos ilustrados en la simulación son en píxeles, la posición deseada se ubica en el centro del área de trabajo (25cm,25cm), que equivale a un valor de (511píxeles,511píxeles) en coordenadas de la imagen. Se simuló el movimiento de la esfera con un desplazamiento con respecto de la posición deseada, de izquierda a derecha y con velocidad constante “5 píxeles por muestra”, por lo que la aceleración es cero. Las señales involucradas en la simulación son:

- iClock: Señal de reloj del módulo, que comparte con el sistema TDI.
- iReset: Sincroniza los sistemas de TDI y de Control, durante el nivel lógico en alto de esta señal, se calcula el centro de masa de la esfera en coordenadas de la imagen en el sistema TDI y en el nivel lógico en bajo, el sistema de control calcula el error estimado así como la fuzificación, evalúa el conjunto de regla, defuzifica la información y la interpreta para poder realizar la acción de control.
- iEje: Es la posición de la esfera dentro del área de trabajo sobre el eje X, dado en coordenadas de la imagen. Bus de datos unidireccional de 10-bits de profundidad.
- oVelocidad: Desplazamiento de la esfera por muestra. Su rango de valores se encuentra entre [-64,63] píxeles o 7-bits de profundidad.
- oError: Cálculo del error estimado de acuerdo a las Ec. (4.18) y (4.19). Su rango de valores está entre [-1024,1023] píxeles u 11-bits de profundidad.
- omuE_PA ... omuE_NA: Grado de pertenencia de los conjuntos difusos de entrada de la variable lingüística “Error”. 8-bits de profundidad por cada variable.
- omuV_PA ... omuV_NA: Grado de pertenencia de los conjuntos difusos de entrada de la variable lingüística “Velocidad”. 8-bits de profundidad por cada variable.
- omuL_PA ... omuL_NA: Grado de pertenencia de los conjuntos difusos de salida de la variable lingüística “Inclinación”. 8-bits de profundidad por cada variable.

Para poder iniciar con el mecanismo de inferencias es necesario tener fuzificadas las variables lingüísticas de entrada, este proceso inicia cuando en la señal de iReset ocurre una transición de un nivel lógico alto a bajo, dicha transición indica que el sistema de TDI a finalizado el cálculo el centro de masa de la esfera en coordenadas de la imagen, por lo que transcurren 5 ciclos de la señal iClock en ser obtenidas las variables lingüísticas de entrada, dicho tiempo

<i>Inclinación</i>		Velocidad				
		NA	NM	Z	PM	PA
Error estimado	NA	0	0	0	0	0
	NM	0	0	16	16	0
	Z	0	0	170	85	0
	PM	0	0	0	0	0
	PA	0	0	0	0	0

Tabla 4.4: Reglas evaluadas para “*Error* = -4” y “*Velocidad* = 5”.

está representado entre los marcadores 10.02us y 10.22us, ver Fig. 4.19.

Una vez obtenidas las variables de entrada estas son fuzificadas y evaluadas por el mecanismo de inferencia, ver zona sombreada de la Fig. 4.19. Ya que el grado de pertenencia de cada conjunto difuso se asoció con una localidad de memoria ROM, la fuzificación consume un ciclo de la señal iClock, dejando el tiempo restante (dos ciclos de la señal iClock) hasta el marcador de +120.0ns, para evaluar las 25 reglas y permitir que los resultados arrojados se establezcan, esto último debido al proceso concurrente en que fue proyectado el mecanismo de inferencia.

Como es evidente en la Fig. 4.19, a partir de la marca de los 10.22us solo dos ciclos de la señal de iClock serían suficientes para poder obtener el grado de pertenencia de los conjuntos difusos de entrada y de salida, pero como sabemos, la simulación es solo una aproximación para describir el comportamiento del modelo real y en la práctica se requirió de un tiempo muerto (con longitud de un ciclo de la señal iClock) para asegurar la correcta lectura de los datos.

Para el caso de la simulación, la Tabla 4.4 muestra evaluadas las 25 reglas con la función MIN, donde las regiones en escala de gris representan los conjuntos difusos de salida. Aun evaluadas las reglas todavía no se ha determinado el valor de los conjuntos difusos de salida, puesto que se le han asignado varios, es por ello que a los consecuentes de las reglas que tengan en común el mismo conjunto de salida (regiones en escala de grises) se les aplica la función MAX y de esta manera es inferido su grado de pertenencia. También es posible observar en la Tabla 4.4 la activación de las reglas dado el movimiento de la esfera.

4.4.4. Defuzificación

Debido a que las salidas del controlador todavía son difusas, es necesario defuzificarlas, para contar con una señal de control que los actuadores puedan interpretar, el método que se utilizó fue el promedio de los centros (4.20), que básicamente opera como sigue: el grado de pertenencia de los conjuntos difusos NA ... PA están etiquetados como $\mu I_1 \dots \mu I_5$, donde estos valores son proporcionados por el mecanismo de inferencia con las variables $omuI_NA \dots omuI_PA$, ver Fig 4.19.

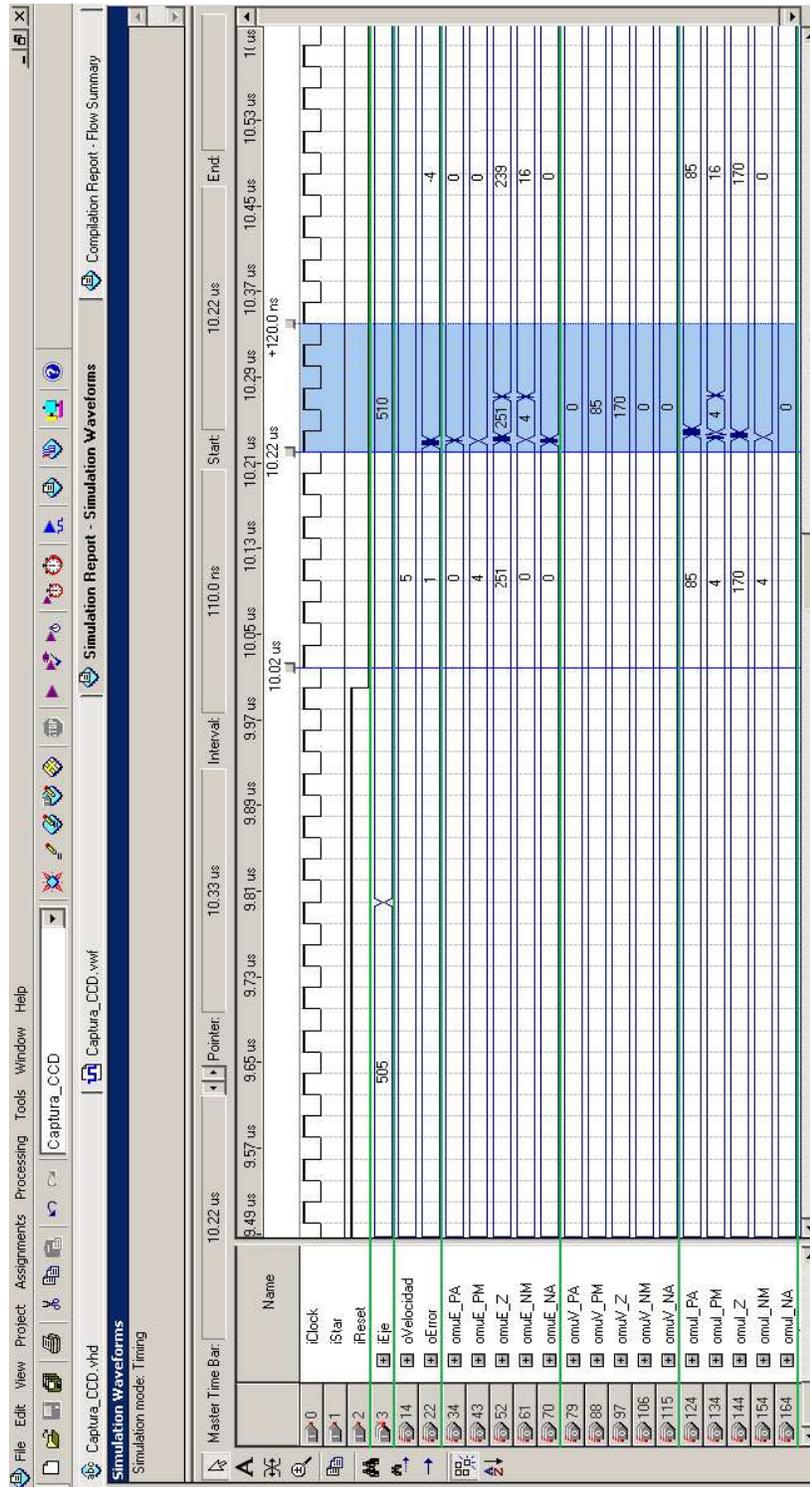


Figura 4.19: Simulación del Mecanismo de inferencia MAX-MIN.

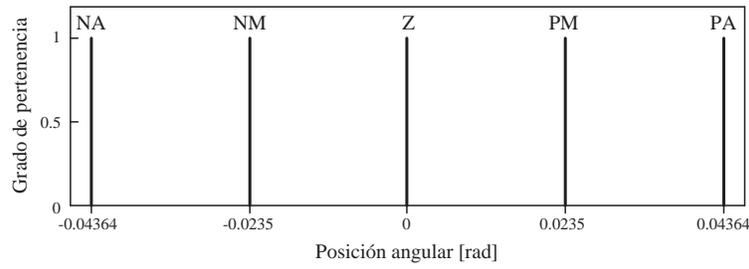


Figura 4.20: Distribución de los conjuntos difusos de salida

$$oqI = \frac{\sum_{i=1}^5 \mu I_i \cdot w_i}{\sum_{i=1}^5 \mu I_i} \quad (4.20)$$

Mientras que los valores de w_i son los puntos centrales de las funciones de pertenencia de los conjuntos difusos de salida, con: $w_1 = I_{NA} = -0,04364[rad]$, $w_2 = I_{NM} = -0,0235[rad]$, $w_3 = I_Z = 0,00001[rad]$, $w_4 = I_{PM} = 0,0235[rad]$ y $w_5 = I_{PA} = 0,04364[rad]$. La distribución y forma de las funciones de pertenencia de salida se ilustran en la Fig. 4.20. Como anteriormente se mencionó, la señal de control debe ser adecuada para accionar los actuadores, es por ellos que los valores de w_i están dados en ciclos de la señal iClock, tiempo equivalente al ancho del pulso de la señal de control estudiada en la sección 4.2.1: $w_1 = -4273$, $w_2 = -2300$, $w_3 = 1$, $w_4 = 2300$, $w_5 = 4273$.

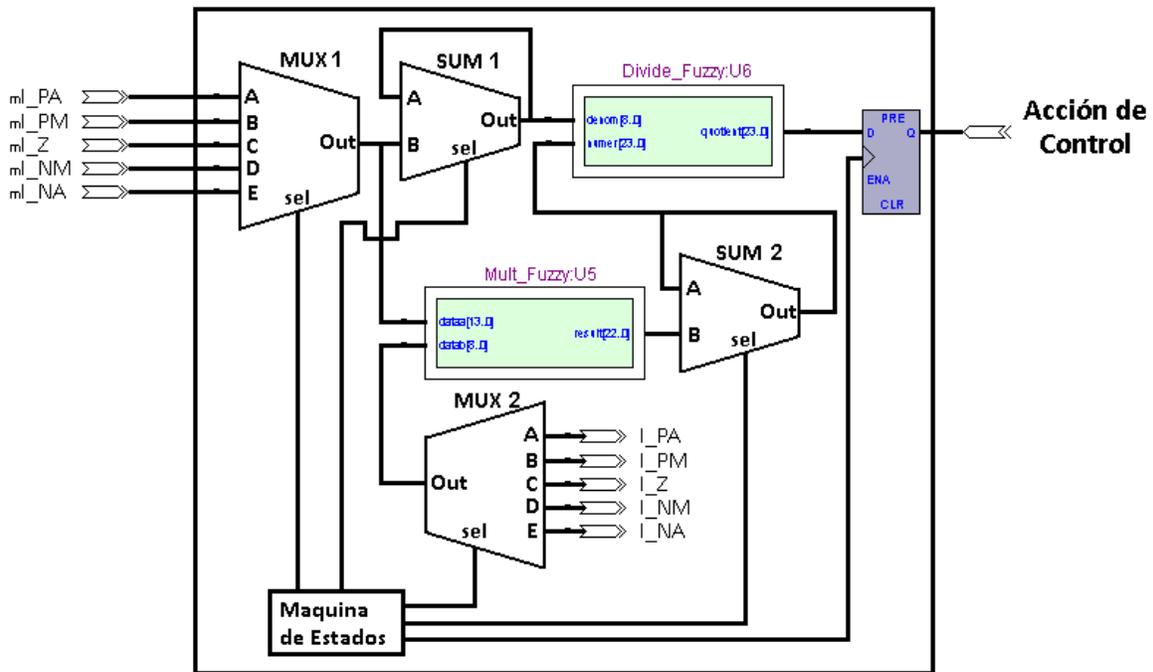


Figura 4.21: Defuzificador promedio de centros.

Dicho lo anterior, es posible proyectar el defuzificador de promedio de los centros en una arquitectura en paralelo o serie, si se considera que las operaciones aritméticas de multiplicación y división para realizar una implementación en paralelo de (4.20), consume importantes recursos del FPGA respecto a su contraparte serie y no existe una diferencia significativa en el tiempo de procesamiento, por estas razones se decidió utilizar la implementación serie mostrada en la Fig. 4.21. Donde el defuzificador se sincronizó con una máquina de estados, los componentes “MUX_2”, “Mult_Fuzzy:U5” y “SUM_2” conforman el numerador de (4.20), mientras que el componentes “SUM_1” constituye el denominador. La máquina de estados selecciona las entradas de los multiplexores “MUX_1 y MUX_2” desde “A” hasta “E”, en cada estado se realizan las operaciones “SUM_1 y SUM_2”, donde “SUM_1” representa la sumatoria de los grados de pertenencia de los conjuntos difusos de salida, por su parte “SUM_2” es la sumatoria del producto entre los grados de pertenencia y los centros de las funciones de pertenencia. Una vez procesadas las entradas al defuzificador, la máquina de estados habilita la salida del componente “Divide_Fuzzy:U6” que corresponde a la salida de control defuzificada.

En la Fig. 4.22 se muestra la simulación completa del controlador difuso, podemos apreciar entre las marcas $+320.0ns$ y $+729.114ns$ el procesamiento del componente defuzificador, donde las señales $oqL_{NA} \dots oqL_{PA}$ representan la multiplicación entre el grado de pertenencia de los conjuntos difusos de salida ($omuL_{NA} \dots omuL_{PA}$) y los centros de las funciones de pertenencia ($w_{-1} \dots w_{-5}$), oqL_{Prom_Num} y oqL_{Prom_Den} es el resultado del numerador y denominador de (4.20) y oqI es la salida defuzificada del controlador, la cual es interpretada por el componente que maneja los servomotores. El tiempo de procesamiento del componente defuzificador tiene una duración de $409.114ns$ y se aprecia su naturaleza secuencial en la forma que se obtienen los resultados de las señales $oqL_{NA} \dots oqL_{PA}$.

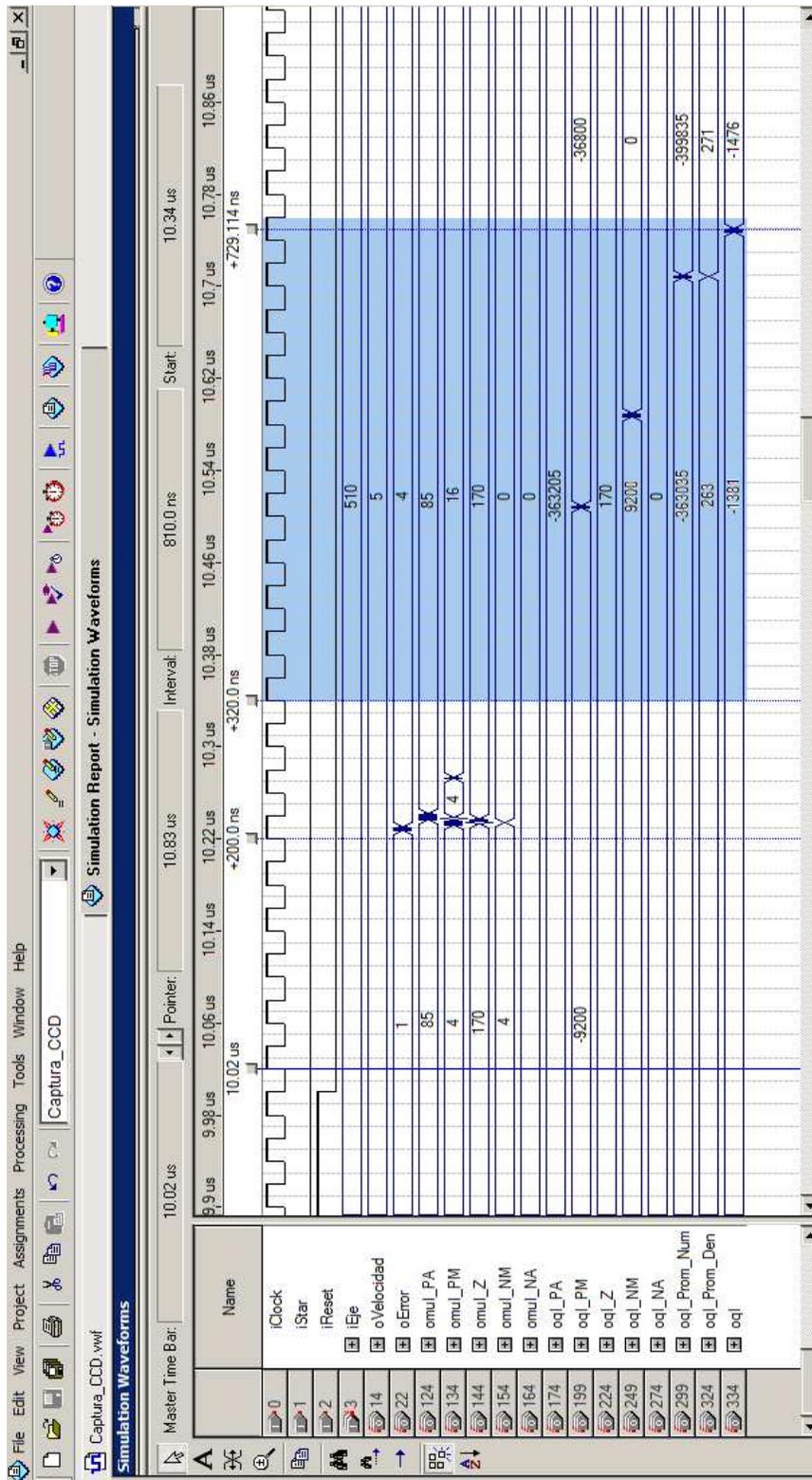


Figura 4.22: Simulación del componente Defuzificador.

Capítulo 5

Experimentos y resultados.

5.1. Módulo de visión

Para el sistema de visión se colocó el sensor digital de imágenes a una distancia fija sobre el plano de 0.638m. El tamaño del plano es de 0.5m x 0.5m, ver Figura 5.1. El **sensor digital de imágenes CMOS** modelo MT9M011, incorpora varias funciones de configuración, una de estas funciones especifica el tamaño de la imagen menor o igual al formato SXGA, la cámara fue configurada con una resolución de 1024H x 1024V para que la forma del plano coincida con la forma de la imagen, de esta manera se cubre toda el área de trabajo.

Debido al filtro de color Bayer [5] en el sensor, sería necesario interpolar para obtener los planos de color verde, rojo y azul, con los que se forma una imagen a color. Para evitar esto, se utilizó el fondo negro y la esfera de color blanco, ya que el color blanco es la superposición de todos los colores.

Se utilizó un **FPGA** Altera Cyclone II EP2C35F672C6 (el dispositivo se encuentra montado en la tarjeta DE2 de Altera Corporation) [6], en donde se implementaron las entidades que describen el comportamiento del módulo de visión. La primera tarea que realiza el módulo, es la de configurar el sensor digital de imágenes a través de una interfaz serie de dos vías.



Figura 5.1: Sistema plano-esfera

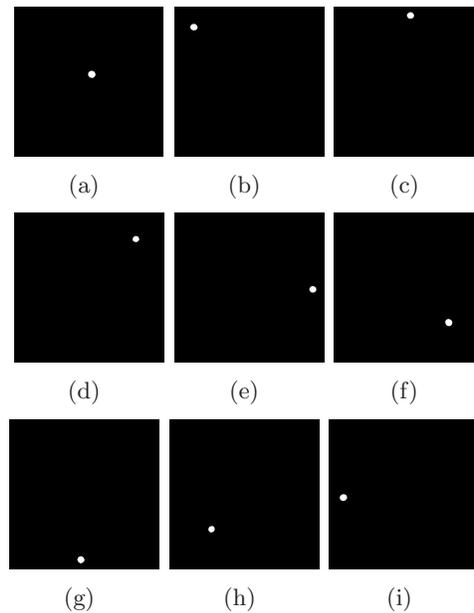


Figura 5.2: Imágenes del sistema plano-esfera procesadas por el módulo de visión

Una vez configurado el sensor, los datos de la imagen son enviados al **FPGA** en un barrido progresivo. Después, el módulo de captura se sincroniza con la cámara para recibir los datos y aplicar la función de umbral para segmentar. El ruido se elimina de la imagen a través de una transformación morfológica de erosión. Conforme este último módulo aplica el elemento estructurante a cada dato de la imagen, se detectan los **píxeles** que pertenecen a la esfera para determinar su centroide en el plano.

En la etapa de diseño de los componentes que conforman al módulo de visión, la imagen resultante de cada componente es almacenada en una memoria RAM embebida en el **FPGA**. Con ayuda de Quartus II Versión 7.2 y con la herramienta In-System Memory Content Editor, se visualizó la imagen y verificó su correcto funcionamiento. En la Figura 5.2, se muestran las imágenes resultantes del módulo de visión y en la Tabla 5.1, se presenta el cálculo de los centroides de algunas posiciones de la esfera sobre el plano.

5.2. Control difuso tipo Mamdani

5.2.1. Introducción

En capítulos previos se presentó el modelo de la dinámica del sistema PE y el procedimiento de diseño del control. Un controlador apropiado fue diseñado basado en el conocimiento del experto, con ayuda de simulaciones del modelo obtenido, se corroboró su funcionamiento y en consecuencia se probó el desempeño del controlador en el modelo real.

Los experimentos mostrados en esta sección se clasificaron de acuerdo a la base de conocimiento con la que se evaluó el controlador, el punto de consigna esta representados por una x encerrada en un círculo, ver Fig. 5.3, que se localiza en el centro del plano (0.25cm,0.25cm). La dinámica de los experimentos es llevar el balón (ilustrado por la

Cálculo del centroide basado en imagen		
Imagen	Esperado $(x, y)^*$	Módulo de visión $(x, y)^*$
1	(534,462)	(534,462)
2	(131,142)	(132,143)
3	(526,62)	(526,63)
4	(832,182)	(832,181)
5	(946,526)	(946,525)
6	(788,752)	(788,752)
7	(508,958)	(508,958)
8	(288,752)	(289,752)
9	(104,536)	(104,536)

*.Unidades dadas en píxeles

Tabla 5.1: Cálculo de centroide de Fig. 5.2

secuencia de círculos) de algún punto próximo a los bordes del plano hasta el centro del mismo. Los *experimentos A y B* fueron evaluados en el modelo real, mientras que el conjunto de reglas del *experimento C* lo fue en simulación. En los apéndices **D**, **E** y **F** se encuentra el resto de las gráficas resultado de los experimentos.

5.2.2. Experimento tipo A

La primer inquietud que se tubo para observar el comportamiento del controlador, fue que el balón se ubicara y equilibrara en el centro del área de trabajo, pues no se presentan situaciones críticas para el controlador. Fueron siete casos en los que se puso a prueba el sistema PE, la posición inicial del balón debía de colocarse en el borde del plano, de esta forma se alcanza la máxima distancia de desplazamiento. Ya que se tiene controladores independientes para los ejes de rotación del plano, se experimento la interacción entre ellos, ubicando la posición inicial del balón a 0° , 45° , 90° , 115° y 180° con respecto a la horizontal de la posición deseada. Al ubicar el balón en 115° , ver Figura 5.4, el comportamiento de ambos controladores es semejante, por lo que la trayectoria de la esfera hasta el punto de consiga

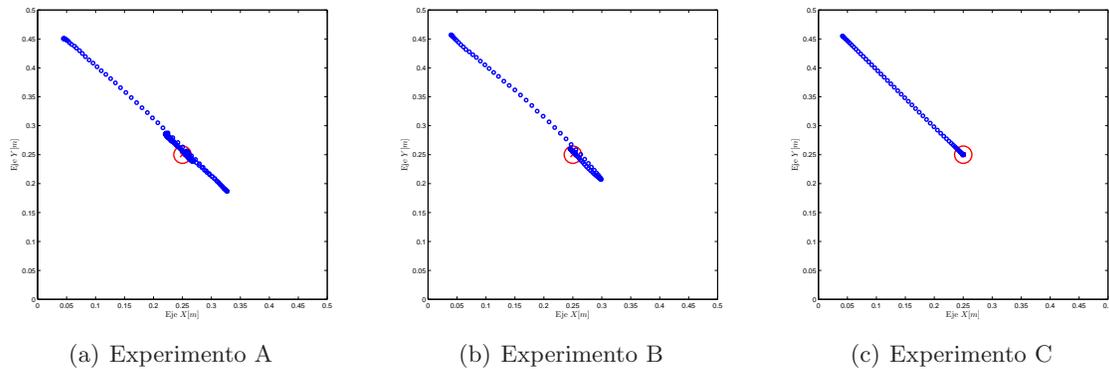


Figura 5.3: Clasificación de los experimentos.

<i>Inclinación</i>		Velocidad				
		NA	NM	Z	PM	PA
Error estimado	NA	<i>NA</i>	<i>PM</i>	<i>PM</i>	<i>PA</i>	<i>PA</i>
	NM	<i>NM</i>	<i>Z</i>	<i>PM</i>	<i>PM</i>	<i>PA</i>
	Z	<i>NM</i>	<i>NM</i>	<i>Z</i>	<i>PM</i>	<i>PM</i>
	PM	<i>NA</i>	<i>NM</i>	<i>NM</i>	<i>Z</i>	<i>PM</i>
	PA	<i>NA</i>	<i>NA</i>	<i>NM</i>	<i>NM</i>	<i>PA</i>

Tabla 5.2: Conjunto de reglas del experimento A.

dibuja una recta. En la Figura 5.5 son graficadas para ambos eje el comportamiento del error de la posición, velocidad de la esfera y la posición angular del plano contra el tiempo. Donde se puede apreciar, ver Fig. 5.5(b), que existe un sobrepaso máximo de 7.715cm para el eje X y 6.348cm para el eje Y. El sobrepaso es originado por el incremento de la velocidad de la esfera con el pico máximo a una distancia de (9.766cm,-6.348cm) de la posición deseada, a partir de esa posición el plano comienza a inclinarse para desacelerar la esfera, hasta llegar a la máxima inclinación del plano en error 0cm y el controlador continua balaceando la esfera hasta estabilizarla. Por su parte en la simulación, ver Fig. 5.5(a), el sobrepaso fue de 7.421cm para el eje X y 7.211cm para el eje Y, mientras que la velocidad máxima se registro a una distancia de (7.192cm,-6.892cm).

El conjunto de reglas que se evaluarón en este experimento se presenta en la Tabla 5.2. Al observar la Fig. 5.5 y las reglas se puede concluir que el sobrepaso es debido a que los conjuntos de salida que se activan son NM, Z y PM, salidas no suficientes para desacelerar apropiadamente la esfera, tal es el caso del valle que se presenta alrededor de los dos segundos de haber transcurrido el experimento, donde la *señal de la posición angular del plano* decrece

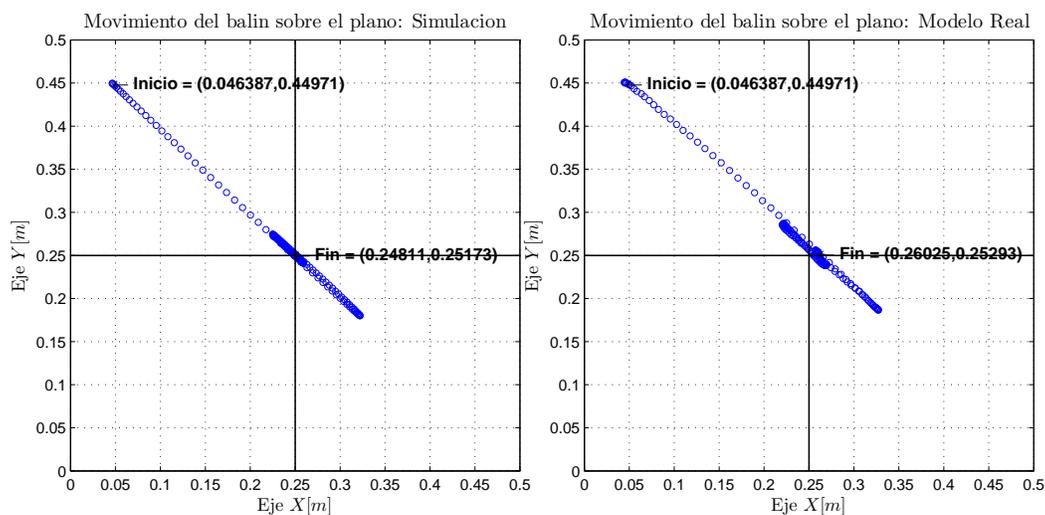
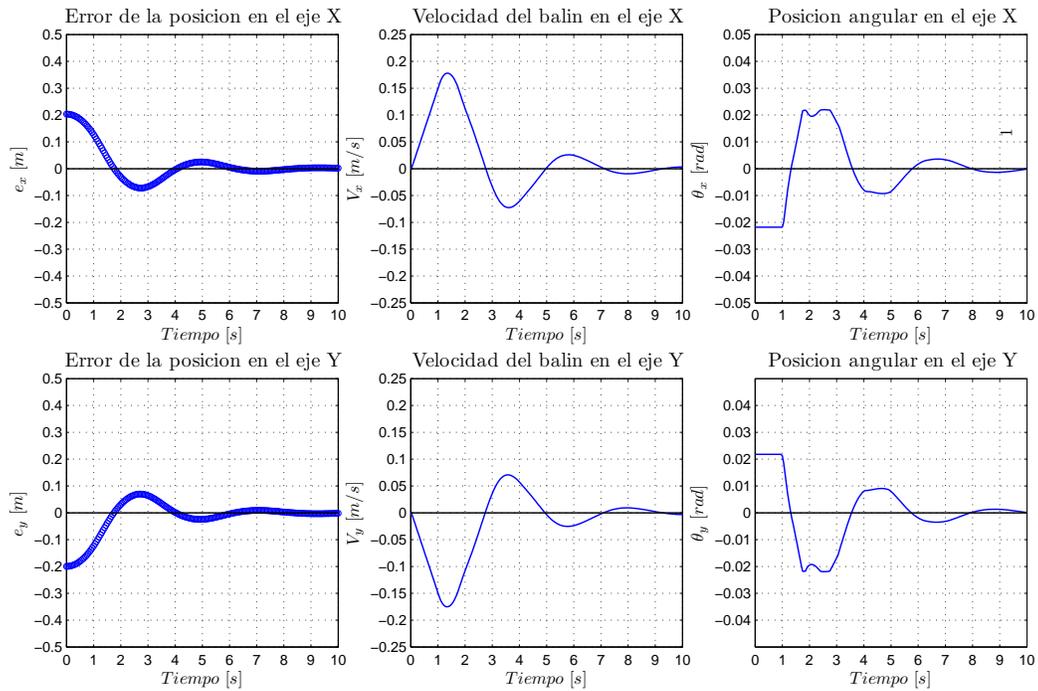
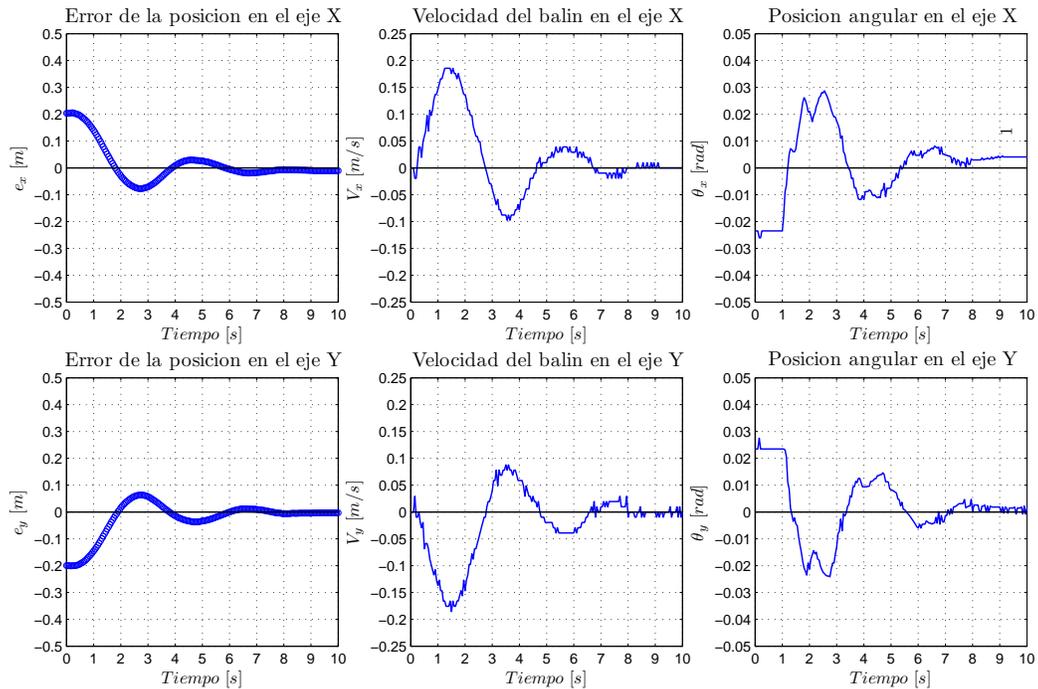


Figura 5.4: Comparación del movimiento de la esfera entre la simulación y el modelo real.



(a) Simulación.



(b) Modelo real.

Figura 5.5: Gráficas del Error, Velocidad y Señal de control.

<i>Inclinación</i>		Velocidad				
		NA	NM	Z	PM	PA
Error estimado	NA	<i>NA</i>	<i>PM</i>	<i>PM</i>	<i>PA</i>	<i>PA</i>
	NM	<i>NA</i>	<i>NM</i>	<i>PM</i>	<i>PA</i>	<i>PA</i>
	Z	<i>NA</i>	<i>NA</i>	<i>Z</i>	<i>PA</i>	<i>PA</i>
	PM	<i>NA</i>	<i>NA</i>	<i>NM</i>	<i>PM</i>	<i>PA</i>
	PA	<i>NA</i>	<i>NA</i>	<i>NM</i>	<i>NM</i>	<i>PA</i>

Tabla 5.3: Conjunto de reglas del experimento B.

favoreciendo el sobrepaso, esto debido principalmente a la activación de las reglas 7 y 18, considerando la numeración de las reglas de izquierda a derecha y de arriba hacia abajo. La forma y distribución de los conjuntos difusos de entrada y salida son como los mostrados en las Fig. 4.13 y 4.20 del capítulo anterior.

5.2.3. Experimento tipo B

Con la finalidad de mejorar el comportamiento del controlador, se elaboró el conjunto de reglas que se presentaron en el capítulo anterior, ver Tabla 5.3. Apoyándonos en el experimento anterior, se trató de resolver el problema de sobrepaso al modificar la base de conocimiento, la forma y distribución de los conjuntos difusos de entrada y salida no fueron alterados. Como se observa en la Fig. 5.6, el sobrepaso se vio disminuido en el modelo real, pero no en igual proporción en la simulación, lo destacado es la disminución del tiempo en que la esfera se estabiliza en la posición deseada, esto con respecto al experimento A. La pronta estabilización de la esfera en el modelo real en comparación con la simulación, puede ser debida a que el modelo matemático en el cual se basó la simulación no se consideró la

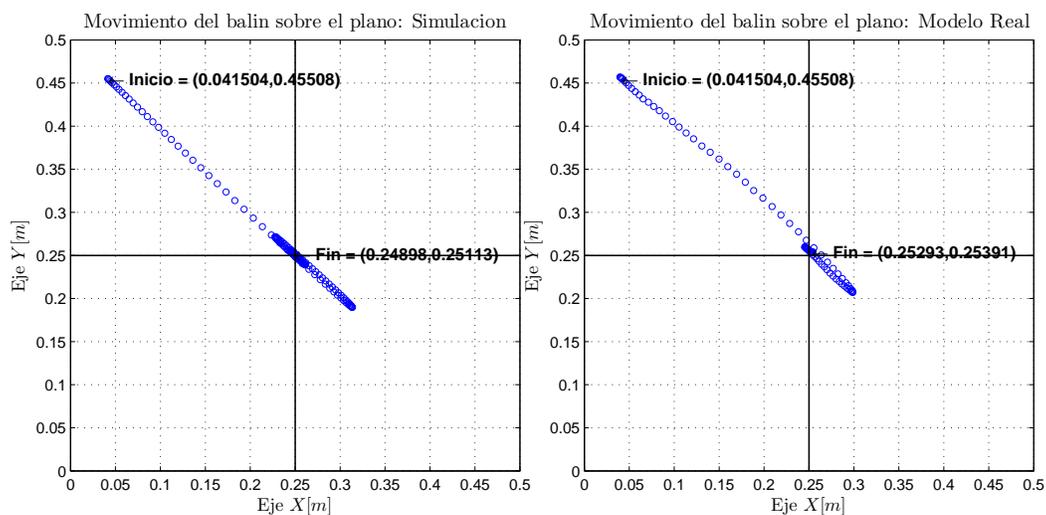
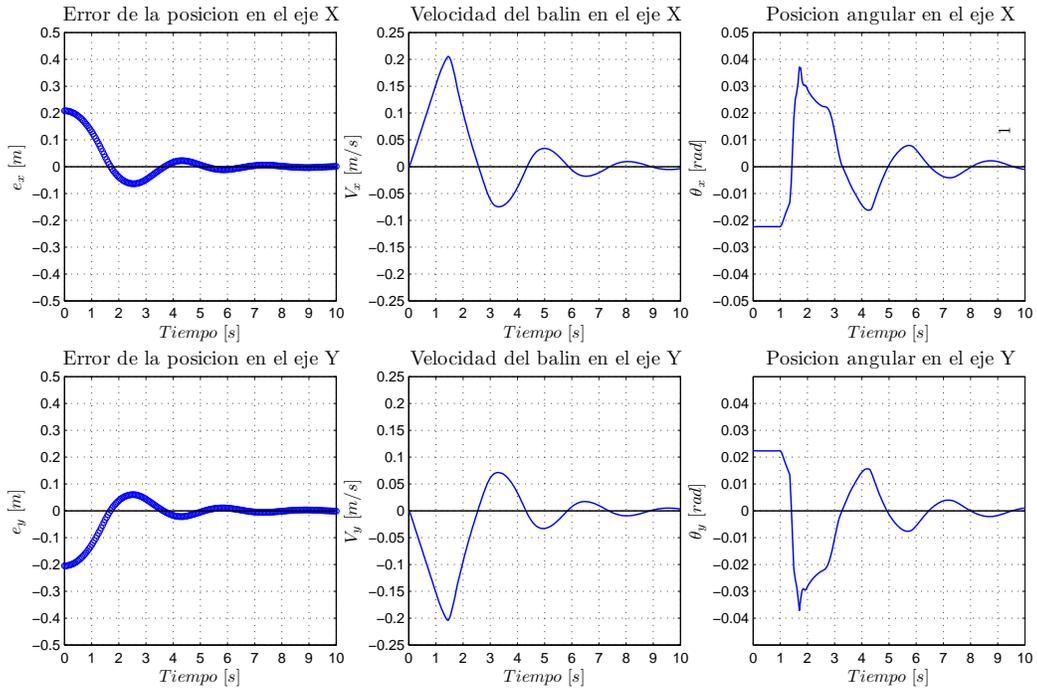
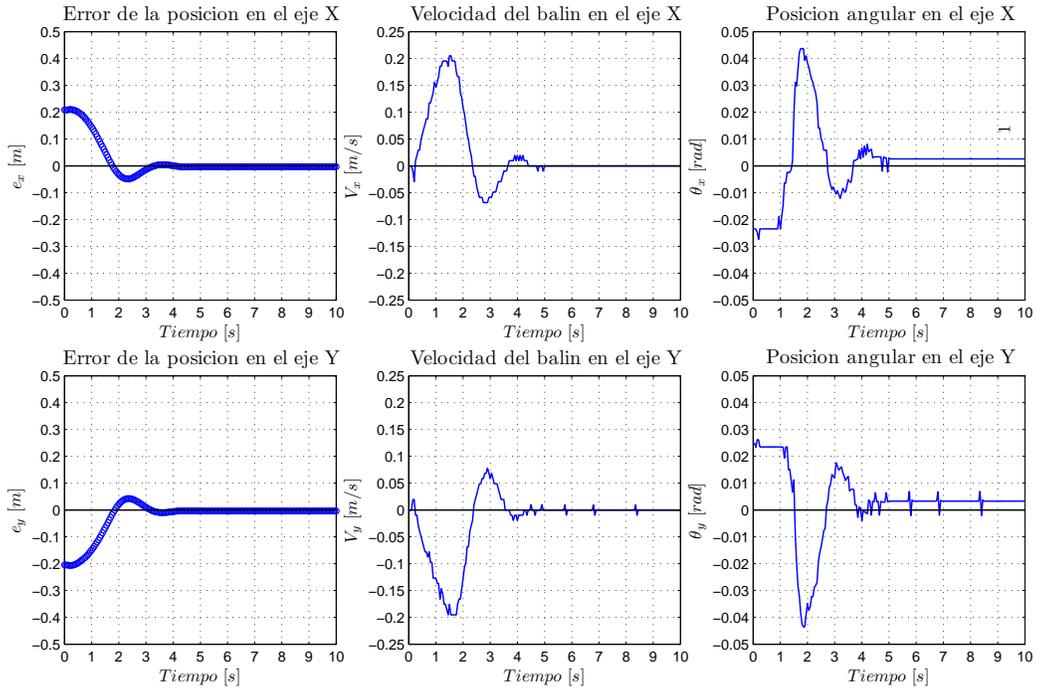


Figura 5.6: Comparación entre el movimiento de la simulación y el experimento real.



(a) Simulación.



(b) Modelo real.

Figura 5.7: Gráficas del Error, Velocidad y Señal de control.

fricción entre el plano y la esfera, así como el tiempo de respuesta de los actuadores.

A diferencia del experimento A, la magnitud del sobrepaso es de 4.883cm en el eje X y 4.297cm en el eje Y. La velocidad máxima se presentó a (5.127cm,5.664cm) de la posición deseada. Por su parte en la simulación el sobrepaso es de 6.36cm en el eje X y 6.023cm en el eje Y. La velocidad máxima se presentó a (4.667cm,-4.333cm) de la posición deseada. En las gráficas de posición angular de la Fig. 5.7, tanto para la simulación como del modelo real, la señal de control es máxima cuando la esfera intercepta la posición deseada, lo que ayuda a desacelerar la esfera pero aún no se ha evitado el sobrepaso, ya que la señal de control comienza a decrecer al cruzar la posición deseada.

Obtenidos estos resultados se decidió evaluar el controlador desplazando el punto de consigna a las posiciones (12.5cm,12.5cm) y (0.4375cm,0.4375cm), las gráficas de estos experimentos se encuentran en el apéndice E.

5.2.4. Experimento tipo C

En los experimentos A y B, la forma para mejorar el desempeño del controlador a sido modificar solo la base de conocimiento, lo cual no fue suficiente para evitar el sobrepaso de la esfera, por esta razón en la presente sección se propuso modificar también la forma y distribución de los conjuntos difusos de la variable lingüística de entrada *Velocidad* y de salida *Inclinación*.

Como se ha hecho notar en los experimentos anteriores, el controlador comienza a desacelerar la esfera muy cerca de la posición deseada y alcanza la señal de control su valor máximo al interceptar la posición deseada e inmediatamente comienza a decrecer su valor. La estrategia a seguir es desplazar el pico de la señal de control para anticipar la desaceleración de la esfera y así evitar el sobrepaso. Una medida para lograr lo anterior es modificar la base de conocimiento, ver Tabla 5.4. Las primeras modificaciones fueron incrementar la acción de control para las situaciones donde la esfera este cerca de la posición deseada y su velocidad sea muy baja (reglas 8 y 18) y hacer de esta manera que la esfera se aproxime al objetivo, ya que la interacción con las demás reglas pueden hacer lo contrario. La siguiente modificación fue al conjunto de reglas ubicadas en la esquina superior izquierda de la Tabla 5.4, así

<i>Inclinación</i>		Velocidad				
		NA	NM	Z	PM	PA
Error estimado	NA	<i>NM</i>	<i>Z</i>	<i>PM</i>	<i>PA</i>	<i>PA</i>
	NM	<i>NM</i>	<i>NM</i>	<i>PA</i>	<i>PA</i>	<i>PA</i>
	Z	<i>NA</i>	<i>NA</i>	<i>Z</i>	<i>PA</i>	<i>PA</i>
	PM	<i>NA</i>	<i>NA</i>	<i>NA</i>	<i>PM</i>	<i>PM</i>
	PA	<i>NA</i>	<i>NA</i>	<i>NM</i>	<i>Z</i>	<i>PM</i>

Tabla 5.4: Conjunto de reglas del experimento C.

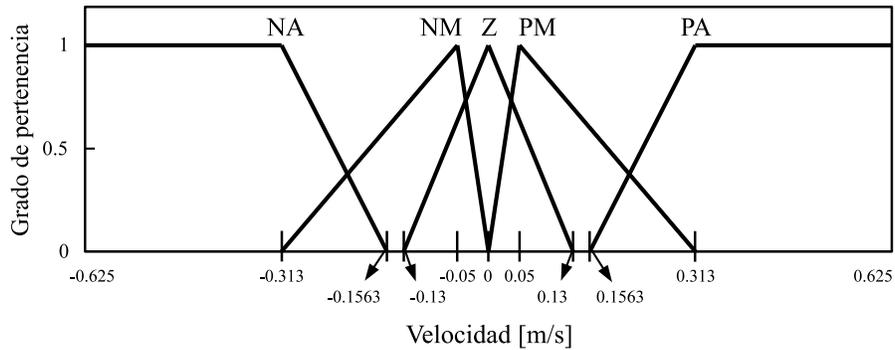


Figura 5.8: Conjuntos modificados de la Variable lingüística *Velocidad*.

como de la esquina inferior izquierda. Este grupo de reglas ayudan a reducir la velocidad máxima de la esfera (en comparación con los experimentos anteriores), como consecuencia los movimientos de la esfera son más suaves. Al modificar la base de conocimiento, también se incrementó la magnitud de los conjuntos difusos de salida, ver Fig. 5.9. Por último, para poder desplazar el pico de la señal de control se modificó los conjuntos de la variable lingüística *Velocidad*, ver Fig. 5.8.

La simulación con el nuevo conjunto de modificaciones resultó en la eliminación del sobrepaso de la esfera, ver Fig. 5.10. Como era deseado, el pico de la señal de control fue desplazado, lo que favoreció a la respuesta sobreamortiguada del controlador. Es necesario destacar que al iniciar el experimento, la velocidad de la esfera fue decreciendo de forma más gradual y la magnitud del pico de velocidad fue menor en comparación con los experimentos A y B. La velocidad máxima de la esfera se alcanzó a (5.69cm, -6.076cm) de la posición deseada y la máxima acción de control se ubicó a (2.878cm, -3.193cm) de la posición deseada, ver Fig. 5.11, que corresponde al comportamiento propuesto.

Se podría seguir manipulando la base de reglas y los conjuntos difusos para continuar mejorando la respuesta del controlador, pero se demostró que la simulación es una herramienta confiable para poder predecir el comportamiento del controlador, es por ello

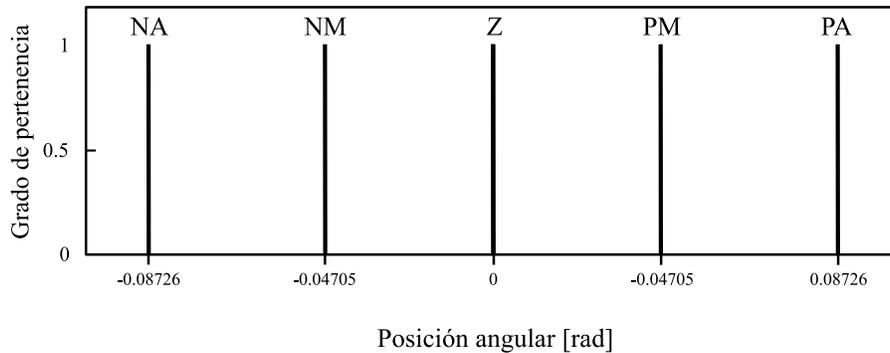


Figura 5.9: Conjuntos modificados de la Variable lingüística *Inclinación*.

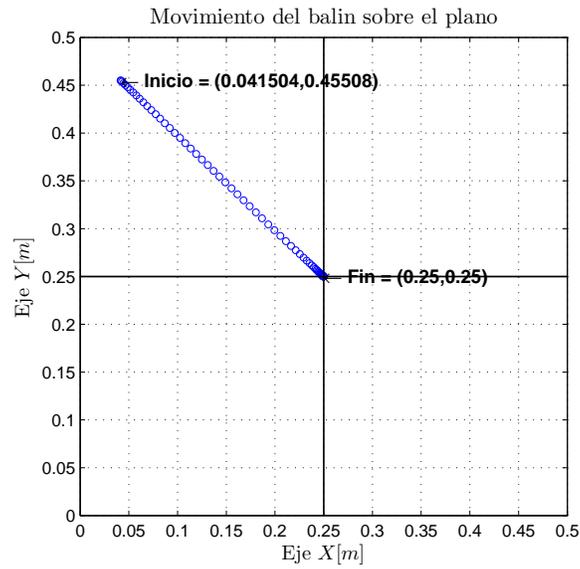


Figura 5.10: Comparación entre el movimiento de la simulación y el experimento real

que en esta sección no se llevo a cabo el experimento en el modelo real. En el apéndice F esta contenido el total de gráficas resultado de este experimento.

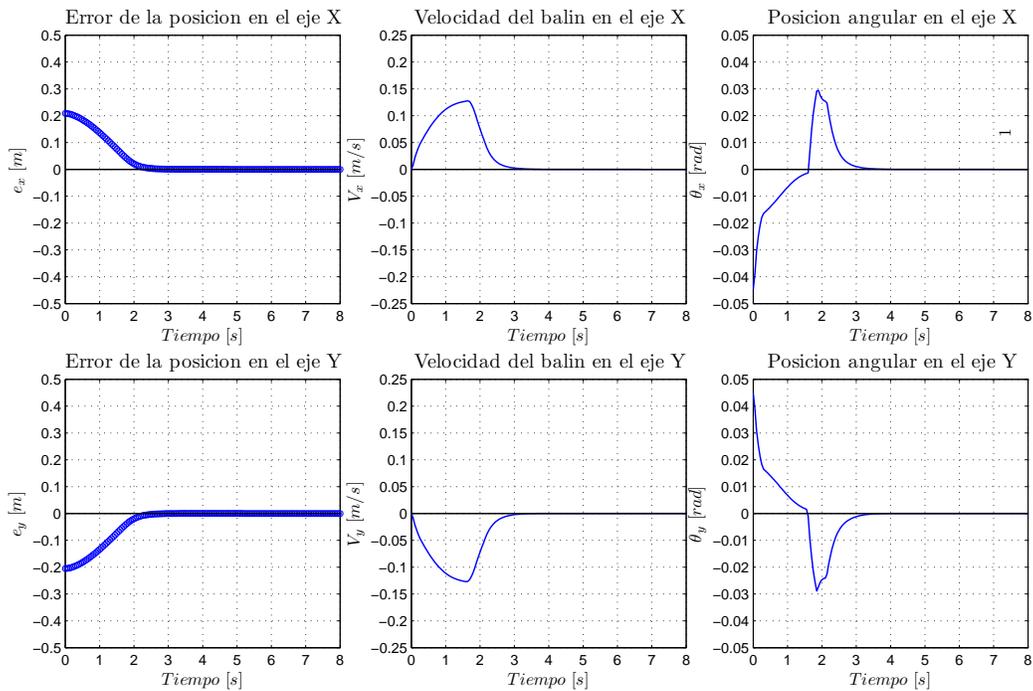


Figura 5.11: Gráficas del Error, Velocidad y Señal de control para la simulación del sistema PE.

5.2.5. Resultados

Recursos utilizados del FPGA

Los recursos utilizados en el FPGA se muestran en la Tabla 5.5, como se puede apreciar solo se utilizó el 15% del número de **elementos lógicos** disponibles en el FPGA. Este FPGA también dispone de multiplicadores embebidos para facilitar el diseño y además para mejorar el desempeño de operaciones aritméticas. Los componentes que calcularon el centro de masa de la esfera, así como el defuzificador emplearon dichos multiplicadores embebidos de los cuales se utilizaron cuatro. Las funciones de pertenencia que se utilizaron para fuzificar las entradas del controlador emplearon 20480 bits de memoria embebida en el dispositivo, lo que equivalió al 4% de la disponible.

Total de elementos lógicos	5077/33,216 (15%)
Total de pins	178/475 (37%)
Total de bits de memoria	20480/483480 (4%)
Multiplicadores embebidos de 9-bit	4/70 (6%)

Tabla 5.5: Tabla de uso de recurso en el FPGA.

Resumen de resultados de los experimentos

En esta sección se resume en tablas, información importante de los tres tipos de experimentos, tal como la posición inicial y final, tiempo que tarda la esfera en estabilizarse y el error.

La Tabla 5.6 contiene información de los experimentos tipo A, en ella se observa que el tiempo máximo que tarda en estabilizarse la esfera es de 8.9s y el tiempo mínimo es de 5.95s, a su vez el error máximo es de 1.025cm y el mínimo de 0.98mm que es equivalente a dos píxeles de error. Los resultados que se muestran en la Tabla 5.6 fueron los arrojados por el modelo real, ya que es de nuestro interés analizar su comportamiento. Solo se evaluaron siete casos para el experimento tipo A, ya que su comportamiento no es el esperado, pero sirvió de referencia para poner a punto la base de conocimiento del experimento tipo B. En este último, el tiempo que tarda la esfera en estabilizarse se redujo alrededor de un 50%, se tiene como tiempo máximo 4.75s y como mínimo 3.5s, ver Tabla 5.7. Los experimentos del uno al ocho tienen como posición deseada $(0.25cm, 0.25cm)$, los experimentos del nueve al doce $(0.125cm, 0.125cm)$ y los restantes $(0.4375cm, 0.4375cm)$, el experimento 16 presenta el máximo desplazamiento desde la posición inicial hasta la final.

Con la finalidad de evaluar un nuevo conjunto de reglas y disposición de los conjunto difusos, se decidió realizar solo la simulación del sistema, ya que se cuenta con más información de su comportamiento. El experimento tipo C, fue evaluado a partir de las posiciones iniciales y deseadas del experimento tipo B, con la finalidad de tener un punto de comparación. Como se observa en la Tabla 5.8, se obtienen resultados superiores a los del experimento B,

Experimento	Posición inicial (m)	Posición final (m)	Tiempo (s)	Error (m)
1	(0.046387,0.44971)	(0.26025,0.25293)	7.6	(-0.01025,-0.0029)
2	(0.24805,47656)	(0.24316,0.24609)	6.4	(0.0068,0.0039)
3	(0.49707,0.24854)	(0.25391,0.259771)	5.95	(-0.00391,-0.009771)
4	(0.4834,0.0083)	(0.26025,25732)	8.9	(-0.01025,-0.00732)
5	(0.24756,0.0053711)	(0.25537,0.25732)	7.3	(-0.00537,-0.00732)
6	(0.027344,0.027344)	(0.25488,0.25049)	8.55	(-0.00488,-0.00049)
7	(0.014648,0.25244)	(0.24561,0.25098)	6.7	(0.00439,-0.00098)

Tabla 5.6: Resumen de las gráficas del experimento A.

además de haber evitado el sobrepaso. Al ser una simulación el error que se obtuvo tiende a cero y el tiempo máximo que se tarda en estabilizarse la esfera es de 4.65s y el tiempo mínimo es de 2.7s.

Experimento	Posición inicial (m)	Posición final (m)	Tiempo (s)	Error (m)
1	(0.041504,0.45508)	(0.25293,0.25391)	4.25	(-0.00293,-0.00391)
2	(0.25732,0.47705)	(0.25293,0.25391)	4	(-0.00293,-0.00391)
3	(0.4707,0.47021)	(0.25537,0.25098)	4.3	(-0.00537,-0.00098)
4	(0.48047,0.25195)	(0.25293,0.25244)	4.75	(-0.00293,-0.00244)
5	(0.46484,0.022461)	(0.25049,0.24854)	4.1	(-0.00049,0.0015)
6	(0.24951,0.020996)	(0.25244,0.25244)	4.2	(-0.00244,-0.00244)
7	(0.03222,0.03808)	(0.25098,0.25342)	4.25	(-0.00098,-0.00342)
8	(0.01855,0.25049)	(0.25244,0.25293)	4.5	(-0.00244,-0.00293)
9	(0.23926,0.24707)	(0.12207,0.12451)	3.5	(0.00293,0.00049)
10	(0.041016,0.46045)	(0.12793,0.12842)	4.8	(-0.00293,-0.00342)
11	(0.27441,0.02588)	(0.12451,12598)	3.85	(0.00049,-0.00098)
12	(0.47217,0.23584)	(0.12598,0.12549)	4.7	(-0.00098,-0.00049)
13	(0.29053,0.0625)	(0.43994,0.44189)	4.2	(-0.00244,-0.00439)
14	(0.25244,0.1084)	(0.43896,0.44043)	4.3	(-0.00146,-0.00293)
15	(0.049316,0.0625)	(0.44336,0.44141)	4.55	(-0.00586,-0.00391)
16	(0.025879,0.24805)	(0.44043,0.44238)	4.75	(-0.00293,-0.00488)
17	(0.25391,0.48535)	(0.44238,0.44189)	3.95	(-0.00488,-0.00439)

Tabla 5.7: Resumen de las gráficas del experimento B.

Experimento	Posición inicial (m)	Posición final (m)	Tiempo (s)	Error (m)
1	(0.041504,0.45508)	(0.25,0.25)	3.15	(0,0)
2	(0.25732,0.47705)	(0.25,0.25)	3.2	(0,0)
3	(0.4707,0.47021)	(0.25,0.25)	3.15	(0,0)
4	(0.48047,0.25195)	(0.25,0.25)	3.25	(0,0)
5	(0.46484,0.022461)	(0.25,0.25)	3.1	(0,0)
6	(0.24951,0.020996)	(0.25,0.25)	3.5	(0,0)
7	(0.03222,0.03808)	(0.25,0.25)	3.25	(0,0)
8	(0.01855,0.25049)	(0.25,0.25)	3.4	(0,0)
9	(0.23926,0.24707)	(0.125,0.125)	2.7	(0,0)
10	(0.041016,0.46045)	(0.125,0.125)	3.85	(0,0)
11	(0.27441,0.02588)	(0.125,0.125)	2.7	(0,0)
12	(0.47217,0.23584)	(0.125,0.125)	3.15	(0,0)
13	(0.29053,0.0625)	(0.4375,0.4375)	4.4	(0,0)
14	(0.25244,0.1084)	(0.4375,0.4375)	4.1	(0,0)
15	(0.049316,0.0625)	(0.4375,0.4375)	4.5	(0,0)
16	(0.025879,0.24805)	(0.4375,0.4375)	4.65	(0,0)
17	(0.25391,0.48535)	(0.4375,0.4375)	3.25	(0,0)

Tabla 5.8: Resumen de las gráficas del experimento C.

Capítulo 6

Conclusiones y trabajo futuro.

6.1. Conclusiones.

El procesamiento de imágenes binarias implementado en FPGA utilizado para el cálculo del centroide de la esfera dentro del plano, presenta una mínima desviación de un píxel con respecto de la posición esperada, debido a que las pruebas se realizaron en condiciones de iluminación naturales en el laboratorio.

Todo el procesamiento de la imagen capturada por la cámara se realiza en línea. El término en línea, nos dice que conforme se leen los datos de la imagen, estos son procesados en el FPGA sin necesidad de almacenarla en su totalidad, sólo se necesita almacenar tres líneas de la imagen en un registro de desplazamiento, por lo que para la obtención del centroide, se tiene un retraso aproximado de dos líneas de la imagen y dos píxeles, lo cual representa una ventaja con respecto a equipos convencionales (PCs).

Tanto la obtención del grado de pertenencia, como el mecanismo de inferencia de las 25 reglas, se llevaron a cabo en procesos concurrentes y la defuzificación se realizó en un proceso serie, ya que su costo computacional no es elevado y así se disminuye el consumo de recursos empleados en el FPGA. Con lo que se obtuvo un tiempo de procesamiento para el controlador difuso de 706.5 ns.

La metodología planteada en este trabajo de tesis y su implementación en un dispositivo programable a podido resolver la tarea que impone el sistema PE. Los resultados arrojados en el Capítulo 5 muestran que el controlador difuso tipo Mamdani discreto coloca la esfera en cualquier posición dentro de un espacio de trabajo de 0.40m x 0.40m, con un error en el peor caso del 7mm con respecto de la posición deseada. El espacio de trabajo se ve reducido por la respuesta transitoria oscilatoria que tiene un pico de sobrepaso máximo de 5cm. El tiempo promedio que tarda en estabilizarse el sistema es de 4seg.

6.2. Trabajos Futuros.

- Como trabajo futuro, se puede agregar al controlador presentado en esta tesis un controlador difuso que supervise el seguimiento de trayectorias de la esfera sobre el plano.
- Con base en la presente tesis, agregar algoritmos de aprendizaje basados en la experiencia del sistema PE y de esta forma llevar a cabo comparativas de su desempeño.
- Desarrollar un controlador difuso tipo Takagi-Sugeno para esta plataforma y hacer una comparativa de desempeño con el controlador presentado en esta tesis.
- Incrementar la capacidad de procesamiento de imágenes en el FPGA, para poder realizar tareas más complejas como el seguimiento o evasión de obstáculos sobre el área de trabajo del sistema PE.
- Así como desarrollar nuevas estrategias de control y arquitecturas de procesamiento en FPGA, para verificar su desempeño en el sistema PE.

Referencias

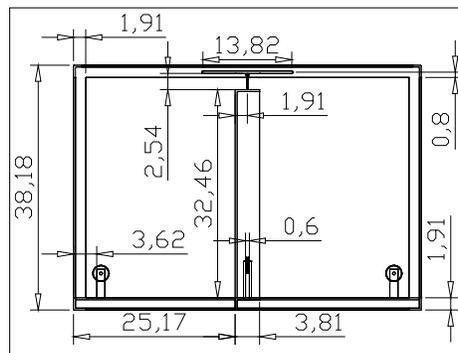
- [1] Wai Lun Lo A. B. Rad, P. T. Chan and C. K. Mok. An online learning fuzzy controller. *IEEE Transactions on Industrial Electronics*, 50(5):1016–1021, Octubre 2003.
- [2] Rajko Sveciko Andrej Knuplez, Amor Chowdhury. Modeling and control design for the ball and plate system. *ICIT-IEEE*, pages 1064–1067, 2003.
- [3] Sánchez Martínez Miguel Angel. Diseño de un circuito comparador de imágenes. Master’s thesis, CINVESTAV del I. P. N., Departamento de ingeniería eléctrica, Sección de computación, México D.F., Julio 2005.
- [4] Shorya Awtar and Kevin C. Craig. [Mechatronic design of a ball on plate balancing system.](#)
- [5] B. E. Bayer. Color imaging array. US Patent No. 3971065.
- [6] Altera Corporation. [DE2 User Manual.](#)
- [7] Alessandr Astolfi Daniele Casagrande and Thomas Parisini. A stabilizing time-switching control strategy for the rolling sphere. *44th IEEE Conference on Decision and Control, and the European Control Conference*, pages 3297–3302, December 2005.
- [8] Víctor Manuel Saldana García. Control visual de un sistema mesa esfera con cámara fija. Master’s thesis, CINVESTAV del I. P. N., Departamento de ingeniería eléctrica, Sección de computación, México D.F., Marzo 1999.
- [9] Ce Ding Qing Gu Hongrui Wang, Yantao Tian and Feng Guo. Output regulation of the ball and plate system with a nonlinear velocity observer. *7th World Congress on Intelligent Control and Automation, Chongqing, China*, pages 2164–2169, June 2008.
- [10] Chang-Kuo Chen Teh-Lu Liao Hsin-Chieh Chen, Yung-Ching Hung and Chun-Kuo Chen. Images processing algorithms realized by discrete-time cellular neural networks and their circuit implementations. *Chaos, Solitons and Fractals*, 29(5):1100–1108, September 2006.
- [11] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187, February 1962.
- [12] HUMUSOFT. [Ball and Plate Model.](#)
- [13] HUMUSOFT. [Real Time Toolbox for use with MATLAB.](#)

- [14] H. Lipken J. A. Piepmeier, C. V. McMurray. Traking a moving target with model independent visual servoing: a predictive estimation approach. *Proceeding of IEEE International Conference on Robotics and Automation, Leuven, Belgium*, pages 2652–2657, May 1998.
- [15] Petri Kuosmanen Jari Kostamo, Heikki Hy^ootyniemi. Ball balancing system: an educational device for demostrating optimal control. *Proceedings 2005 IEEE International Symposium on Computational Intelligence in Robotic and Automation*, pages 379–384, June 2005.
- [16] J. Su M. Bai, H. Lu and Y. Tian. Motion control of ball and plate system using supervisory fuzzy controller. *In Proc. 6th World Congree on Intelligent Control and Automation*, 2006.
- [17] Mohsen Saadat Miad Moarref and Gholamreza Vossoughi. Mechatronic design and position control of a novel ball and plate system. *16th Mediterranean Conference on Control and Automation-IEEE*, pages 1071–1076, June 2008.
- [18] Richard E. Woods Rafael C. Gonzalez and Steven L. Eddins. *Digital Image Processing using Matlab*. Prentice Hall, 2004.
- [19] Hitec RCD. [Servomotor HS-805BB](#).
- [20] Roger Y. Tsai Reimar K. Lenz. Techniques for calibration of the scale factor and image center for high accuracy 3-d machine vision methodology. *IEEE transaction on pattern and machine intelligence*, 10(5):713–720, september 1988.
- [21] Peter I. Corke Seth Hutchinson, Gregory D. Hager. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, October 1996.
- [22] IEEE Computer Society. *IEEE Standard VHDL Language Reference Manual*. January 2000.
- [23] Micron Technology. [MT9M011: 1/3-Inch Megapixel CMOS Active-Pixel Digital Image Sensor](#).
- [24] Terasic. [TRDB_DC2 User Guide](#).
- [25] Arturo Téllez Velázquez. Arquitectura de un controlador lógico difuso usando lógica combinatoria. Master’s thesis, Instituto Politécnico Nacional. Centro de Investigación en Computación, Junio 2008.
- [26] Li-Xin Wang. *A course in fuzzy systems and control: Desing of Fuzzy Systems from Input-Output Data*. Prentice-Hall, Upper Saddle River, N.J., 1997.
- [27] Shujie Teng Xingzhe Fan, Naiyao Zhang. Trajectory planning and tracking of ball and plate system using hierarchical fuzzy control scheme. *Fuzzy Sets and Systems, Elsevier*, March 2003.
- [28] Fernando Pardo Carpio y José A. Boluda Grau. *VHDL: Lenguaje para síntesis y modelado de circuitos*. Alfaomega RA-MA, 1999.

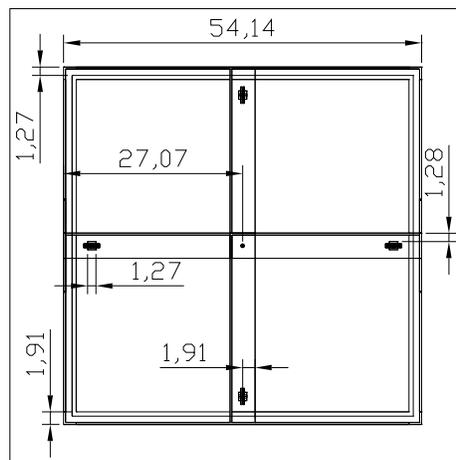
- [29] Lotfi A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning - i. *Inf. Sci.*, 8(3):199–249, 1975.

Apéndice A

CAD del Prototipo Plano-Esfera

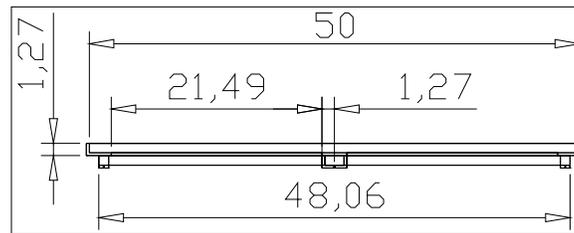


(a) Vista frontal

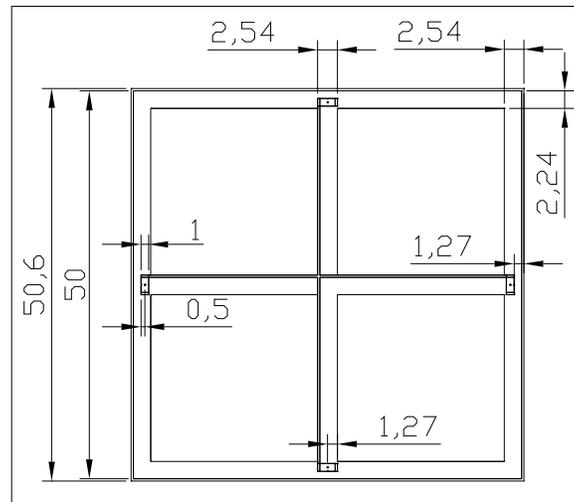


(b) Vista superior

Figura A.1: CAD de la base del Prototipo Plano-Esfera



(a) Vista frontal



(b) Vista superior

Figura A.2: CAD de la base del area de trabajo

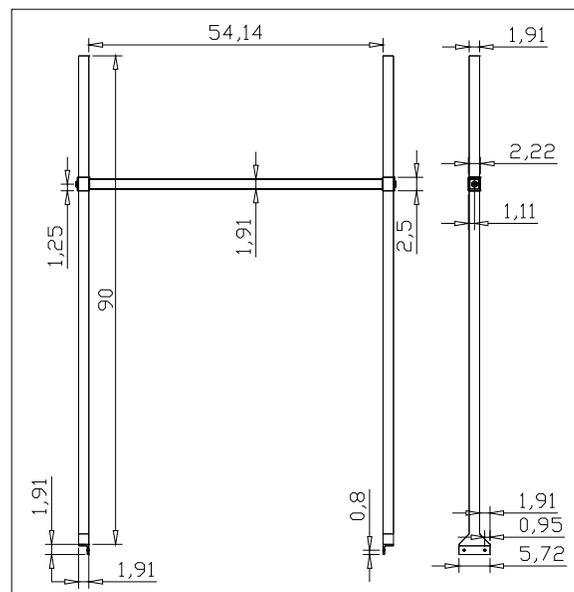


Figura A.3: CAD del soporte de la cámara

Apéndice B

Componentes del diseño en hardware

Se utilizó un **FPGA** Altera Cyclone II EP2C35F672C6 (el dispositivo se encuentra montado en la tarjeta DE2 de Altera Corporation) [6], en donde se implementaron las entidades que describen el comportamiento del módulo de visión, ver Fig. B.1. La primera tarea que realiza el FPGA, es la de configurar el sensor digital de imágenes a través de una interfaz serie de dos vías, la entidad **I2C_Control** es la encargada de generar las señales que configuran la cámara. Por su parte la entidad **I2C_Config** contiene los parámetros a configurar como lo son el ancho de las filas y columnas, la posición de inicio de las filas y columnas, el tiempo de exposición, entre otros.

Una vez configurado el sensor, los datos de la imagen son enviados al **FPGA** en un barrido progresivo. La entidad **Captura_CCD** se sincroniza con la cámara para recibir los datos de la imagen y aplicar la función de umbral para segmentarla. Obtenida la imagen binarizada esta misma entidad elimina los píxeles que no pertenecen al objeto de interés (esfera) a través de una transformación morfológica de erosión. Conforme es aplicado el elemento estructurante a cada dato de la imagen, se detectan los **píxeles** que pertenecen a la esfera para determinar su centroide en el plano a través de las entidades **Eje:U4** y **Eje:U5**. Obtenida la posición de la esfera sobre el plano, la entidad **Mamdani_Eje_X** y **Mamdani_Eje_Y** fusifican las entradas lingüísticas, infiere el valor de los conjuntos difusos de salida a partir de la base de conocimiento y defusifica la salida del controlador. Una vez determinada la salida del controlador esta es interpretada por la entidad **Servo** que genera las señales apropiadas para los servomotores y las envía a través de los puertos de salida **GPIO_0[1]** y **GPIO_0[0]**.

Apéndice C

Programas

Anexo en CD.

Apéndice D

Gráficas del experimento A

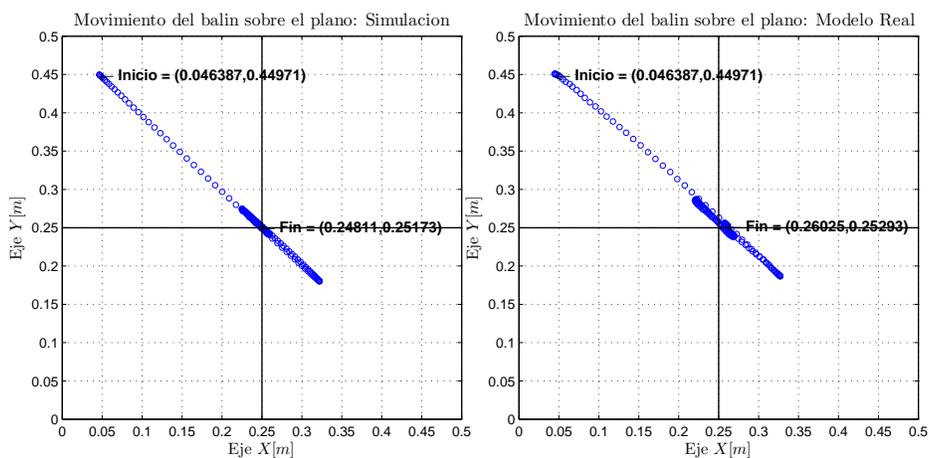


Figura D.1: Comparación entre el movimiento de la simulación y el modelo real, experimento A1.

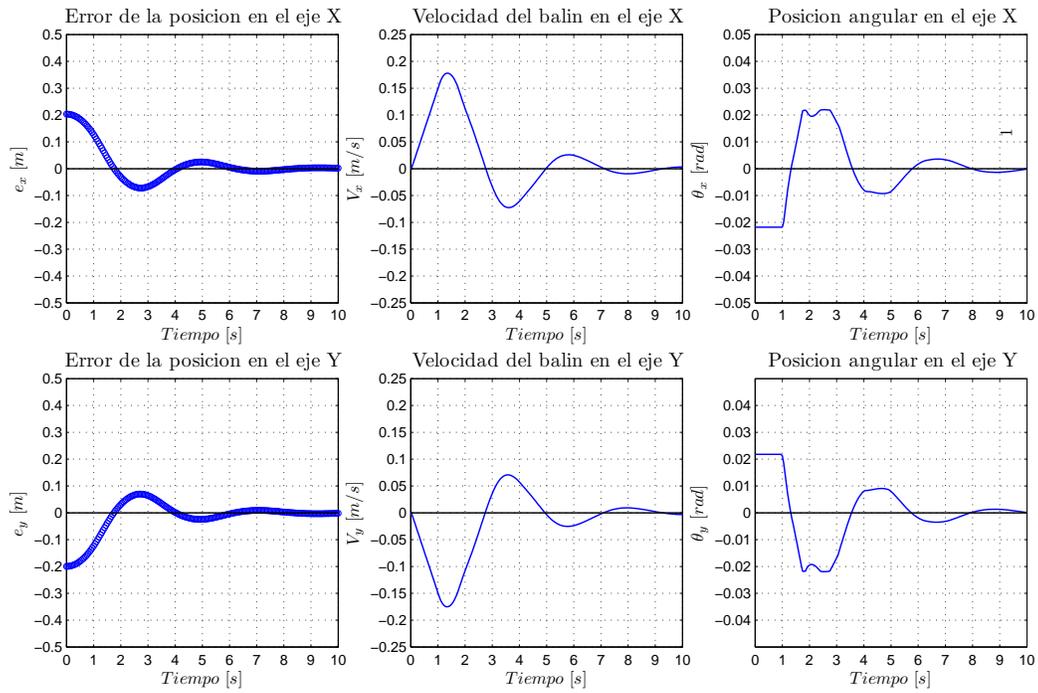


Figura D.2: Gráficas del Error, Velocidad y Señal de control, simulación A1.

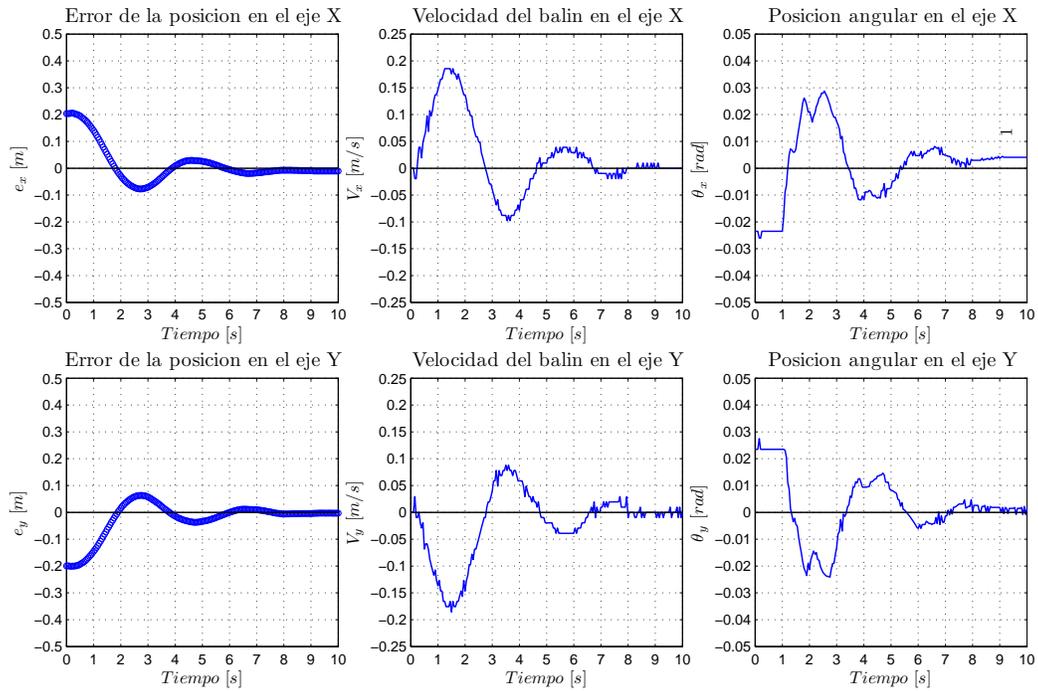


Figura D.3: Gráficas del Error, Velocidad y Señal de control, modelo real A1.

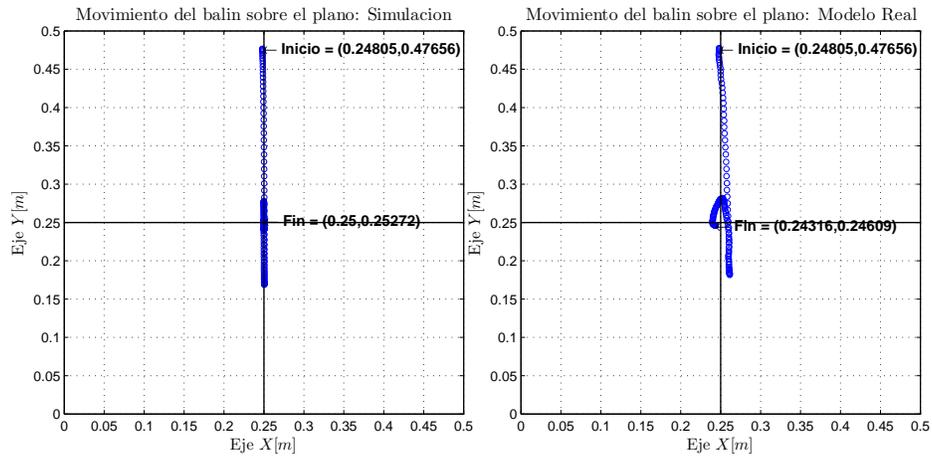


Figura D.4: Comparación entre el movimiento de la simulación y el modelo real, experimento A2.

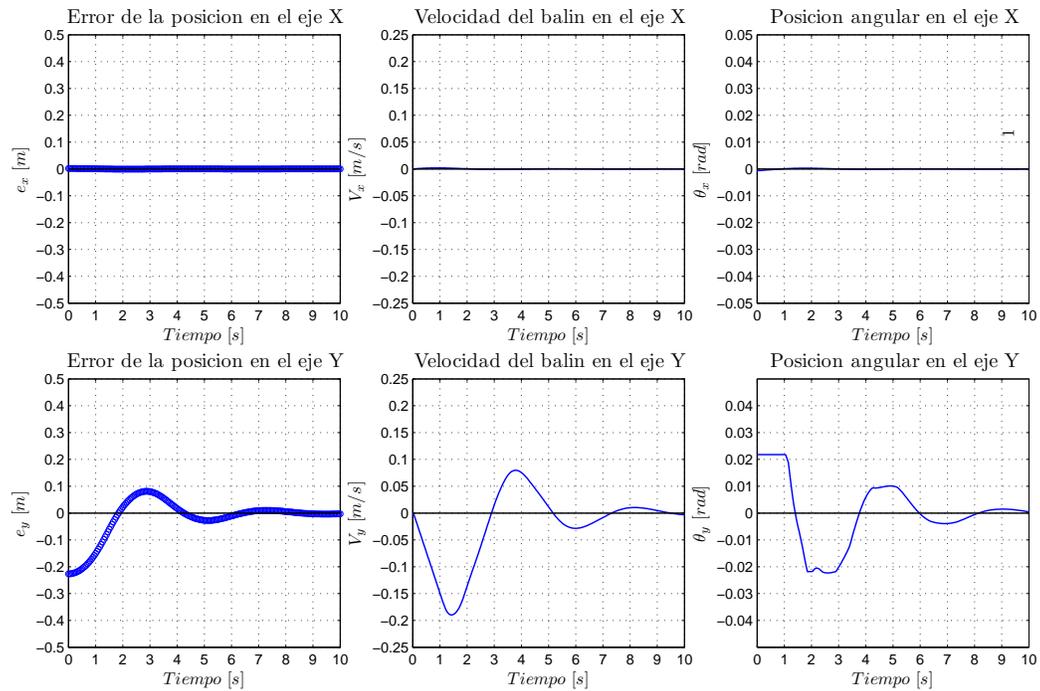


Figura D.5: Gráficas del Error, Velocidad y Señal de control, simulación A2.

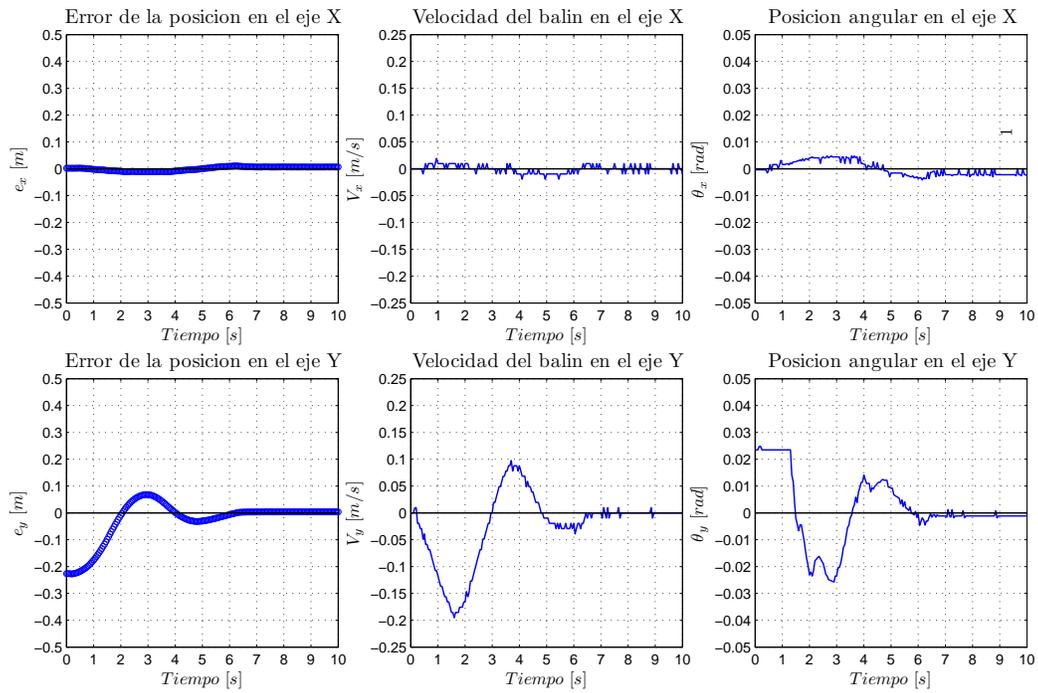


Figura D.6: Gráficas del Error, Velocidad y Señal de control, modelo real A2.

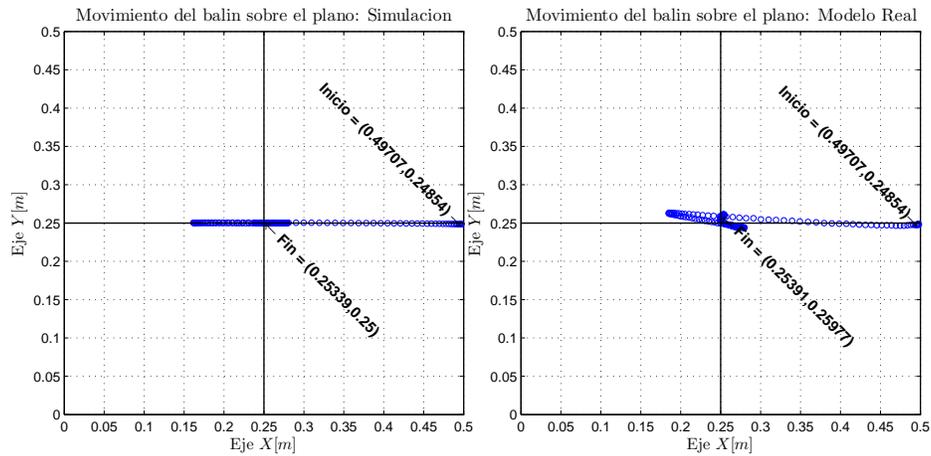


Figura D.7: Comparación entre el movimiento de la simulación y el modelo real, experimento A3.

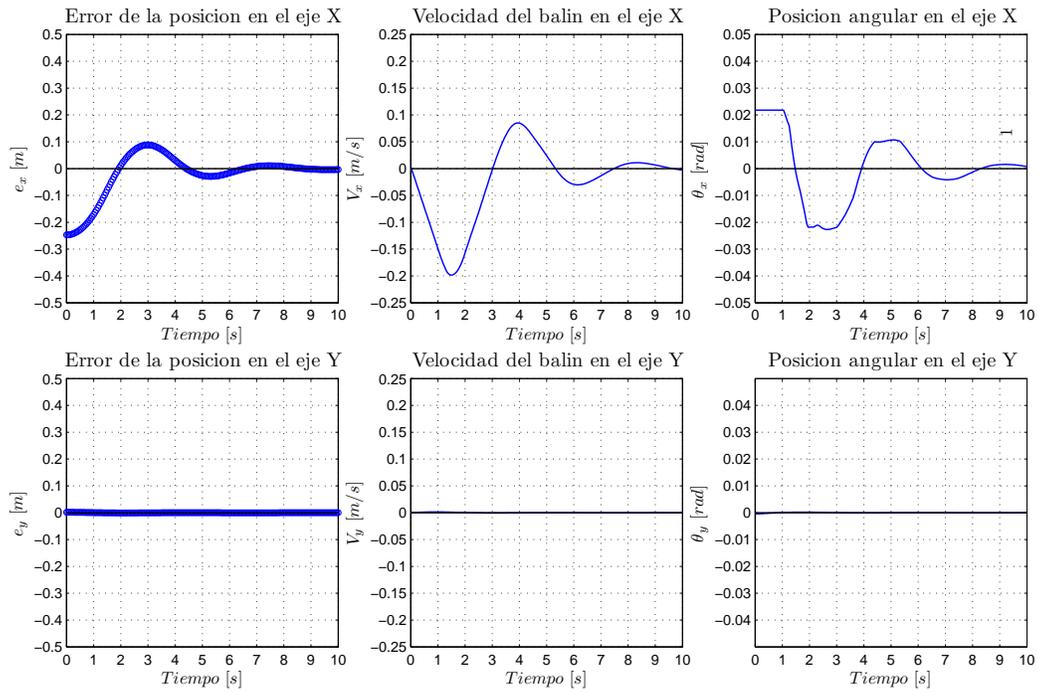


Figura D.8: Gráficas del Error, Velocidad y Señal de control, simulación A3.

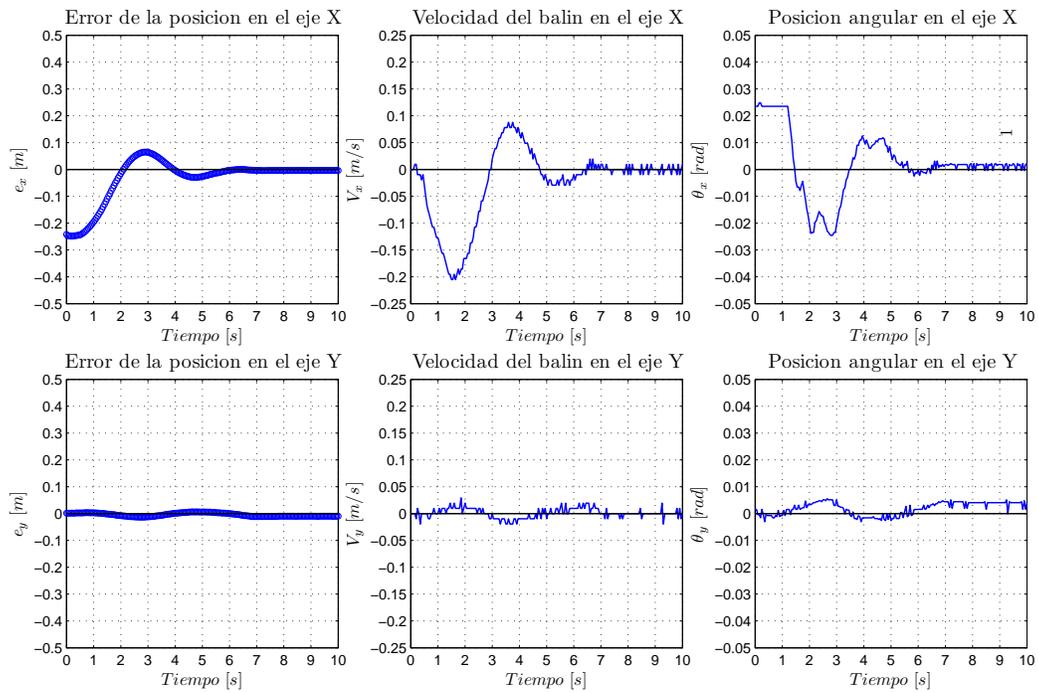


Figura D.9: Gráficas del Error, Velocidad y Señal de control, modelo real A3.

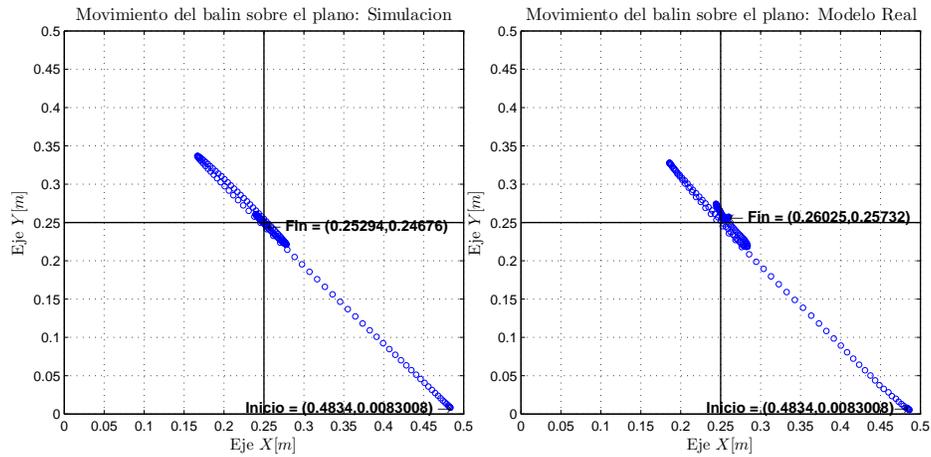


Figura D.10: Comparación entre el movimiento de la simulación y el modelo real, experimento A4.

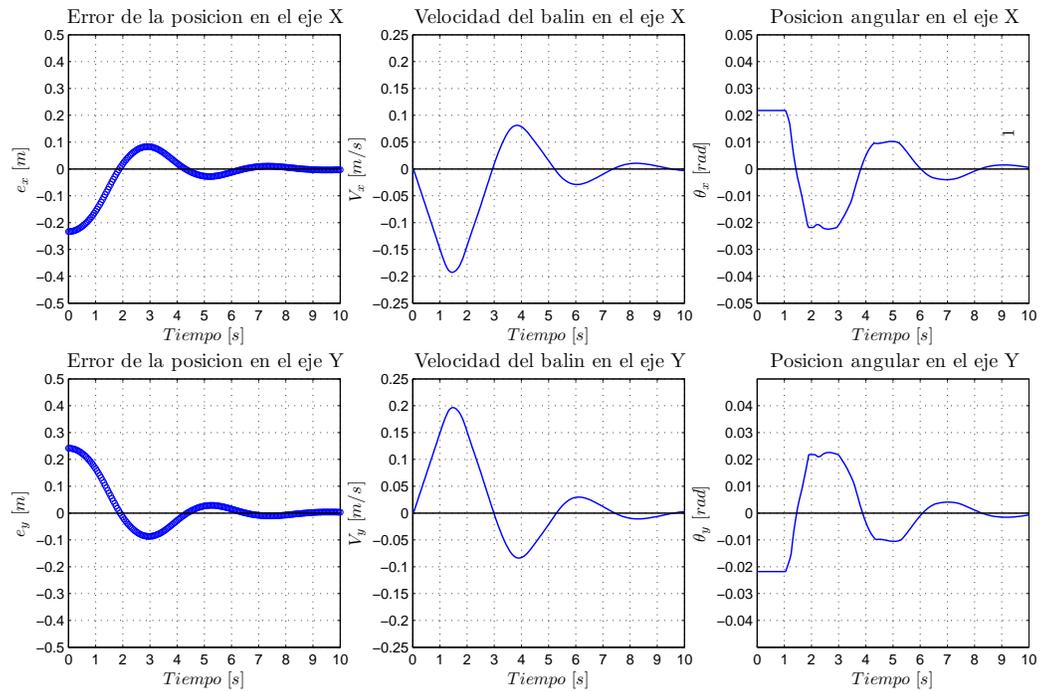


Figura D.11: Gráficas del Error, Velocidad y Señal de control, simulación A4.

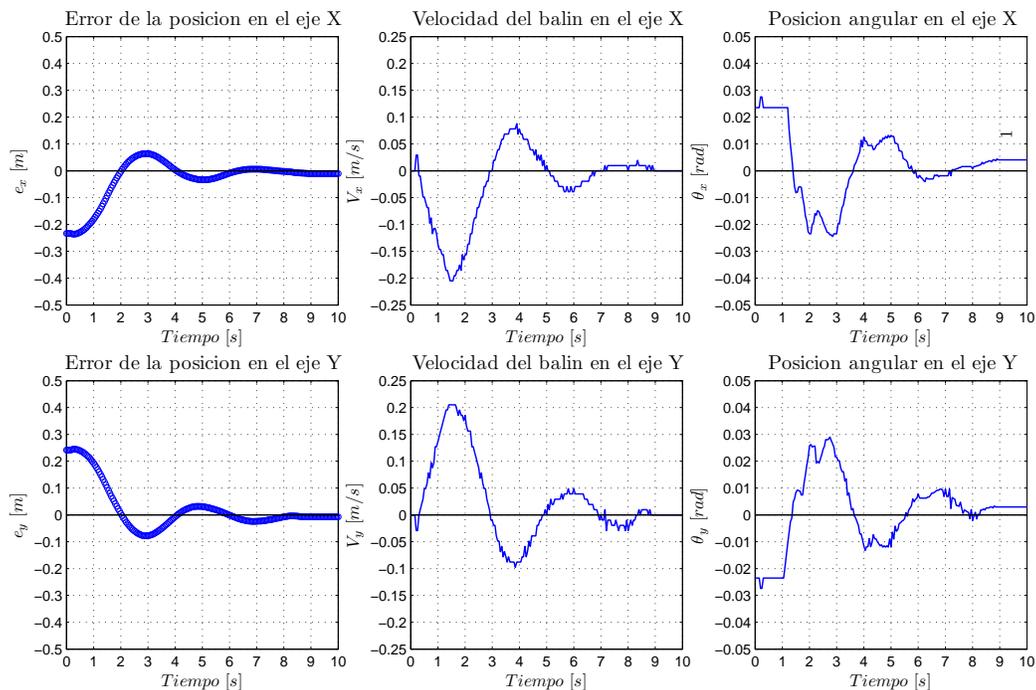


Figura D.12: Gráficas del Error, Velocidad y Señal de control, modelo real A4.

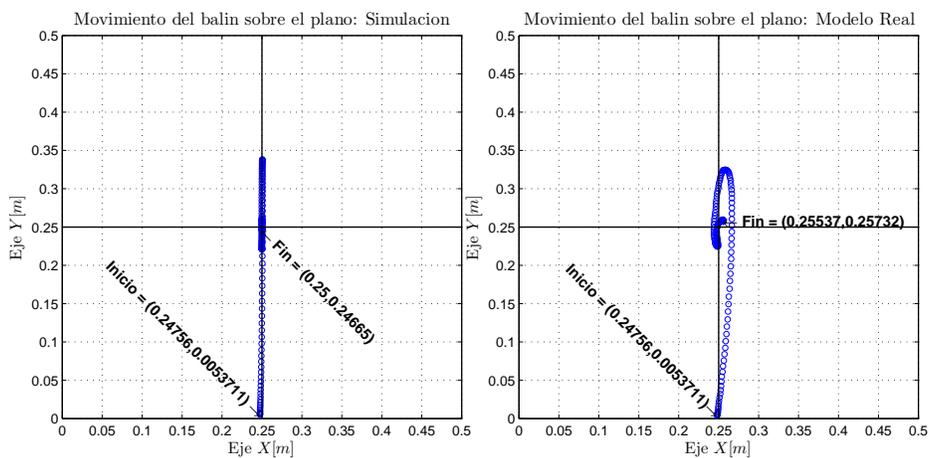


Figura D.13: Comparación entre el movimiento de la simulación y el modelo real, experimento A5.

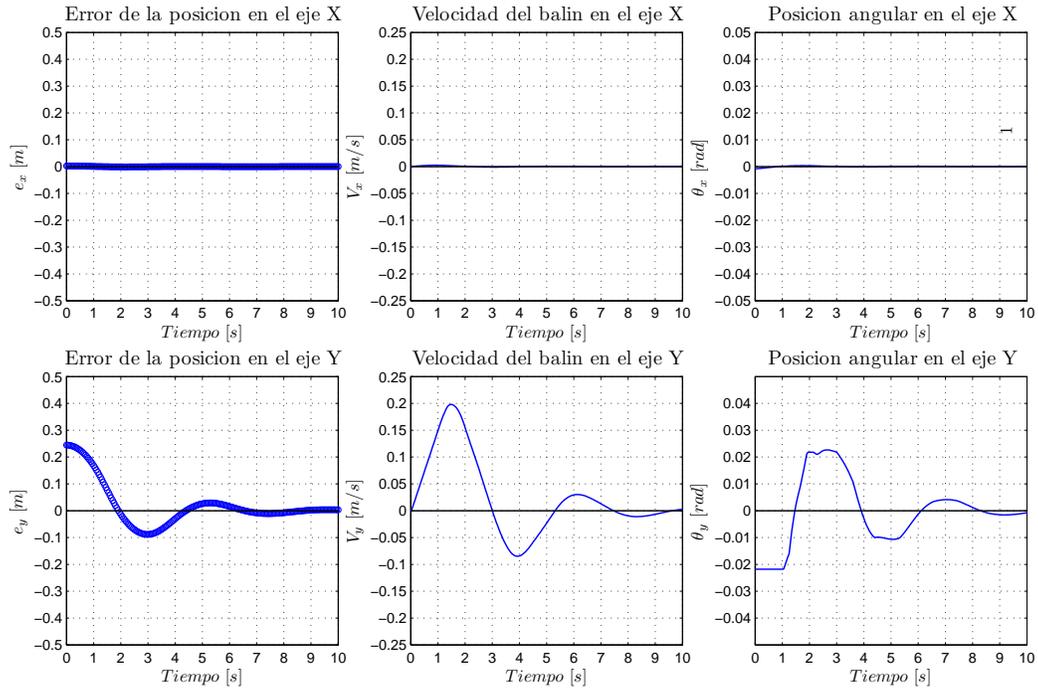


Figura D.14: Gráficas del Error, Velocidad y Señal de control, simulación A5.

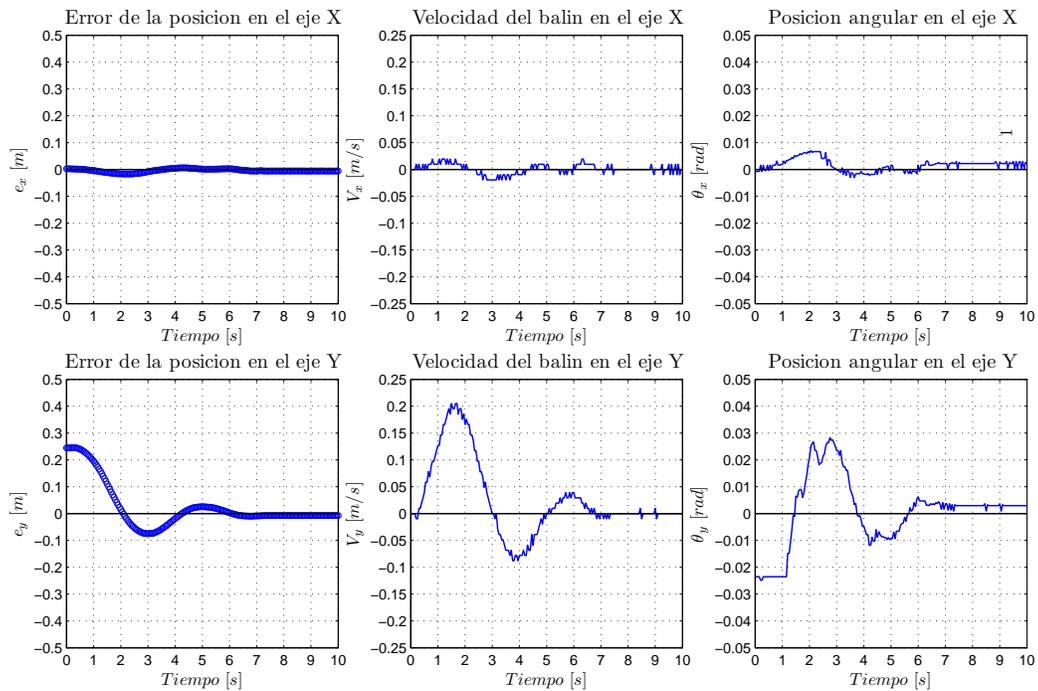


Figura D.15: Gráficas del Error, Velocidad y Señal de control, modelo real A5.

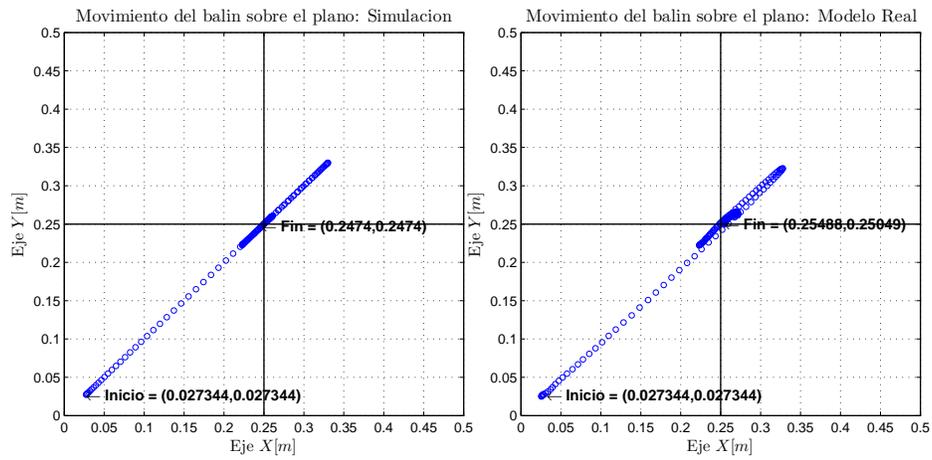


Figura D.16: Comparación entre el movimiento de la simulación y el modelo real, experimento A6.

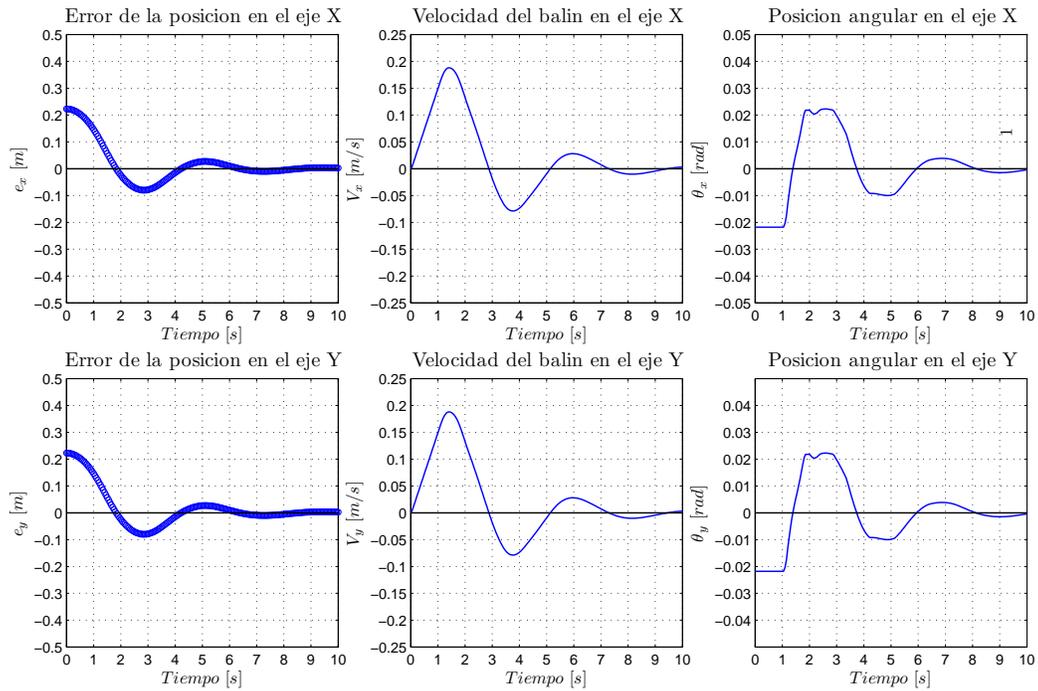


Figura D.17: Gráficas del Error, Velocidad y Señal de control, simulación A6.

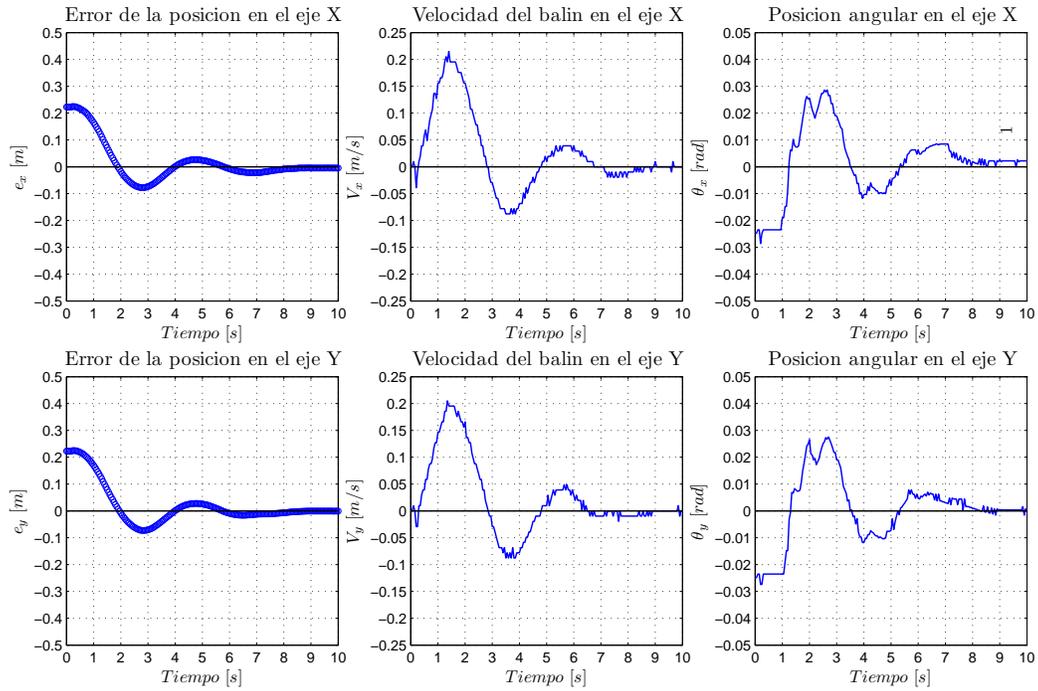


Figura D.18: Gráficas del Error, Velocidad y Señal de control, modelo real A6.

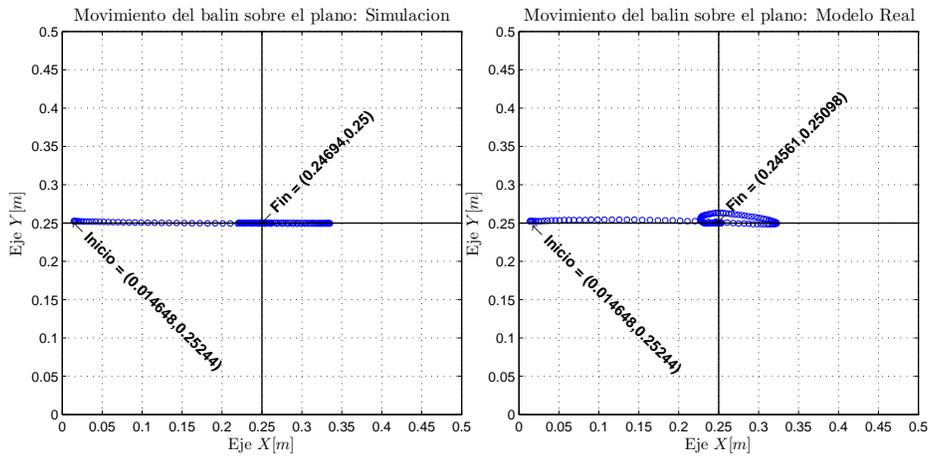


Figura D.19: Comparación entre el movimiento de la simulación y el modelo real, experimento A7.

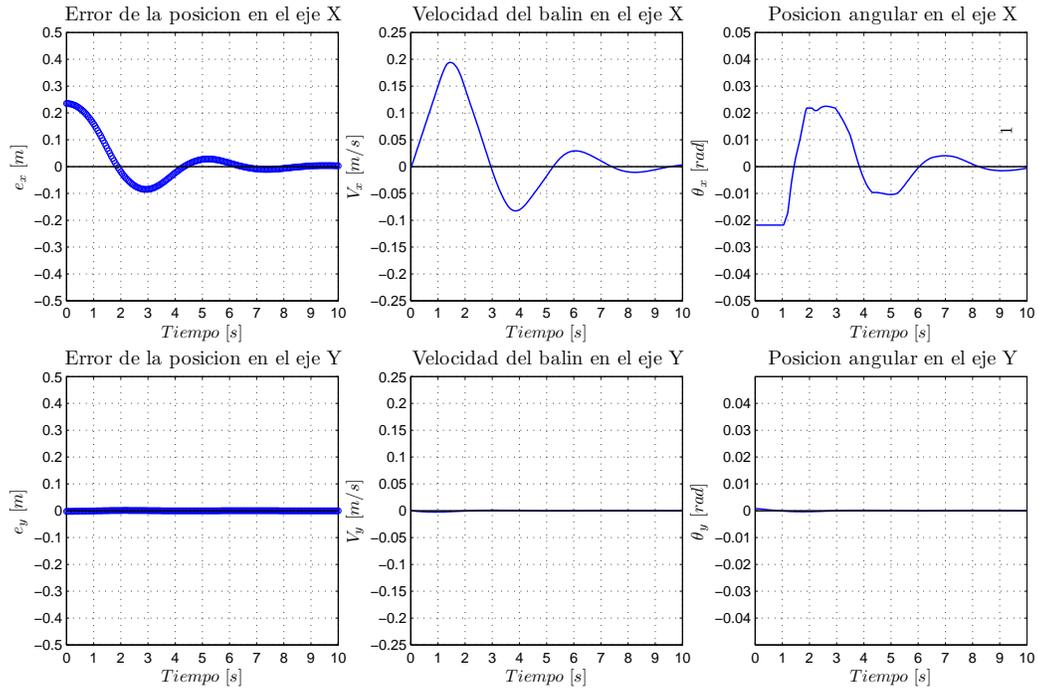


Figura D.20: Gráficas del Error, Velocidad y Señal de control, simulación A7.

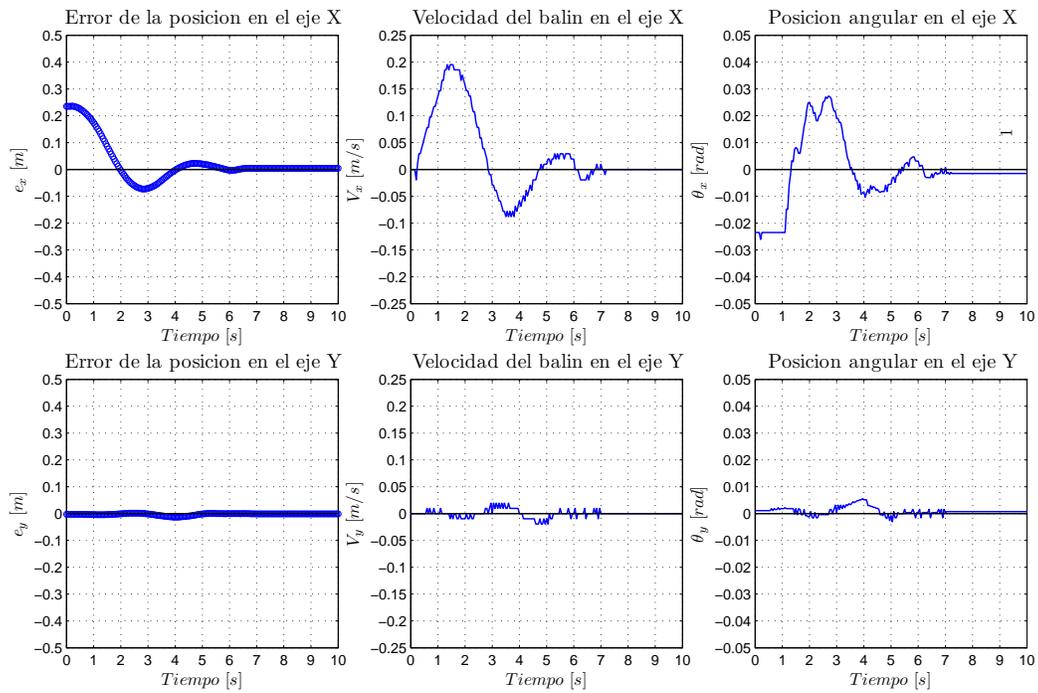


Figura D.21: Gráficas del Error, Velocidad y Señal de control, modelo real A7.

Apéndice E

Gráficas del experimento B

Los experimentos del uno al ocho tienen como posición deseada $(0.25cm, 0.25cm)$, los experimentos del nueve al doce $(0.125cm, 0.125cm)$ y los restantes $(0.4375cm, 0.4375cm)$.

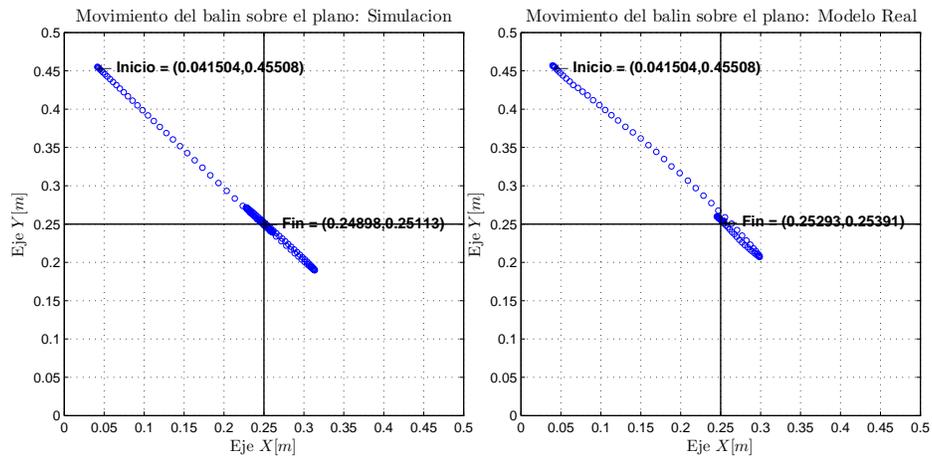


Figura E.1: Comparación entre el movimiento de la simulación y el modelo real, experimento B1.

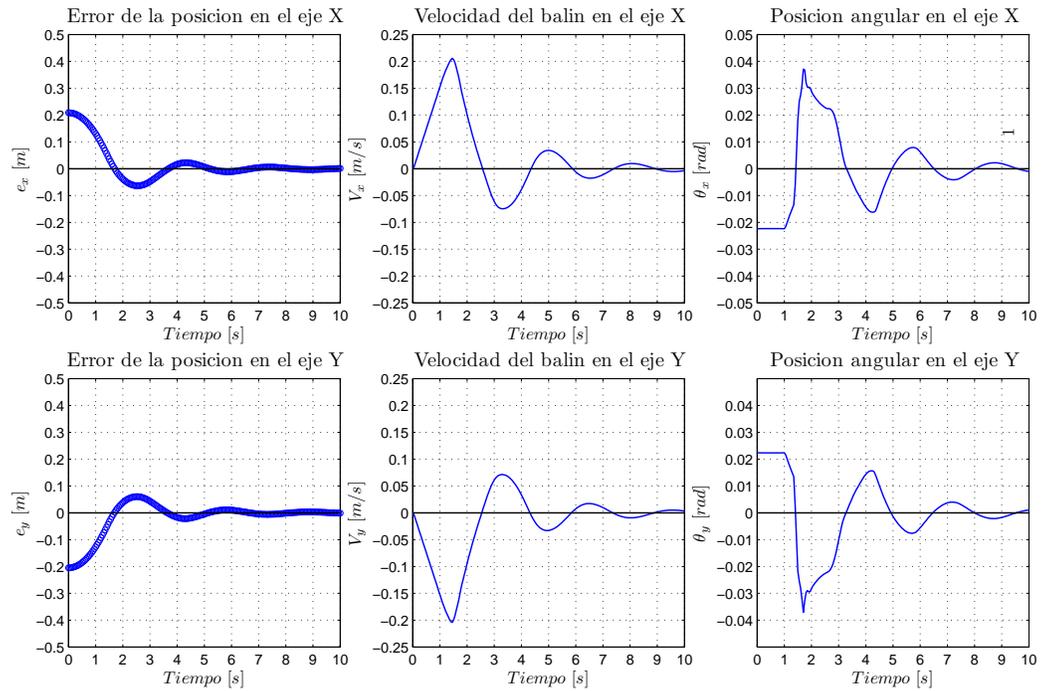


Figura E.2: Gráficas del Error, Velocidad y Señal de control, simulación B1.

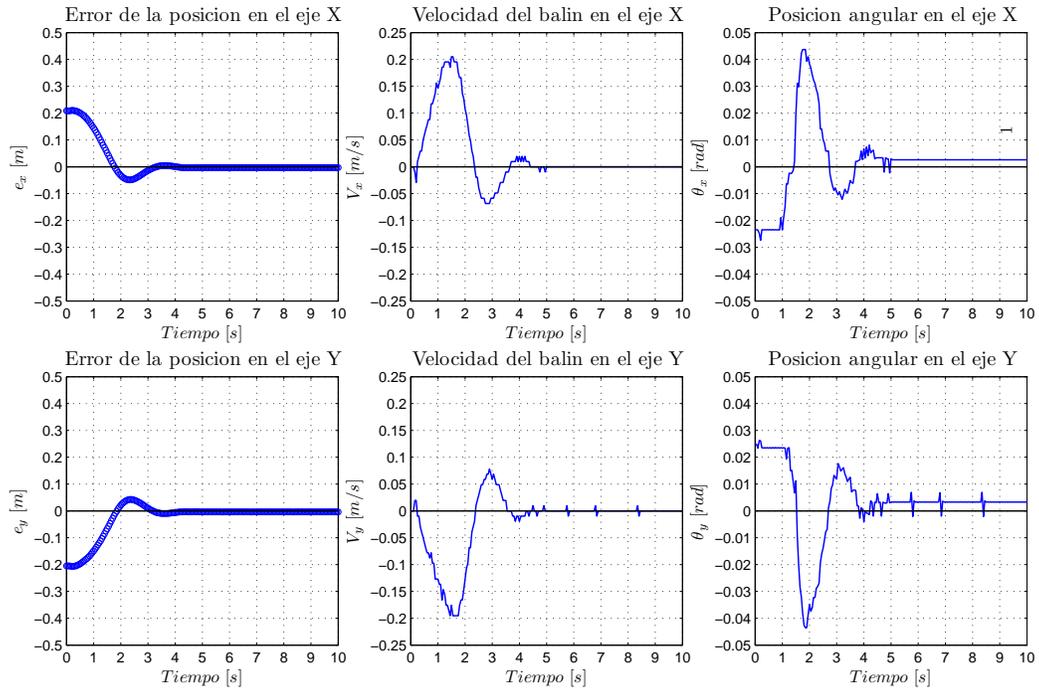


Figura E.3: Gráficas del Error, Velocidad y Señal de control, modelo real B1.

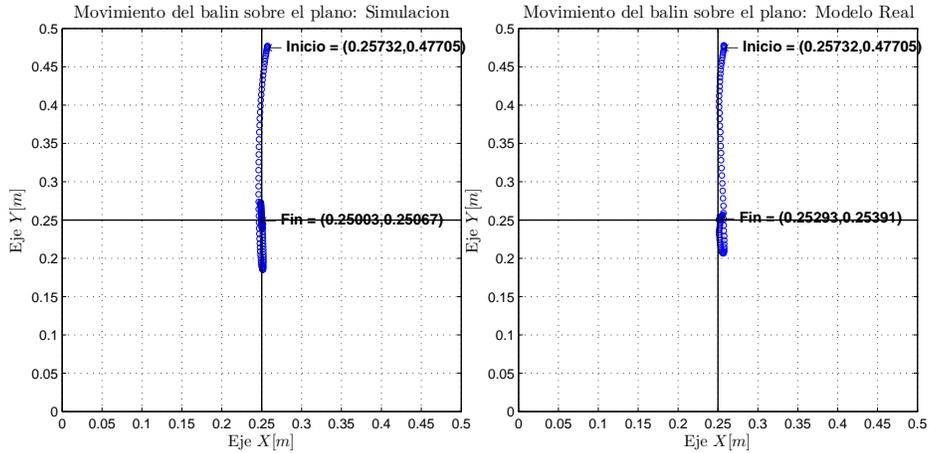


Figura E.4: Comparación entre el movimiento de la simulación y el modelo real, experimento B2.

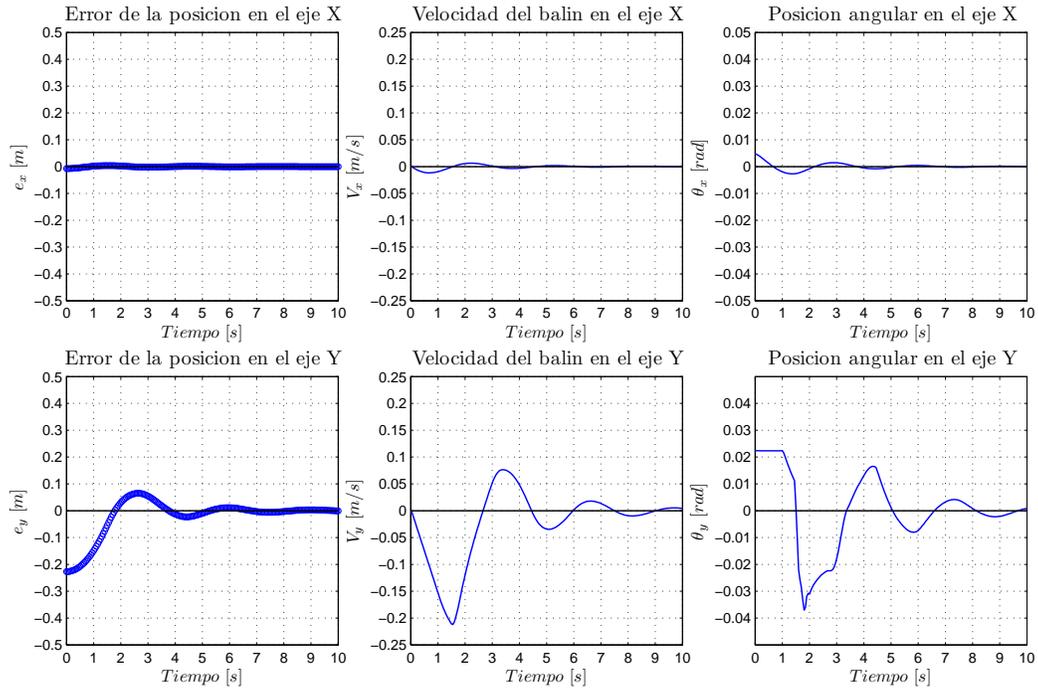


Figura E.5: Gráficas del Error, Velocidad y Señal de control, simulación B2.

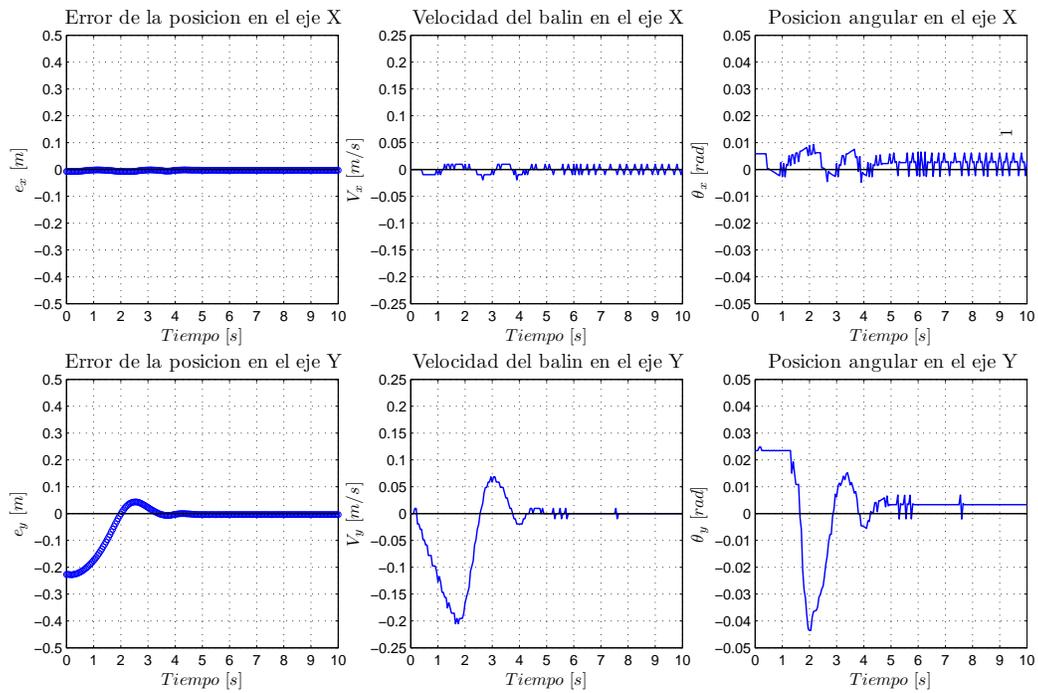


Figura E.6: Gráficas del Error, Velocidad y Señal de control, modelo real B2.

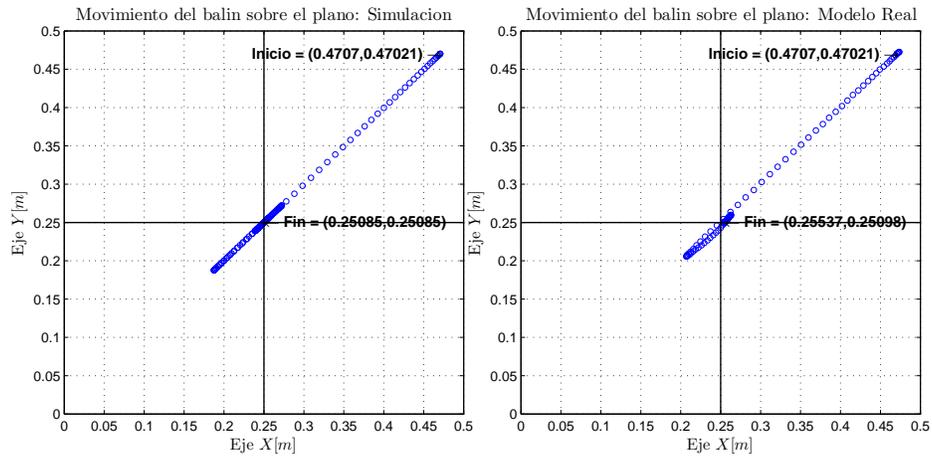


Figura E.7: Comparación entre el movimiento de la simulación y el modelo real, experimento B3.

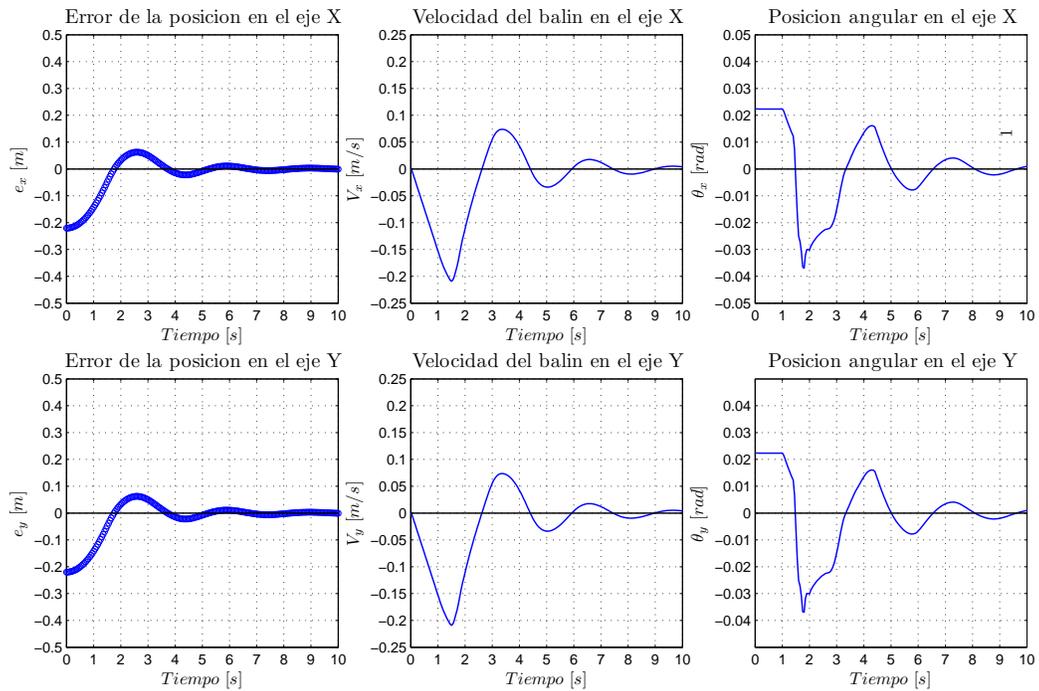


Figura E.8: Gráficas del Error, Velocidad y Señal de control, simulación B3.

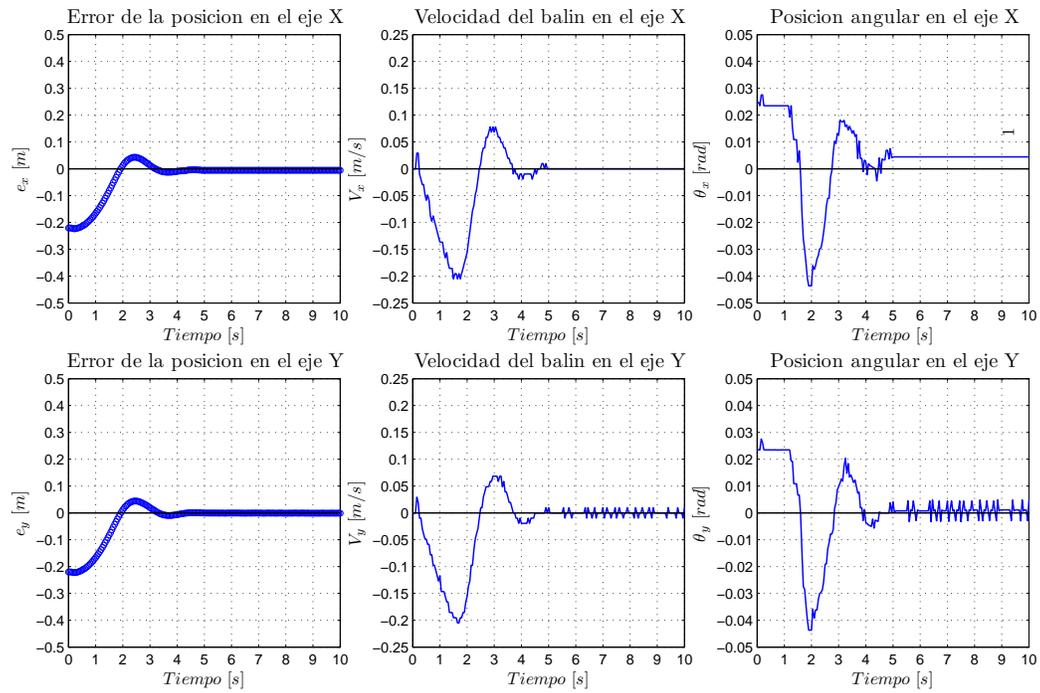


Figura E.9: Gráficas del Error, Velocidad y Señal de control, modelo real B3.

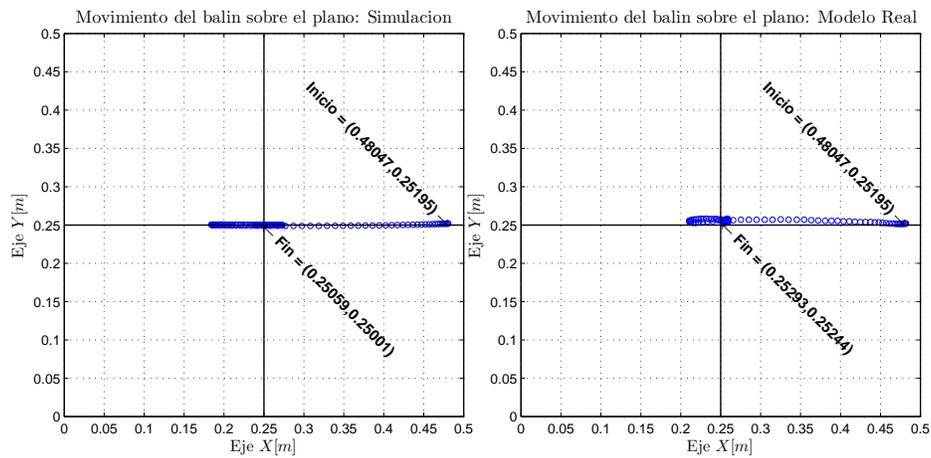


Figura E.10: Comparación entre el movimiento de la simulación y el modelo real, experimento B4.

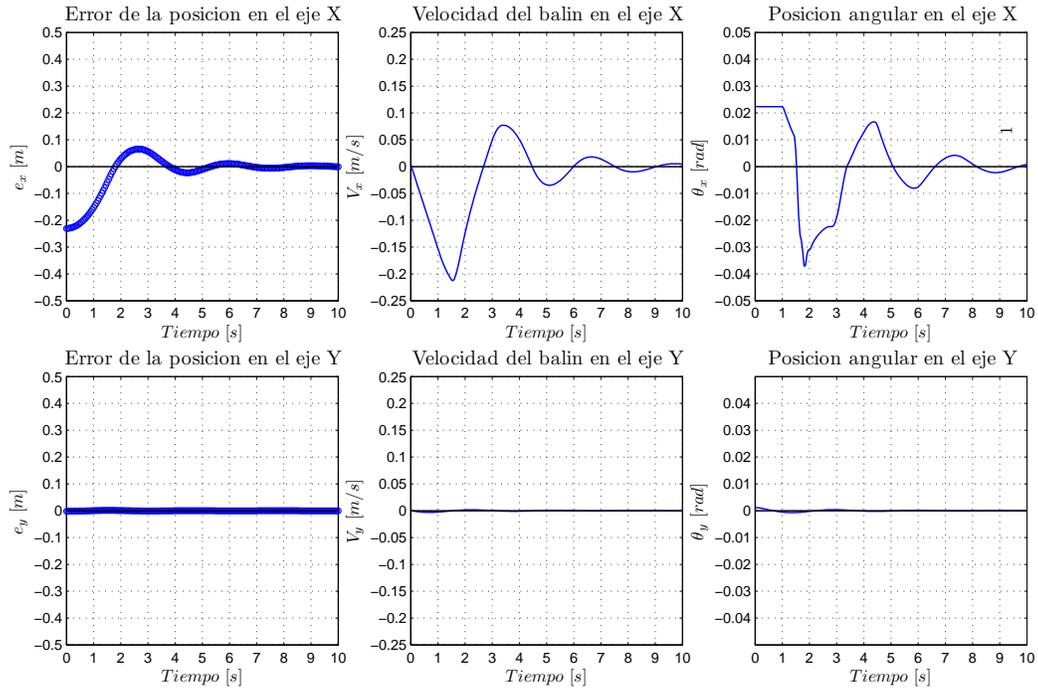


Figura E.11: Gráficas del Error, Velocidad y Señal de control, simulación B4.

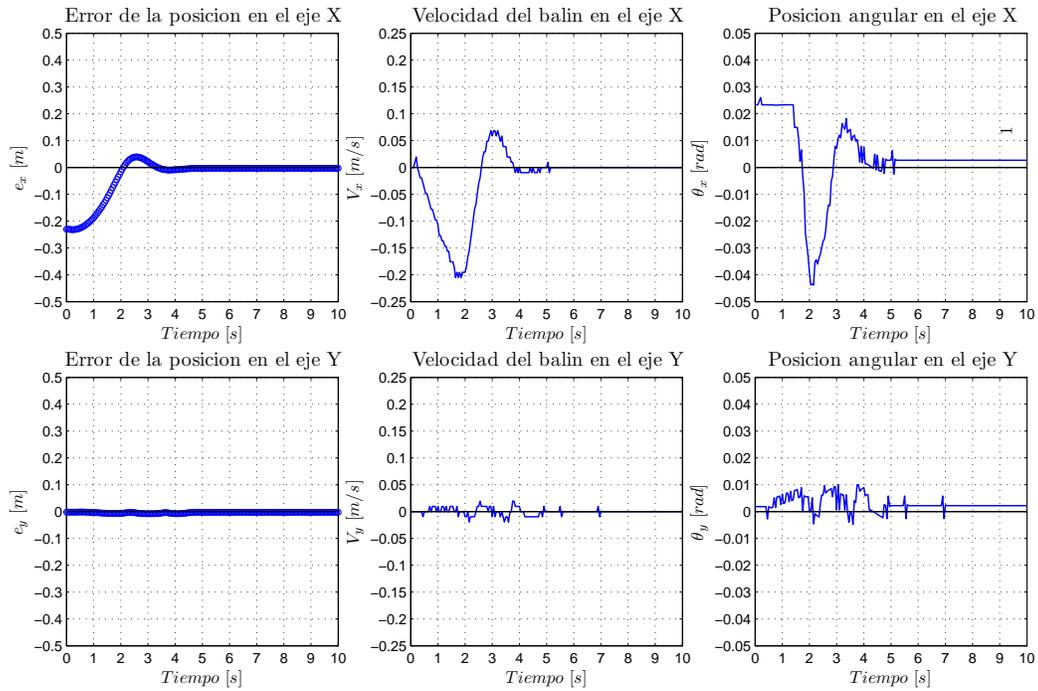


Figura E.12: Gráficas del Error, Velocidad y Señal de control, modelo real B4.

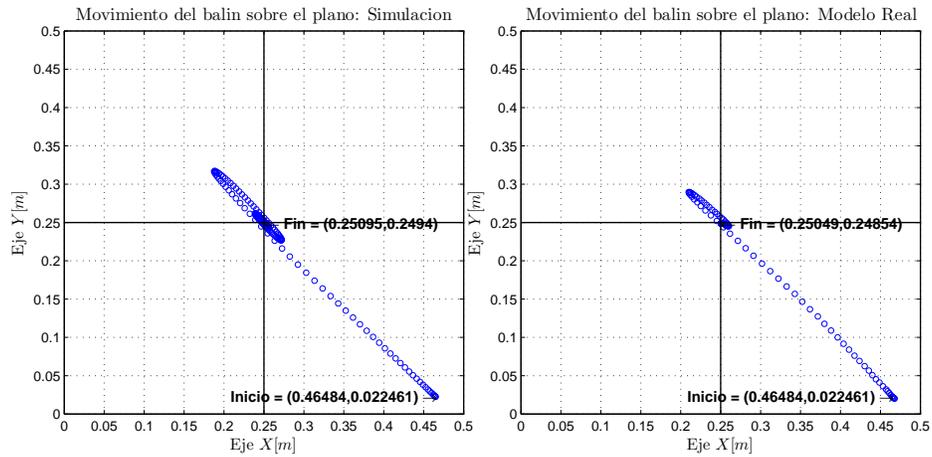


Figura E.13: Comparación entre el movimiento de la simulación y el modelo real, experimento B5.

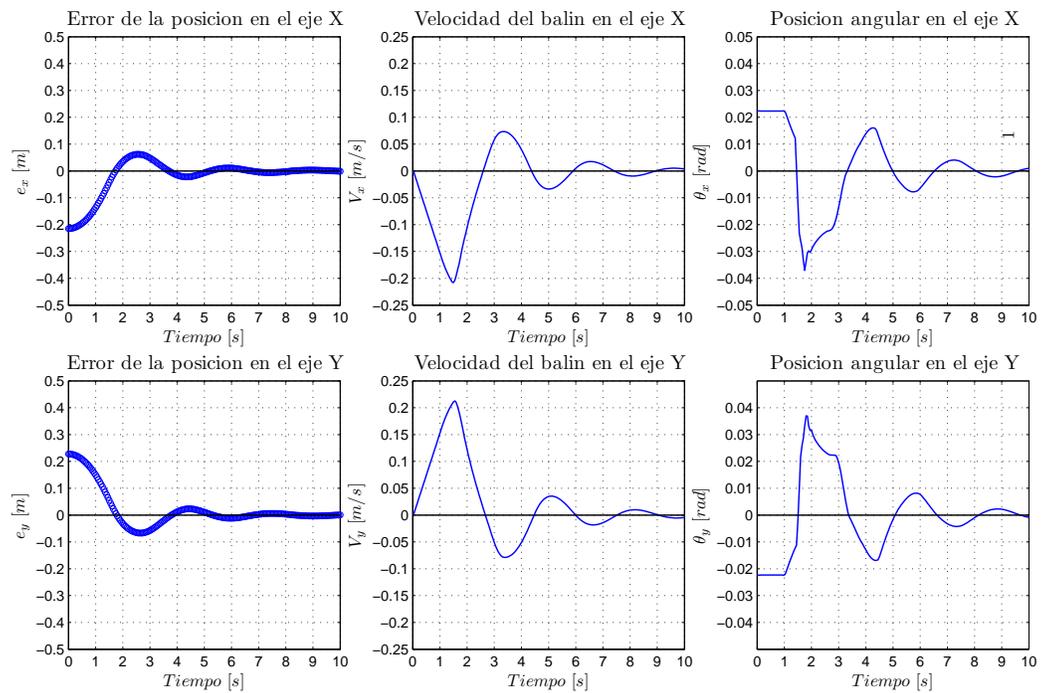


Figura E.14: Gráficas del Error, Velocidad y Señal de control, simulación B5.

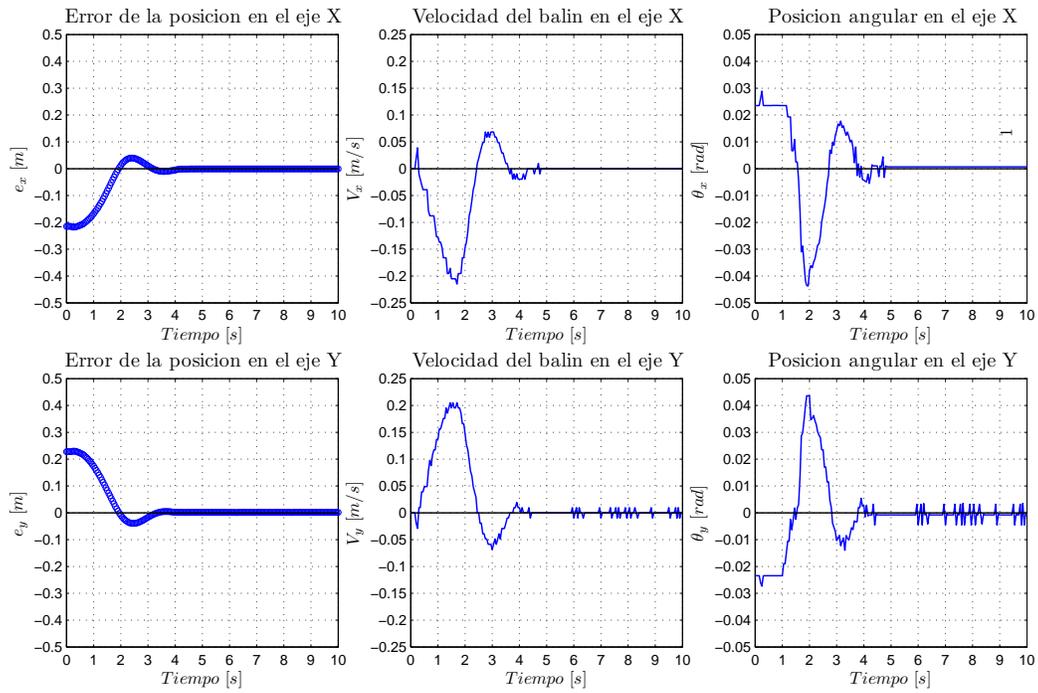


Figura E.15: Gráficas del Error, Velocidad y Señal de control, modelo real B5.

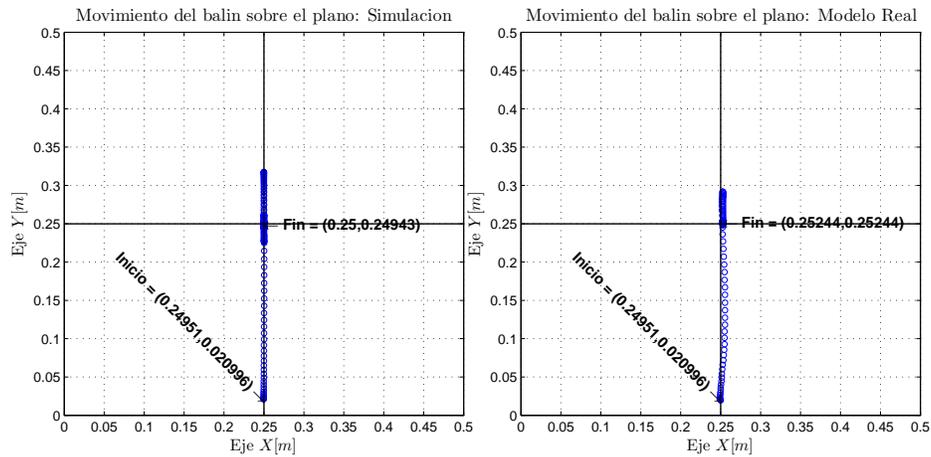


Figura E.16: Comparación entre el movimiento de la simulación y el modelo real, experimento B6.

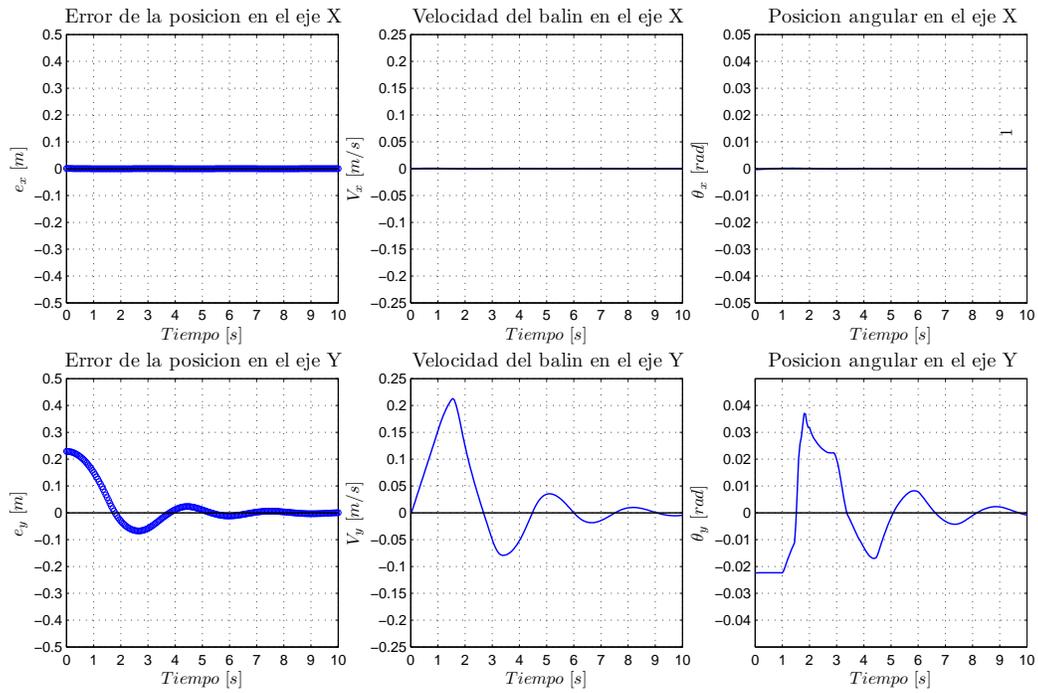


Figura E.17: Gráficas del Error, Velocidad y Señal de control, simulación B6.

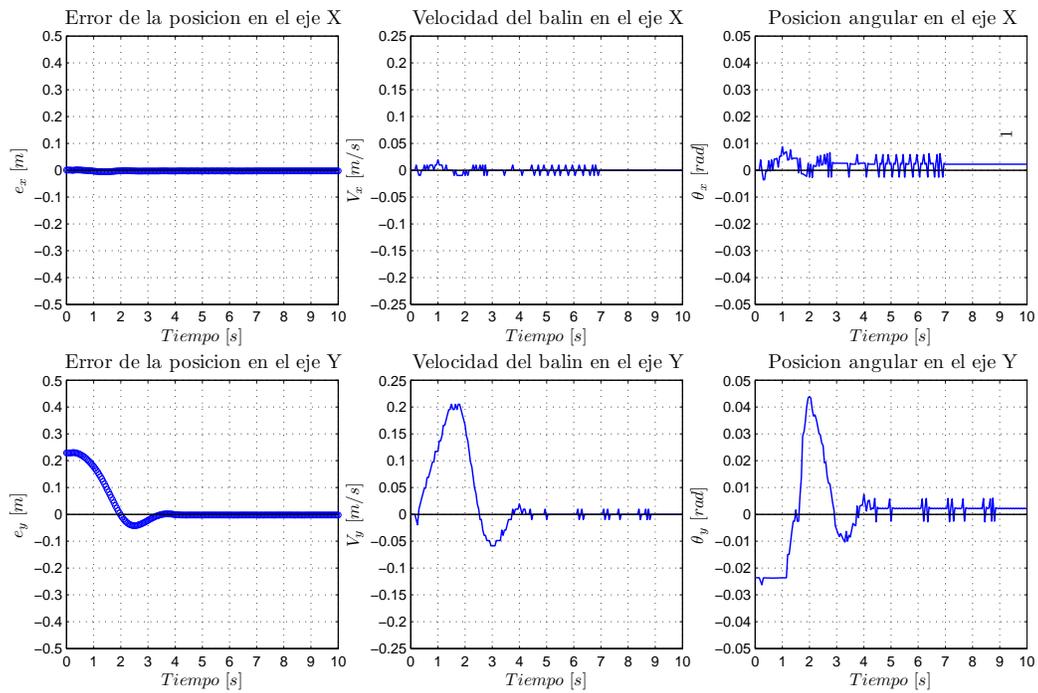


Figura E.18: Gráficas del Error, Velocidad y Señal de control, modelo real B6.

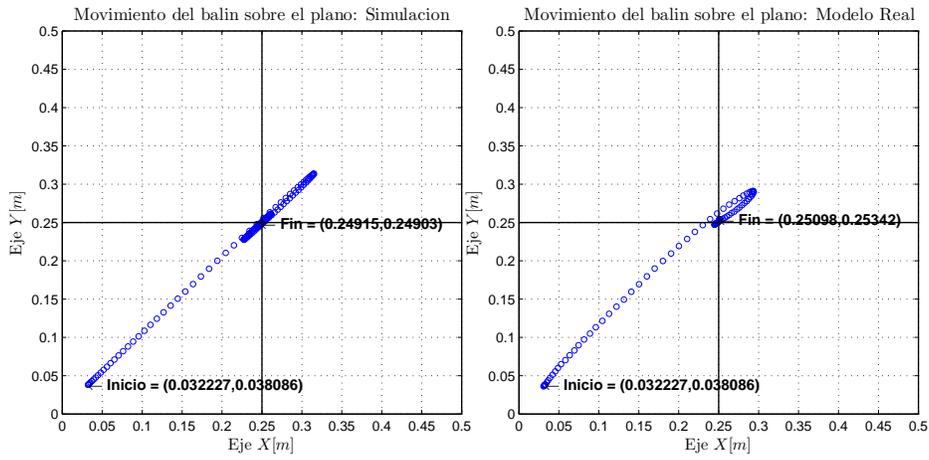


Figura E.19: Comparación entre el movimiento de la simulación y el modelo real, experimento B7.

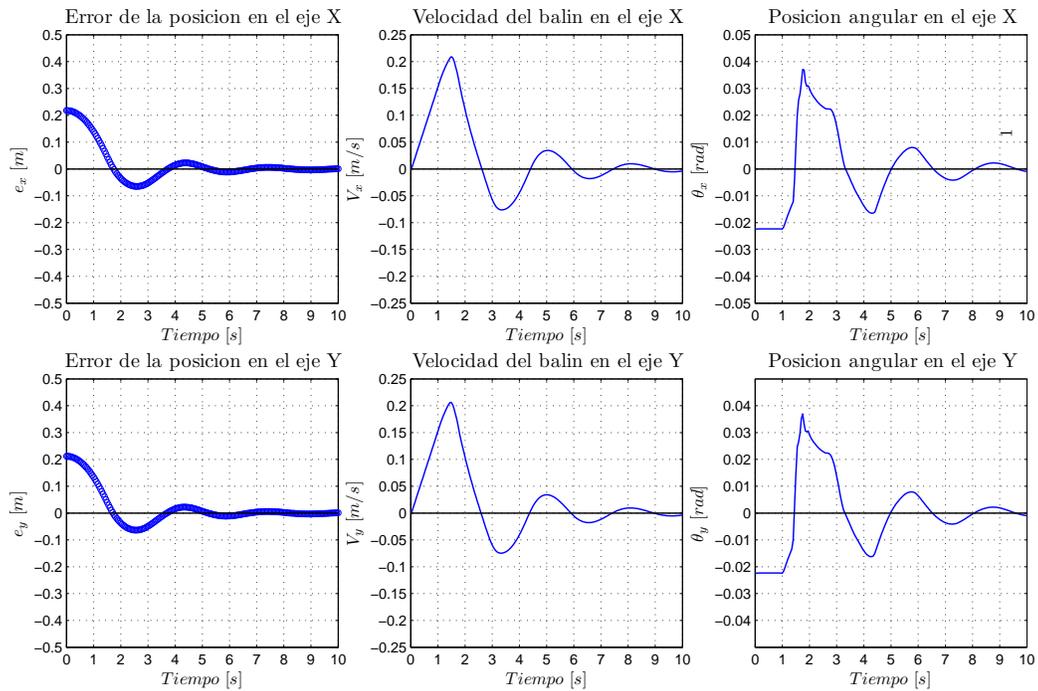


Figura E.20: Gráficas del Error, Velocidad y Señal de control, simulación B7.

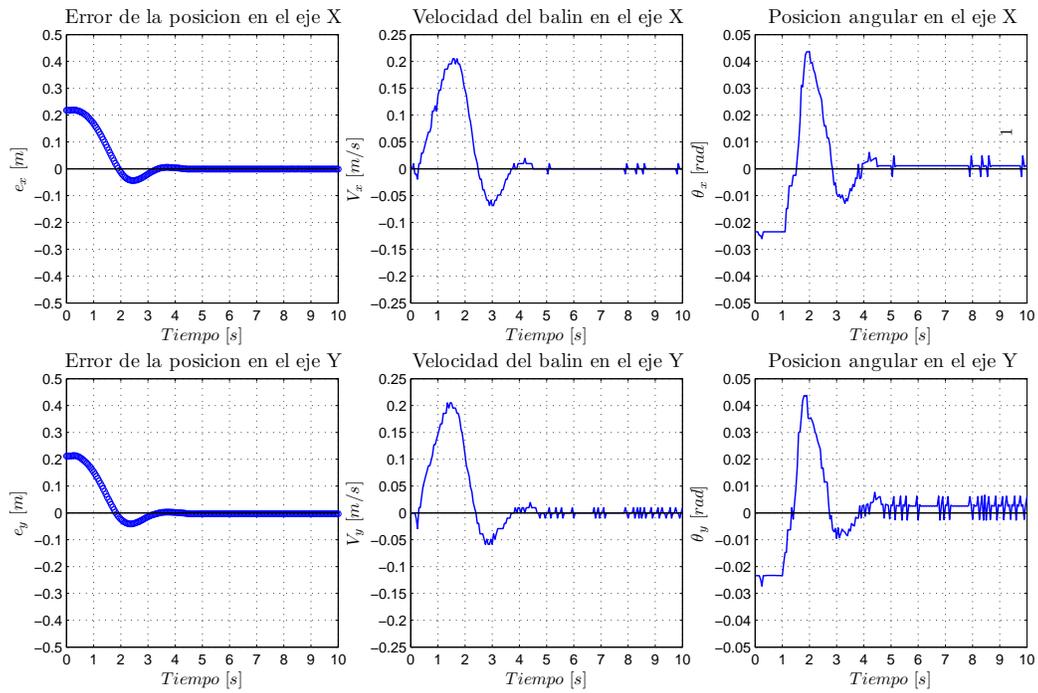


Figura E.21: Gráficas del Error, Velocidad y Señal de control, modelo real B7.

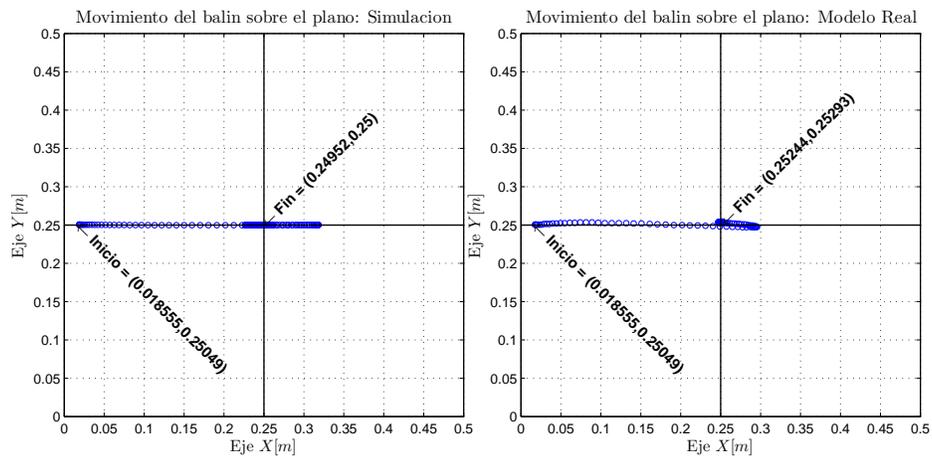


Figura E.22: Comparación entre el movimiento de la simulación y el modelo real, experimento B8.

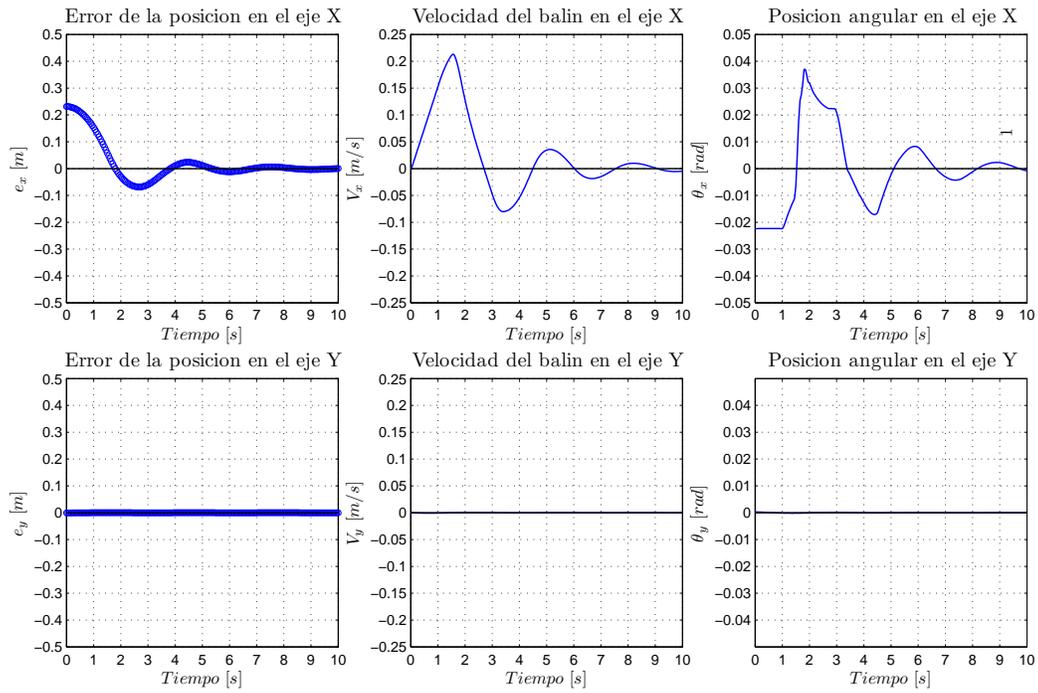


Figura E.23: Gráficas del Error, Velocidad y Señal de control, simulación B8.

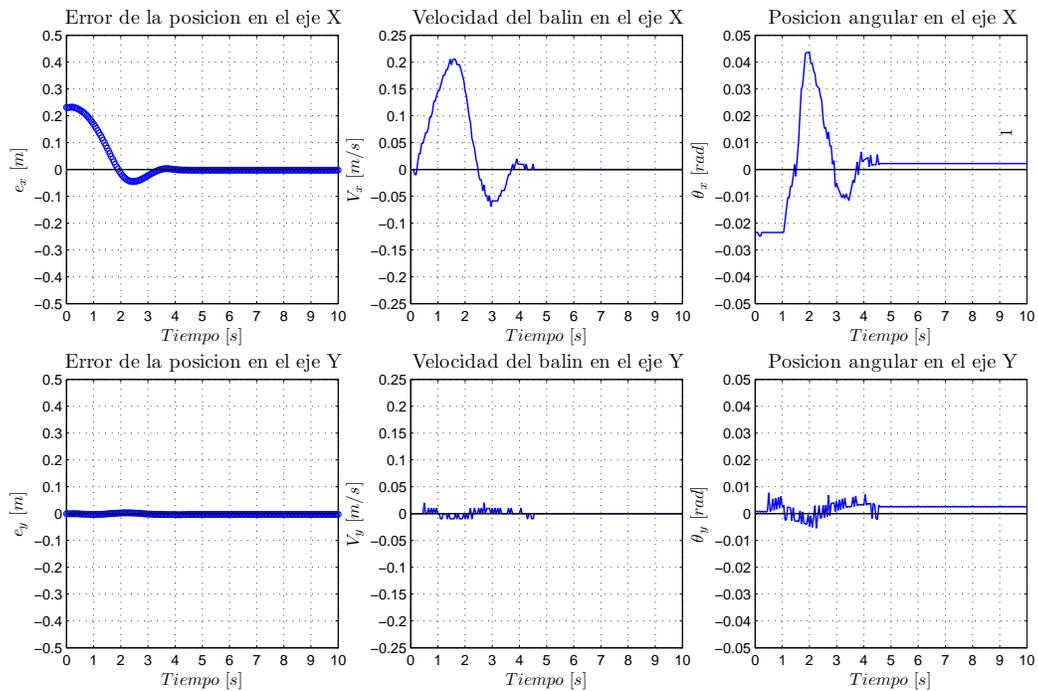


Figura E.24: Gráficas del Error, Velocidad y Señal de control, modelo real B8.

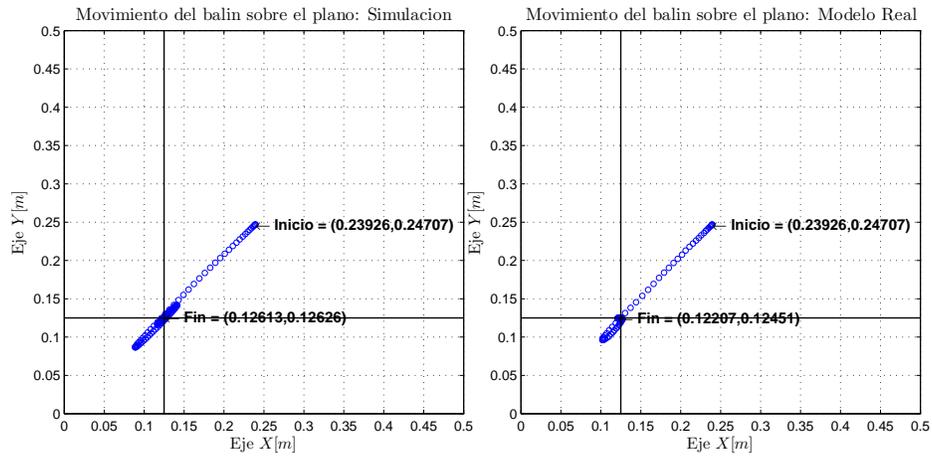


Figura E.25: Comparación entre el movimiento de la simulación y el modelo real, experimento B9.

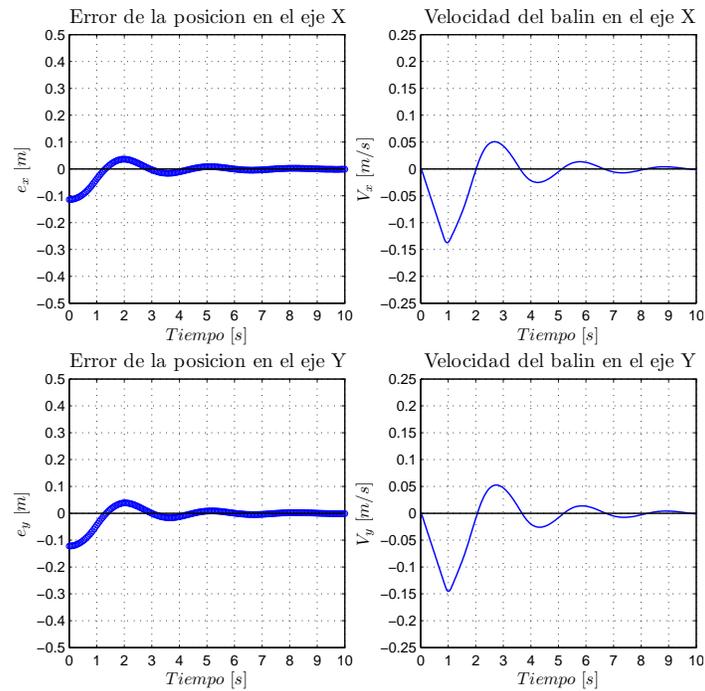


Figura E.26: Gráficas del Error, Velocidad y Señal de control, simulación B9.

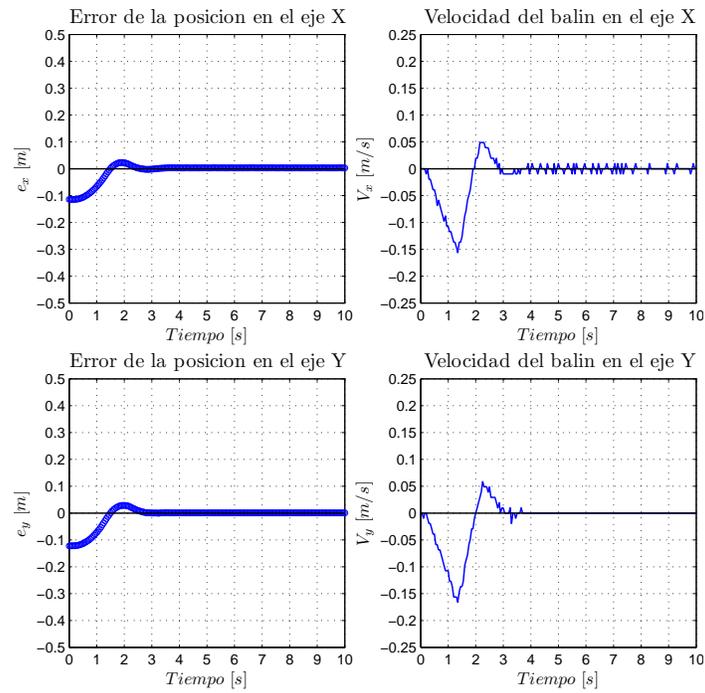


Figura E.27: Gráficas del Error, Velocidad y Señal de control, modelo real B9.

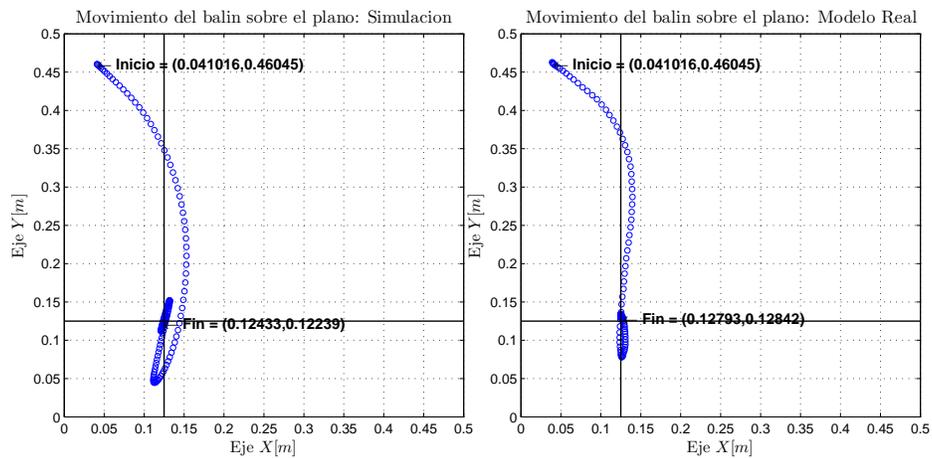


Figura E.28: Comparación entre el movimiento de la simulación y el modelo real, experimento B10.

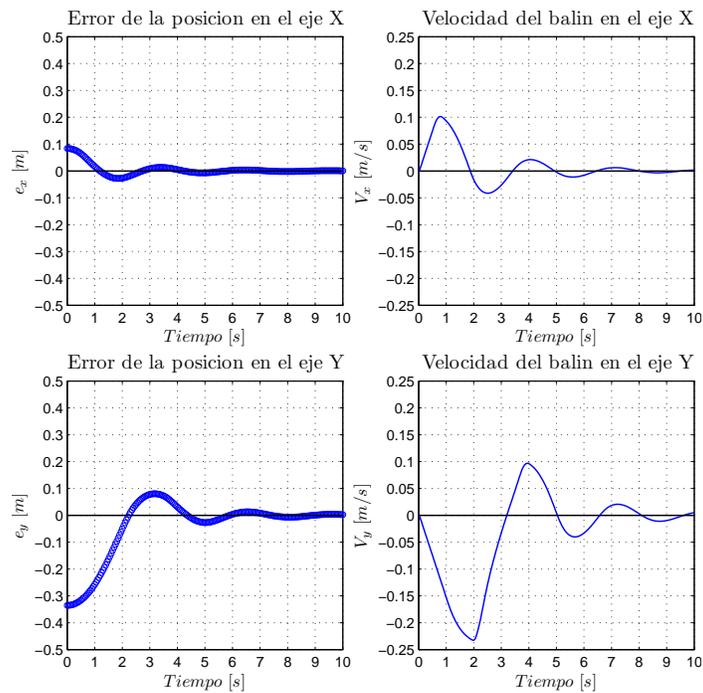


Figura E.29: Gráficas del Error, Velocidad y Señal de control, simulación B10.

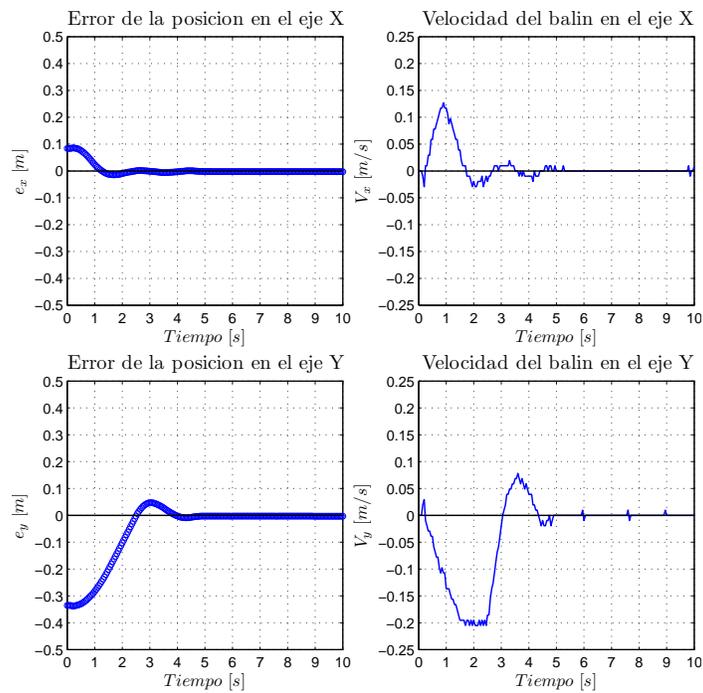


Figura E.30: Gráficas del Error, Velocidad y Señal de control, modelo real B10.

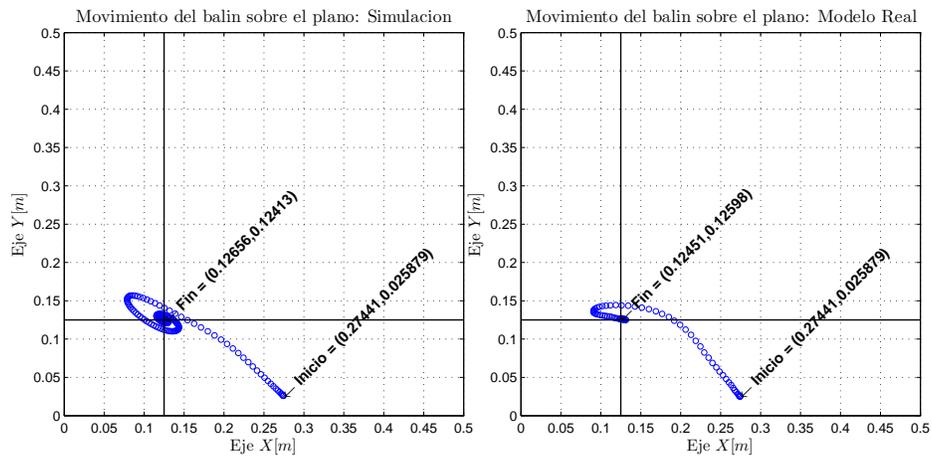


Figura E.31: Comparación entre el movimiento de la simulación y el modelo real, experimento B11.

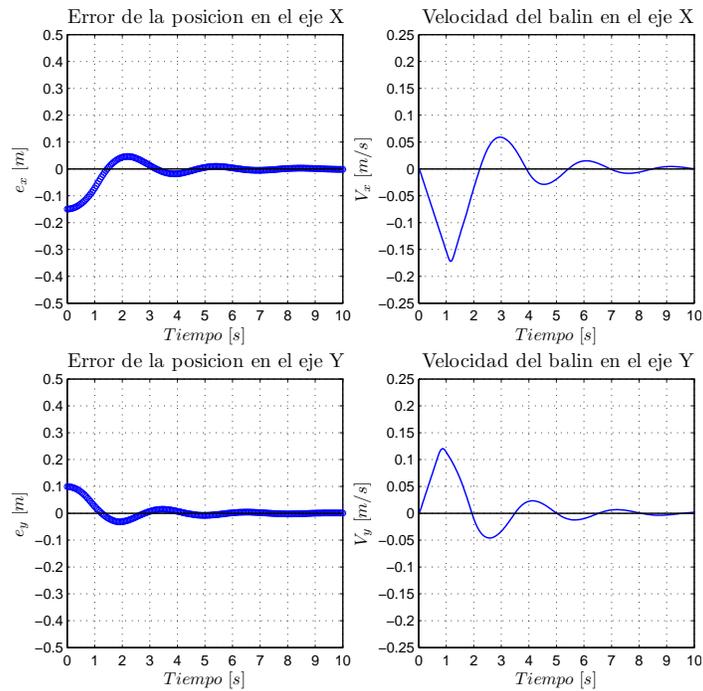


Figura E.32: Gráficas del Error, Velocidad y Señal de control, simulación B11.

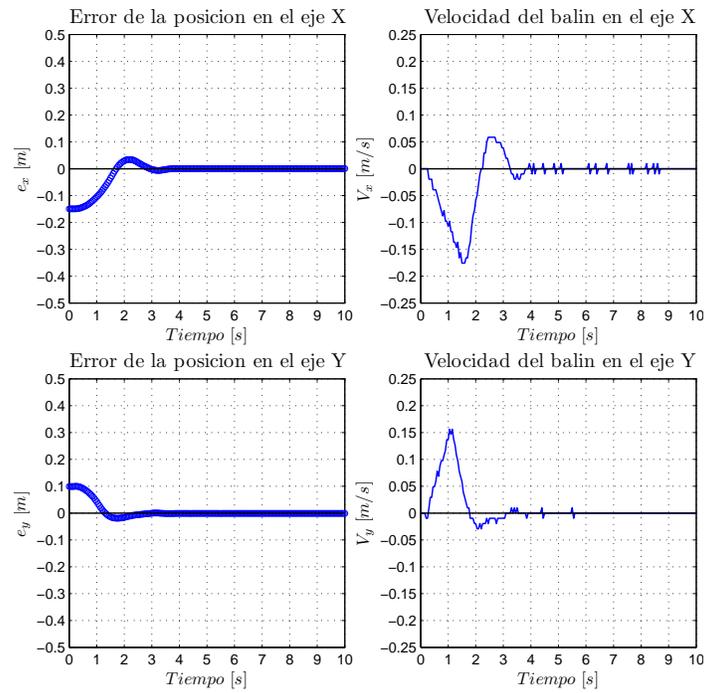


Figura E.33: Gráficas del Error, Velocidad y Señal de control, modelo real B11.

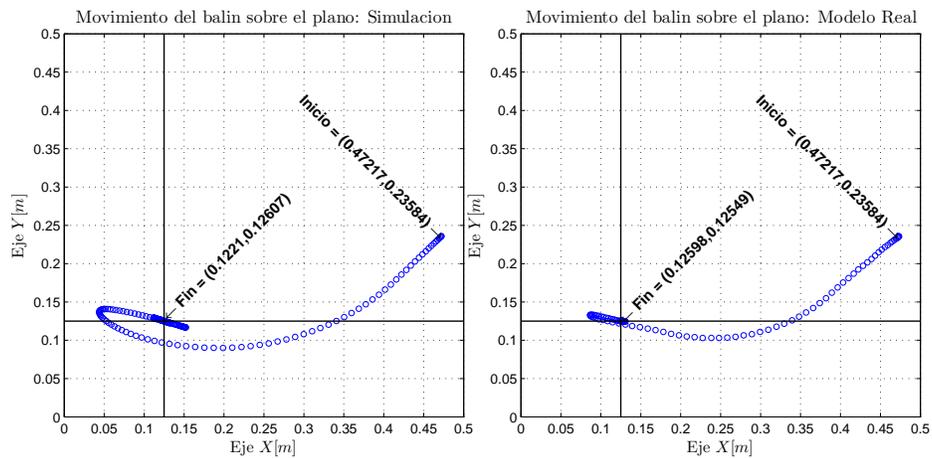


Figura E.34: Comparación entre el movimiento de la simulación y el modelo real, experimento B12.

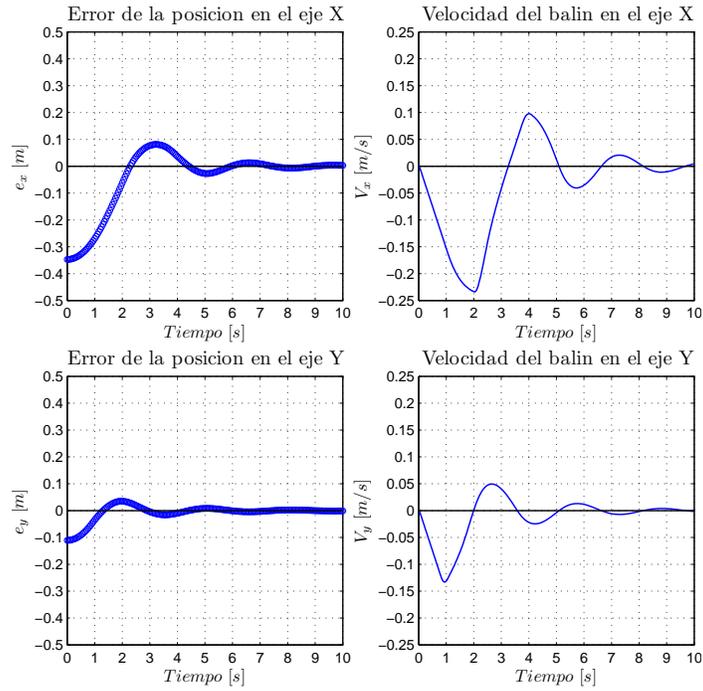


Figura E.35: Gráficas del Error, Velocidad y Señal de control, simulación B12.

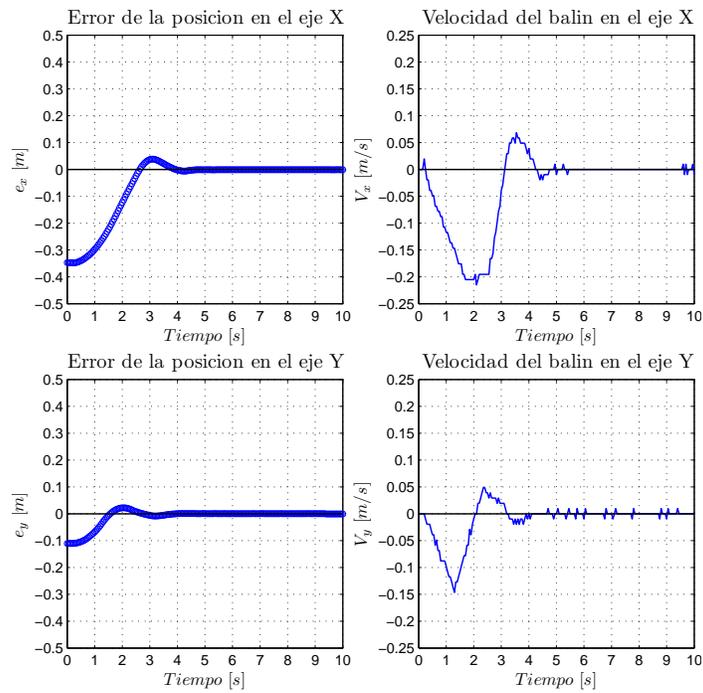


Figura E.36: Gráficas del Error, Velocidad y Señal de control, modelo real B12.

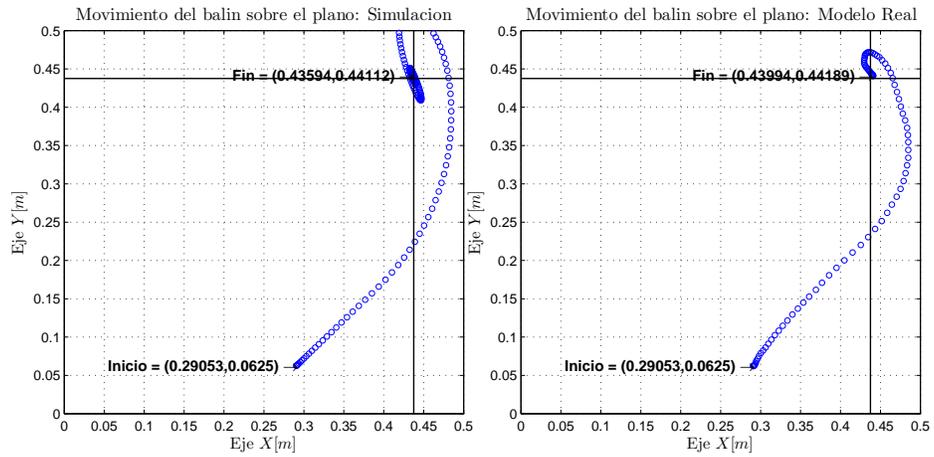


Figura E.37: Comparación entre el movimiento de la simulación y el modelo real, experimento B13.

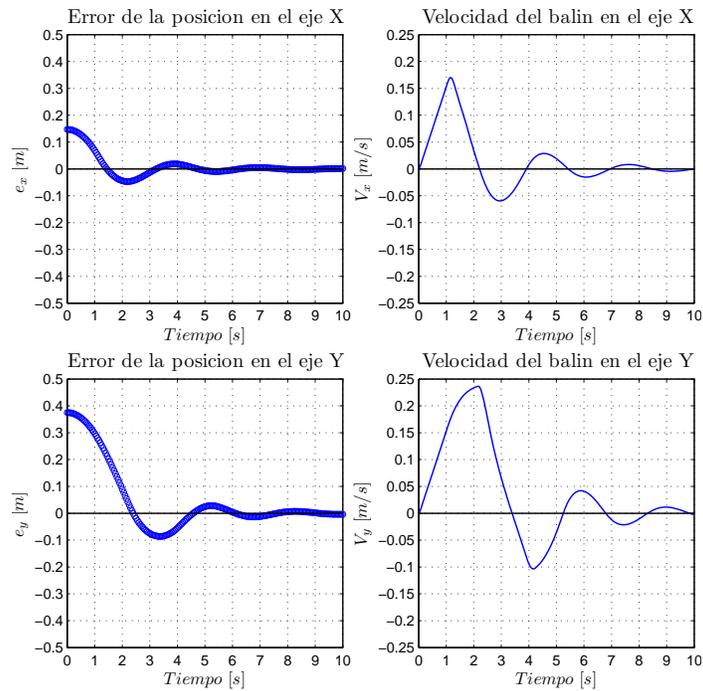


Figura E.38: Gráficas del Error, Velocidad y Señal de control, simulación B13.

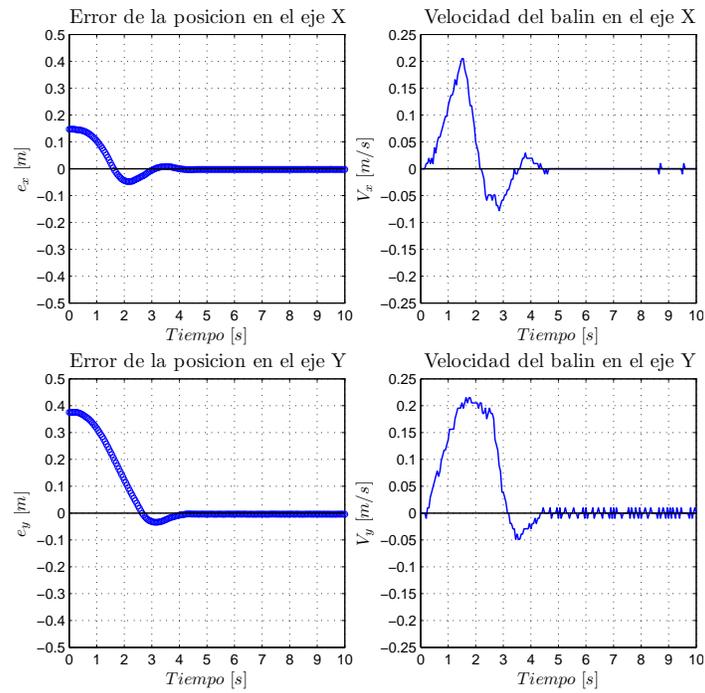


Figura E.39: Gráficas del Error, Velocidad y Señal de control, modelo real B13.

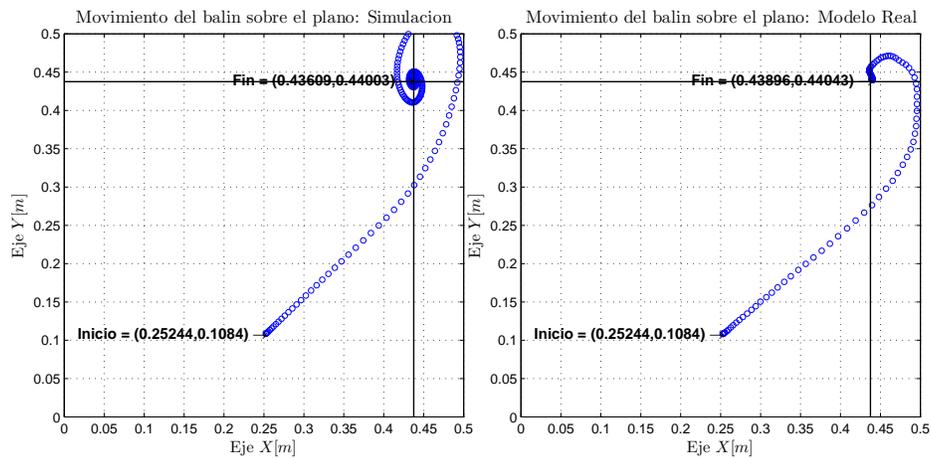


Figura E.40: Comparación entre el movimiento de la simulación y el modelo real, experimento B14.

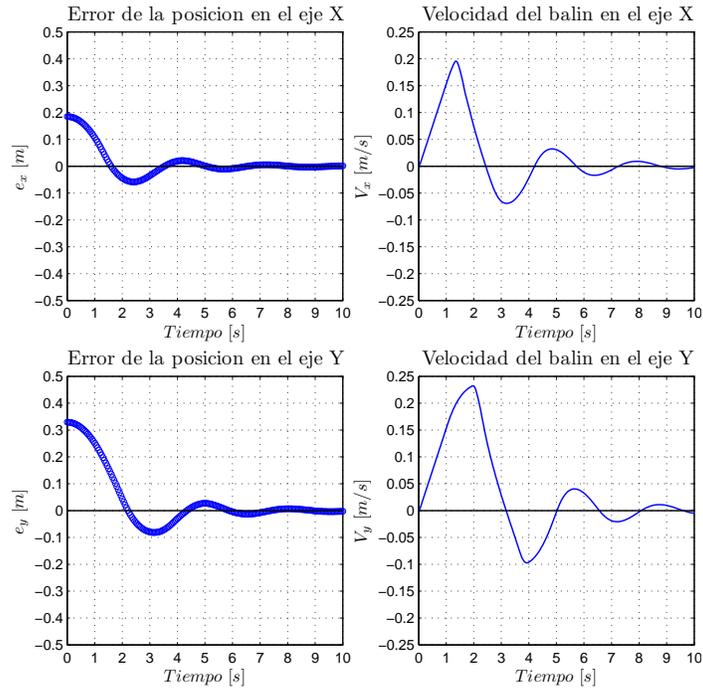


Figura E.41: Gráficas del Error, Velocidad y Señal de control, simulación B14.

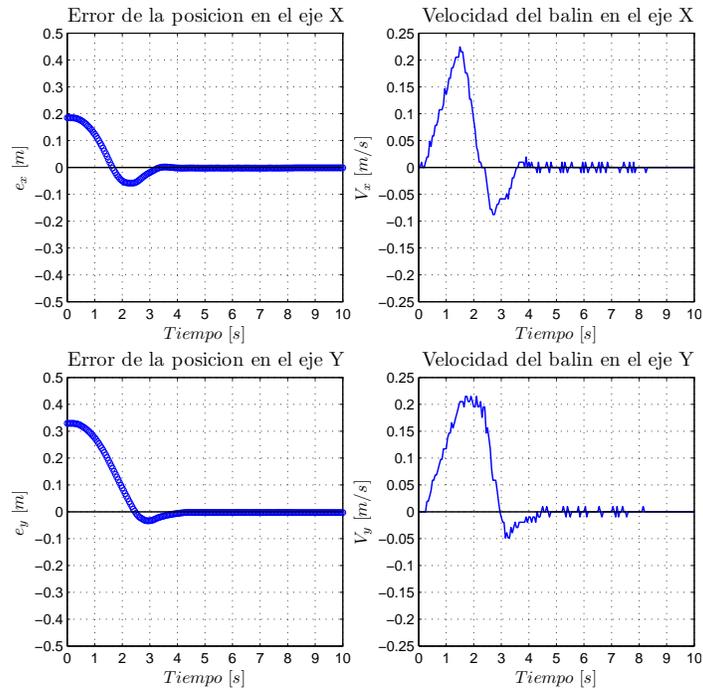


Figura E.42: Gráficas del Error, Velocidad y Señal de control, modelo real B14.

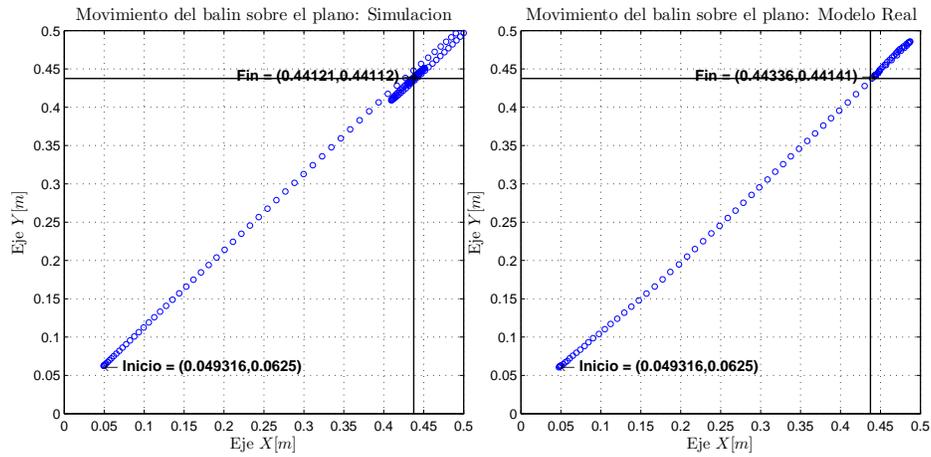


Figura E.43: Comparación entre el movimiento de la simulación y el modelo real, experimento B15.

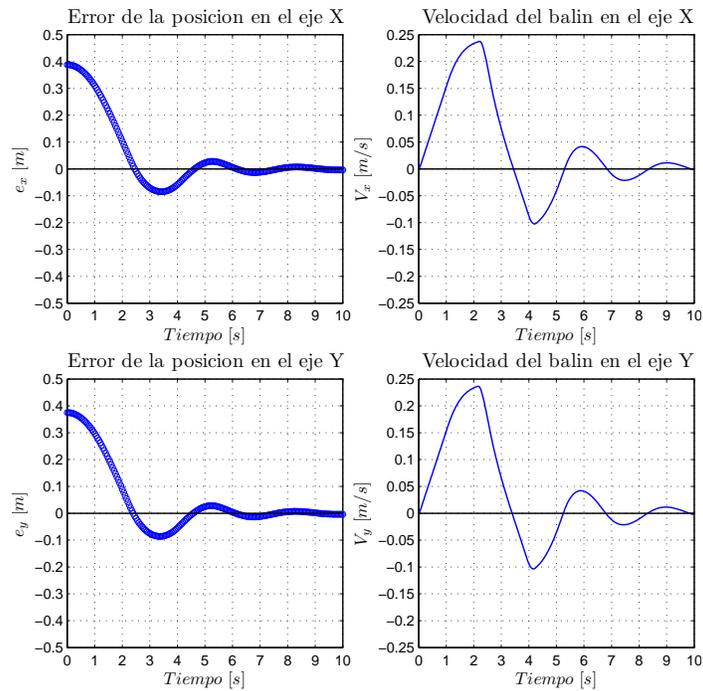


Figura E.44: Gráficas del Error, Velocidad y Señal de control, simulación B15.

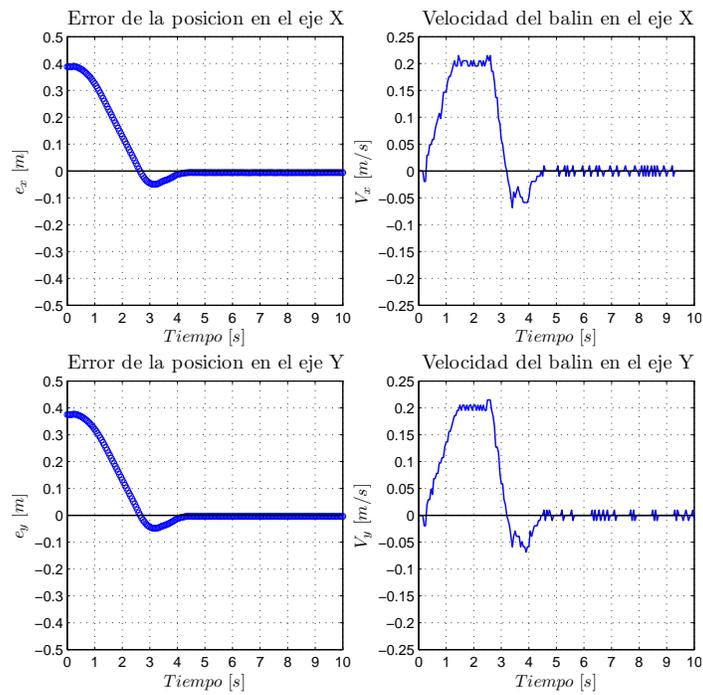


Figura E.45: Gráficas del Error, Velocidad y Señal de control, modelo real B15.

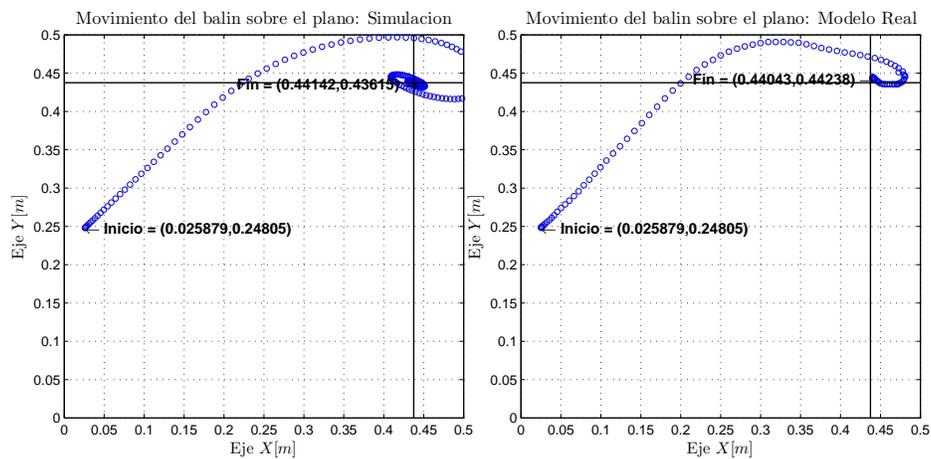


Figura E.46: Comparación entre el movimiento de la simulación y el modelo real, experimento B16.

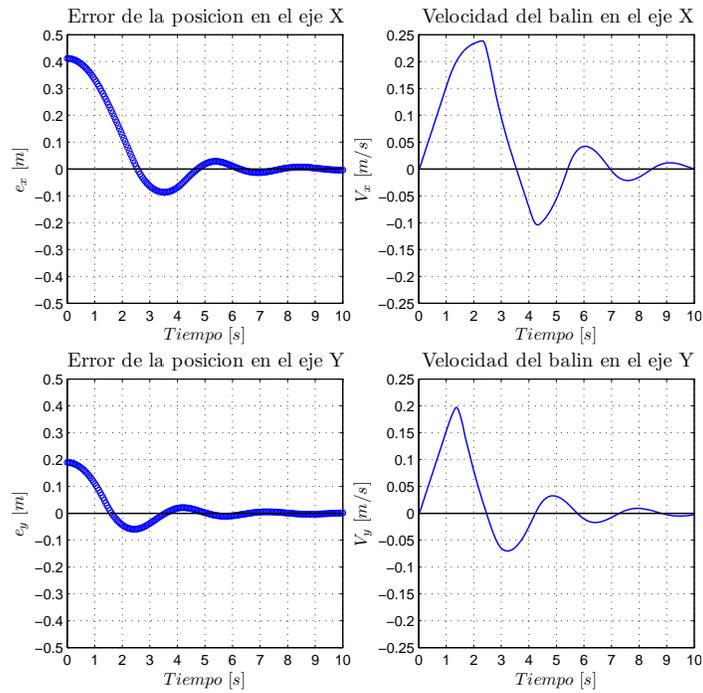


Figura E.47: Gráficas del Error, Velocidad y Señal de control, simulación B16.

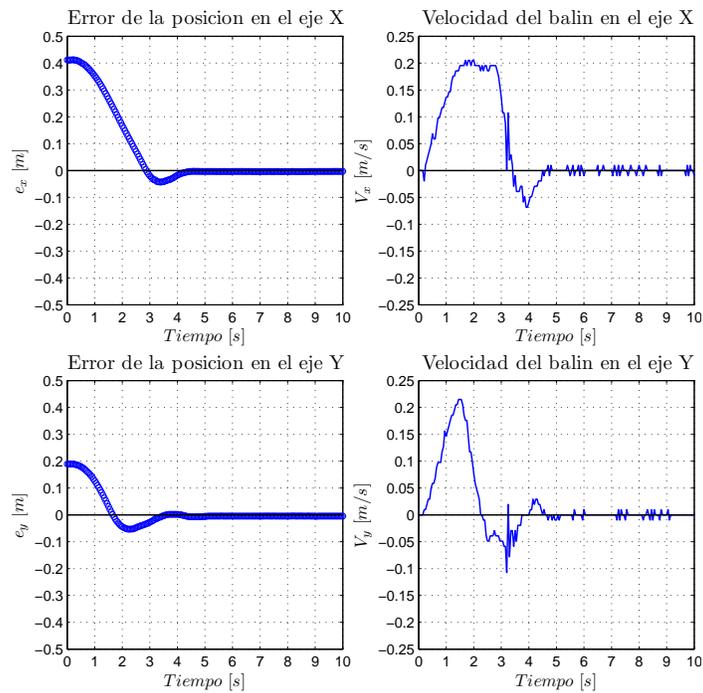


Figura E.48: Gráficas del Error, Velocidad y Señal de control, modelo real B16.

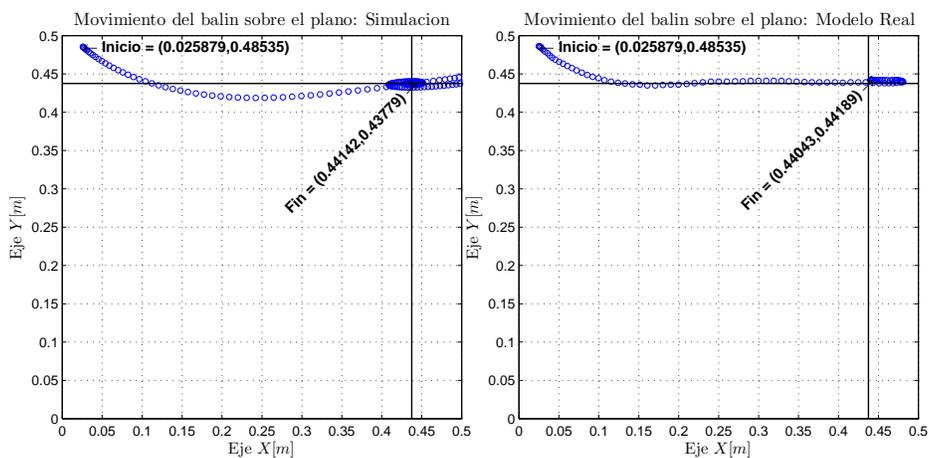


Figura E.49: Comparación entre el movimiento de la simulación y el modelo real, experimento B17.

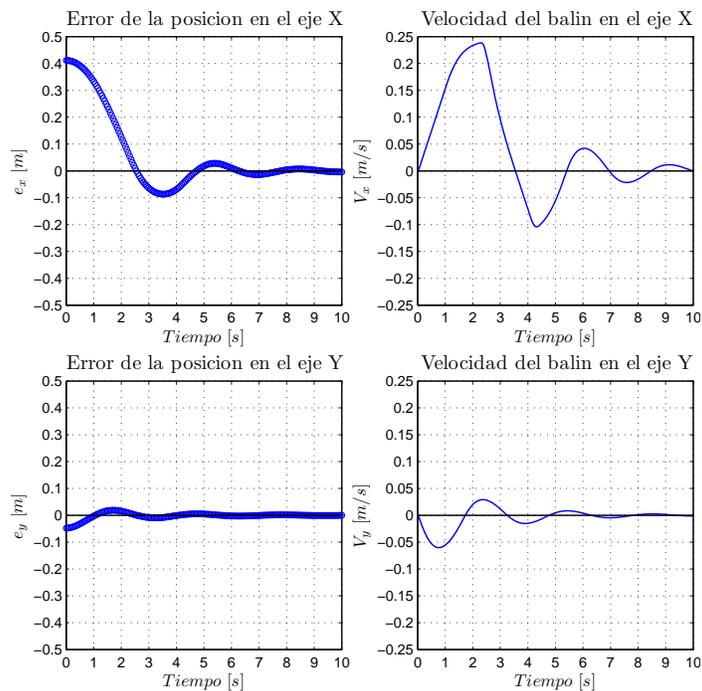


Figura E.50: Gráficas del Error, Velocidad y Señal de control, simulación B17.

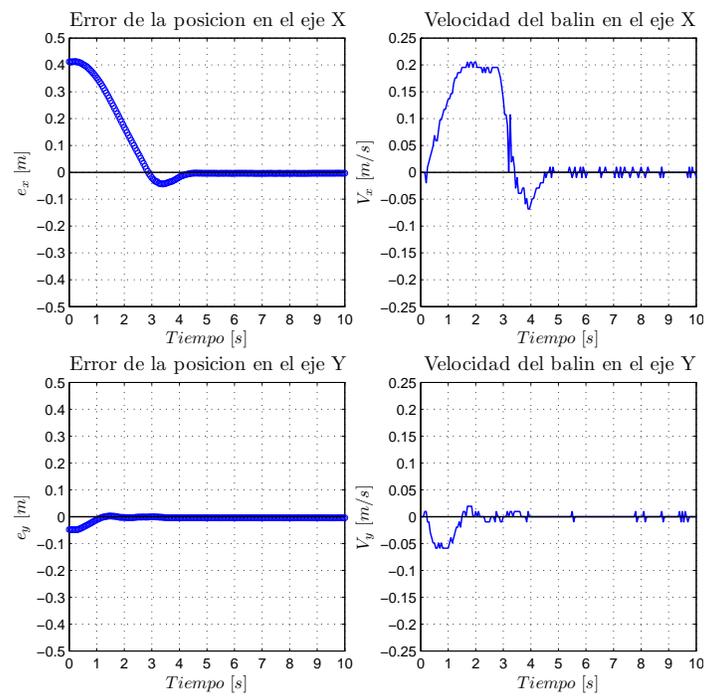


Figura E.51: Gráficas del Error, Velocidad y Señal de control, modelo real B17.

Apéndice F

Gráficas del experimento C

Los experimentos del uno al ocho tienen como posición deseada $(0.25\text{cm}, 0.25\text{cm})$, los experimentos del nueve al doce $(0.125\text{cm}, 0.125\text{cm})$ y los restantes $(0.4375\text{cm}, 0.4375\text{cm})$.

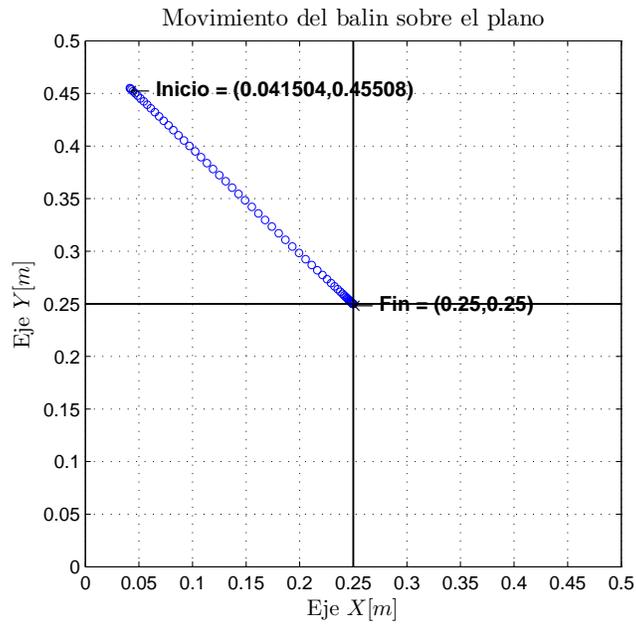


Figura F.1: Comparación entre el movimiento de la simulación y el modelo real, experimento C1.

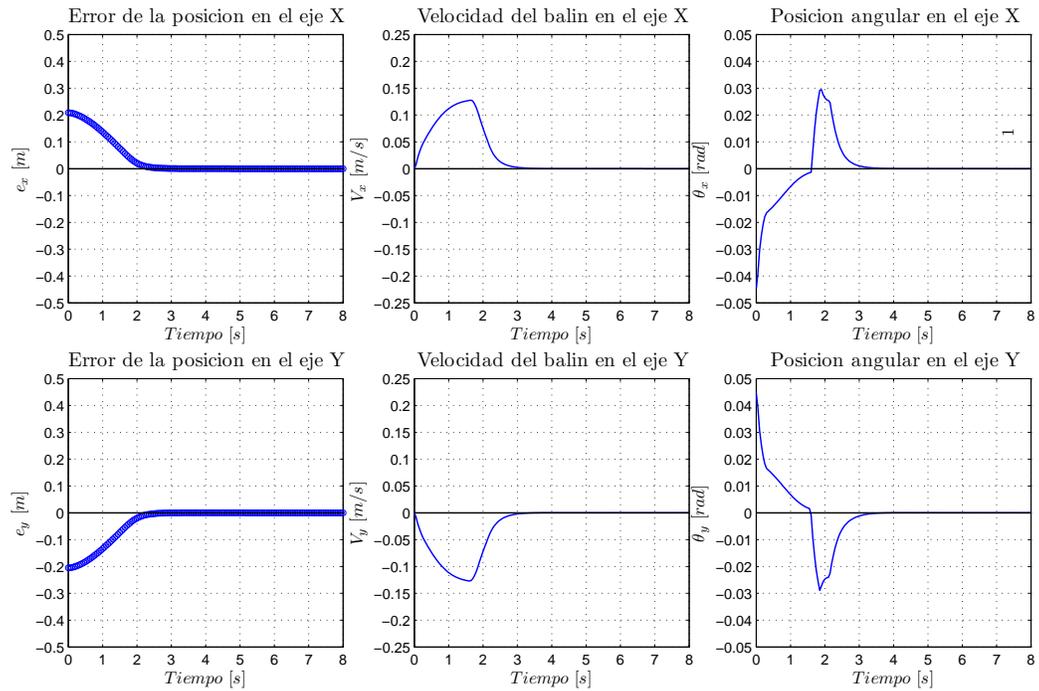


Figura F.2: Gráficas del Error, Velocidad y Señal de control, simulación C1.

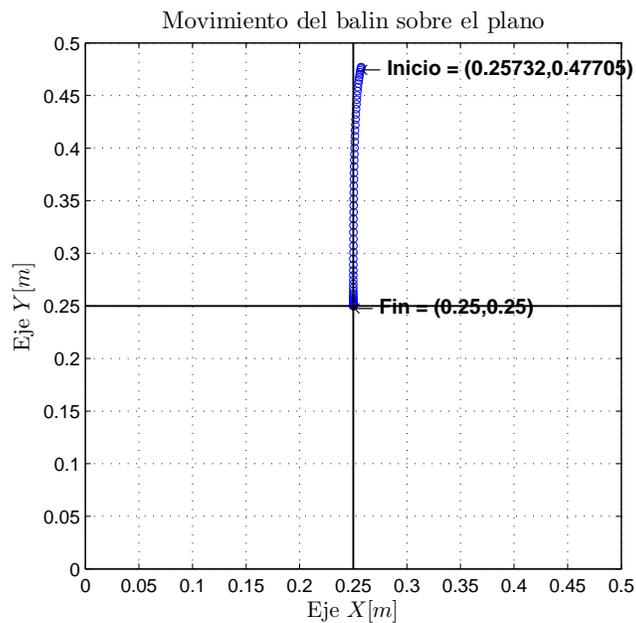


Figura F.3: Comparación entre el movimiento de la simulación y el modelo real, experimento C2.

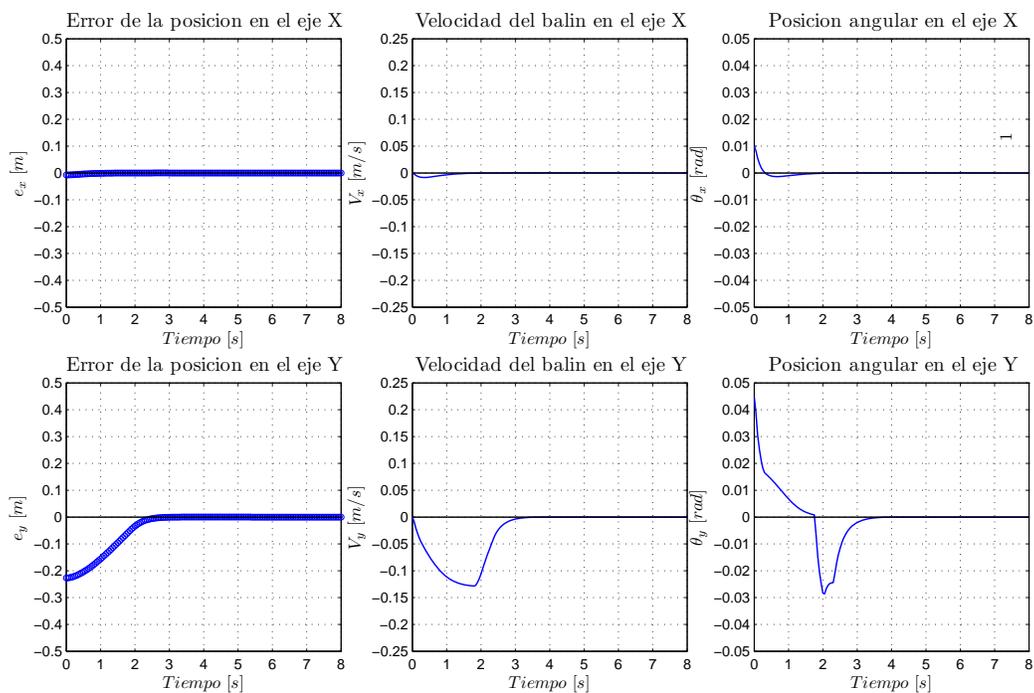


Figura F.4: Gráficas del Error, Velocidad y Señal de control, simulación C2.

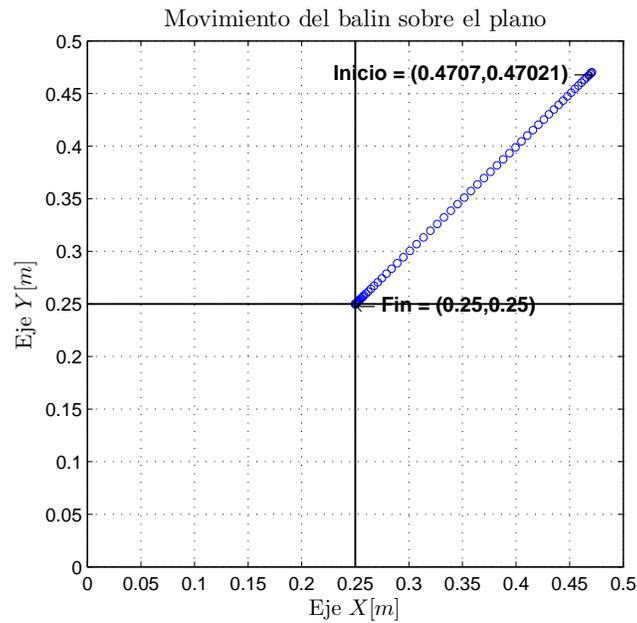


Figura F.5: Comparación entre el movimiento de la simulación y el modelo real, experimento C3.

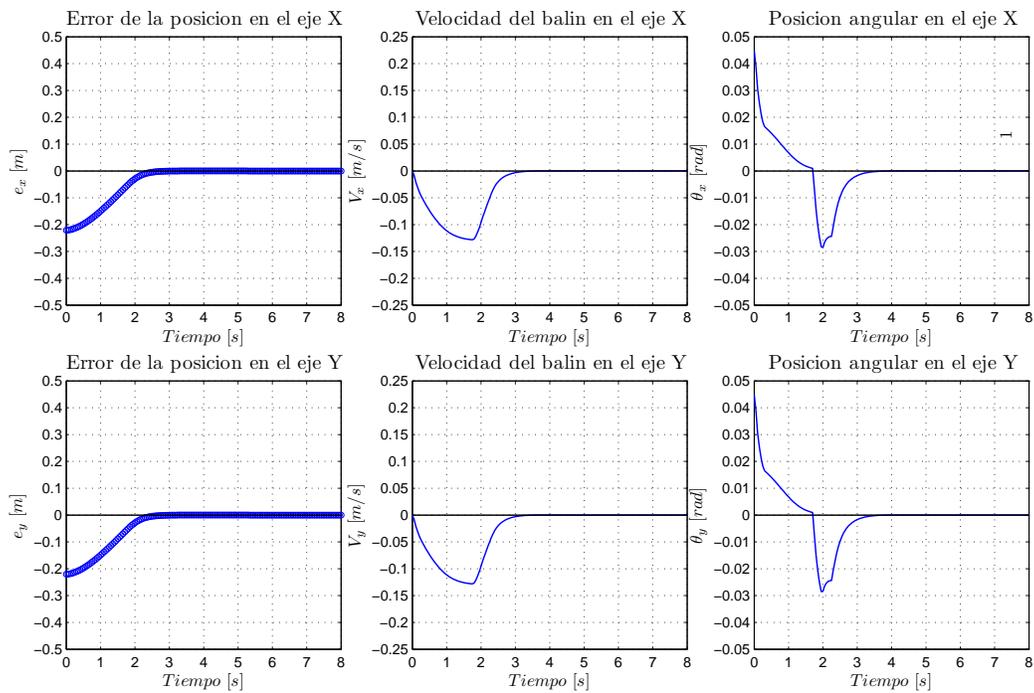


Figura F.6: Gráficas del Error, Velocidad y Señal de control, simulación C3.

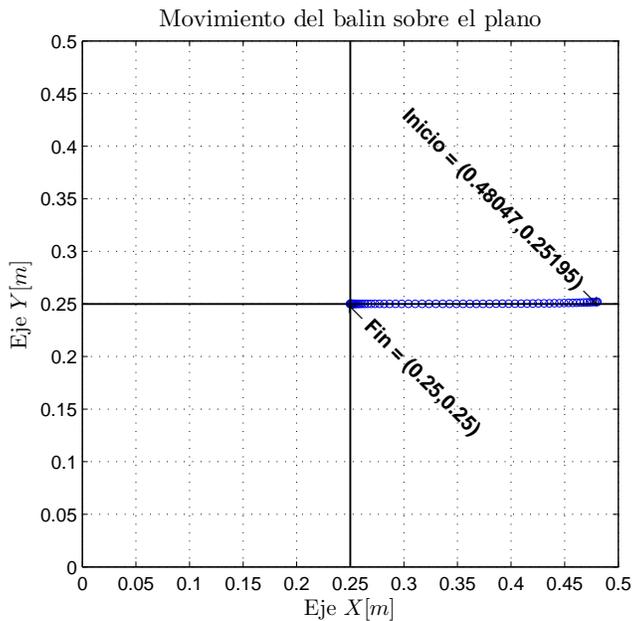


Figura F.7: Comparación entre el movimiento de la simulación y el modelo real, experimento C4.

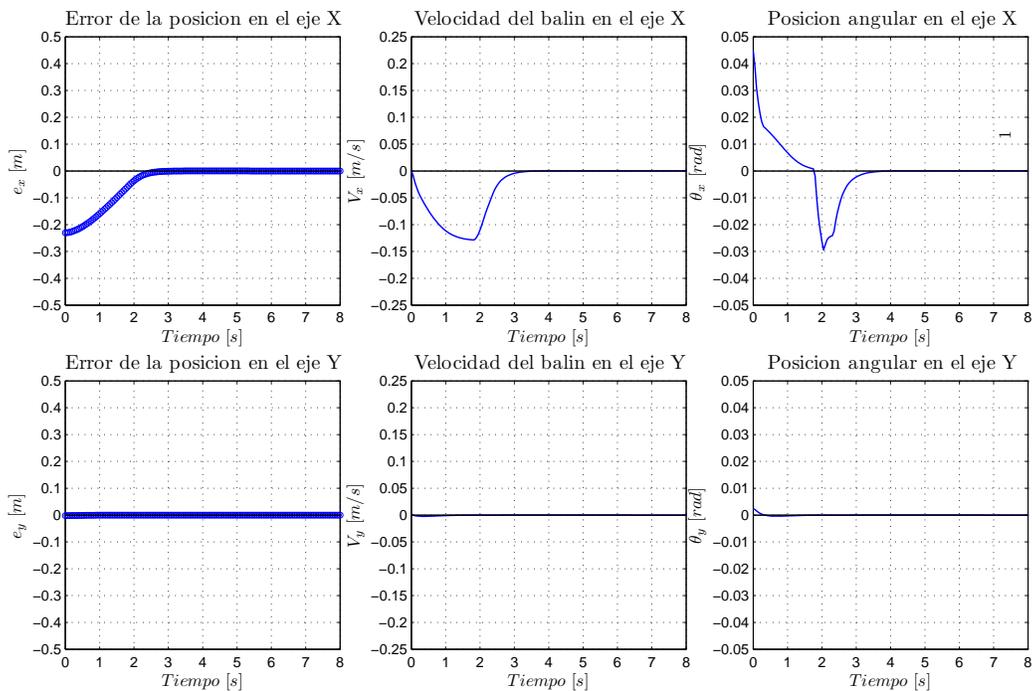


Figura F.8: Gráficas del Error, Velocidad y Señal de control, simulación C4.

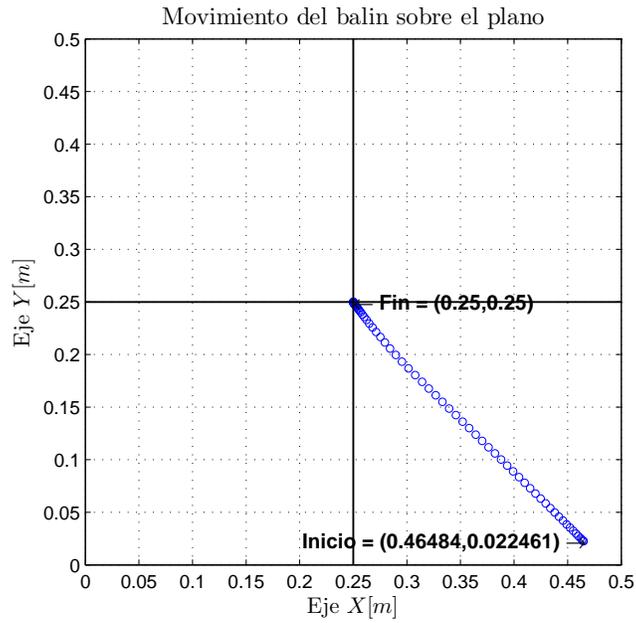


Figura F.9: Comparación entre el movimiento de la simulación y el modelo real, experimento C5.

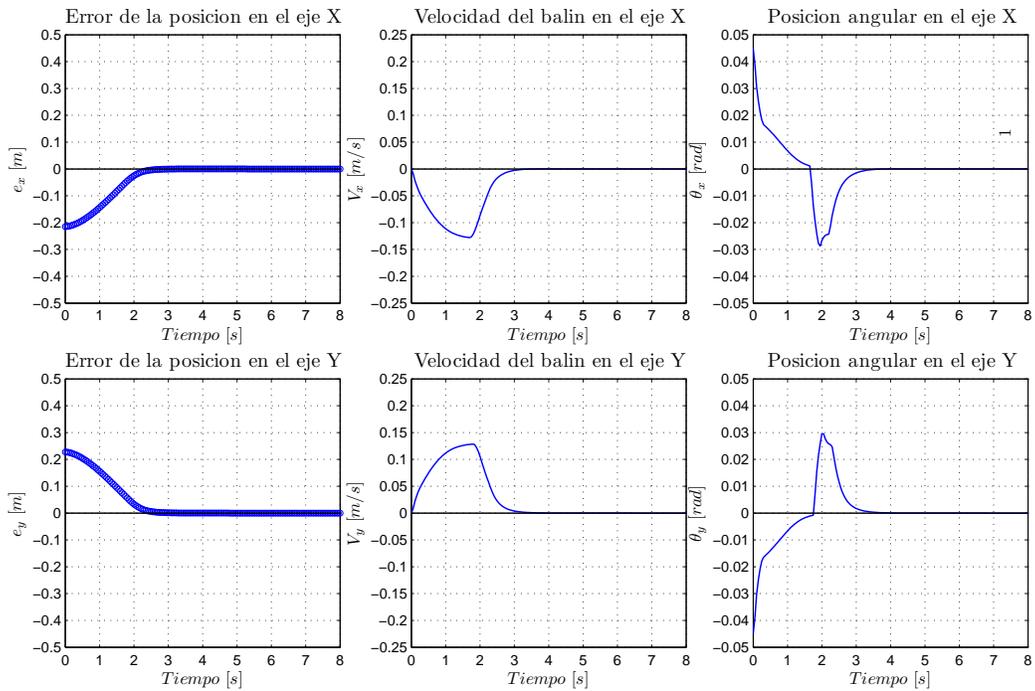


Figura F.10: Gráficas del Error, Velocidad y Señal de control, simulación C5.

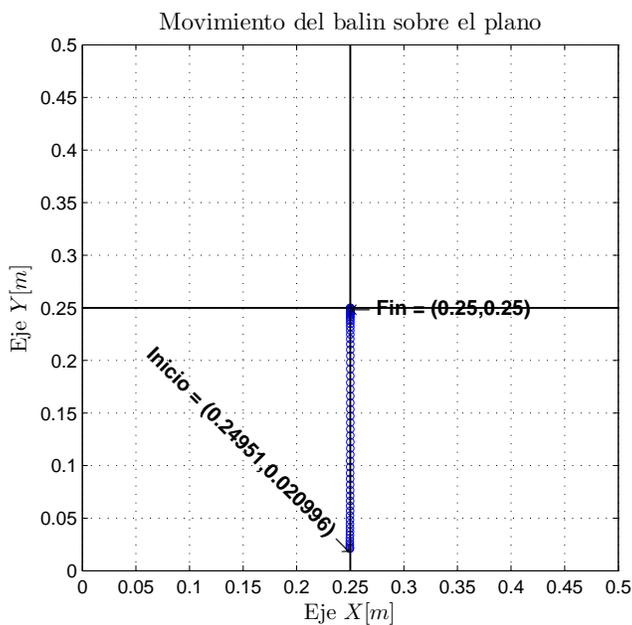


Figura F.11: Comparación entre el movimiento de la simulación y el modelo real, experimento C6.

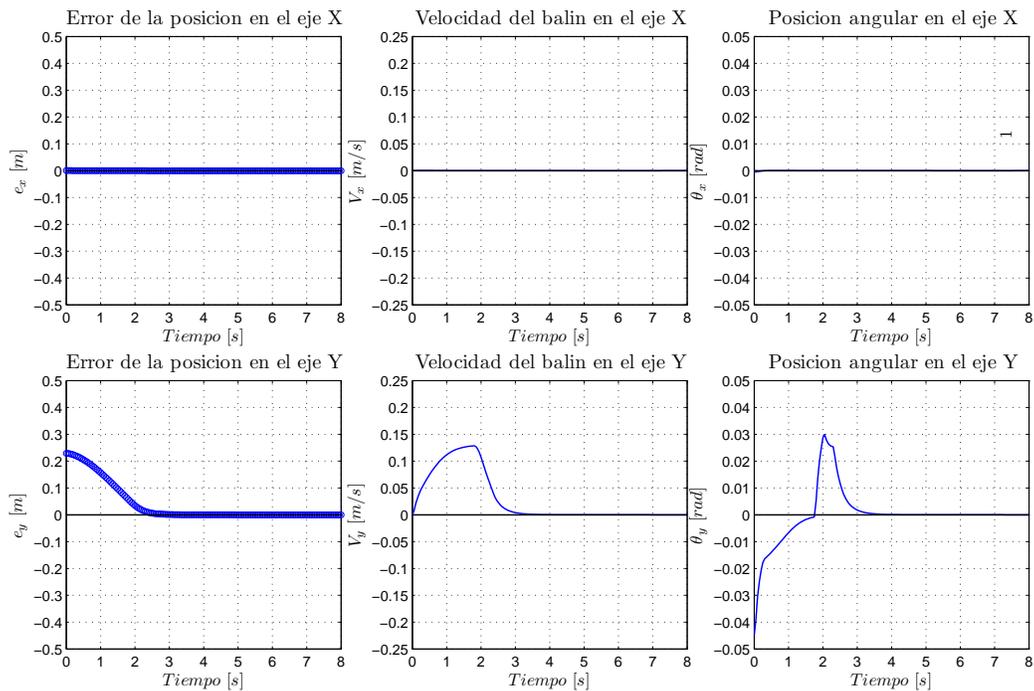


Figura F.12: Gráficas del Error, Velocidad y Señal de control, simulación C6.

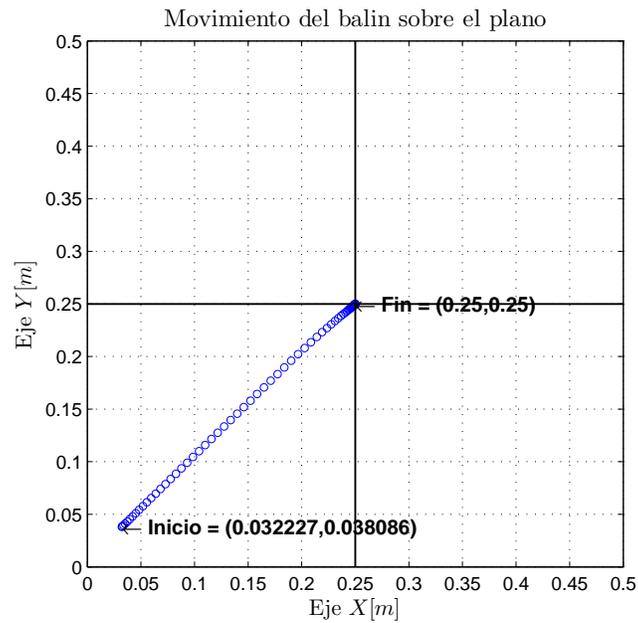


Figura F.13: Comparación entre el movimiento de la simulación y el modelo real, experimento C7.

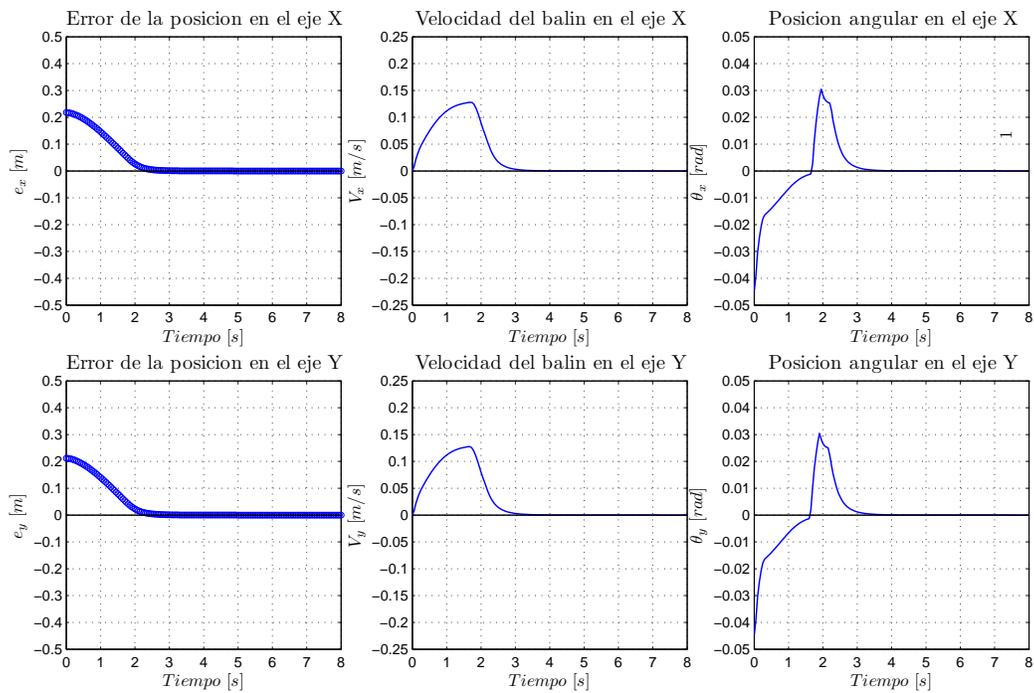


Figura F.14: Gráficas del Error, Velocidad y Señal de control, simulación C7.

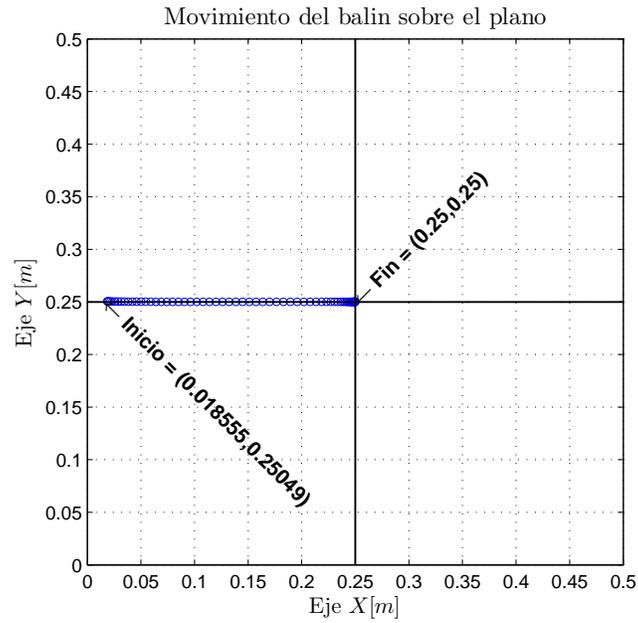


Figura F.15: Comparación entre el movimiento de la simulación y el modelo real, experimento C8.

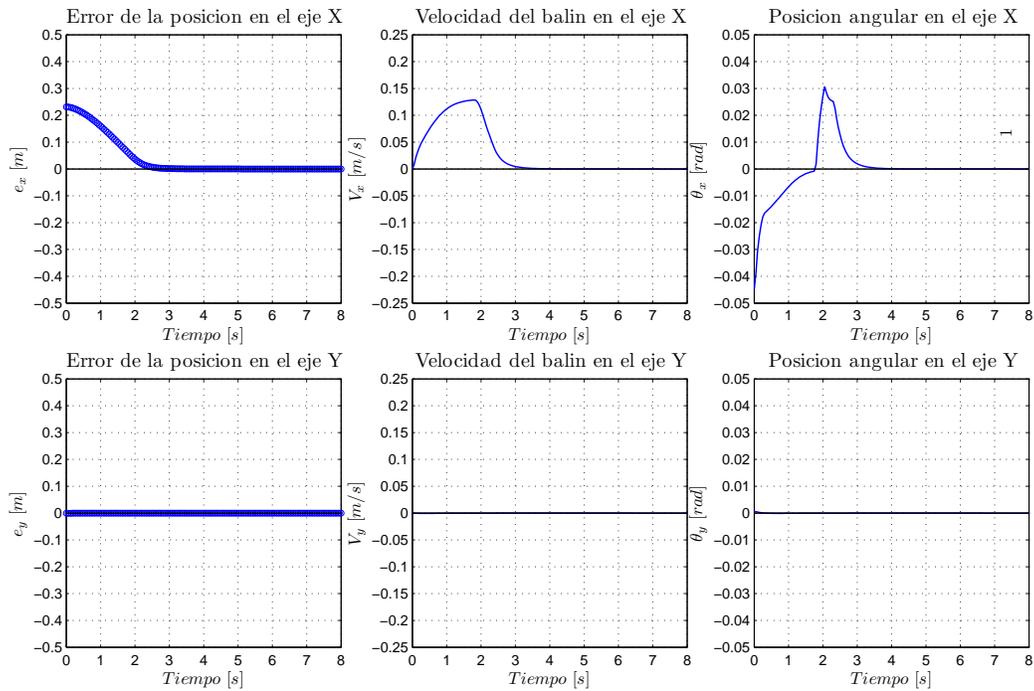


Figura F.16: Gráficas del Error, Velocidad y Señal de control, simulación C8.

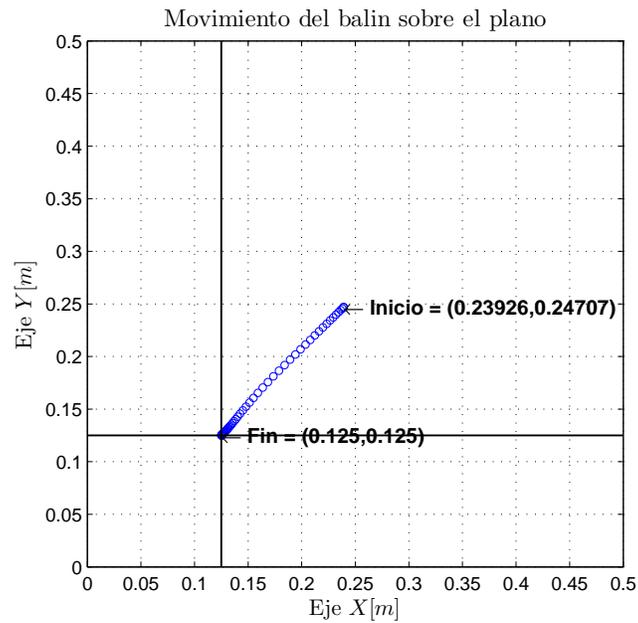


Figura F.17: Comparación entre el movimiento de la simulación y el modelo real, experimento C9.

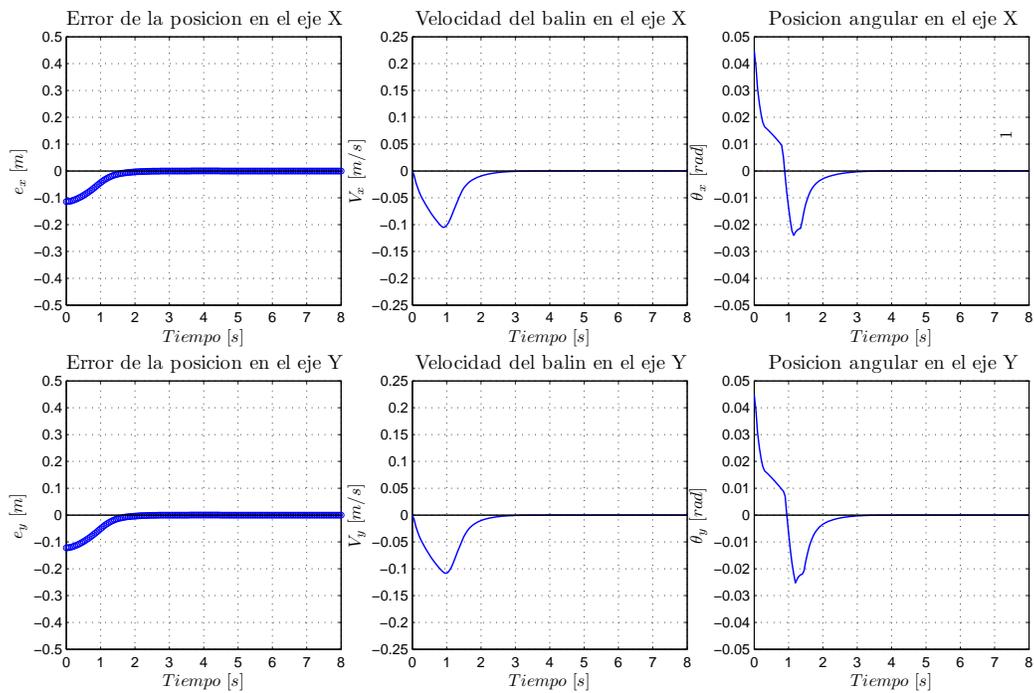


Figura F.18: Gráficas del Error, Velocidad y Señal de control, simulación C9.

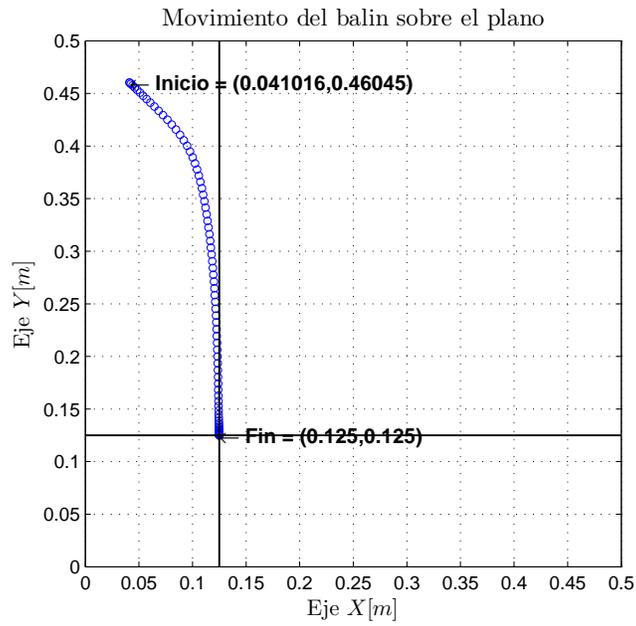


Figura F.19: Comparación entre el movimiento de la simulación y el modelo real, experimento C10.

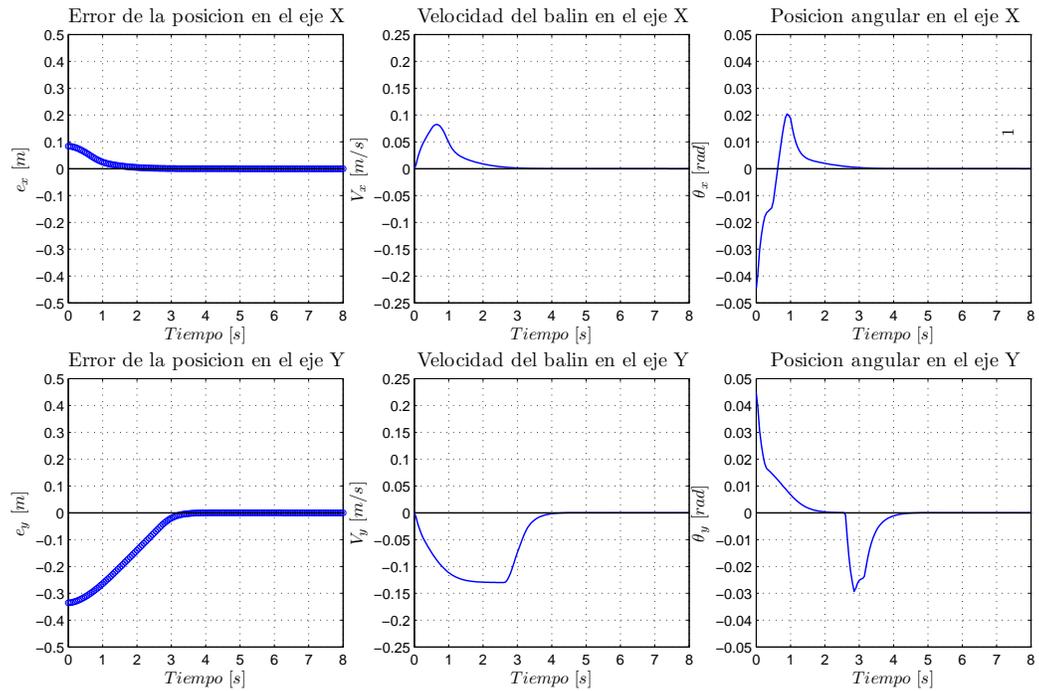


Figura F.20: Gráficas del Error, Velocidad y Señal de control, simulación C10.

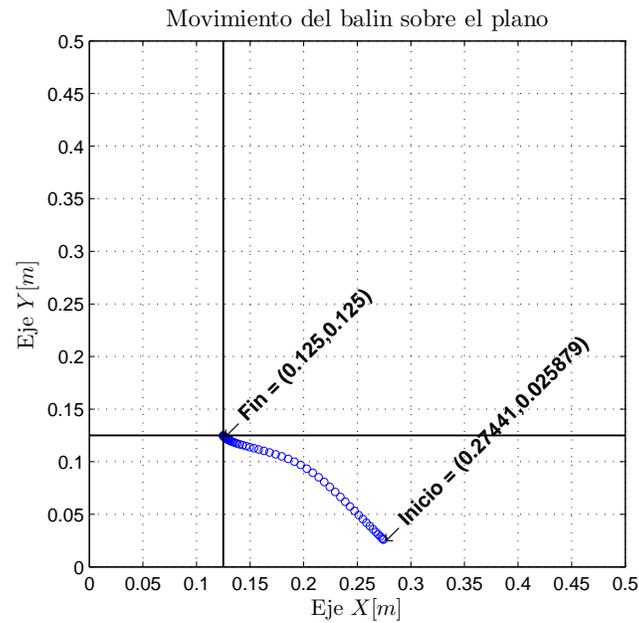


Figura F.21: Comparación entre el movimiento de la simulación y el modelo real, experimento C11.

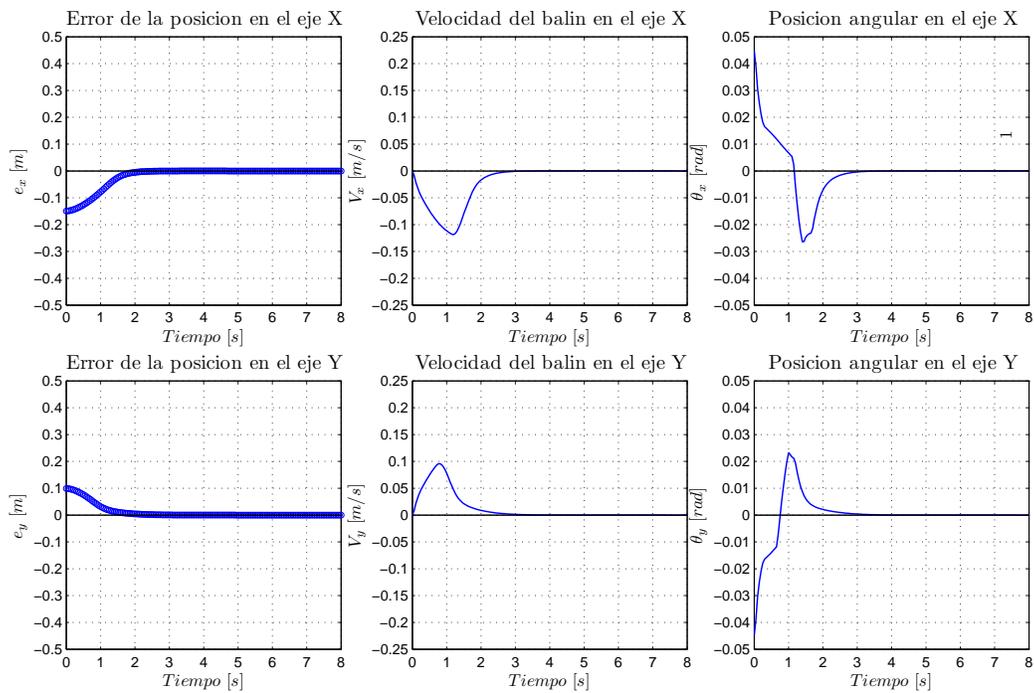


Figura F.22: Gráficas del Error, Velocidad y Señal de control, simulación C11.

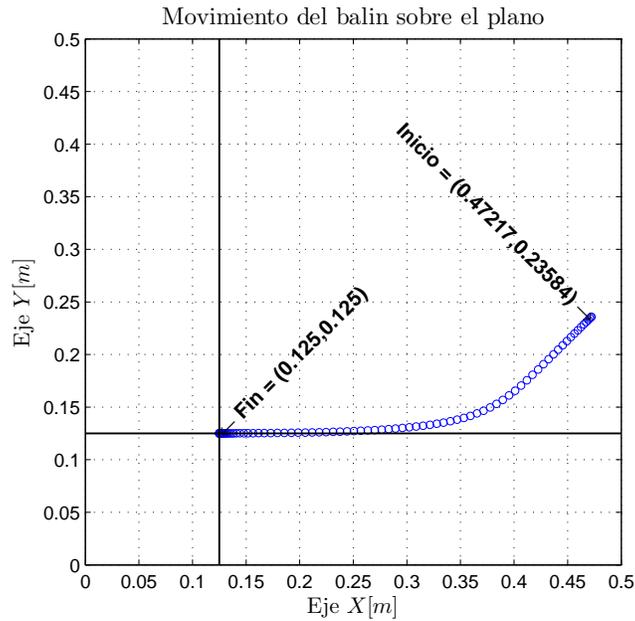


Figura F.23: Comparación entre el movimiento de la simulación y el modelo real, experimento C12.

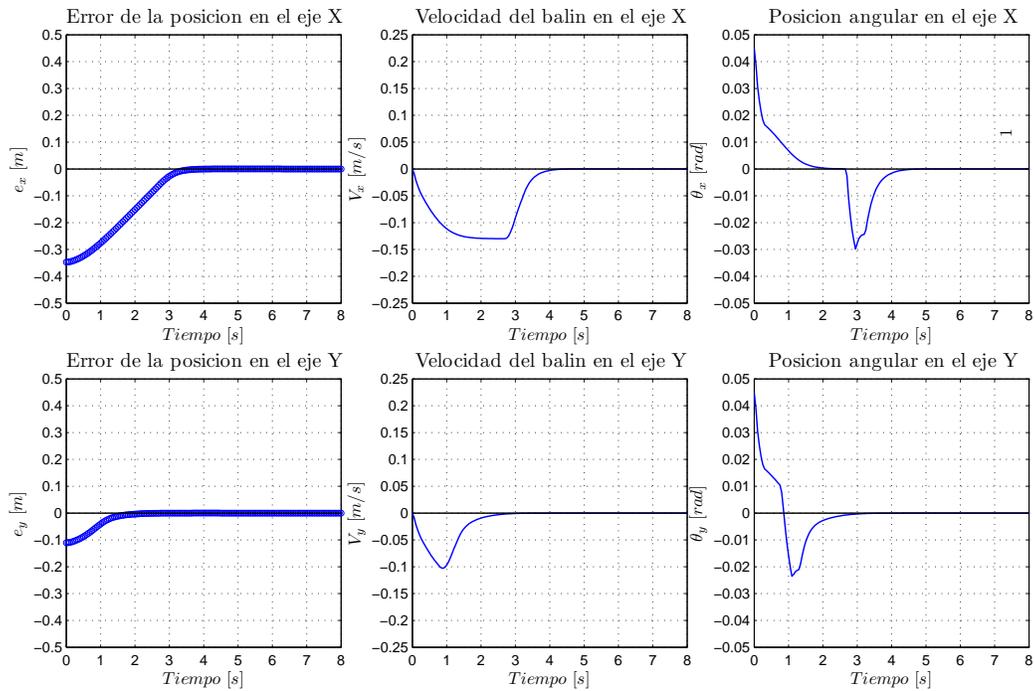


Figura F.24: Gráficas del Error, Velocidad y Señal de control, simulación C12.

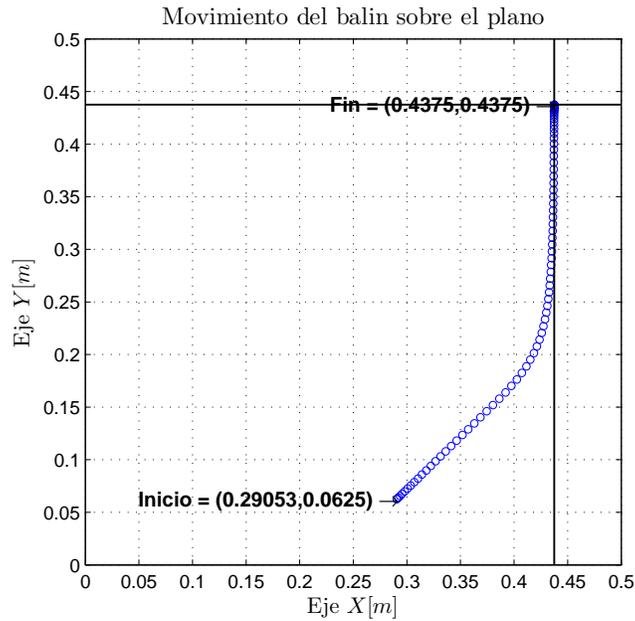


Figura F.25: Comparación entre el movimiento de la simulación y el modelo real, experimento C13.

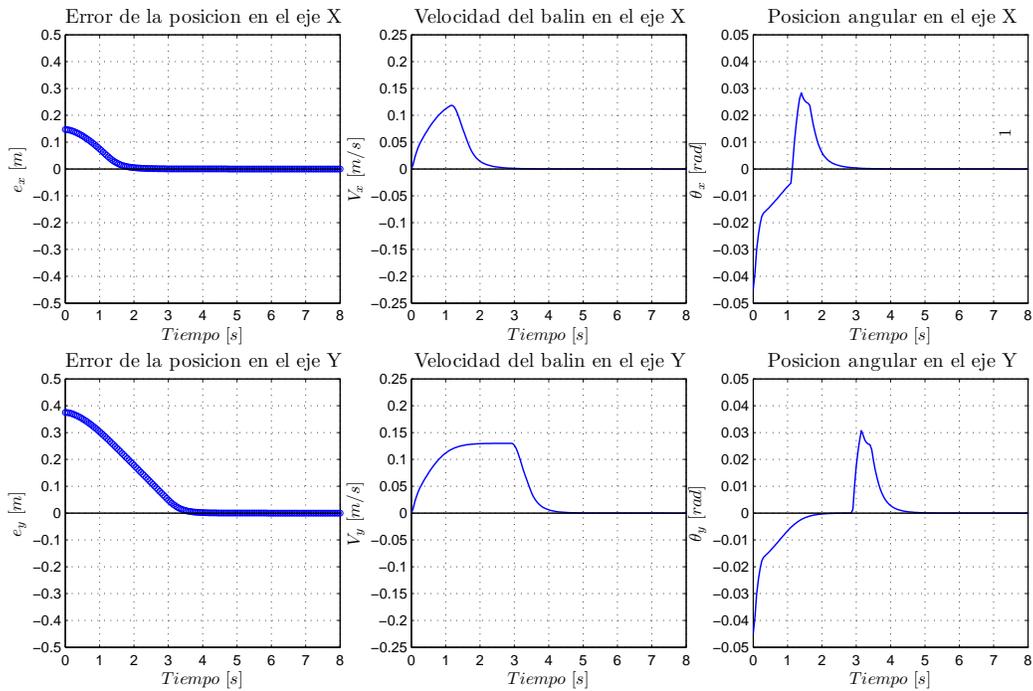


Figura F.26: Gráficas del Error, Velocidad y Señal de control, simulación C13.

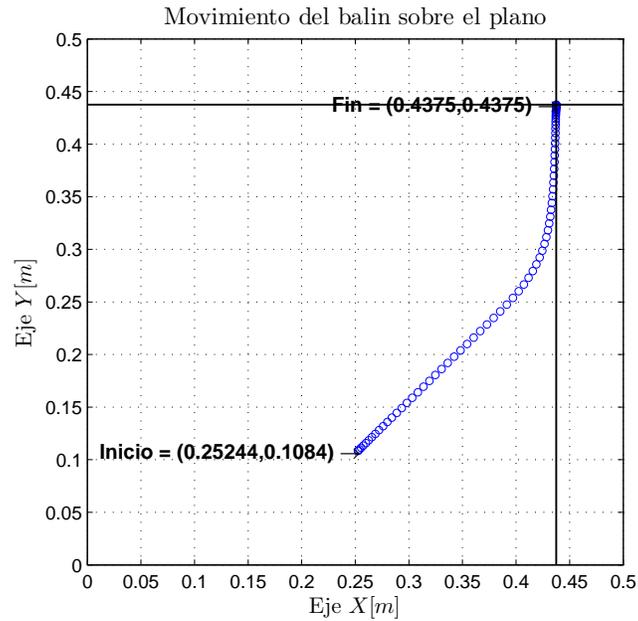


Figura F.27: Comparación entre el movimiento de la simulación y el modelo real, experimento C14.

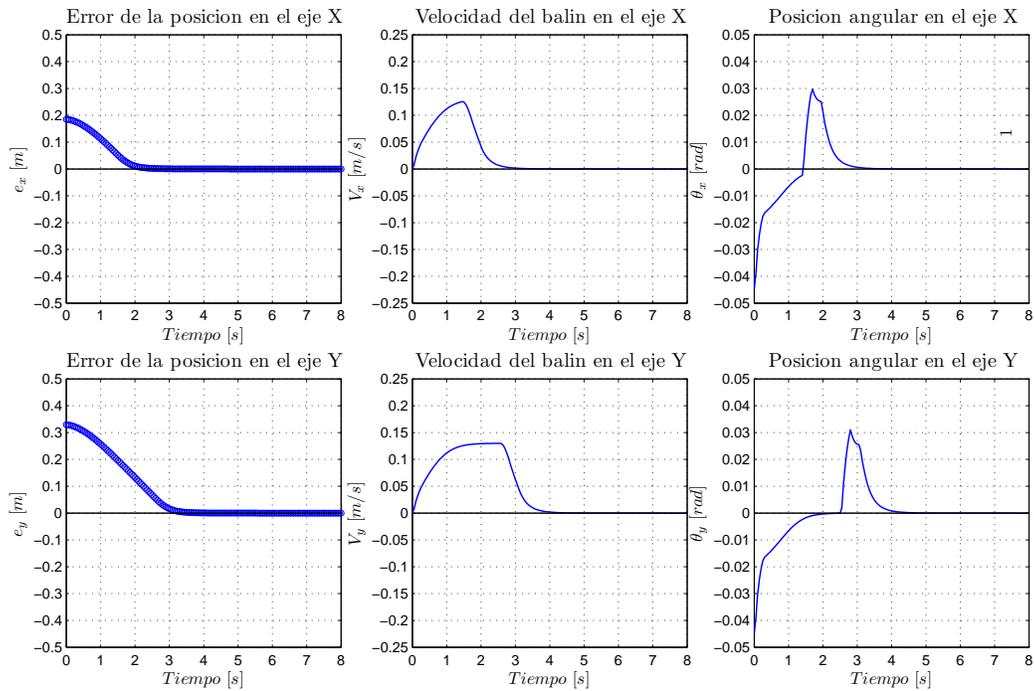


Figura F.28: Gráficas del Error, Velocidad y Señal de control, simulación C14.

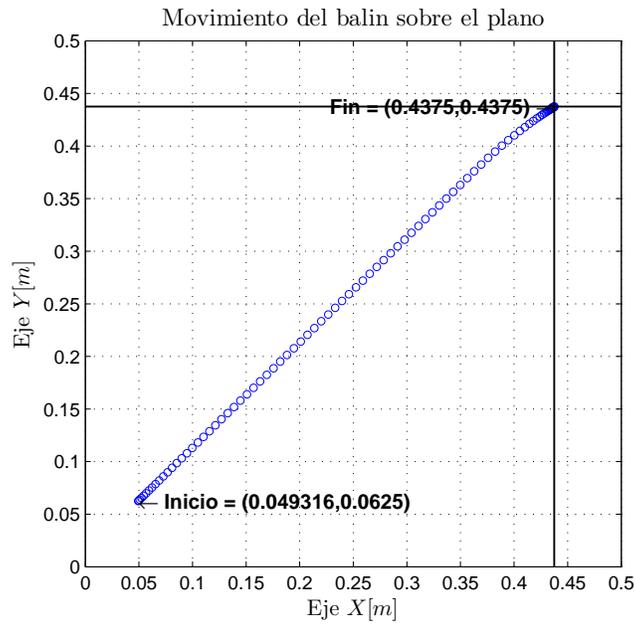


Figura F.29: Comparación entre el movimiento de la simulación y el modelo real, experimento C15.

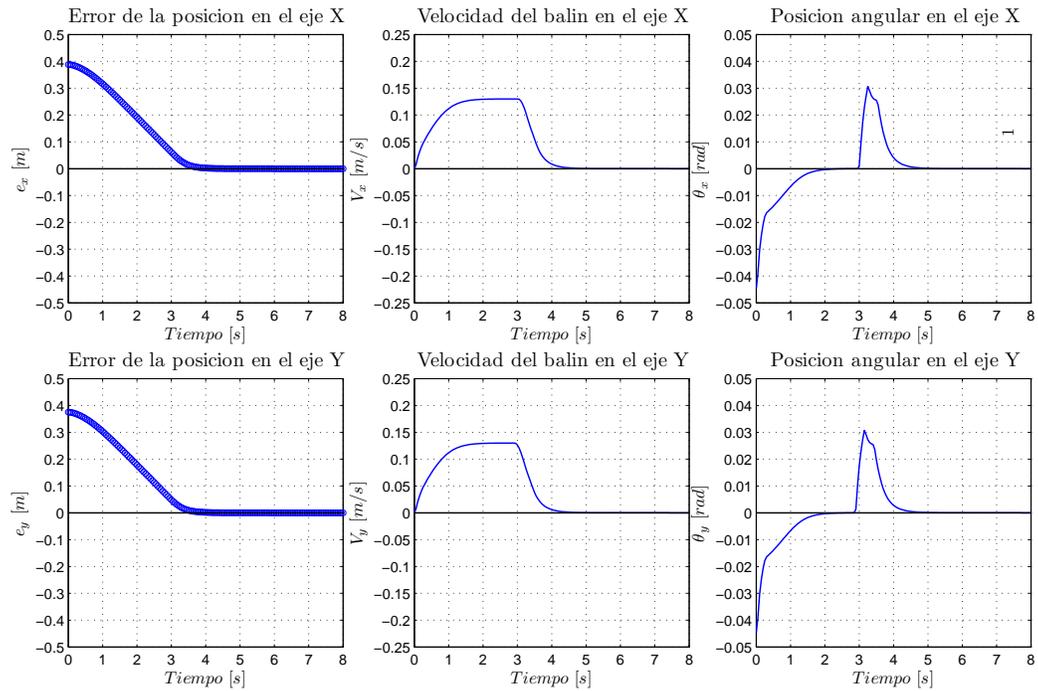


Figura F.30: Gráficas del Error, Velocidad y Señal de control, simulación C15.

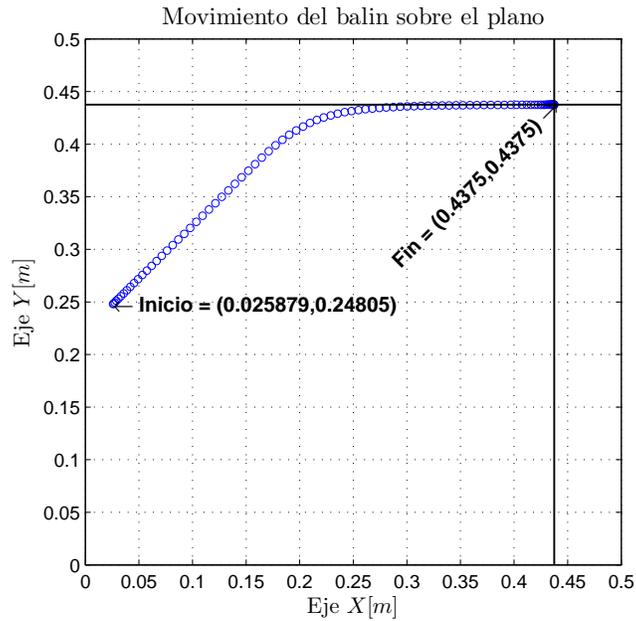


Figura F.31: Comparación entre el movimiento de la simulación y el modelo real, experimento C16.

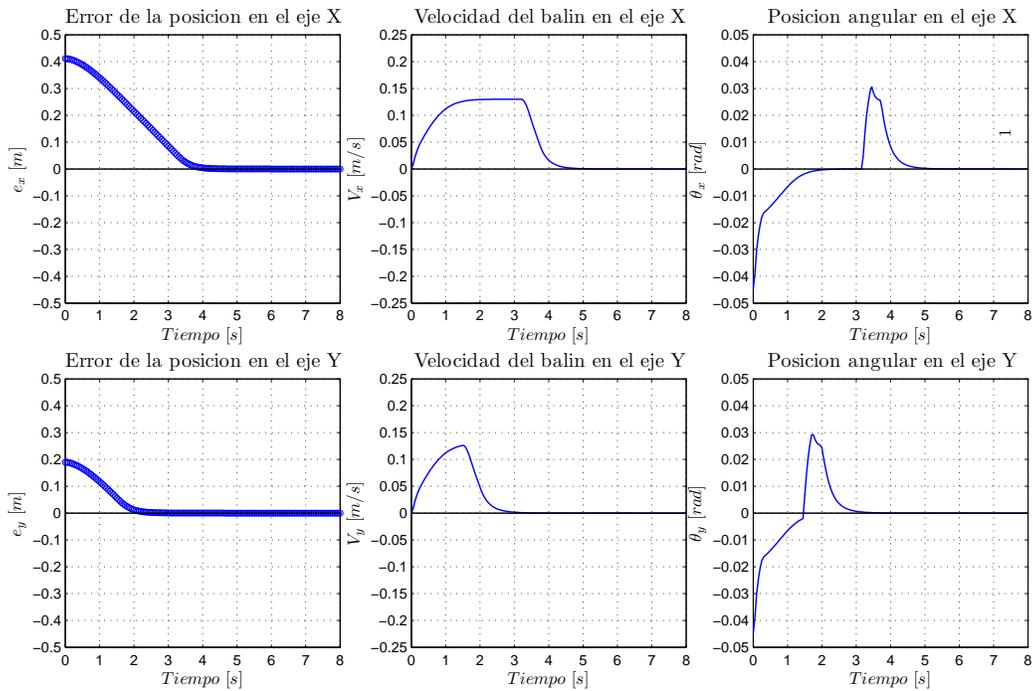


Figura F.32: Gráficas del Error, Velocidad y Señal de control, simulación C16.

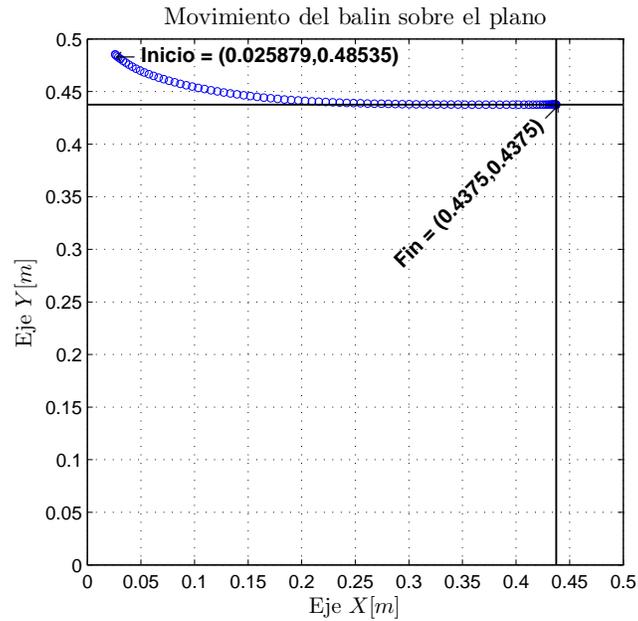


Figura F.33: Comparación entre el movimiento de la simulación y el modelo real, experimento C17.

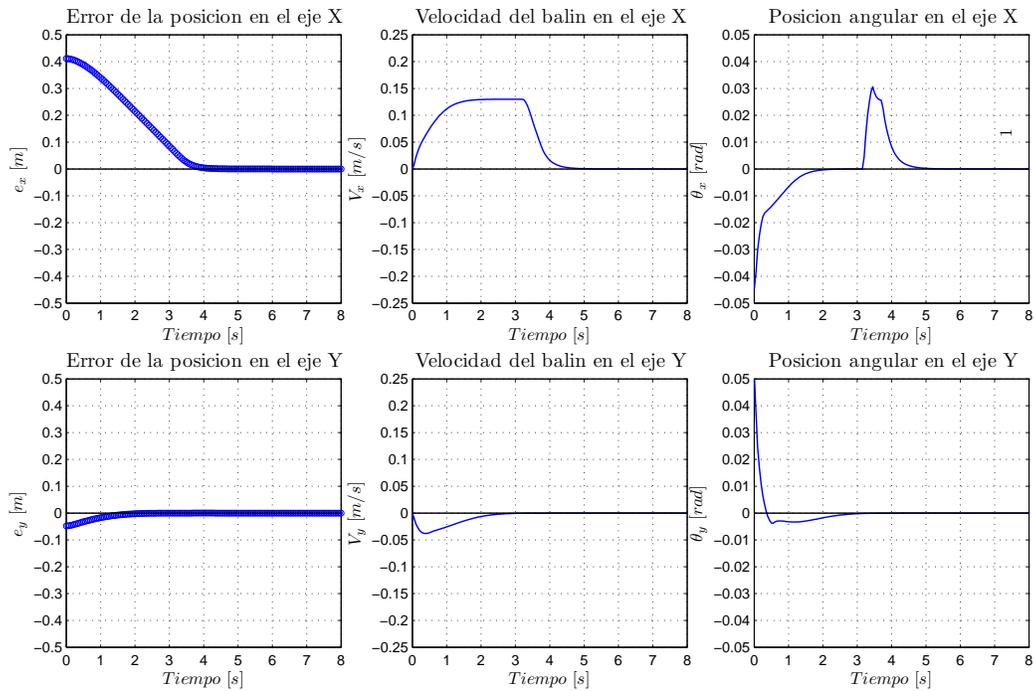


Figura F.34: Gráficas del Error, Velocidad y Señal de control, simulación C17.