



**I
N
A
O
E**

**Control de movimiento de un robot
manipulador SCARA de cuatro grados de
libertad mediante técnica de control
adaptativa basado en redes neuronales
artificiales.**

por:

Cesar Alejandro De La Paz Arteaga

Tesis presentada en cumplimiento parcial de los requisitos para el
grado de:

Maestro en ciencias en la Especialidad de Electrónica

Instituto Nacional de Astrofísica, Óptica y Electrónica

Marzo 2024

Tonantzintla, Puebla

Supervisada por:

Director: Dr. Israel Cruz Vega - Coord. Electrónica

**Co-Director: Dr. Iván Alejandro Gutiérrez Giles - Coord. Cienc.
Comp.**

©INAOE 2024

Todos los derechos reservados

El autor otorga a INAOE el permiso para reproducir y distribuir
copias de esta tesis en su totalidad o en partes mencionando la
fuente.



Agradecimientos

Quiero agradecer a mis asesores el Dr. Israel Cruz Vega y al Dr. Iván Alejandro Gutiérrez Giles por el respaldo y confianza brindados al permitirme desarrollar el presente trabajo.

Así mismo, agradezco a mis sinodales el Dr. José de Jesús Rangel Magdaleno, el Dr. Juan Manuel Ramírez Cortés y el Dr. Luis Hernández Martínez por las valiosas aportaciones que ayudaron a mejorar mi trabajo.

De la misma manera, agradezco al Instituto Nacional de Astrofísica Óptica y Electrónica por permitirme formar parte de su comunidad estudiantil, así como desarrollar mi Tesis.

Así mismo, al CONAHCYT, por su gran soporte económico que me brindó durante estos años de estudio.

Al personal de coordinación docente por su paciencia y comprensión durante mi formación.

Dedicatoria

A las personas que siempre me han apoyado para lograr mis metas: Mis padres, Guillermo Moreno Torres y Raquel Arteaga Rivera, por sus grandes consejos, cariño y atención.

Mis abuelos, Zenón Arteaga Valdés y María Clementina Rivera González, por su amor incondicional y por todas las enseñanzas brindadas.

A mi hermana, Brianda Karime Moreno Arteaga, la cual me brinda la motivación necesaria para continuar esforzándome.

A la persona que me ha brindado toda la confianza y el apoyo moral para no desistir de mis objetivos: Zaira Itzel Vargas Domínguez.

A mis amigos y compañeros, gracias por alentarme y ayudarme de diversas maneras en mi estancia en este instituto.

Resumen

El enfoque principal de esta tesis se dirige hacia el control del movimiento de un brazo robótico dentro de los límites físicos establecidos por su diseño, sin interactuar con su entorno directo y siguiendo una trayectoria específica. Las técnicas de control desarrolladas para sistemas no lineales se consideran de gran importancia en este contexto con un interés por optimizar el rendimiento y el tiempo de respuesta mediante la incorporación de redes neuronales artificiales. Este trabajo de tesis se lleva a cabo en un brazo robótico rígido clasificado como SCARA que posee 4 grados de libertad. Primero se realiza la construcción del prototipo, se analizan las características del robot manipulador seleccionado, así como el modelado del comportamiento dinámico mediante la metodología Euler-Lagrange. Después se describe la principal técnica de control basada en control adaptativo y redes neuronales artificiales. Este trabajo también presenta la comparación con otras técnicas de control convencional como control PID y control adaptable convencional. Finalmente, se realizan pruebas experimentales y simulaciones con el robot propuesto, con el objetivo de determinar la efectividad y el rendimiento de la técnica de control propuesta no solo de forma experimental sino con resultados gráficos y con medidas de error comparado con los otros enfoques de control presentados.

Abstract

The main focus of this thesis is directed toward controlling the movement of a robotic arm within the physical limits established by its design without interacting with its direct environment and following a specific trajectory. Control techniques developed for nonlinear systems are essential in this context, with an interest in optimizing performance and response time by incorporating artificial neural networks. This thesis is carried out on a rigid robotic arm classified as SCARA with 4 degrees of freedom. After constructing the prototype, the characteristics of the selected manipulator robot are analyzed, and the modeling of the dynamic behavior using the Euler-Lagrange methodology is performed. Afterwards, the primary control technique to be implemented, based on adaptive control and artificial neural networks, is described. This work also compares conventional control techniques, such as PID and traditional adaptive control. Finally, experimental tests and simulations are carried out with the proposed robot, to determine the effectiveness and performance of the proposed control technique graphically, experimentally and with error measurements compared to the other control approaches presented.

Índice general

Agradecimientos	III
Dedicatoria	v
Resumen	VII
Abstract	IX
1. Introducción	3
1.1. Estado del Arte	4
1.2. Planteamiento del problema	9
1.3. Justificación	9
1.4. Hipótesis	10
1.5. Objetivos	10
1.5.1. Objetivo General	10
1.5.2. Objetivos Específicos	10
1.6. Organización del documento	11

2. Marco Teórico	13
2.1. Introducción	13
2.2. Características del robot manipulador SCARA	14
2.3. Modelo cinemático directo	17
2.4. Modelo cinemático inverso	22
2.5. Modelo dinámico del manipulador	24
2.5.1. Propiedades del modelo dinámico	27
2.5.2. Descripción y aplicación de la técnica de Euler–Lagrange	28
2.6. Control de retroceso no lineal (<i>backstepping</i>)	35
2.7. Estructura de la red neuronal	38
2.8. Parámetros dinámicos	41
2.9. Conclusiones	42
3. Algoritmos de Control	43
3.1. Introducción	43
3.2. Control PID	44
3.3. Control Adaptable convencional	44
3.4. Control adaptativo basado en redes neuronales artificiales	49
4. Resultados	53
4.1. Introducción	53
4.2. Simulaciones	54

4.2.1.	Simulación del control PID	54
4.2.2.	Simulación del control adaptable	55
4.2.3.	Simulaciones del control adaptativo basado en redes neurona- les artificiales.	57
4.2.4.	Posiciones sin perturbación	58
4.2.5.	Errores sin perturbación	62
4.2.6.	Posiciones con perturbación	65
4.2.7.	Errores con perturbación	68
4.3.	Experimentos	71
4.3.1.	Posiciones sin compensador de fricción	73
4.3.2.	Errores sin compensador de fricción	77
4.3.3.	Posiciones con compensador de fricción	80
4.3.4.	Errores con compensador de fricción	83
4.4.	Análisis de regresión	86
5.	Conclusiones y observaciones	87
5.1.	Conclusiones generales	87
5.2.	Trabajo a futuro	89
A.	Convención de Denavit Hartenberg	91
B.	Robot físico de cuatro grados de libertad	93

Índice de figuras

2.1. Robot manipulador SCARA y su espacio de trabajo [1].	15
2.2. Modelo de cinemática directa e inversa [2].	17
2.3. Esquema de cinemática directa [2].	18
2.4. Modelo funcional SCARA.	19
2.5. Sistema de coordenadas generalizadas x, y, z	20
2.6. Esquema de cinemática inversa [2].	23
2.7. Esquema del modelo dinámico directo.	26
2.8. Esquema del modelo dinámico inverso.	27
2.9. Diagrama de red neuronal artificial que estima las no linealidades del sistema.	39
2.10. Diagrama de red neuronal artificial que estima la zona muerta.	40
4.1. Esquema de PID en Simulink.	54
4.2. Esquema de Adaptable en Simulink.	55
4.3. Esquema de control Adaptable basado en redes neuronales artificiales, en Simulink.	57

4.4. Control PID, regulación: $q_d(---)$, $q(- - -)$	59
4.5. Control PID, seguimiento: $q_d(---)$, $q(- - -)$	59
4.6. Control adaptable convencional, regulación: $q_d(---)$, $q(- - -)$	60
4.7. Control adaptable convencional, seguimiento: $q_d(---)$, $q(- - -)$	60
4.8. Control adaptativo basado en redes neuronales artificiales, regulación: $q_d(---)$, $q(- - -)$	61
4.9. Control adaptativo basado en redes neuronales artificiales, seguimien- to: $q_d(---)$, $q(- - -)$	61
4.10. Control PID, regulación: $e_d(---)$, $e(- - -)$	62
4.11. Control PID, seguimiento: $e_d(---)$, $e(- - -)$	62
4.12. Control adaptable convencional, regulación: $e_d(---)$, $e(- - -)$	63
4.13. Control adaptable convencional, seguimiento: $e_d(---)$, $e(- - -)$	63
4.14. Control adaptativo basado en redes neuronales artificiales, regulación: $e_d(---)$, $e(- - -)$	64
4.15. Control adaptativo basado en redes neuronales artificiales, seguimien- to: $e_d(---)$, $e(- - -)$	64
4.16. Control PID, regulación: $q_d(---)$, $q(- - -)$	65
4.17. Control PID, seguimiento: $q_d(---)$, $q(- - -)$	65
4.18. Control adaptable convencional, regulación: $q_d(---)$, $q(- - -)$	66
4.19. Control adaptable convencional, seguimiento: $q_d(---)$, $q(- - -)$	66
4.20. Control adaptativo basado en redes neuronales artificiales, regulación: $q_d(---)$, $q(- - -)$	67

4.21. Control adaptativo basado en redes neuronales artificiales, seguimiento: $q_d(---)$, $q(- - -)$	67
4.22. Control PID, regulación: $e_d(---)$, $e(- - -)$	68
4.23. Control PID, seguimiento: $e_d(---)$, $e(- - -)$	68
4.24. Control adaptable convencional, regulación: $e_d(---)$, $e(- - -)$	69
4.25. Control adaptable convencional, seguimiento: $e_d(---)$, $e(- - -)$	69
4.26. Control adaptativo basado en redes neuronales artificiales, regulación: $e_d(---)$, $e(- - -)$	70
4.27. Control adaptativo basado en redes neuronales artificiales, seguimiento: $e_d(---)$, $e(- - -)$	70
4.28. Control PID, regulación: $q_d(---)$, $q(- - -)$	74
4.29. Control PID, seguimiento: $q_d(---)$, $q(- - -)$	74
4.30. Control adaptable convencional, regulación: $q_d(---)$, $q(- - -)$	75
4.31. Control adaptable convencional, seguimiento: $q_d(---)$, $q(- - -)$	75
4.32. Control adaptativo basado en redes neuronales artificiales, regulación: $q_d(---)$, $q(- - -)$	76
4.33. Control adaptativo basado en redes neuronales artificiales, seguimiento: $q_d(---)$, $q(- - -)$	76
4.34. Control PID, regulación: $e_d(---)$, $e(- - -)$	77
4.35. Control PID, seguimiento: $e_d(---)$, $e(- - -)$	77
4.36. Control adaptable convencional, regulación: $e_d(---)$, $e(- - -)$	78
4.37. Control adaptable convencional, seguimiento: $e_d(---)$, $e(- - -)$	78

4.38. Control adaptativo basado en redes neuronales artificiales, regulación: $e_d(---), e(- - -)$	79
4.39. Control adaptativo basado en redes neuronales artificiales, seguimien- to: $e_d(---), e(- - -)$	79
4.40. Control PID, regulación: $q_d(---), q(- - -)$	80
4.41. Control PID, seguimiento: $q_d(---), q(- - -)$	80
4.42. Control adaptable convencional, regulación: $q_d(---), q(- - -)$	81
4.43. Control adaptable convencional, seguimiento: $q_d(---), q(- - -)$	81
4.44. Control adaptativo basado en redes neuronales artificiales, regulación: $q_d(---), q(- - -)$	82
4.45. Control adaptativo basado en redes neuronales artificiales, seguimien- to: $q_d(---), q(- - -)$	82
4.46. Control PID, regulación: $e_d(---), e(- - -)$	83
4.47. Control PID, seguimiento: $e_d(---), e(- - -)$	83
4.48. Control adaptable convencional, regulación: $e_d(---), e(- - -)$	84
4.49. Control adaptable convencional, seguimiento: $e_d(---), e(- - -)$	84
4.50. Control adaptativo basado en redes neuronales artificiales, regulación: $e_d(---), e(- - -)$	85
4.51. Control adaptativo basado en redes neuronales artificiales, seguimien- to: $e_d(---), e(- - -)$	85
B.1. Robot manipulador SCARA visualizado desde la parte superior.	94
B.2. Diseño de piezas en 3D para el robot manipulador SCARA.	94

B.3. Piezas fabricadas para el robot manipulador SCARA, a base de filamento PLA.	95
B.4. Diagrama de conexión del protocolo de comunicación SPI.	97
B.5. Esquema electrónico general para el robot manipulador SCARA de cuatro grados de libertad.	98
B.6. Conexiones eléctricas del robot manipulador.	98
B.7. Robot manipulador SCARA de cuatro grados de libertad para pruebas experimentales.	99
C.1. Aplicación para la dirección del robot manipulador SCARA, desarrollada en ©Visual Studio.	101

Índice de tablas

2.1. Parámetros Denavit–Hartenberg.	21
2.2. Parámetros asignados a cada elemento de la matriz de inercia.	32
2.3. Parámetros asignados a cada elemento de la matriz de Coriolis y fuerza centrífuga.	33
2.4. Parámetros asignados a cada elemento de la matriz de constantes de fricción.	34
2.5. Parámetros asignados a cada elemento del vector de fuerzas gravita- cionales.	34
2.6. Notación de los parámetros del robot manipulador.	41
3.1. Elementos del regresor Y	45
3.2. Elementos del vector de parámetros θ	46
3.3. Elementos del regresor modificado Y_r	48
4.1. Ganancias de sintonización para la simulación del control PID.	55
4.2. Ganancias de sintonización para la simulación del control adaptable.	56

4.3. Ganancias de sintonización para la simulación del control adaptable mediante redes neuronales artificiales.	58
4.4. Análisis del RMSE de los datos de simulación para cada control implementado.	71
4.5. Ganancias de sintonización para la experimentación con el control PID.	72
4.6. Ganancias y parámetros de adaptación para la experimentación con el control adaptable convencional.	73
4.7. Ganancias y parámetros de adaptación para la experimentación con el control adaptativo basado en redes neuronales artificiales.	73
4.8. Análisis de RMSE de los datos de experimentación para cada control implementado.	86
B.1. Elementos mecánicos, eléctricos y electrónicos del robot manipulador SCARA para cuatro grados de libertad.	96

Lista de abreviaturas

ML Machine Learning

ANN Artificial Neural Networks

FFN Feed Forward Network.

RNN Recurrent Neural Network.

ITAE Integral time absolute error.

IMC Internal mode control

RMSE Root Mean Square Error.

DC Direct Current.

SCARA Selective Compliance Assembly Robot Arm.

PID Proportional - Integral - Derivative Control.

PCB Printed Circuit Board.

PWM Pulse Width Modulation.

PLA Polylactic Acid.

USB Universal Serial Bus.

RBF Radial Basis Functions.

DOF Degrees of freedom.

SPI Serial Peripheral Interface.

Capítulo 1

Introducción

Desde el siglo 20, la robótica se ha hecho muy importante como una herramienta de automatización flexible para cumplir con tareas repetitivas y peligrosas que reducen riesgos y la fatiga de los trabajadores. Al mismo tiempo, el uso de automatización mediante la robótica incrementa la producción eficiente y consistente [3]. La primera implementación de robots inició en 1937, con un robot Cartesiano [4]. A partir de entonces, el robot Cartesiano ha sido ampliamente usado en tareas de automatización industrial tales como en agricultura, almacenamiento automatizado y sistemas de recuperación, entre otras [5, 6]. Sin embargo, el rango de operación y de precisión del robot Cartesiano es limitada comparada con otros tipos de robots [7]. Como resultado de investigación para mejora continua, se propuso en 1962 el primer robot cilíndrico [8]. Los robots cilíndricos son normalmente empleados para la industria de ensamblaje tales como soldadura y manejo de equipos de fundición a presión [9]. El robot cilíndrico tiene mejor precisión comparada al robot Cartesiano, pero el rango de movimiento es limitado a la forma cilíndrica. Por lo tanto, el robot esférico y el robot articulado son empleados para mejorar el rango de movimiento [10]. Debido a las mejoras hechas, los robots esférico y articulado se usan ampliamente en aplicaciones que requieren alto alcance espacial, tales como aplica-

ciones militares, médicas, de manejo de materiales y manufactura [11, 12, 13]. Y por último, se encuentra el brazo robótico articulado de compliancia selectiva (SCARA, del inglés Selective Compliance Articulated Robot Arm), el cual tiene una alta precisión, exactitud y al mismo tiempo eficiencia energética debido a que solo una de las articulaciones opera contra la gravedad. Por lo tanto el robot SCARA es una buena opción para la industria que requiere alta precisión y menos consumo energético. Para alcanzar una alta exactitud, el robot SCARA requiere de control avanzado para desempeñarse y controlar comportamientos dinámicos no lineales [14, 15].

En este trabajo de tesis se abordan técnicas de control no lineales para lograr un mejor desempeño en la tarea propuesta. En particular, se empleará control adaptable usando redes neuronales artificiales, que han demostrado ser algoritmos de aproximación universal donde se requiere esencialmente de datos del sistema para poder aproximar el comportamiento no lineal de éste. En este trabajo además de la validación y el estudio comparativo mediante simulaciones numéricas también se considera la construcción de un prototipo de robot SCARA con el fin de obtener una plataforma física que permita conocer los problemas asociados a los sistemas físicos. Por último, se comparará el algoritmo basado en redes neuronales artificiales con técnicas de control convencionales como el control PID y el control adaptable convencional.

1.1. Estado del Arte

En esta sección se aborda un breve estudio del estado del arte orientado a las técnicas de control existentes para determinar las ventajas y desventajas de algunas de éstas. Una de las técnicas predominantes en muchos sistemas de control automático es el **control PID**, que es un sistema de control en lazo cerrado formulado como sigue:

$$u(t) = k_p e(t) + k_i \int e(t) dt + k_d \frac{d(e)}{dt} \quad (1.1)$$

donde $u(t)$ es la señal de control, $e(t)$ es la señal de error, k_p es la ganancia proporcional, k_d es la ganancia derivativa y k_i es la ganancia integral. Cada una de estos valores de ganancias es crucial dado que afectará el comportamiento tanto transitorio como en estado estacionario de la salida. La salida es retroalimentada al sistema y comparada con la señal de referencia $r(t)$ para producir la señal de error, que se deriva y se integra para generar la entrada de control con las tres partes mencionadas. El control PID es simple y ha sido usado ampliamente en muchas aplicaciones de la robótica [16]. Sin embargo, el control PID tiene solo buena estabilidad en un sistema que se comporta de manera lineal o cerca de la zona lineal [17]. Por lo tanto, en la mayoría de los casos, el desempeño de un control PID no será suficiente cuando el sistema se comporta de manera no lineal [18, 19]. Además, el control PID presenta un desempeño insuficiente para contrarrestar los requisitos de operación de alta velocidad y alta precisión que surgen normalmente cuando se utilizan robots del tipo SCARA[20]. Para superar la respuesta lineal del control PID, muchos enfoques se orientan a cambiar activamente las ganancias del control PID basado en sus condiciones de operación, teniendo así un sistema de control mediante ganancias programadas [21]. A continuación se mencionan algunos casos de la aplicación del control PID en el robot SCARA. En muchos de los casos se ha usado en robots SCARA para reducir los errores de posicionamiento y en las acciones del manipulador. Un controlador PID fue diseñado para el control de trayectoria de un robot SCARA bajo variación de carga y condiciones de trayectoria variable minimizando la medida de desempeño basado en la integral del error absoluto ponderado en el tiempo (ITAE derivado del inglés: *integral time absolute error*), para estimar el valor de inercia. Se encontró que la vibración del manipulador fue reducida por la aplicación del control PID [22]. Los autores en [23] , desarrollaron un control PID basado en control de modelo interno (IMC, del inglés *internal mode control*) para un robot SCARA de dos eslabones demostrando un control preciso de las posiciones angulares del robot sin error de estado estacionario. Sin embargo, este controlador fue realizado solo en simulación con Matlab/Simulink.

Dentro de los métodos de inteligencia computacional que pretenden establecer sistemas de control no lineal se encuentra el **control difuso**. El control difuso se deriva de la lógica matemática generalizada, que a diferencia de la lógica booleana, define distintos valores dentro del intervalo $[0, 1]$. La señal en este intervalo de valores puede ser obtenida de las variables de entrada y se asocia a funciones de membresía que describen con etiquetas el comportamiento de una variable. Por ejemplo, en el caso de la señal de error podemos usar etiquetas como cero, poco positivo, muy positivo, poco negativo, muy negativo o etiquetas como alto, bajo, medio, etc. De esta manera, el conocimiento de un experto, en términos de las variables que afectan al sistema, puede ser estructurado por reglas del tipo causa-efecto (Si-Entonces) [24]. Un ejemplo de implementación de un control difuso se presenta en [25], donde las ganancias del controlador PID son activamente ajustadas de acuerdo al promedio ponderado del sistema de inferencia difuso, mejorando un esquema por ganancias fijas que requieren pre-ajuste y no son adaptables al comportamiento no-lineal del robot SCARA. También en [26], se muestran las mejoras sobre el control PID usando un control difuso mejorando la suavidad de transición y la estabilidad del robot. Otra ventaja es que la implementación de un control difuso es simple aunque requiere un buen ajuste de las reglas y funciones de membresía. El control difuso, no fue bien aceptado por la falta en un principio de elementos matemáticos que se encuentran en los modelos matemáticos analíticos, pero se han ido incorporando la formalidad matemática para analizar estabilidad, aproximación universal y otros conceptos importantes en teoría de control [27].

Otro método en inteligencia computacional muy usado para control de robots son las **Redes Neuronales Artificiales (RNA)**. Las RNA han sido empleadas no sólo en investigación sino en la industria para mejorar la robustez, la exactitud y así superar diferentes condiciones operativas de los sistemas, en donde es una tarea complicada obtener modelos analíticos de éstos. Las RNA tratan de imitar la capacidad de aprendizaje y de adaptabilidad de un cerebro humano, al tener una

gran cantidad de elementos de procesamiento (neuronas), interconexiones y capas internas de estos elementos neuronales. En [28], se observó que una RNA bien entrenada puede alcanzar una mejor precisión de control que un PID. De igual manera en [20], la red neuronal bien entrenada estima la posición y el error de movimiento para activamente ajustar las ganancias de un controlador PID y así, tratar con el comportamiento no-lineal del robot SCARA. En [29], una RNA fue empleada para calcular los coeficientes no-lineales de la ecuación de Euler-Lagrange de un robot SCARA, se concluyó que es posible obtener un modelo basado en una red neuronal de propagación hacia adelante (feedforward) basado sólo en información de las entradas y las salidas del robot, aunque se pretende mejorar el algoritmo de aprendizaje para eliminar el error al principio del seguimiento de trayectoria. Finalmente, una de las desventajas de estos algoritmos computacionales, es la falta de estructuras matemáticas convencionales para análisis matemático e interpretabilidad. Para algunos sistemas se requiere una cantidad considerable de datos de entrenamiento, como en el caso de las redes neuronales profundas, o se requieren mejores algoritmos para evitar sobre-entrenamiento o bajo-entrenamiento (overfitting and underfitting), o no se pueden lograr implementaciones en hardware en tiempo real debido a la cantidad de recursos computacionales [30]. Existen otros métodos en inteligencia computacional que son usados esencialmente para optimización heurística. Por ejemplo, en [31], un algoritmo basado en optimización de enjambre de partículas (PSO, siglas del inglés: *Particle Swarm Optimization*) fue empleado para definir las ganancias óptimas de un control PID para un robot SCARA y otros casos similares mostrados en [32, 33]. Otros trabajos usan otro tipo de metaheurísticas como algoritmos genéticos, por ejemplo en [34], se emplea un algoritmo genético para optimizar los parámetros de un control difuso, es decir, sus funciones de membresía y sus reglas.

Dentro de las técnicas relevantes en control avanzado no lineal se encuentra el **control adaptable**. El control adaptable usualmente contiene cuatro bloques, que son la planta (modelo matemático del sistema bajo estudio), el modelo de referencia

(modelo matemático de comportamiento deseado), la ley de control y el mecanismo de adaptación, en donde los componentes de control son diseñados de acuerdo a la planta. El mecanismo de adaptación el cual es la ley de control continuamente ajustará los parámetros del controlador de acuerdo a las condiciones de operación del ambiente y teniendo en cuenta al modelo de referencia [35]. El control adaptable puede desempeñarse bien con parámetros desconocidos [36]. Dado que la incertidumbre del ambiente puede predecirse por el controlador, el desempeño del controlador en términos de exactitud es mejor que un controlador de ganancias fijas tal como el control PID, especialmente con parámetros desconocidos. La habilidad de auto-adaptación en ambientes desconocidos de un controlador adaptable puede garantizar la estabilidad y convergencia del controlador [37].

Como se mencionó anteriormente, diferentes técnicas de control se han presentado de acuerdo a la tarea que realiza el robot, desde un control lineal como el control PID, que sigue siendo uno de los más utilizados y por lo tanto es un estándar de comparación, hasta técnicas de control no lineales basadas en algoritmos computacionales o de conocimiento estructurado por medio de un experto que no requieren modelos matemáticos. También se han propuesto técnicas de control no lineal capaces de reaccionar a disturbios o cambios en condiciones de operación que han presentado excelentes resultados como el control adaptable. La motivación de este trabajo es principalmente implementar técnicas de control donde se fusiona lo mejor de cada técnica como es el caso del control adaptable. Sin embargo, hay ciertos elementos no-lineales que no son fáciles de modelar por medio de ecuaciones analíticas, pudiendo entonces usar las ventajas de algoritmos computacionales que aproximen los elementos desconocidos en estructuras de control analíticas ampliamente probadas.

1.2. Planteamiento del problema

Los robots manipuladores se encuentran dentro de los diversos sistemas de control de movimiento que están sujetos a no-linealidades tales como fricción, zonas muertas, efectos elásticos, entre otros [1]. En muchas ocasiones no existen modelos analíticos de estos efectos no-lineales e ignorarlos para simplificar el diseño del sistema de control conlleva a errores en estado estacionario, respuestas transitorias deficientes y un desempeño limitado durante la operación [38, 39]. Es relevante en la investigación de control de robots continuar con implementaciones de leyes de control no-lineal que mejoran el desempeño y la precisión en tareas como seguimiento de trayectorias o de regulación [40, 41]. Las técnicas de control lineal como el control PID, limitan los rangos de operación y tienen una baja calidad de desempeño sobre todo en zonas de operación distintas a las que fueron diseñadas [42, 43].

1.3. Justificación

Al desarrollar técnicas de control no lineal, resulta complicado tener conocimiento exacto de modelos matemáticos y parámetros que definen el comportamiento de los sistemas físicos [44, 45], por lo que muchas veces se recurre a emplear algoritmos computacionales que tienen la propiedad de aproximar el comportamiento global o de algunos parámetros del sistema a través del entrenamiento de estos algoritmos con datos extraídos directamente del sistema bajo estudio. Es por eso, que en este trabajo de tesis se quiere abordar no solo el uso de técnicas de control no lineal, sino tener una fusión con algoritmos computacionales que permitan mejorar el desempeño y la precisión de las tareas asignadas al robot, así como tener una comparativa con las técnicas ya conocidas y, finalmente, la implementación para conocer las limitaciones físicas de estos sistemas empleados en tareas de automatización.

1.4. Hipótesis

El control adaptable con redes neuronales artificiales proporcionará mejor precisión y desempeño para el control de seguimiento de trayectoria y de regulación de un robot SCARA, donde los términos no lineales del robot manipulador son aproximados mediante redes neuronales del tipo de funciones de base radial (RBF, siglas del inglés: *Radial Basis Functions*).

1.5. Objetivos

1.5.1. Objetivo General

Implementar y comparar leyes de control lineales y no lineales combinadas con técnicas de aprendizaje automático para aproximación de elementos no lineales que mejoren el desempeño y precisión de las tareas del robot.

1.5.2. Objetivos Específicos

1. Diseñar, imprimir, ensamblar e instrumentar un robot SCARA de 4 grados de libertad.
2. Modelar e identificar los parámetros del robot SCARA
3. Estudiar el estado del arte para proponer técnicas de control que mejoren el desempeño del robot SCARA.
4. Implementar, comparar y discutir las mejoras y limitaciones de las técnicas de control empleadas.

1.6. Organización del documento

Este trabajo se encuentra organizado de la siguiente manera: en el capítulo 1 se presentó la introducción así como los objetivos perseguidos en este trabajo de tesis y un breve estudio del estado del arte. En el capítulo 2 se presentan las consideraciones teóricas que respaldan este trabajo de investigación tales como los modelos cinemático y dinámico del robot, así como la descripción de la técnica de aprendizaje automático empleada en la presente Tesis. En el capítulo 3 se menciona la metodología de este trabajo. Posteriormente, en el capítulo 4 se presentan el desarrollo de las técnicas de control seleccionadas, en simulación e implementadas en el robot manipulador SCARA de cuatro grados de libertad. Finalmente, en el capítulo 5 se genera discusión acerca de las técnicas de control implementadas en esta tesis, así como los resultados obtenidos, definiendo la conclusión general y los trabajos futuros.

Capítulo 2

Marco Teórico

2.1. Introducción

Un robot manipulador se compone de una serie de eslabones conectados mediante uniones con el propósito de realizar una o varias tareas. Así mismo, contiene un conjunto de motores ubicados en las articulaciones los cuales son mecanismos controlados para llevar a cabo una tarea específica. En la mayoría de los casos, en el extremo del robot se encuentra una herramienta que le permite interactuar con el entorno, denominado efector final [2].

Los robots manipuladores son sistemas altamente complejos que requieren el estudio de ecuaciones diferenciales no lineales para comprender su funcionamiento. A pesar de su naturaleza, se suelen involucrar una amplia variedad de formulaciones que se abordan como problemas de control. La importancia de contar con un modelo dinámico de un robot manipulador radica en que este modelo puede ser utilizado en las etapas de diseño mecánico, lo que permite estudiar el robot antes de su construcción física.

Además, la utilidad de los parámetros dinámicos se manifiesta en la imple-

mentación de sistemas de control que se basan en el modelo dinámico del robot manipulador. La precisión de estos esquemas de control depende en gran medida de la exactitud de los parámetros del modelo dinámico.

En este capítulo, se presentarán fundamentos teóricos relacionados con los modelos dinámicos de los robots manipuladores. Con este propósito, la sección 2.2 introduce una metodología para obtener el modelo del robot manipulador SCARA de cuatro grados de libertad.

2.2. Características del robot manipulador SCARA

Dado que un robot manipulador es un sistema mecánico complejo cuya descripción analítica requiere de ecuaciones diferenciales no lineales, esto nos permite situar puntos de interés los cuales consisten en el diseño de sistemas controladores de seguimiento para realizar movimiento de alta velocidad y exactitud, como por lo general los controladores de seguimiento se basan en el modelo dinámico del robot manipulador, es de vital importancia contar con un modelo que describa fielmente los efectos dinámicos de su comportamiento real [3].

Los robots manipuladores son manejados por actuadores eléctricos, hidráulicos o neumáticos, a los cuales es necesario aplicar cierta fuerza (pares / torques) por medio de las uniones del robot. La dinámica de un robot manipulador describe como es la respuesta de movimiento debida a estas fuerzas. Para realizar un modelo de robots manipuladores es necesario incluir algunas disciplinas de la física, tales como propiedades de materiales, dinámica, electrónica y teoría de control.

Existen varias ventajas al utilizar el modelo dinámico del robot manipulador, una de ellas es la utilidad en la construcción mecánica. Además, es necesario para obtener resultados de simulación del controlador propuesto, ya que, el comportamiento de estos esquemas depende de la exactitud en el valor numérico de los parámetros

del modelo dinámico con respecto a los valores reales.

En este trabajo se considera la implementación de un robot manipulador SCARA (siglas del ingles: *Selective Compliant Articulated Robot for Assembly*), el cual es altamente empleado en distintos ámbitos industriales, debido a su alta precisión en tareas de posicionamiento y manipulación, ideales para aplicaciones que requieren movimientos exactos y repetibles. También mantienen un diseño compacto y ocupan menos espacio en comparación con otros tipos de robots. También cabe destacar que este tipo de manipuladores cuenta con un espacio de trabajo relativamente amplio en el plano horizontal.

En la Figura 2.1 Se visualiza el esquema de un robot manipulador de tipo SCARA.

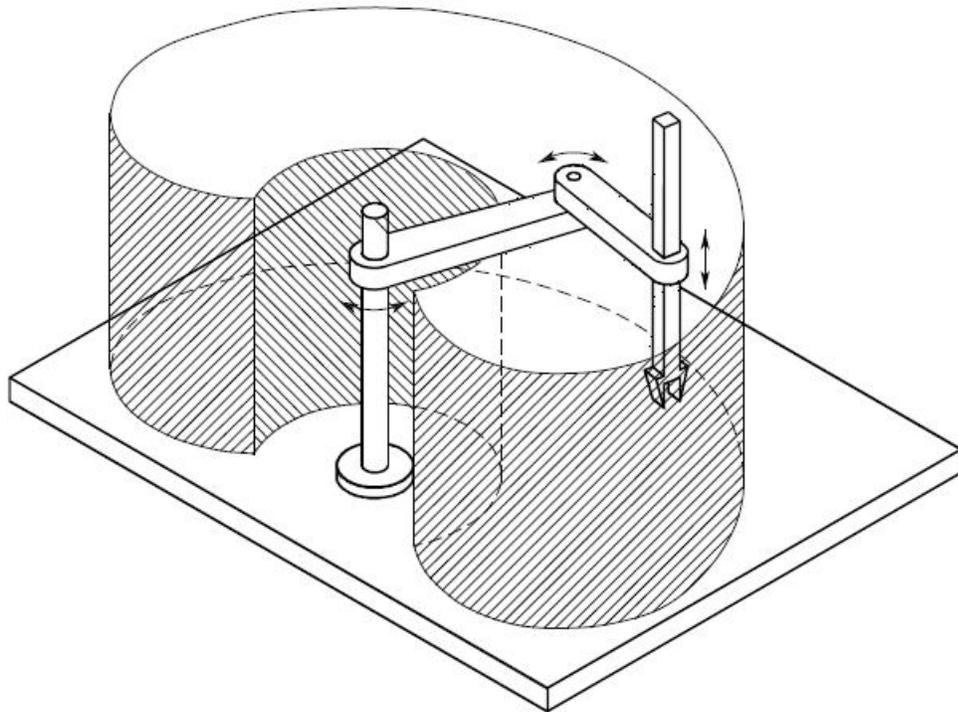


Figura 2.1: Robot manipulador SCARA y su espacio de trabajo [1].

Los grados de libertad de un robot (DOF, siglas del ingles: *Degrees of freedom*), se refieren al número de movimientos independientes que cada articulación o compo-

nente del robot puede realizar. Estos grados de libertad determinan la versatilidad y capacidad de movimiento del robot en su entorno [46].

El robot SCARA, generalmente cuenta con tres grados de libertad: dos articulaciones rotacionales que proporcionan movimiento en el plano horizontal, junto con una articulación prismática que permite el desplazamiento vertical del efector final. Sin embargo, en este caso en particular, se ha incorporado un cuarto grado de libertad adicional, que habilita el giro horizontal del efector final. Como resultado, el robot manipulador en uso se clasifica como un brazo robótico SCARA de cuatro grados de libertad, de los que tres grados son rotacionales y uno es prismático [47].

La cinemática del brazo robótico se enfoca en el análisis geométrico del movimiento del robot en relación con un sistema de coordenadas de referencia fijo, sin tener en cuenta las fuerzas o momentos que generan ese movimiento. En otras palabras, se centra en describir de manera analítica cómo el robot se desplaza en el espacio a medida que varía el tiempo [2]. Esto incluye el estudio de las relaciones entre las variables que describen la posición y orientación del extremo final del robot y las variables articulares que controlan sus movimientos. La descripción cinemática del robot consiste en calcular los modelos cinemáticos directo e inverso, como se muestra en la Figura 2.2.

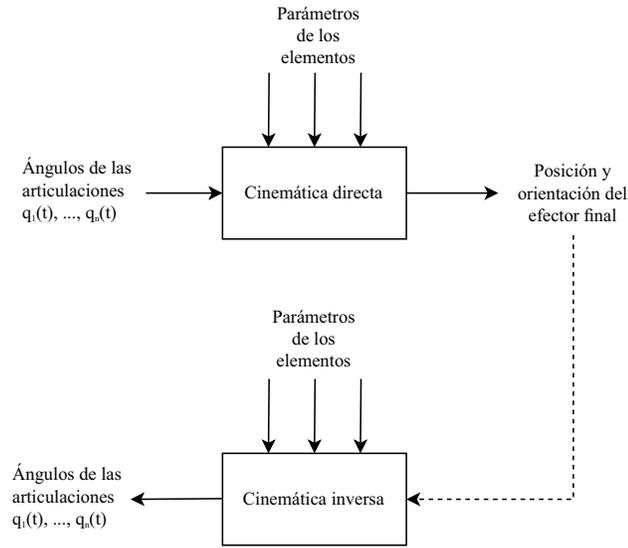


Figura 2.2: Modelo de cinemática directa e inversa [2].

2.3. Modelo cinemático directo

Las variables independientes en un brazo robótico corresponden a las posiciones articulares y las tareas de posicionamiento se suelen dar en términos de las coordenadas de la tarea, que suelen representarse en el espacio Cartesiano.

El modelo cinemático directo es el visualizado en la Figura 2.3, donde se muestran los ángulos de las articulaciones de la ecuación 2.1 y los parámetros respectivos de la geometría del robot, donde n es el número de grados de libertad, donde se tiene la condición de que para un conjunto de coordenadas de referencia se obtiene la orientación y posición del efector final del manipulador.

$$q(t) = q_1(t), q_2(t), \dots, q_n(t)^T \quad (2.1)$$

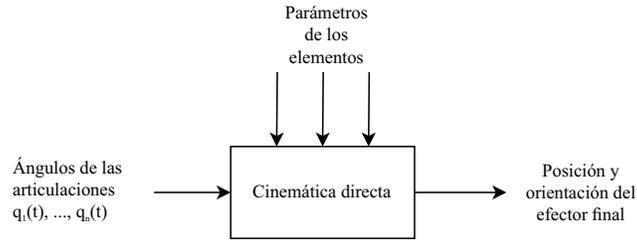


Figura 2.3: Esquema de cinemática directa [2].

Para este método se desarrollan el conjunto de convenciones de Denavit–Hartenberg (D–H) que proporcionan un procedimiento sistemático para desarrollar el análisis de cinemática directa [46]. En esta situación cada matriz homogénea se representa como un producto de cuatro transformaciones básicas, ecuación 2.2:

$$A_i = R_{z,\theta_i} * T_{z,d_i} * T_{x,a_i} * R_{x,\alpha_i} , \quad (2.2)$$

Que se denominan:

- La longitud del eslabón, (a_i)
- El giro del eslabón, (α_i)
- El desplazamiento del eslabón, (d_i)
- El ángulo de giro de la articulación, (θ_i)

Las cuales se expresan en un producto matricial:

$$A_i = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

Resultando en una matriz de transformación homogénea denominada A_i :

$$A_i = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

El número de matrices homogéneas de paso " A_i " depende de los grados de libertad del robot manipulador. Para poder definir los parámetros de Denavit–Hartenberg se requieren una serie de pasos clave especificados a continuación [48].

Este método se puede revisar en el anexo A.

1. Representación funcional.

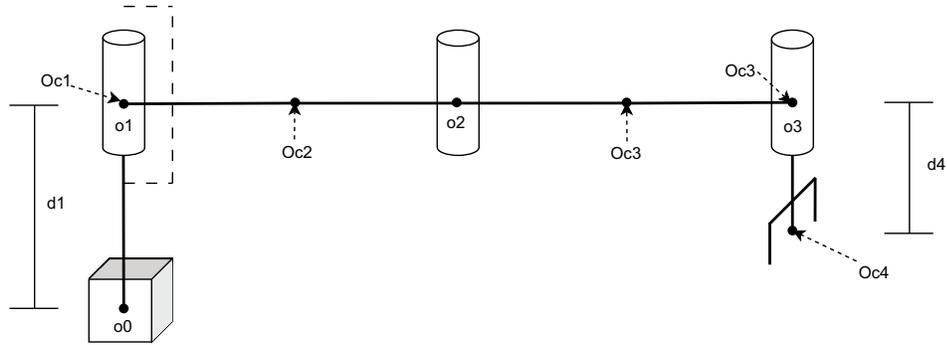


Figura 2.4: Modelo funcional SCARA.

Se posiciona al robot manipulador en estado inicial, en este caso de manera horizontal y/o paralela al eje de las abscisas, esto comúnmente se realiza como un estándar característico de este método el cual se explica a continuación.

2. Sistema de coordenadas generalizadas.

Para asignar la orientación inicial del robot y especificar las coordenadas generalizadas se deben tener en cuenta las siguientes consideraciones:

- No se puede desplazar ni rotar respecto al eje de las ordenadas o "y".
- Los ejes z_i coinciden con los ejes de acción de las articulaciones (Rotación y traslación en el eje z).
- $x_i \perp z_i$ y x_{i-1} .

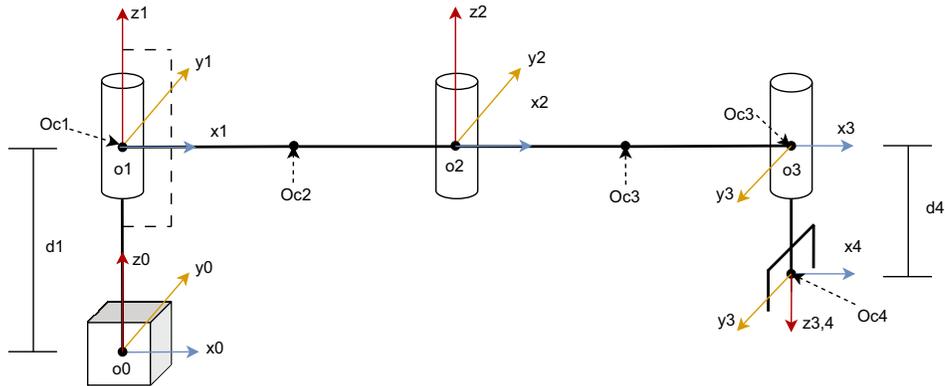


Figura 2.5: Sistema de coordenadas generalizadas x, y, z .

3. Tabla de parámetros.

Para este caso se utilizan un conjunto de cuatro parámetros, de los cuales se contemplan tres constantes y uno variable por cada articulación, esta última denotada por \mathbf{q} especificada en la Tabla 2.1. Por lo que en base al diagrama de la Figura 2.5 se obtienen:

- Giros en z (θ_i).
- Desplazamientos en z (d_i).
- Giros en x (α_i).

- Desplazamientos en x (a_i).

Articulación (i)	z		x		q
	θ_i	d_i	a_i	α_i	
1	0	q1	0	0	d1
2	q2	0	a2	0	d2
3	q3	0	a3	180°	d3
4	a4	d4	0	0	d4

Tabla 2.1: Parámetros Denavit–Hartenberg.

4. Sustituyendo los parámetros de Denavit–Hartenberg en las matrices homogéneas de paso respectivas de cada articulación a partir de la ecuación 2.4:

$$A_1 = \begin{bmatrix} c_{\theta_1} & -s_{\theta_1}c_{\alpha_1} & s_{\theta_1}s_{\alpha_1} & a_1c_{\theta_1} \\ s_{\theta_1} & c_{\theta_1}c_{\alpha_1} & -c_{\theta_1}s_{\alpha_1} & a_1s_{\theta_1} \\ 0 & s_{\alpha_1} & c_{\alpha_1} & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

$$A_2 = \begin{bmatrix} c_{\theta_2} & -s_{\theta_2}c_{\alpha_2} & s_{\theta_2}s_{\alpha_2} & a_2c_{\theta_2} \\ s_{\theta_2} & c_{\theta_2}c_{\alpha_2} & -c_{\theta_2}s_{\alpha_2} & a_2s_{\theta_2} \\ 0 & s_{\alpha_2} & c_{\alpha_2} & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

$$A_3 = \begin{bmatrix} c_{\theta_3} & -s_{\theta_3}c_{\alpha_3} & s_{\theta_3}s_{\alpha_3} & a_3c_{\theta_3} \\ s_{\theta_3} & c_{\theta_3}c_{\alpha_3} & -c_{\theta_3}s_{\alpha_3} & a_3s_{\theta_3} \\ 0 & s_{\alpha_3} & c_{\alpha_3} & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_3 & s_3 & 0 & a_3c_3 \\ s_3 & -c_3 & 0 & a_3s_3 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.7)$$

$$A_4 = \begin{bmatrix} c_{\theta_4} & -s_{\theta_4}c_{\alpha_4} & s_{\theta_4}s_{\alpha_4} & a_4c_{\theta_4} \\ s_{\theta_4} & c_{\theta_4}c_{\alpha_4} & -c_{\theta_4}s_{\alpha_4} & a_4s_{\theta_4} \\ 0 & s_{\alpha_4} & c_{\alpha_4} & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_4 & s_4 & 0 & 0 \\ s_4 & -c_4 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

5. Definición de la matriz de transformación del robot manipulador [2].

$$T_0^4 = A_0 * A_1 * A_2 * A_3 \quad (2.9)$$

Obteniendo:

$$\begin{bmatrix} c_{234} & s_{234} & 0 & a_3c_{23} + a_2c_2 \\ s_{234} & -c_{234} & 0 & a_3s_{23} + a_2c_2 \\ 0 & 0 & -1 & q_1 - d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.10)$$

2.4. Modelo cinemático inverso

En el caso contrario, en el método de cinemática inversa se especifica una orientación y posición para el efector final del robot manipulador, así como también, se ingresan algunos parámetros de los elementos respectivos de un sistema de coordenadas de referencia, como datos de entrada. A partir de lo anterior se podrán obtener las configuraciones de las variables articulares para dicha orientación [2]. Este método se puede observar en la Figura 2.6.

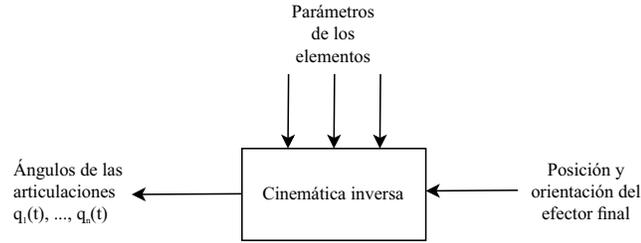


Figura 2.6: Esquema de cinemática inversa [2].

Teniendo la matriz de transformación del robot manipulador con la siguiente estructura:

$$T = T_0^4 = \begin{bmatrix} nx & ox & ax & px \\ ny & oy & ay & py \\ nz & oz & az & pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.11)$$

Esta matriz fue determinada en el cálculo cinemático directo previamente visto. Ahora, se obtienen el punto de orientación del efector final en términos de las coordenadas x , y y z deseadas para el efector final [49], denotadas en la ecuación 2.12 como px , py y pz , respectivamente:

$$\begin{cases} px = a_3 c_{23} + a_2 c_2 \\ py = a_3 s_{23} + a_2 s_2 \\ pz = q_1 - d_4 \end{cases} \quad (2.12)$$

Por lo que se obtienen las cuatro variables articulares definidas para el robot manipulador SCARA:

$$q_{d1} = pz + d_4 \quad (2.13)$$

$$D = \frac{(px)^2 + (py)^2 - (a_3)^2 - (a_2)^2}{2a_2a_3} \quad (2.14)$$

$$q_{d3} = \arctan 2(\pm\sqrt{1 - D^2}, D) \quad (2.15)$$

$$q_{d2} = \arctan 2(py, px) - \arctan 2(a_3s_3, a_3c_3 + a_2) \quad (2.16)$$

$$q_{d4} = q_2 + q_3 - \varphi_d \quad (2.17)$$

Donde φ_d es la orientación deseada en el plano para el efector final.

Las variables determinadas corresponden a una prismática q_1 y tres articulares q_2 , q_3 y q_4 .

2.5. Modelo dinámico del manipulador

Las ecuaciones de movimiento de un manipulador son un conjunto de expresiones matemáticas que describen el comportamiento dinámico el robot. Estas ecuaciones son fundamentales para diversas aplicaciones, como la simulación por computadora del movimiento del robot, el diseño de estrategias de control adecuadas y la evaluación de la estructura del brazo robótico.

El objetivo principal del control basado en computadora de un manipulador es asegurar que su respuesta dinámica se ajuste a ciertos criterios de rendimiento y objetivos específicos.

En última instancia, el rendimiento dinámico de un manipulador está estre-

chamente relacionado con la efectividad de los algoritmos de control utilizados y con la precisión de su modelo dinámico. Por lo tanto, el problema de control implica la obtención de modelos dinámicos a partir del robot manipulador físico para después especificar las leyes o estrategias de control para el mismo, lo que implica el proceso de modelar y evaluar las propiedades y el comportamiento dinámico de los robots controlados por computadora.

El modelo dinámico de un robot se puede derivar utilizando principios físicos bien establecidos, como las leyes de la mecánica Newtoniana y Lagrangiana. Este enfoque implica la formulación de las ecuaciones de movimiento dinámico que describen el comportamiento de las distintas articulaciones del manipulador en función de los parámetros geométricos e inerciales de sus componentes, incluyendo las fuerzas externas que interactúan con éste [3].

Existen métodos convencionales los cuales se pueden aplicar sistemáticamente para desarrollar las ecuaciones de movimiento del robot manipulador. Algunos de estos métodos son: Newton–Euler (N-E) y Euler–Lagrange (E-L), entre otros. Estas formulaciones tienen cierta equivalencia de acuerdo a que describen el comportamiento dinámico del mismo brazo manipulador. Sin embargo, existen diferencias en su estructura, esto debido principalmente y enfatizando a las razones de aplicación y objetivos de control [2].

Por un lado, se pueden obtener tiempos de cálculo rápido en la evaluación de los pares de las articulaciones nominales para controlar el manipulador. Otras permiten facilitar el análisis y síntesis de control. Además, existen algunos métodos que mejoran la simulación en computadora del movimiento del robot.

La obtención del modelo dinámico de un manipulador utilizando la formulación de Euler-Lagrange es un proceso simple y sistemático. Suponiendo el movimiento del cuerpo rígido y excluyendo la dinámica de los dispositivos de control electrónico, sin embargo, para este trabajo se contempla la fricción de los engranajes y partes mecáni-

cas del brazo robótico. A partir de esto, las ecuaciones de movimiento resultantes son un conjunto de ecuaciones diferenciales no lineales acopladas de segundo orden.

Las ecuaciones de Euler–Lagrange describen de forma explícita la dinámica del robot. Estas ecuaciones son útiles para analizar y diseñar estrategias de control avanzadas que operan en el espacio de las variables de articulación.

A partir de lo anterior, se logran estudiar e implementar dos tipos de esquemas altamente relacionados, definidos como dinámica directa y dinámica inversa para el sistema del robot manipulador SCARA de 4 grados de libertad.

El primero, es característico dadas las fuerzas y pares deseados, se utilizan las ecuaciones dinámicas para determinar las aceleraciones de las variables articulares, para posteriormente realizar un conjunto de integraciones y obtener las velocidades y coordenadas generalizadas del mismo, Figura 2.7.

El segundo, obtenidas las coordenadas generalizadas deseadas y sus primera derivada temporal \mathbf{q} y $\dot{\mathbf{q}}$, se calculan las fuerzas y pares generalizados, es decir, la respuesta de la ley de control τ .

Si bien, en las dos situaciones anteriores se necesitan calcular previamente las matrices \mathbf{M} , \mathbf{C} , y \mathbf{G} , definidos como la matriz de inercia, la matriz de Coriolis y el vector de pares gravitacionales, respectivamente. Adicionalmente, se puede añadir la matriz diagonal \mathbf{D} correspondiente a los coeficientes de fricción.

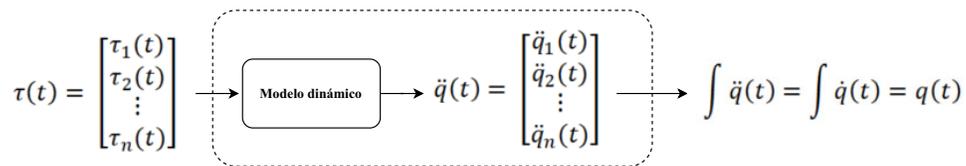


Figura 2.7: Esquema del modelo dinámico directo.

De la misma manera, en la Figura 2.8 se observa la representación del modelo

dinámico inverso.



Figura 2.8: Esquema del modelo dinámico inverso.

Se puede observar que en el modelo dinámico directo se tiene como entrada el torque o fuerzas aplicadas y en su salida las aceleraciones de las variables articulares de acuerdo a la dinámica del manipulador. Por otro lado, en el modelo dinámico inverso, ingresan las variables articulares, tanto posiciones, velocidades y aceleraciones deseadas para obtener a la salida los torques requeridos para lograr dicho movimiento.

2.5.1. Propiedades del modelo dinámico

A pesar de la complejidad de la ecuación que describe el comportamiento dinámico del robot manipulador, esta presenta varias propiedades fundamentales que pueden ser aprovechadas para simplificar el diseño de sistemas de control [3].

1. La matriz de inercia $\mathbf{M}(\mathbf{q})$ es una matriz simétrica definida positiva, lo que significa que $\mathbf{M}(\mathbf{q})$ es igual a su traspuesta y todos sus valores son positivos:

$$\mathbf{M}(\mathbf{q}) = \mathbf{M}(\mathbf{q})^T > 0 \quad (2.18)$$

Esta propiedad es especialmente relevante porque garantiza la existencia de la matriz inversa de $\mathbf{M}(\mathbf{q})$, que también es simétrica y definida positiva.

$$\mathbf{M}(\mathbf{q})^{-1} \quad (2.19)$$

2. La matriz $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ y la derivada en el tiempo $\dot{\mathbf{M}}(\mathbf{q})$ de la matriz de inercia cumplen con la siguiente relación:

$$\dot{\mathbf{q}}^T \left[\frac{1}{2} \dot{\mathbf{M}}(\mathbf{q}) - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \right] \dot{\mathbf{q}} = 0 \in \mathbb{R}, \forall \mathbf{q}, \dot{\mathbf{q}} \in \mathbb{R}^n \quad (2.20)$$

3. La matriz $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ satisface:

$$\mathbf{C}(\mathbf{q}, 0) = 0 \quad \forall \mathbf{q} \in \mathbb{R}^n \quad (2.21)$$

4. Para el caso de robots provistos únicamente de articulaciones rotacionales, existe una constante positiva k_{c1} que satisface la siguiente desigualdad:

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} \leq k_{c1} \|\dot{\mathbf{q}}\| \|\mathbf{q}\| \quad (2.22)$$

Para todo $\dot{\mathbf{q}}, \mathbf{q} \in \mathbb{R}^n$.

Estas propiedades son fundamentales para simplificar el análisis de sistemas de control para robots manipuladores.

2.5.2. Descripción y aplicación de la técnica de Euler–Lagrange

Un método convencional para generar el modelo dinámico de un robot manipulador que consta de $n + 1$ eslabones rígidos conectados en serie es utilizar las ecuaciones de movimiento de Euler–Lagrange (E–L), las cuales se describen a continuación para el robot manipulador de tipo SCARA de cuatro grados de libertad.

Considerando un robot manipulador de n grados de libertad, formado por eslabones rígidos conectados por articulaciones libres de elasticidad, la energía total

$\epsilon(\mathbf{q}, \dot{\mathbf{q}})$ del robot está dada por la suma de la energía cinética $\kappa(\mathbf{q}, \dot{\mathbf{q}})$ y la energía potencial $v(\mathbf{q}, \dot{\mathbf{q}})$:

$$\epsilon(\mathbf{q}, \dot{\mathbf{q}}) = \kappa(\mathbf{q}, \dot{\mathbf{q}}) + v(\mathbf{q}, \dot{\mathbf{q}}) \quad (2.23)$$

Donde $\mathbf{q} \in \mathbb{R}^n$ es el vector de posiciones y $\dot{\mathbf{q}} \in \mathbb{R}^n$ es el vector de velocidades articulares.

Esta relación de energías nos permite formular el concepto de Lagrangiano [50], el cual es una función de la que se puede determinar la evolución temporal, las leyes de conservación, entre otras propiedades físicas fundamentales de un sistema dinámico. Matemáticamente, el Lagrangiano $L(\mathbf{q}, \dot{\mathbf{q}})$ de un robot manipulador es definido como la diferencia entre la energía cinética $\kappa(\mathbf{q}, \dot{\mathbf{q}})$ y potencial $u(\mathbf{q}, \dot{\mathbf{q}})$, expresado de la siguiente manera:

$$L(\mathbf{q}, \dot{\mathbf{q}}) = \kappa(\mathbf{q}, \dot{\mathbf{q}}) - u(\mathbf{q}, \dot{\mathbf{q}}) \quad (2.24)$$

La energía cinética $\kappa(\mathbf{q}, \dot{\mathbf{q}})$ asociada a un robot manipulador es una función cuadrática del vector de velocidad $\dot{\mathbf{q}}$ que puede ser expresada como:

$$\kappa(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} \quad (2.25)$$

Donde $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ es la matriz de inercia del manipulador, la cual tiene propiedades de ser simétrica y definida positiva. Así mismo, la energía potencial $u(\mathbf{q})$ de un robot manipulador se representa generalmente como una relación de la masa, la gravedad y la distancia respecto a un punto de referencia.

$$u(\mathbf{q}) = \sum_{i=1}^n m_i g h_i(\mathbf{q}) \quad (2.26)$$

Donde g es la constante de aceleración de la gravedad y $h_i(\mathbf{q})$ es la función de altura del centro de masa del eslabón i .

Dados los fundamentos anteriores, y determinado el Lagrangiano para el respectivo diseño, se prosigue a emplear las ecuaciones de movimiento de acuerdo al método de E-L:

$$\frac{d}{dt} \frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} - \frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} = \tau - f(\dot{\mathbf{q}}, \tau) \quad (2.27)$$

Donde $\tau \in \mathbb{R}^n$ es el vector de fuerzas y/o pares generalizados que actúan sobre las articulaciones del robot. El vector de fuerzas o pares de fricción $f(\dot{\mathbf{q}}, \tau) \in \mathbb{R}^n$ presente en las articulaciones del mismo, para el caso en donde se consideren [51].

Dado el vector de coordenadas Cartesianas del efector final del robot o_x, o_y, o_z y su orientación φ :

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{q}) = \begin{bmatrix} o_x \\ o_y \\ o_z \\ \varphi \end{bmatrix} = \begin{bmatrix} a_3 c_{23} + a_2 c_2 \\ a_3 s_{23} + a_2 s_2 \\ q_1 - d_4 \\ q_3 + q_2 - q_4 \end{bmatrix} \quad (2.28)$$

Se calcula el Jacobiano analítico a partir de las derivadas parciales con respecto a q , como se muestra a continuación:

$$\dot{\mathbf{x}} = \frac{\partial \mathbf{f}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} \quad (2.29)$$

Obteniendo:

$$\mathbf{J}_a = \begin{bmatrix} a_3 c_{23} + a_2 c_2 & -a_3 s_{23} - a_2 s_2 & -a_3 s_{23} & 0 \\ a_3 s_{23} + a_2 s_2 & a_3 c_{23} + a_2 c_2 & a_3 c_{23} & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & -1 \end{bmatrix} \quad (2.30)$$

Continuando con la formulación E-L, se substituyen las energías en la expresión del Lagrangiano:

$$L(\mathbf{q}, \dot{\mathbf{q}}) = \kappa(\mathbf{q}, \dot{\mathbf{q}}) - u(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} - u(q) \quad (2.31)$$

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \dot{\mathbf{M}}(\mathbf{q}) \dot{\mathbf{q}} - \frac{1}{2} \frac{\partial}{\partial q} (\dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}}) + \frac{\partial u(q)}{\partial q} = \tau - \mathbf{f}(\dot{\mathbf{q}}) \quad (2.32)$$

Donde $\dot{\mathbf{M}}(\mathbf{q}) \dot{\mathbf{q}} - \frac{1}{2} \frac{\partial}{\partial q} (\dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}})$ corresponde a la matriz de fuerzas centrípetas y de Coriolis $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}$.

Como $\mathbf{g}(\mathbf{q})$ es obtenido mediante el gradiente de la energía potencial, es decir, $\mathbf{g}(\mathbf{q}) = \frac{\partial u(\mathbf{q})}{\partial \mathbf{q}}$, la ecuación simplificada que describe el comportamiento dinámico del robot manipulador es la siguiente:

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{f}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \tau \quad (2.33)$$

Donde $\mathbf{f}(\dot{\mathbf{q}}) \in \mathbb{R}^n$ es el vector de fricción viscosa y de Coulomb que se contempla para este trabajo.

Las unidades para los elementos de $\mathbf{M}(\mathbf{q})$ son Nm – seg²/rad, para $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ y $\mathbf{D}(\dot{\mathbf{q}})$ son Nm – seg/rad y para el vector $\mathbf{g}(\mathbf{q})$ es Nm [52]. Empleando las ecuaciones de movimiento de Euler–Lagrange se obtiene la matriz de inercia que está dada por:

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \quad (2.34)$$

Donde:

$m_{11} = m_1 + m_2 + m_3 + m_4$
$m_{12} = 0$
$m_{13} = 0$
$m_{14} = 0$
$m_{21} = 0$
$m_{22} = I_{zz2} + I_{zz3} + I_{zz4} + lc2^{2*m2} + a_2^{2*m3} + 2*a_2*c_3*lc3*m_3 + lc_3^{2*m3} + a_2^{2*m4} + a_3^{2*m4} + 2*a_2*a_3*c_3*m_4$
$m_{23} = I_{zz3} + I_{zz4} + a_2*c_3*lc3*m_3 + lc_3^{2*m3} + a_3^{2*m4} + a_2*a_3*c_3*m_4$
$m_{24} = -I_{zz4}$
$m_{31} = 0$
$m_{32} = I_{zz3} + I_{zz4} + a_2*c_3*lc3*m_3 + lc_3^{2*m3} + a_3^{2*m4} + a_2*a_3*c_3*m_4$
$m_{33} = I_{zz3} + I_{zz4} + lc_3^{2*m3} + a_3^{2*m4}$
$m_{34} = -I_{zz4}$
$m_{41} = 0$
$m_{42} = -I_{zz4}$
$m_{43} = -I_{zz4}$
$m_{44} = -I_{zz4}$

Tabla 2.2: Parámetros asignados a cada elemento de la matriz de inercia.

Así mismo, la matriz de fuerza centrífuga y de coriolis toma la forma:

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{bmatrix} \quad (2.35)$$

Donde:

$c_{11} = 0$
$c_{12} = 0$
$c_{13} = 0$
$c_{14} = 0$
$c_{21} = 0$
$c_{22} = -s_3 * a_2 * (lc_3 * m_3 + a_3 * m_4) * \dot{q}_3$
$c_{23} = -s_3 * a_2 * (lc_3 * m_3 + a_3 * m_4) * (\dot{q}_2 + \dot{q}_3)$
$c_{24} = 0$
$c_{31} = 0$
$c_{32} = s_3 * a_2 * (lc_3 * m_3 + a_3 * m_4) * q_{2p}$
$c_{33} = 0$
$c_{34} = 0$
$c_{41} = 0$
$c_{42} = 0$
$c_{43} = 0$
$c_{44} = 0$

Tabla 2.3: Parámetros asignados a cada elemento de la matriz de Coriolis y fuerza centrífuga.

La fricción presente en el robot manipulador es modelada como una combinación lineal de la fricción viscosa y de Coulomb:

$$\mathbf{D}(\dot{\mathbf{q}}) = \begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & d_{34} \\ d_{41} & d_{42} & d_{43} & d_{44} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} \quad (2.36)$$

Dónde:

$d_{11} = cf_1$
$d_{12} = 0$
$d_{13} = 0$
$d_{14} = 0$
$d_{21} = 0$
$d_{22} = cf_2$
$d_{23} = 0$
$d_{24} = 0$
$d_{31} = 0$
$d_{32} = 0$
$d_{33} = cf_3$
$d_{34} = 0$
$d_{41} = 0$
$d_{42} = 0$
$c_{43} = 0$
$d_{44} = cf_4$

Tabla 2.4: Parámetros asignados a cada elemento de la matriz de constantes de fricción.

Finalmente, el vector de pares gravitacionales está dado como:

$$g(\mathbf{q}) = \begin{bmatrix} g_{11} \\ g_{21} \\ g_{31} \\ g_{41} \end{bmatrix} \quad (2.37)$$

Dónde:

$g_{11} = g_0 * (m_1 + m_2 + m_3 + m_4)$
$g_{21} = 0$
$g_{31} = 0$
$g_{41} = 0$

Tabla 2.5: Parámetros asignados a cada elemento del vector de fuerzas gravitacionales.

Dado que la matriz de inercia $\mathbf{M}(\mathbf{q})$ multiplica a la aceleración $\ddot{\mathbf{q}}$, la matriz de Coriolis y la matriz de fricciones a la velocidad $\dot{\mathbf{q}}$ y el vector de pares gravitacionales

sólo depende de la posición \mathbf{q} , se puede obtener la aceleración mediante:

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\tau - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{D}(\dot{\mathbf{q}}) - \mathbf{g}(\mathbf{q})) \quad (2.38)$$

2.6. Control de retroceso no lineal (*backstepping*)

La técnica conocida como control de retroceso no lineal (del inglés: *Backstepping*), es un enfoque de diseño recursivo que se utiliza para desarrollar de manera sistemática tanto leyes de control de retroalimentación como funciones de Lyapunov asociadas. Los diseños de *backstepping* están estrechamente relacionados con el método de linearización de retroalimentación (del inglés: *Feedback linearization*). Sin embargo, a diferencia de estos métodos que cancelan todas las no linealidades en el sistema, el enfoque de *backstepping control* permite al diseñador aprovechar las no linealidades "buenas" mientras se controlan las no linealidades "malas" o desestabilizadoras. Esto se logra al agregar amortiguamiento no lineal, lo que proporciona una mayor robustez al sistema [53].

Para el control de retroalimentación de estado se contemplan las variables de estado x_1 y x_2 correspondientes a las posiciones y velocidades de las articulaciones, respectivamente.

Es decir:

$$x_1 = [q_1, q_2, \dots, q_n]^T \quad (2.39)$$

De la misma manera se tiene:

$$x_2 = [\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n]^T \quad (2.40)$$

Entonces, al derivar la variable de estado x_2 se obtiene la aceleración, tal como se muestra en la ecuación 2.41.

$$\dot{x}_2 = M^{-1}[D(\tau) - C(x_1, x_2)x_2 - f_{dis} - G(x_1)] \quad (2.41)$$

Por otro lado, se tiene la señal de error de posición como:

$$e_1 = x_1 - x_d \quad (2.42)$$

De la misma manera:

$$e_2 = x_2 - \alpha \quad (2.43)$$

Al derivar se obtiene:

$$\dot{e}_2 = \dot{x}_2 - \dot{\alpha} \quad (2.44)$$

Se elige la función barrera de Lyapunov:

$$V_1 = \frac{1}{2} \sum_{i=1}^n \ln \frac{b_i^2}{b_i^2 - e_{1i}^2} \quad (2.45)$$

Derivando con respecto al tiempo:

$$\dot{V}_1 = \sum_{i=1}^n \frac{e_{1i}(e_{2i} + \alpha_i - \dot{x}_{di})}{b_i^2 - e_{1i}^2} \quad (2.46)$$

Entonces se emplea la variable de control virtual α :

$$\alpha = - \begin{bmatrix} k_1 e_{11} - \dot{x}_{d1} \\ k_2 e_{12} - \dot{x}_{d2} \\ \vdots \\ k_n e_{1n} - \dot{x}_{dn} \end{bmatrix} \quad (2.47)$$

Sustituyendo la ecuación 2.47 en la ecuación 2.46 se obtiene:

$$\dot{V}_1 = - \sum_{i=1}^n \frac{k_i e_{1i}^2}{b_i^2 - e_{1i}^2} + \sum_{i=1}^n \frac{e_{1i} e_{2i}}{b_i^2 - e_{1i}^2} \quad (2.48)$$

La segunda función candidata de Lyapunov es:

$$V_2 = V_1 + \frac{1}{2} e_2^T M(x_1) e_2 \quad (2.49)$$

Su derivada es:

$$\dot{V}_2 = \dot{V}_1 + e_2^T M(x_1) \dot{e}_2 + e_2^T \frac{1}{2} \dot{M}(x_1) e_2 \quad (2.50)$$

Sustituyendo la ecuación 2.43 en la ecuación 2.50:

$$\dot{V}_2 = \dot{V}_1 + e_2^T \left[M(x_1) (\dot{x}_2 - \dot{\alpha}) + \frac{1}{2} \dot{M}(x_1) e_2 \right] \quad (2.51)$$

Sustituyendo la ecuación 2.41 y la ecuación 2.43 en la ecuación 2.51, obteniendo:

$$\dot{V}_2 = \dot{V}_1 + e_2^T [D(\tau) - f_{dis} - G(x_1) - C(x_1, x_2) \alpha - M(x_1) \dot{\alpha}] \quad (2.52)$$

Ahora, sustituyendo la segunda derivada, correspondiente a la ecuación 2.48 se sustituye en la ecuación 2.52, permitiendo $D(\tau) = \tau - \Delta\tau$, $\Delta\tau$ es el error, resultando en:

$$\dot{V}_2 = - \sum_{i=1}^n \frac{k_i e_{1i}^2}{b_i^2 - e_{1i}^2} + \sum_{i=1}^n \frac{e_{1i} e_{2i}}{b_i^2 - e_{1i}^2} + e_2^T [\tau - \Delta\tau - f_{dis} - G(x_1) - C(x_1, x_2)\alpha - M(x_1)\dot{\alpha}] \quad (2.53)$$

Donde la ley de control τ corresponde a:

$$\tau = -k_2 e_2 + \hat{W}^T S(X) + \hat{W}_\tau^T S(X_\tau) \quad (2.54)$$

Sustituyendo la ecuación 2.54 en la ecuación 2.53 se obtiene:

$$\dot{V}_2 = -e_2^T K_2 e_2 - \sum_{i=1}^n \frac{k_i e_{1i}^2}{b_i^2 - e_{1i}^2} + e_2^T \left(\hat{W}^T S(X) + \hat{W}_\tau^T S(X_\tau) \right) \quad (2.55)$$

2.7. Estructura de la red neuronal

Una red neuronal se caracteriza como un procesador distribuido masivamente en paralelo, compuesto de procesamiento simple y unidades donde se almacena conocimiento en base a la experiencia, el cual se encuentra a disposición. El emplear redes neuronales implica los beneficios tales como: no linealidad, mapeo de entradas y salidas, adaptabilidad, entre otros [54].

Para el algoritmo adaptativo basado en redes neuronales artificiales es indispensable la estructura neuronal que permitirá la estimación de no linealidades del

modelo dinámico en la Figura 2.9, así como también de la zona muerta, Figura 2.10.

La estructura de las redes neuronales empleadas en este documento involucran 2 capas externas: Caracterizadas por la capa de entrada y capa de salida, así como también una capa interna dedicada a las funciones de base radial empleadas en el modelo de control [55]. Para la Figura 2.9, la capa de entrada correspondiente a la matriz Z , se encuentra en términos de los valores de posición, velocidad, el término α y su derivada $\dot{\alpha}$ del robot manipulador, por otra parte, la capa de salida se encuentra definida por el término S , término necesario para el complemento de la ley de control τ ; las correspondientes capas se encuentran interconectadas por un total de 32 nodos.

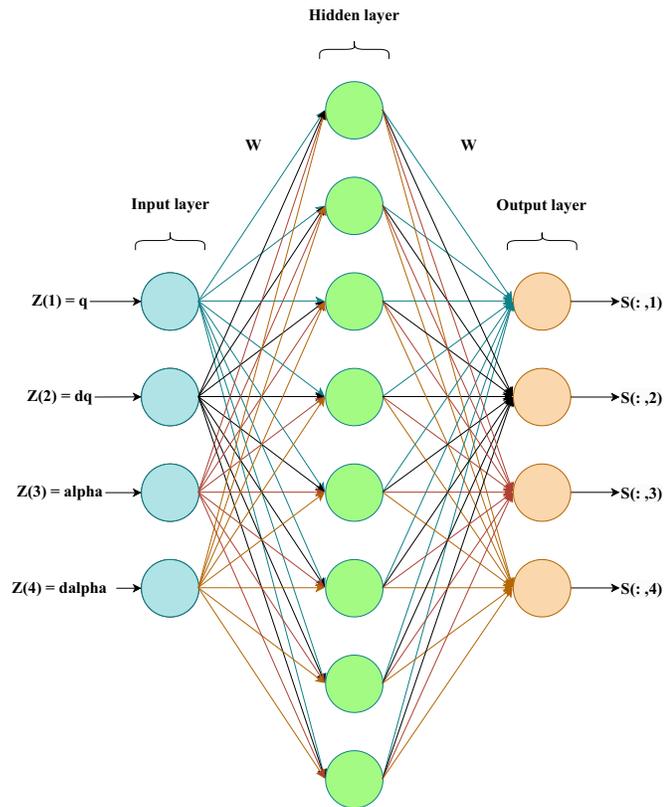


Figura 2.9: Diagrama de red neuronal artificial que estima las no linealidades del sistema.

Por otra parte, la estructura de la segunda red neuronal, es similar a la primera de acuerdo al número de nodos y a la función radial básica empleada, sin embargo,

cabe destacar que en este caso la capa de entrada Z considera un término relacionado con el torque ($\int \dot{\tau}$), el cual se obtiene de la diferencia entre el torque calculado en un instante de tiempo y la derivada del torque (previamente integrado), visualizado en la ecuación 2.56.

$$\dot{\tau} = \tau - \int \dot{\tau} \quad (2.56)$$

Elemento necesario para el término de estimación debido a la zona muerta que se presenta en el proceso de movimiento del robot mediante a partir del torque suministrado a los motores de DC.

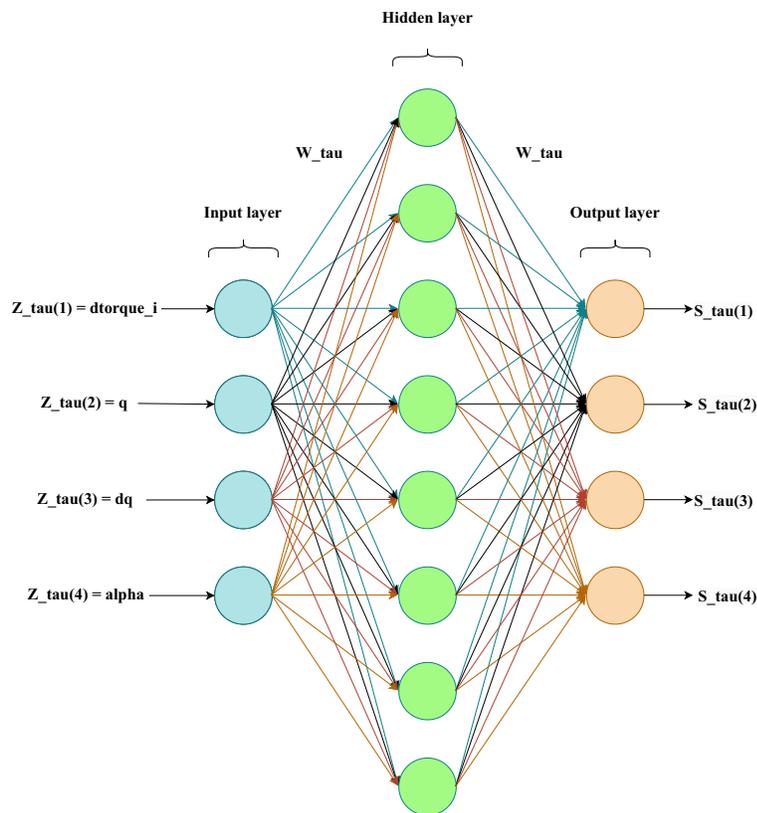


Figura 2.10: Diagrama de red neuronal artificial que estima la zona muerta.

2.8. Parámetros dinámicos

El modelo dinámico del robot descrito en la ecuación 2.33 se caracteriza por los parámetros dinámicos que incluyen: masas, distancias desde el centro de masa hasta el eje de giro, momentos de inercia con respecto al centro de cada eslabón y parámetros de fricción. Estos parámetros varían según la composición física de cada eslabón.

La fricción es un fenómeno natural presente en todos los sistemas mecánicos y se manifiesta como una fuerza que se opone al movimiento. Modelarla con precisión es difícil, lo que a menudo resulta en una falta de precisión en el estado estacionario de los sistemas. Los dos modelos de fricción más comunes son la fricción viscosa y la fricción de Coulomb [56]. Estos modelos presentan linealidad en los coeficientes de fricción. Así mismo, en la Tabla 2.6 se muestran los parámetros físicos estipulados para este robot SCARA de cuatro grados de libertad.

Masas		Momentos de inercia	
m_1	0.93 kg	pr	0.06
m_2	0.68 kg	izz2	$m_2 \frac{(a_2 * a_2 + pr^2)}{12}$
m_3	0.54 kg	izz3	$m_3 \frac{(a_3 * a_3 + pr^2)}{12}$
m_4	0.35 kg	izz4	$\frac{m_4 pr^2}{6}$
Parámetros D-H		Centros de masa	
a_2	0.2 m	Ic_2	$\frac{a_2}{2}$
a_3	0.145 m	Ic_3	$\frac{a_3}{2}$
a_4	0.05 m		
Definiciones de senos y cosenos		Constante gravitacional	
c_3	$\cos(q_3)$	g_0	$9.8 \frac{m}{s^2}$
s_3	$\sen(q_3)$		
Coeficientes de fricción			
cf_1		0.01	
cf_2		0.01	
cf_3		0.01	
cf_4		0.01	

Tabla 2.6: Notación de los parámetros del robot manipulador.

2.9. Conclusiones

En este capítulo se analizaron los métodos empleados para el modelado dinámico del robot, basado en el método de Euler–Lagrange, ya que es una manera sistemática de definir la dinámica del robot, así como también, la posibilidad de determinar las técnicas de control adecuadas para simulaciones numéricas.

Se visualizaron los conceptos de cinemática y dinámica aplicados al robot SCARA de cuatro grados de libertad, los cuales implican una relación tanto geométrica en base a la estructura del robot, así como también la utilidad de parámetros físicos relacionados con las fuerzas que actúan sobre el mismo.

Dada la expresión matricial del modelo dinámico, se definieron los términos de las matrices de inercia, coriolis, fuerza de fricción y gravedad para el diseño a partir de la geometría y características físicas del robot.

Se describió el control *backstepping*, necesario para el análisis del control basado en redes neuronales, presente en este trabajo.

Se observaron los esquemas de redes neuronales artificiales, describiendo de manera resumida sus capas correspondientes, que se emplearan en la técnica de control adaptable.

Capítulo 3

Algoritmos de Control

3.1. Introducción

El control del movimiento de los robots manipuladores es un objetivo fundamental y esto implica la necesidad de establecer una estrategia de control que pueda generar los pares o torques aplicados a las articulaciones, con el fin de posicionarlo en un punto en específico o permitirle al robot el seguimiento de una trayectoria [57].

Uno de los desafíos actuales en el control de movimiento de manipuladores implica la integración de redes neuronales artificiales, de manera que incorporando bucles de aprendizaje se complemente la ley de control estimando las no linealidades del sistema, efectos de zona muerta, entre otros [58]. Este ajuste capacita al robot manipulador para responder de manera inteligente y flexible ante los cambios que puedan surgir en las condiciones deseadas.

En este capítulo, se llevará a cabo el diseño y desarrollo de la técnica de control adaptativa basada en redes neuronales artificiales con la finalidad de poder implementar un control de movimiento para un robot manipulador SCARA de cuatro grados de libertad.

3.2. Control PID

Dentro de los controladores convencionales se contempla el controlador Proporcional Integral Derivativo (PID), debido a su simplicidad para el control de sistemas lineales. Esta técnica depende del cálculo del error, su derivada y su integral con respecto al tiempo, tal como se visualiza en la ecuación 3.1.

$$\tau(t) = -k_p \mathbf{e}(t) - k_i \int \mathbf{e}(t) dt - k_d \frac{d(\mathbf{e})}{dt}, \quad (3.1)$$

Donde:

$$\mathbf{e}(t) = \mathbf{q} - \mathbf{q}_d \quad (3.2)$$

Como se observa, cada término que contribuye al modelo se multiplica por su respectiva ganancia k_p , k_i y k_d . La suma de estas contribuciones dan resultado a la ley de control, determinando el torque τ necesario para posicionar las articulaciones o seguir una trayectoria deseada.

Una de las desventajas de este controlador para el caso de sistemas no lineales es que no contempla las variables realimentadas provenientes del sistema a controlar, sino que únicamente hace uso de los términos basados en los errores para el cálculo de la ley de control, visualizado previamente en la ecuación 3.2.

3.3. Control Adaptable convencional

Existen una serie de técnicas de control para sistemas que aprovechan sus características no lineales, para esto requieren de la generación de una matriz denominada regresor y un vector de parámetros θ [59]. Para el caso del robot manipulador SCA-

RA se genera la matriz regresor a partir de la parametrización basada en su modelo dinámico. Por lo que a partir de la ecuación 3.3, se obtiene el regresor Y .

$$Y = \begin{bmatrix} Y_{1,1} & Y_{1,2} & Y_{1,3} & Y_{1,4} & Y_{1,5} & Y_{1,6} & Y_{1,7} & Y_{1,8} & Y_{1,9} & Y_{1,10} \\ Y_{2,1} & Y_{2,2} & Y_{2,3} & Y_{2,4} & Y_{2,5} & Y_{2,6} & Y_{2,7} & Y_{2,8} & Y_{2,9} & Y_{2,10} \\ Y_{3,1} & Y_{3,2} & Y_{3,3} & Y_{3,4} & Y_{3,5} & Y_{3,6} & Y_{3,7} & Y_{3,8} & Y_{3,9} & Y_{3,10} \\ Y_{4,1} & Y_{4,2} & Y_{4,3} & Y_{4,4} & Y_{4,5} & Y_{4,6} & Y_{4,7} & Y_{4,8} & Y_{4,9} & Y_{4,10} \end{bmatrix} \quad (3.3)$$

Donde:

Regresor (Y)	
$Y_{1,1} = \ddot{q}_1 + g_0$	$Y_{3,1} = 0$
$Y_{1,2} = 0$	$Y_{3,2} = 0$
$Y_{1,3} = 0$	$Y_{3,3} = c_3 * \ddot{q}_2 + s_3 * \dot{q}_2 * \dot{q}_2$
$Y_{1,4} = 0$	$Y_{3,4} = \ddot{q}_2 + \ddot{q}_3$
$Y_{1,5} = 0$	$Y_{3,5} = c_3 * \ddot{q}_2 + s_3 * \dot{q}_2 * \dot{q}_2$
$Y_{1,6} = 0$	$Y_{3,6} = -\ddot{q}_4$
$Y_{1,7} = \dot{q}_1$	$Y_{3,7} = 0$
$Y_{1,8} = 0$	$Y_{3,8} = 0$
$Y_{1,9} = 0$	$Y_{3,9} = \dot{q}_3$
$Y_{1,10} = 0$	$Y_{3,10} = 0$
$Y_{2,1} = 0$	$Y_{4,1} = 0$
$Y_{2,2} = \ddot{q}_2$	$Y_{4,2} = 0$
$Y_{2,3} =$ $2 * c_3 * \ddot{q}_2 + c_3 * \ddot{q}_3 - s_3 * \dot{q}_3 * \dot{q}_2$ $- s_3 * (\dot{q}_2 + \dot{q}_3) * \dot{q}_3$	$Y_{4,3} = 0$
$Y_{2,4} = \ddot{q}_2 + \ddot{q}_3$	$Y_{4,4} = 0$
$Y_{2,5} =$ $2 * c_3 * \ddot{q}_2 + c_3 * \ddot{q}_3 - s_3 * \dot{q}_3 * \dot{q}_2 -$ $s_3 * (\dot{q}_2 + \dot{q}_3) * \dot{q}_3$	$Y_{4,5} = 0$
$Y_{2,6} = -\ddot{q}_4$	$Y_{4,6} = -\ddot{q}_2 - \ddot{q}_3 + \ddot{q}_4$
$Y_{2,7} = 0$	$Y_{4,7} = 0$
$Y_{2,8} = \dot{q}_2$	$Y_{4,8} = 0$
$Y_{2,9} = 0$	$Y_{4,9} = 0$
$Y_{2,10} = 0$	$Y_{4,10} = \dot{q}_4$

Tabla 3.1: Elementos del regresor Y .

Los parámetros empleados son los que se muestran en la ecuación 3.4.

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \\ \theta_8 \\ \theta_9 \\ \theta_{10} \end{bmatrix} \quad (3.4)$$

Donde los elementos θ son los siguientes:

Parámetros θ
$\theta_1 = m_1 + m_2 + m_3 + m_4$
$\theta_2 = a_2^2 * m_3 + m_2 * lc_2^2 + a_2^2 * m_4 + Izz_2$
$\theta_3 = a_2 * m_3 * lc_3$
$\theta_4 = m_3 * lc_3^2 + a_3^2 * m_4 + Izz_3 + Izz_4$
$\theta_5 = a_2 * a_3 * m_4$
$\theta_6 = Izz_4$
$\theta_7 = cf_1$
$\theta_8 = cf_2$
$\theta_9 = cf_3$
$\theta_{10} = cf_4$

Tabla 3.2: Elementos del vector de parámetros θ .

Es posible adquirir estos elementos directamente del modelo dinámico del robot manipulador, específicamente de los términos relacionados con el diseño tales como: masas, dimensiones del robot, centros de masa, momentos de inercia y estimaciones de fricción. Dicho esto, se logra obtener la ley de control mediante el producto de la matriz regresor Y y el vector de parámetros θ , visualizado en la ecuación 3.5.

$$\tau_2 = \mathbf{Y}\theta \quad (3.5)$$

La acción de parametrizar linealmente el modelo del robot SCARA permite obtener la ley de control obteniendo un comportamiento aproximado al deseado, sin embargo, para este análisis se requiere una mayor cantidad de información tal como: la posición, velocidad y aceleración, entre otras. Por ende el objetivo del modelo control adaptable es ajustar los parámetros $\hat{\theta}$ mediante la actualización de los valores de estimación (q_r y \dot{q}_r), partiendo de las variables de posición q y velocidad \dot{q} reales, como se muestra en las ecuaciones 3.6 y 3.7.

$$\dot{q}_r = \dot{q}_d - \lambda(q - q_d) \quad (3.6)$$

$$\ddot{q}_r = \ddot{q}_d - \lambda(\dot{q} - \dot{q}_d) \quad (3.7)$$

Donde \dot{q}_r se calcula a partir de las condiciones deseadas de velocidad y de los errores de posición del sistema, en conjunto con el producto de la ganancia λ . De manera similar sucede con el término de la aceleración estimada \ddot{q}_r donde en este se ve implicada la aceleración deseada del robot manipulador y de su diferencia con el error de velocidad real.

Dicho lo anterior, la parametrización previamente realizada se modifica obteniendo un regresor modificado Y_r estructurado como se observa en la ecuación 3.8.

$$Y_r(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)\theta = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{f}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) \quad (3.8)$$

Entonces, la nueva matriz del regresor modificado Y_r se visualiza en la ecuación 3.9 y sus elementos en la Tabla 3.3.

$$Y_r = \begin{bmatrix} Yr_{1,1} & Yr_{1,2} & Yr_{1,3} & Yr_{1,4} & Yr_{1,5} & Yr_{1,6} & Yr_{1,7} & Yr_{1,8} & Yr_{1,9} & Yr_{1,10} \\ Yr_{2,1} & Yr_{2,2} & Yr_{2,3} & Yr_{2,4} & Yr_{2,5} & Yr_{2,6} & Yr_{2,7} & Yr_{2,8} & Yr_{2,9} & Yr_{2,10} \\ Yr_{3,1} & Yr_{3,2} & Yr_{3,3} & Yr_{3,4} & Yr_{3,5} & Yr_{3,6} & Yr_{3,7} & Yr_{3,8} & Yr_{3,9} & Yr_{3,10} \\ Yr_{4,1} & Yr_{4,2} & Yr_{4,3} & Yr_{4,4} & Yr_{4,5} & Yr_{4,6} & Yr_{4,7} & Yr_{4,8} & Yr_{4,9} & Yr_{4,10} \end{bmatrix} \quad (3.9)$$

Donde:

Regresor modificado (Y_r)	
$Yr_{1,1} = q\ddot{i}_r + g_0$	$Yr_{3,1} = 0$
$Yr_{1,2} = 0$	$Yr_{3,2} = 0$
$Yr_{1,3} = 0$	$Yr_{3,3} = c_3 * q\ddot{2}_r + s_3 * \dot{q}_2 * q\dot{2}_r$
$Yr_{1,4} = 0$	$Yr_{3,4} = q\ddot{2}_r + q\ddot{3}_r$
$Yr_{1,5} = 0$	$Yr_{3,5} = c_3 * q\ddot{2}_r + s_3 * \dot{q}_2 * q\dot{2}_r$
$Yr_{1,6} = 0$	$Yr_{3,6} = -q\ddot{4}_r$
$Yr_{1,7} = q\dot{i}_r$	$Yr_{3,7} = 0$
$Yr_{1,8} = 0$	$Yr_{3,8} = 0$
$Yr_{1,9} = 0$	$Yr_{3,9} = q\dot{3}_r$
$Yr_{1,10} = 0$	$Yr_{3,10} = 0$
$Yr_{2,1} = 0$	$Yr_{4,1} = 0$
$Yr_{2,2} = q\ddot{2}_r$	$Yr_{4,2} = 0$
$Yr_{2,3} =$ $2 * c_3 * q\ddot{2}_r + c_3 * q\ddot{3}_r - s_3 * \dot{q}_3 * q\dot{2}_r$ $- s_3 * (\dot{q}_2 + \dot{q}_3) * q\dot{3}_r$	$Yr_{4,3} = 0$
$Yr_{2,4} = q\ddot{2}_r + q\ddot{3}_r$	$Yr_{4,4} = 0$
$Yr_{2,5} =$ $2 * c_3 * q\ddot{2}_r + c_3 * q\ddot{3}_r - s_3 * \dot{q}_3 * q\dot{2}_r -$ $s_3 * (\dot{q}_2 + \dot{q}_3) * q\dot{3}_r$	$Yr_{4,5} = 0$
$Yr_{2,6} = -q\ddot{4}_r$	$Yr_{4,6} = -q\ddot{2}_r - q\ddot{3}_r + q\ddot{4}_r$
$Yr_{2,7} = 0$	$Yr_{4,7} = 0$
$Yr_{2,8} = q\dot{2}_r$	$Yr_{4,8} = 0$
$Yr_{2,9} = 0$	$Yr_{4,9} = 0$
$Yr_{2,10} = 0$	$Yr_{4,10} = q\dot{4}_r$

Tabla 3.3: Elementos del regresor modificado Y_r .

Para obtener la ley de control de la técnica adaptable primero se especifica el término de slotine–Li denominado como s , donde se establece la diferencia entre las velocidades real y estimada, representado en la ecuación 3.10.

$$s = \dot{q} - \dot{q}_r \quad (3.10)$$

Ahora, se plantea la ley de control mediante la ecuación 3.11.

$$\tau = Y_r \hat{\theta} - k_v s \quad (3.11)$$

Donde se emplean los parámetros $\hat{\theta}$ establecidos a partir de la ley de adaptación, mismos que se actualizan con la expresión de la ecuación 3.12.

$$\dot{\hat{\theta}} = -\gamma(Y_r')s \quad (3.12)$$

Donde posteriormente se van integrando los valores de $\dot{\hat{\theta}}$ resultado de la ecuación 3.13.

$$\hat{\theta} = \hat{\theta} + T\dot{\hat{\theta}} \quad (3.13)$$

3.4. Control adaptativo basado en redes neuronales artificiales

Con anterioridad se analizó la técnica de control adaptable convencional, en la cual requiere primordialmente de la información del modelo dinámico mediante la parametrización lineal del sistema, donde previamente se analizó su matriz denominada regresor (Y_r) y de su vector de parámetros $\hat{\theta}$ donde interactuaban los términos de estimación de velocidad y aceleración. Debido a esto, se implementa la técnica de control adaptativa basado en redes neuronales artificiales la cual se describe a continuación.

Se consideran dos elementos primordiales para el modelo de control adaptativo basado en redes neuronales artificiales: La primera, es a partir de lo previsto en el modelo adaptable convencional donde se aprovechan las variables de control estimadas características de este modelo. La segunda, es la implementación de redes neuronales artificiales (ANN, del inglés: *Artificial Neural Network*) utilizadas para aproximar las no linealidades del sistema dinámico respectivo del robot manipulador SCARA para cuatro grados de libertad, adicionando mediante una segunda estructura neuronal la aproximación de la zona muerta. Con este método se pretendió una técnica de control sin la necesidad de la parametrización lineal a partir de la matriz regresor "Y" y la información en constante actualización de los parámetros θ que convencionalmente se utiliza en el algoritmo de control adaptable convencional.

Ahora bien, se sigue manteniendo la naturaleza del controlador adaptable, por lo que se lleva a cabo el cálculo de las variables de aproximación de velocidad y aceleración, mediante las ecuaciones 3.14 y 3.15.

$$\alpha_1 = \dot{q}_d - k_1(q - q_d) \quad (3.14)$$

$$\dot{\alpha}_1 = \ddot{q}_d - k_1(\dot{q} - \dot{q}_d) \quad (3.15)$$

Donde α_1 y $\dot{\alpha}_1$, corresponde a la aproximación de las variables de velocidad y aceleración del sistema. Donde k_1 se relaciona directamente con la ganancia Λ y esta a su vez se determina a partir de la ecuación 3.16.

$$\Lambda = k_1 = \frac{k_p}{k_2} = \frac{k_p}{k_d} \quad (3.16)$$

La variable que determina la relación entre la velocidad real y velocidad aproximada al modelo se define para este caso como e_2 , por lo que se obtiene la ecua-

ción 3.17.

$$e2 = \dot{q} - \alpha_1 \quad (3.17)$$

Por otra parte, de acuerdo a la red neuronal artificial de la Figura 2.9 mantiene una matriz Z en la capa externa de entrada, misma que se puede visualizar ampliamente en la ecuación 3.18.

$$Z = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \\ \dot{q}_1 & \dot{q}_2 & \dot{q}_3 & \dot{q}_4 \\ \alpha_{11} & \alpha_{12} & \alpha_{13} & \alpha_{14} \\ \dot{\alpha}_{11} & \dot{\alpha}_{12} & \dot{\alpha}_{13} & \dot{\alpha}_{14} \end{bmatrix} \quad (3.18)$$

Esta matriz se constituye a partir de cuatro parámetros mismos que se representan por la posición real, velocidad real, la velocidad aproximada y la aceleración aproximada. Del mismo modo, se cuenta con una matriz Z_τ la cual tiene la contribución del torque ($\int \dot{\tau}$), visualizado en la ecuación 3.19.

$$Z_\tau = \begin{bmatrix} \int \dot{\tau}_1 & \int \dot{\tau}_2 & \int \dot{\tau}_3 & \int \dot{\tau}_4 \\ q_1 & q_2 & q_3 & q_4 \\ \dot{q}_1 & \dot{q}_2 & \dot{q}_3 & \dot{q}_4 \\ \alpha_{11} & \alpha_{12} & \alpha_{13} & \alpha_{14} \end{bmatrix} \quad (3.19)$$

Sin embargo, las ecuaciones anteriores mantienen el aporte del diseño del control adaptable convencional, ya que estos términos dentro de las matrices Z y Z_τ son similares. Con la diferencia de la aportación del torque contribuyendo al efecto de zona muerta.

Así mismo, la parte de interés se define a partir de las funciones de base ra-

dial mediante funciones gaussianas. Estas funciones permiten simplificar el cálculo que se haría en el caso de la parametrización lineal, evitando el uso del regresor y vector de parámetros, y acoplando funciones de base radial en la red neuronal para aproximarlos.

La función de base radial de la red neuronal se presenta en la ecuación 3.20.

$$S_n(Z_n) = \exp \left[\frac{-(Z_n - \mu_n)'(Z_n - \mu_n)}{\eta_n^2} \right] \quad (3.20)$$

Del mismo modo, para la salida de la red neuronal utilizada para la estimación de zona muerta se obtiene:

$$S_{\tau n}(Z_{\tau n}) = \exp \left[\frac{-(Z_{\tau n} - \mu_{\tau n})'(Z_{\tau n} - \mu_{\tau n})}{\eta_n^2} \right] \quad (3.21)$$

Finalizando con la ley de control para este modelo mediante la ecuación 3.22.

$$\tau = k_2 e_2 + (W'S + W'_\tau S_\tau)' \quad (3.22)$$

Por lo que $W'S$ y $W'_\tau S_\tau$ corresponden a las contribuciones de la red neuronal artificial, evitando el uso de $Y\hat{\theta}$, y aportando la compensación estimando la zona muerta.

Debido a la necesidad de entrenamiento de la red neuronal artificial, se provee de la actualización de pesos W mediante las ecuaciones 3.23 y 3.24.

$$d_W = -\gamma(Se_2 + \sigma W) \quad (3.23)$$

$$d_{W_\tau} = -\gamma_\tau(S_\tau e_2 + \sigma_\tau W_\tau) \quad (3.24)$$

Capítulo 4

Resultados

4.1. Introducción

Dada la naturaleza científica se recurrió al análisis de los algoritmos para el sistema de control del robot SCARA mediante el empleo de diversas herramientas computacionales, tales como el software ©Matlab, y su complemento ©Simulink, para poder llevar a cabo simulaciones inherentes a su sistema de control bajo las diversas estrategias, de manera que se modele el comportamiento del sistema. Para la simulación se considerarán parámetros de diseño, condiciones iniciales, algunas estimaciones de fricción y señales de perturbación; para obtener un comportamiento lo más cercano a la realidad.

Mediante la experimentación con el robot manipulador SCARA de cuatro grados de libertad se llevaran a cabo implementación de un conjunto de estrategias de control tales como: control PID, control adaptable convencional y control adaptativo basado en redes neuronales artificiales. Para los cuales se propondrán pruebas de regulación y seguimiento de trayectorias.

Con lo anterior se obtendrán resultados mediante el análisis de las variables de

movimiento del robot, en el tiempo transitorio y estacionario, obteniendo los errores de posición en regulación y errores de seguimiento (del inglés: *Tracking error*) para el caso de análisis de trayectorias. Finalmente se compararán los resultados de las técnicas de control mediante la herramienta matemática "error cuadrático medio".

4.2. Simulaciones

4.2.1. Simulación del control PID

La Figura 4.1 especifica los bloques del modelo de control PID.

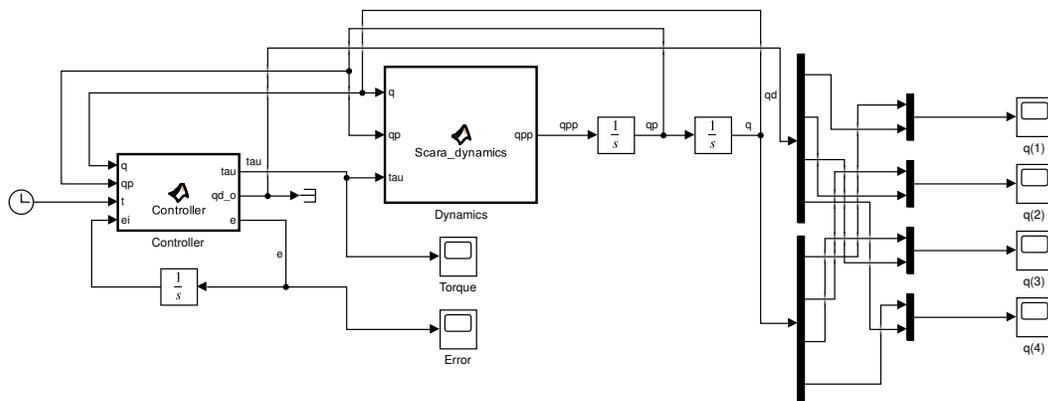


Figura 4.1: Esquema de PID en Simulink.

El modelo consta de dos bloques nombrados como "*Dynamics* y *Controller*" donde se encuentra el modelo dinámico del robot manipulador SCARA de cuatro grados de libertad y el algoritmo de control para estrategia PID, respectivamente. El bloque de la planta del sistema, cuenta con tres entradas (q , qp y τ), definidas como la posición y velocidad (de retroalimentación) y la ley de control τ obtenida por el bloque "*Controller*", determinando la aceleración (obtenida a partir del despeje del modelo dinámico del robot), resultando movimiento, mismo que es posible adecuarlo mediante una doble integración, permitiendo obtener la velocidad y posición real del

sistema. Esta información es útil para retroalimentar a los bloques "Dynamics" y para actualizar la ley de control, por lo que en el bloque "Controller" se tiene dicha información como entrada, así como también el error de integración (obtenido a partir del error de posición $q - qd$, que ingresa a un bloque integrador).

Aunado a lo anterior, el bloque "Controller" permite la salida de la ley de control, calculada a partir del modelo de control PID previamente analizado en la ecuación 3.1.

Para la sintonización del controlador PID se recurrió a la utilización de las ganancias de la Tabla 4.1.

Ganancias de sintonización	
k_p	[80, 6, 3.33, 0.1]
k_i	[4.5, 2.5, 0.4, 0.7]
k_d	[8, 2, 0.9, 0.01]

Tabla 4.1: Ganancias de sintonización para la simulación del control PID.

4.2.2. Simulación del control adaptable

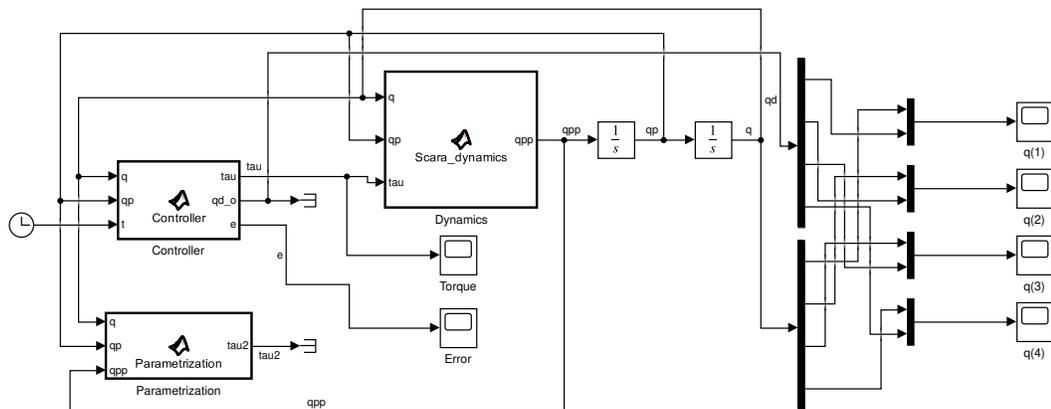


Figura 4.2: Esquema de Adaptable en Simulink.

En esta implementación se ordena el modelo de manera similar a partir de los bloques de "Controller" y "Dynamics", sin embargo, ahora, se tiene un tercer

bloque donde se lleva a cabo la parametrización lineal del modelo dinámico del robot manipulador SCARA, como se mencionó en el Capítulo 3, Sección 3.3, donde requiere de la información en términos de aceleraciones, velocidades y posiciones reales del robot.

Se genera la matriz regresor Y y mediante el vector de parámetros $\hat{\theta}$, utilizadas en el bloque "Controller" bajo la naturaleza del mismo controlador, definiendo la ley de control adaptable que será utilizado en el bloque del modelo dinámico del robot manipulador. Para la sintonización del controlador adaptable convencional se recurrió a la utilización de las ganancias de sintonización y ganancias de adaptación de la Tabla 4.2.

Ganancias de sintonización y de adaptación		
k_v	[8, 2, 0.9, 0.01]	$k_v = k_d$
λ	[2.5, 2, 5, 10]	$\lambda = \frac{k_p}{k_2}$
γ	[1, 1, 0.1, 0.1, 0.1, 0.001, 1, 1, 1, 1]	

Tabla 4.2: Ganancias de sintonización para la simulación del control adaptable.

4.2.3. Simulaciones del control adaptativo basado en redes neuronales artificiales.

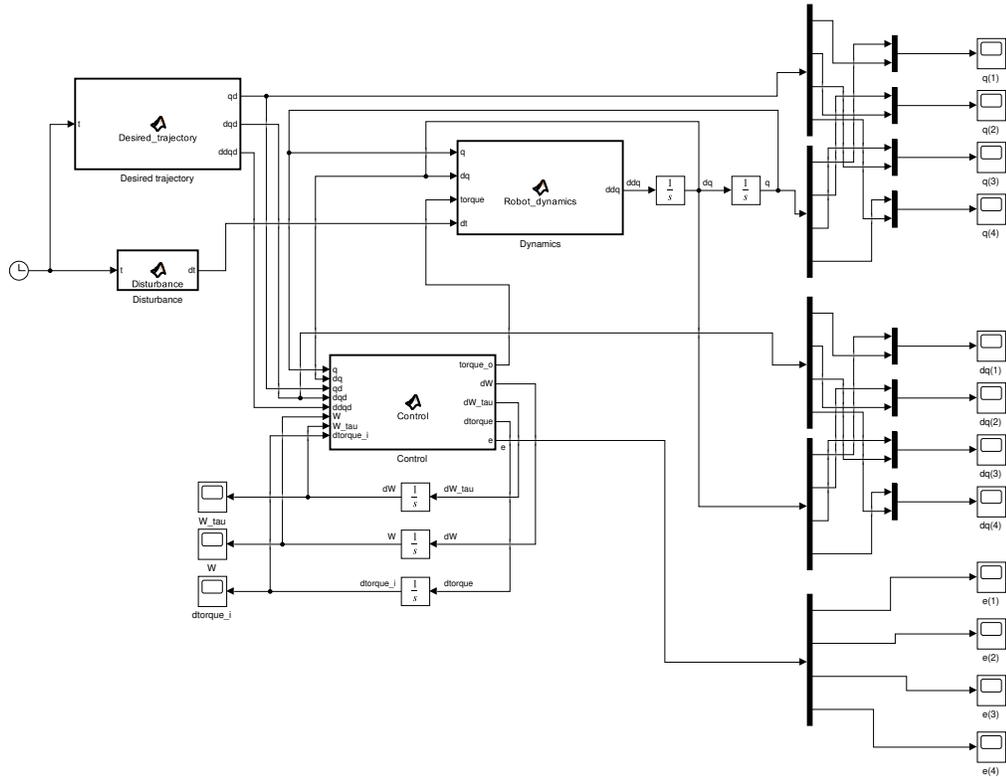


Figura 4.3: Esquema de control Adaptable basado en redes neuronales artificiales, en Simulink.

En el modelo adaptativo basado en redes neuronales artificiales tiene un conjunto de bloques en los cuales se aprecia nuevamente la etapa de "Dynamics", también el bloque "Control" que determina la ley de control τ , así como también los bloques del modelo de referencia donde se adquirirán las posiciones, velocidades y aceleraciones deseadas.

El bloque de control contiene un conjunto de variables que aportan información a la etapa adaptable y a la red neuronal del mismo. Estos parámetros son las variables reales de posición y velocidad del robot, así como la posición, velocidad y aceleración

deseadas. A su vez, se emplea W , W_τ que corresponden a los pesos de la red neuronal, en conjunto con la variable $intd_\tau$ utilizada para la red neuronal que complementa el término del torque en base a la zona muerta.

Para este modelo, se emplea el bloque de "Disturbance" para incluir la perturbación de manera que se pueda observar la característica robusta del sistema.

Para el controlador adaptativo basado en redes neuronales artificiales se recurrió a utilizar las ganancias de sintonización, de adaptación y parámetros de la red neuronal de la Tabla 4.3.

Ganancias de sintonización y parámetros ANN		
k_2	[8, 2, 0.9, 0.01]	$k_2 = k_v$
k_1	[10, 3, 3, 10]	$k_1 = \lambda$
γ	[0.55 * (ones(1,32))]	
γ_τ	[0.55 * (ones(1,32))]	
σ	0.02	
η	100	

Tabla 4.3: Ganancias de sintonización para la simulación del control adaptable mediante redes neuronales artificiales.

Como adición, es necesario expresar que para visualizar las señales de interés para cada uno de los controladores, se hizo uso de pequeños osciloscopios del mismo software ©Simulink. De la misma manera, algunos bloques de integración que permiten la retroalimentación del conjunto de variables del bloque integrador.

4.2.4. Posiciones sin perturbación

En la Figura 4.4 se observan las variables de posición y el valor de referencia para regulación, resultado del control PID.

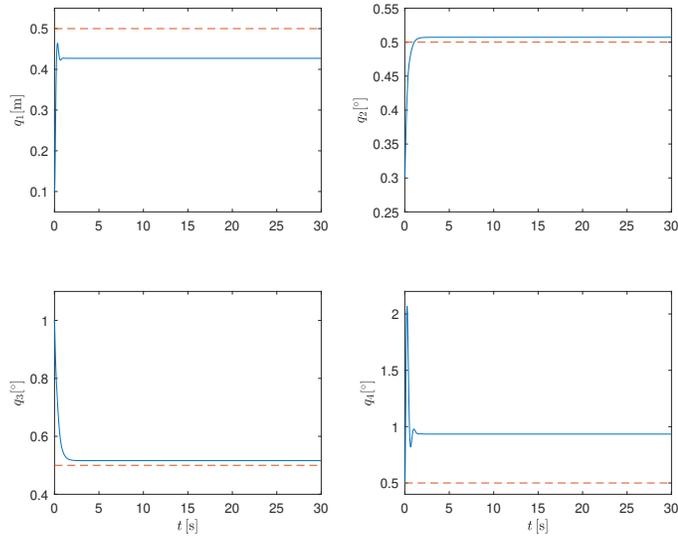


Figura 4.4: Control PID, regulación: q_d (- -), q (- - -).

En la Figura 4.5 se observan las variables de posición y el valor de referencia para seguimiento, resultado del control PID.

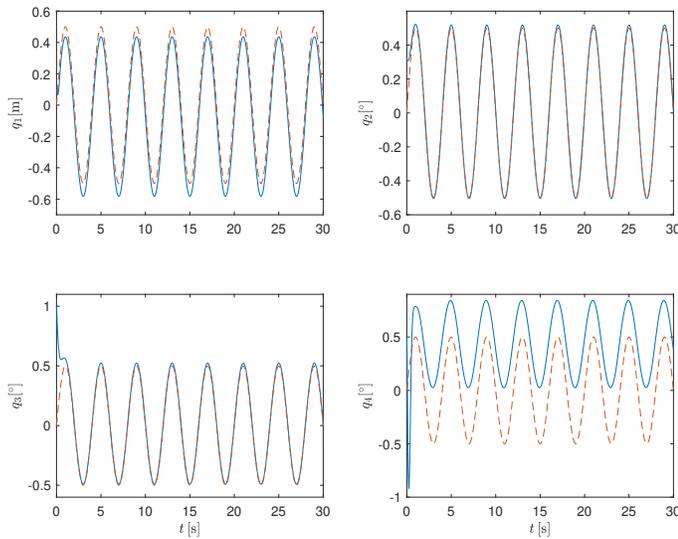


Figura 4.5: Control PID, seguimiento: q_d (- -), q (- - -).

En la Figura 4.6 se observan las variables de posición y el valor de referencia para regulación, resultado del control adaptable convencional.

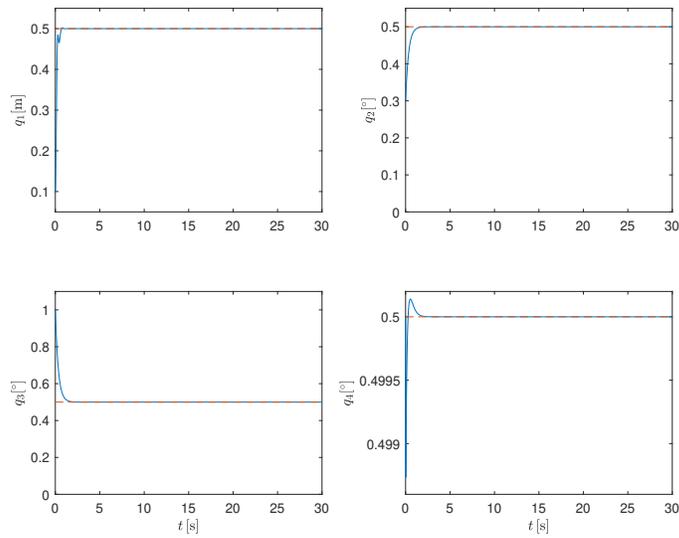


Figura 4.6: Control adaptable convencional, regulación: q_d (- -), q (- - -).

En la Figura 4.7 se observan las variables de posición y el valor de referencia para seguimiento, resultado del control adaptable convencional.

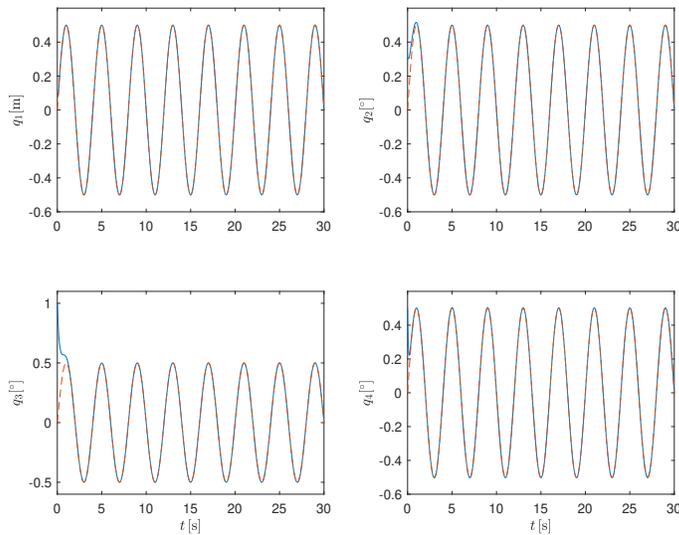


Figura 4.7: Control adaptable convencional, seguimiento: q_d (- -), q (- - -).

En la Figura 4.8 se observan las variables de posición y el valor de referencia para regulación, resultado del control adaptativo basado en redes neuronales artificiales.

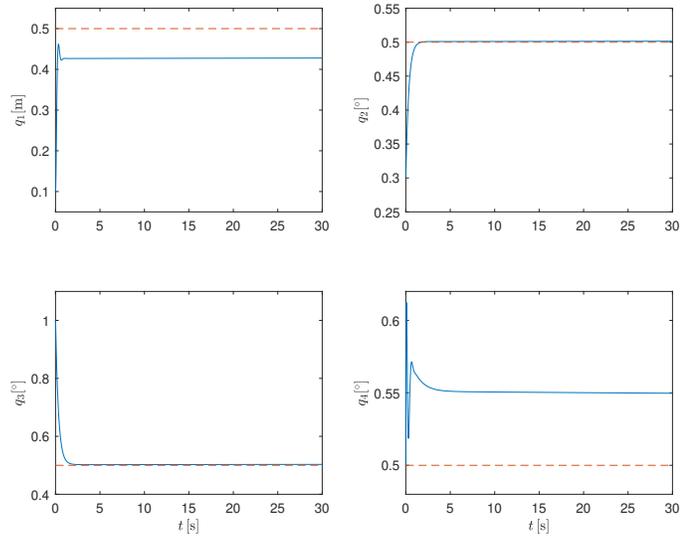


Figura 4.8: Control adaptativo basado en redes neuronales artificiales, regulación:
 q_d (---), q (— — —).

En la Figura 4.9 se observan las variables de posición y valor de referencia para seguimiento, resultado del control adaptativo basado en redes neuronales artificiales.

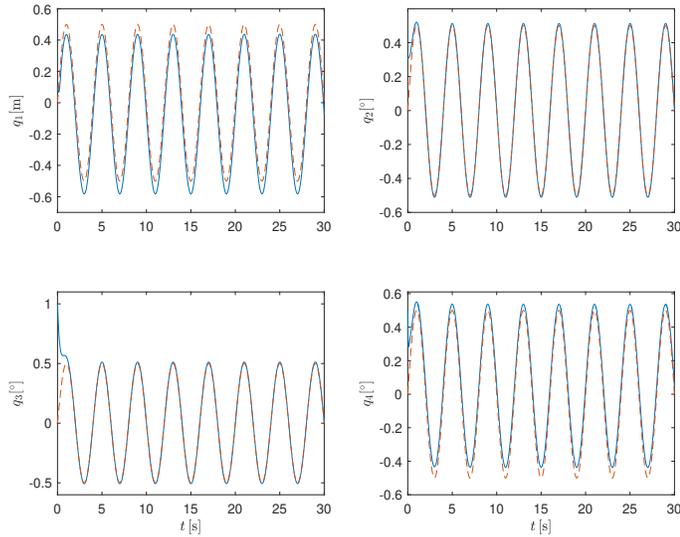


Figura 4.9: Control adaptativo basado en redes neuronales artificiales, seguimiento:
 q_d (---), q (— — —).

4.2.5. Errores sin perturbación

En la Figura 4.10 se observan los errores de posición, para regulación, resultado del control PID.

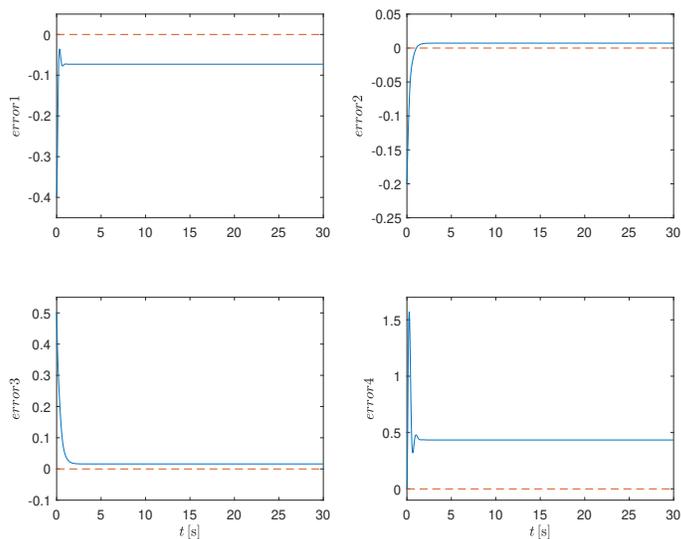


Figura 4.10: Control PID, regulación: e_d (---), e (— — —).

En la Figura 4.11 se observan los errores de posición, para seguimiento, resultado del control PID.

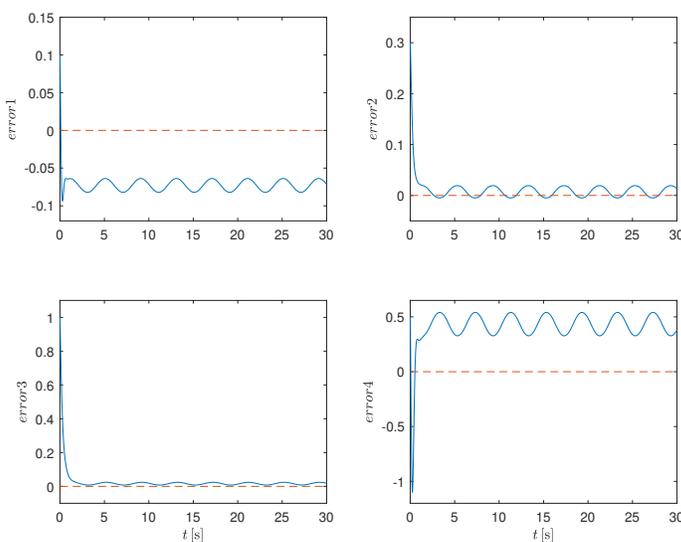


Figura 4.11: Control PID, seguimiento: e_d (---), e (— — —).

En la Figura 4.12 se observan los errores de posición, para regulación, resultado del control adaptable convencional.

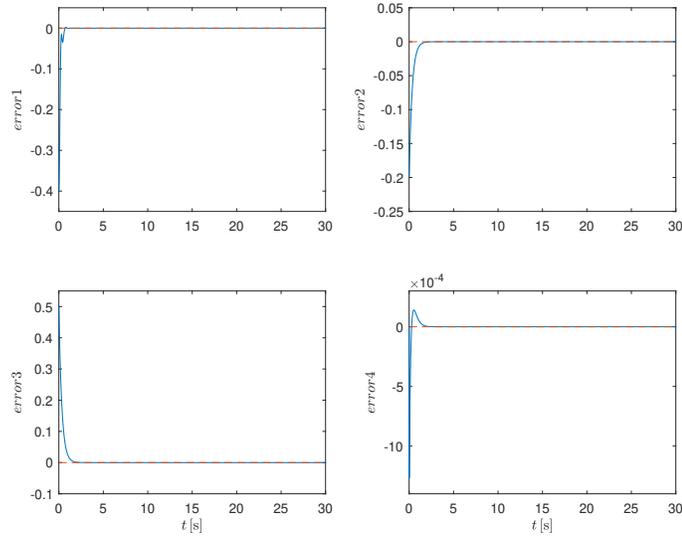


Figura 4.12: Control adaptable convencional, regulación: e_d (---), e (— — —).

En la Figura 4.13 se observan los errores de posición, para seguimiento, resultado del control adaptable convencional.

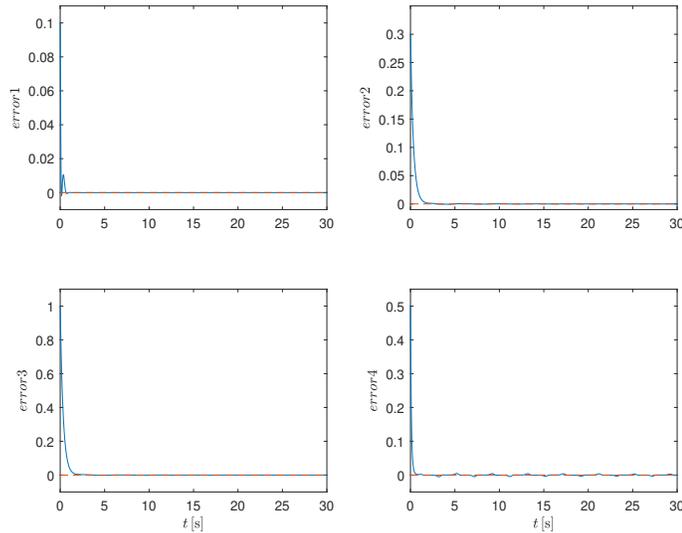


Figura 4.13: Control adaptable convencional, seguimiento: e_d (---), e (— — —).

En la Figura 4.14 se observan los errores de posición, para regulación, resultado

del control adaptativo basado en redes neuronales artificiales.

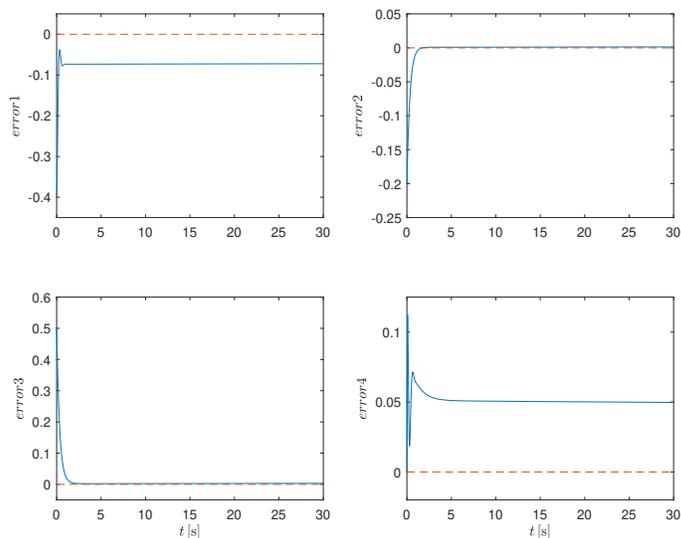


Figura 4.14: Control adaptativo basado en redes neuronales artificiales, regulación:
 e_d (---), e (— — —).

En la Figura 4.15 se observan los errores de posición, para seguimiento, resultado del control adaptativo basado en redes neuronales artificiales.

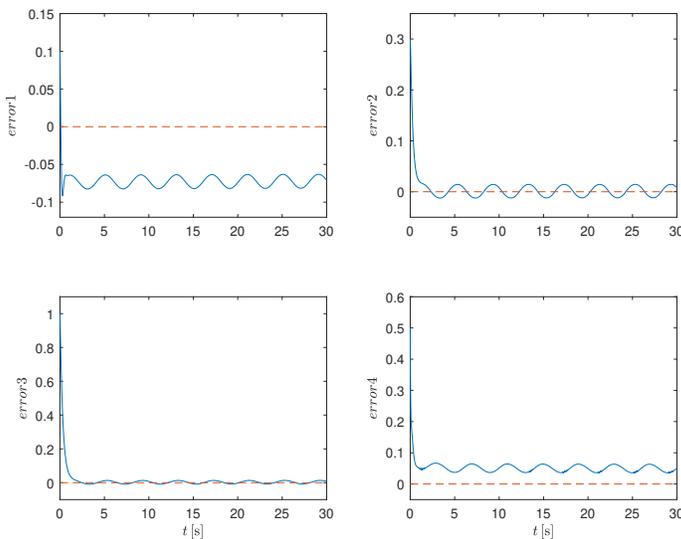


Figura 4.15: Control adaptativo basado en redes neuronales artificiales, seguimiento:
 e_d (---), e (— — —).

4.2.6. Posiciones con perturbación

En la Figura 4.16 se observan las variables de posición y el valor de referencia para regulación, resultado del control PID.

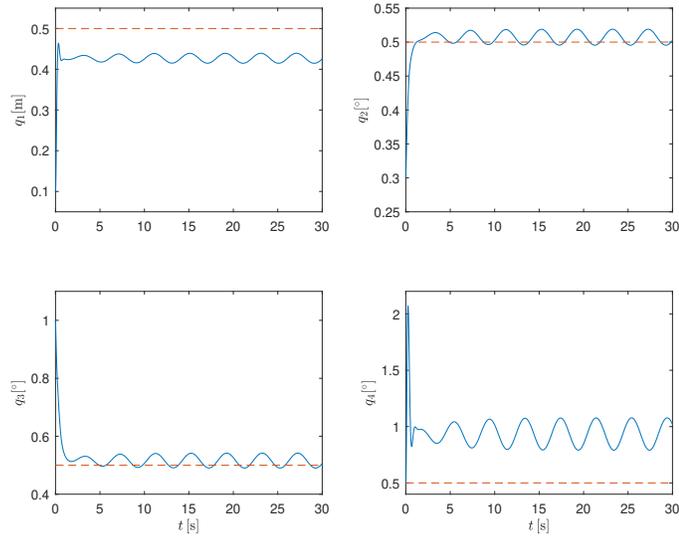


Figura 4.16: Control PID, regulación: q_d (---), q (— — —).

En la Figura 4.17 se observan las variables de posición y el valor de referencia para seguimiento, resultado del control PID.

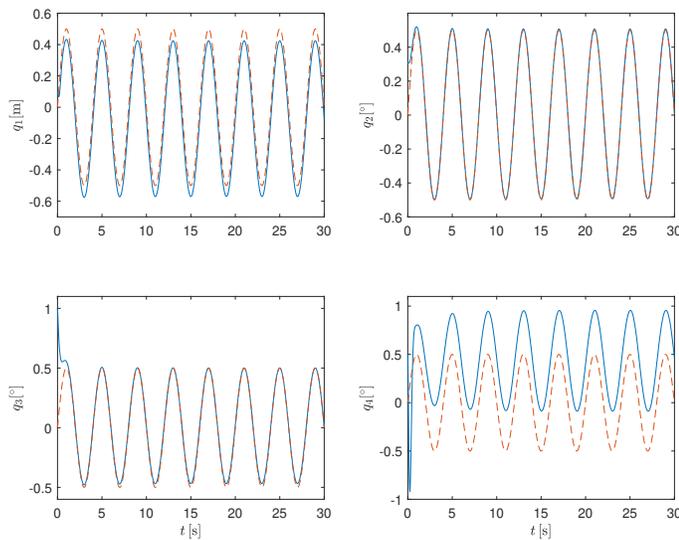


Figura 4.17: Control PID, seguimiento: q_d (---), q (— — —).

En la Figura 4.18 se observan las variables de posición y el valor de referencia para regulación, resultado del control adaptable convencional.

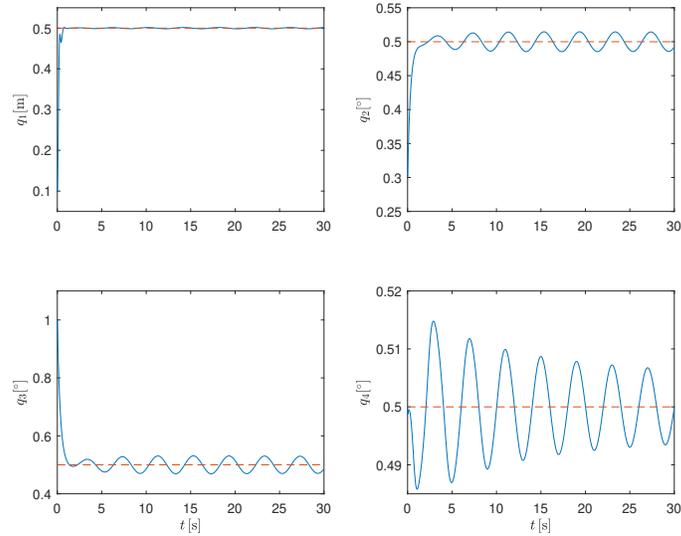


Figura 4.18: Control adaptable convencional, regulación: q_d (- - -), q (- - -).

En la Figura 4.19 se observan las variables de posición y el valor de referencia para seguimiento, resultado del control adaptable convencional.

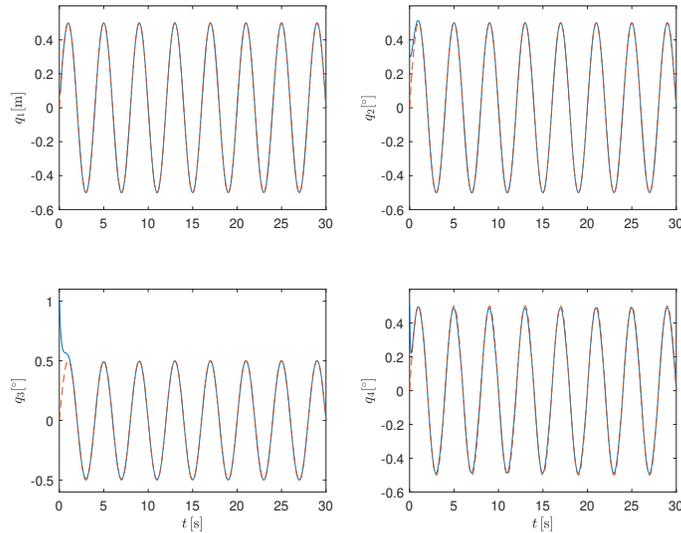


Figura 4.19: Control adaptable convencional, seguimiento: q_d (- - -), q (- - -).

En la Figura 4.20 se observan las variables de posición y el valor de referen-

cia para regulación, resultado del control adaptativo basado en redes neuronales artificiales.

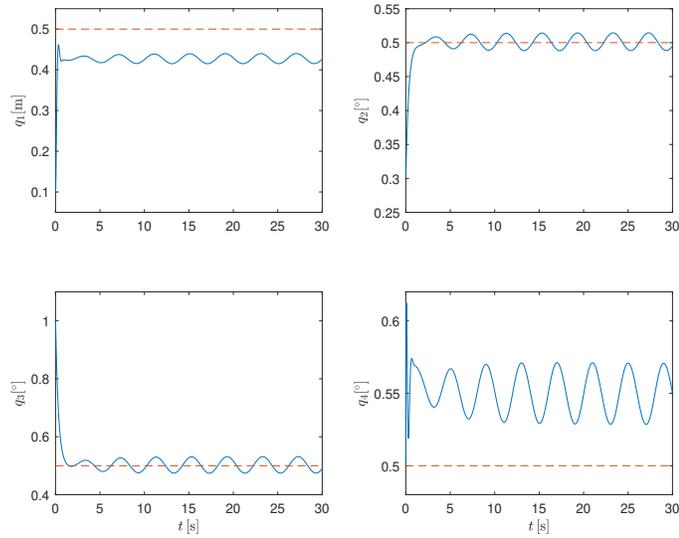


Figura 4.20: Control adaptativo basado en redes neuronales artificiales, regulación:
 q_d (---), q (— — —).

En la Figura 4.21 se observan las variables de posición y el valor de referencia para seguimiento, resultado del control adaptativo basado en redes neuronales artificiales.

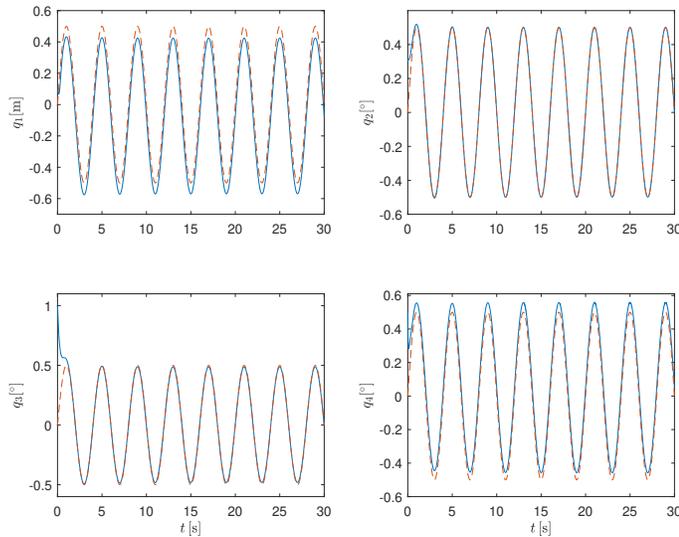


Figura 4.21: Control adaptativo basado en redes neuronales artificiales, seguimiento:
 q_d (---), q (— — —).

4.2.7. Errores con perturbación

En la Figura 4.22 se observan los errores de posición para regulación, resultado del control PID.

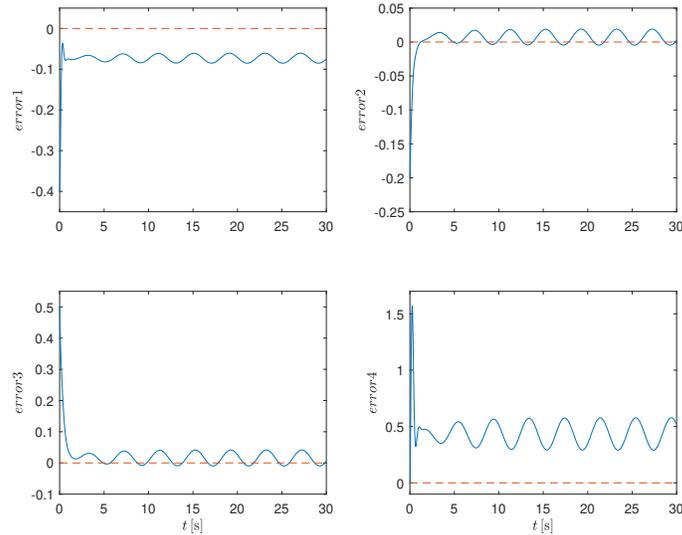


Figura 4.22: Control PID, regulación: e_d (---), e (- - -).

En la Figura 4.23 se observan los errores de posición para seguimiento, resultado del control PID.

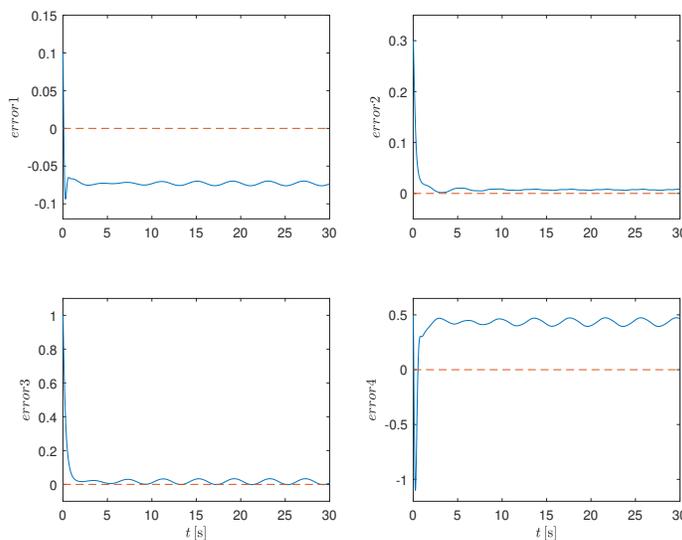


Figura 4.23: Control PID, seguimiento: e_d (---), e (- - -).

En la Figura 4.24 se observan los errores de posición para regulación, resultado del control adaptable convencional.

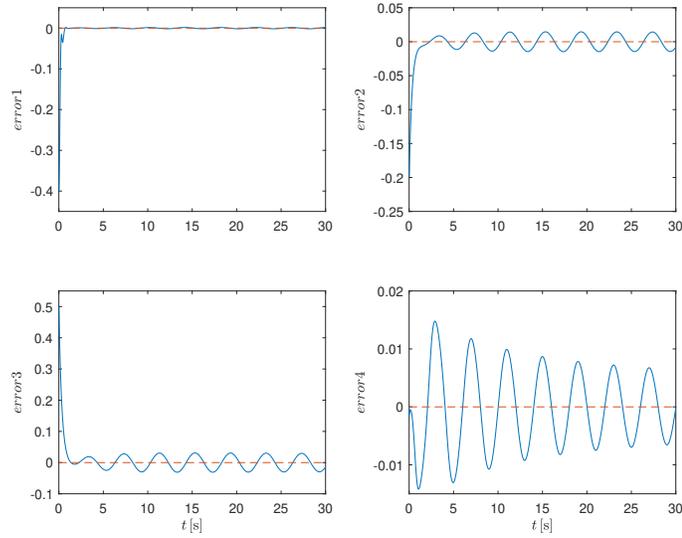


Figura 4.24: Control adaptable convencional, regulación: e_d (---), e (— — —).

En la Figura 4.25 se observan los errores de posición para seguimiento, resultado del control adaptable convencional.

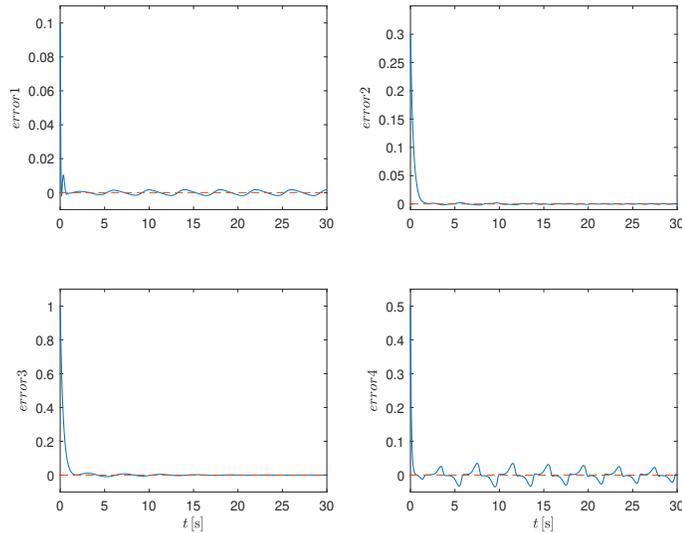


Figura 4.25: Control adaptable convencional, seguimiento: e_d (---), e (— — —).

En la Figura 4.26 se observan los errores de posición para regulación, resultado

del control adaptativo basado en redes neuronales artificiales.

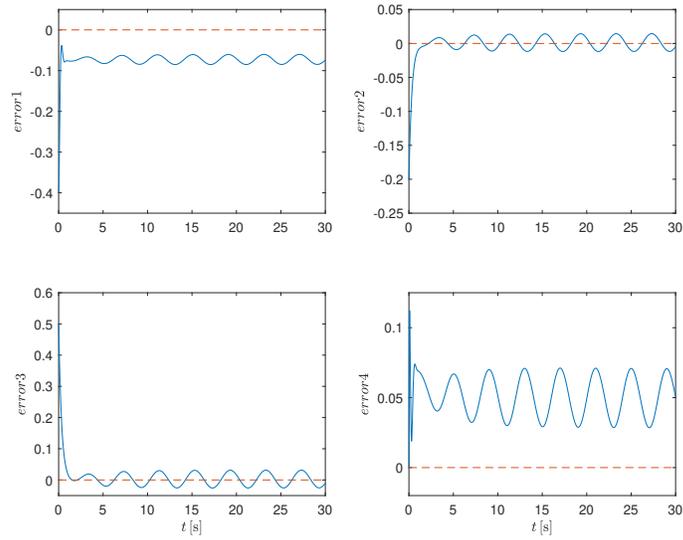


Figura 4.26: Control adaptativo basado en redes neuronales artificiales, regulación:
 e_d (---), e (—)

En la Figura 4.27 se observan los errores de posición para seguimiento, resultado del control adaptativo basado en redes neuronales artificiales.

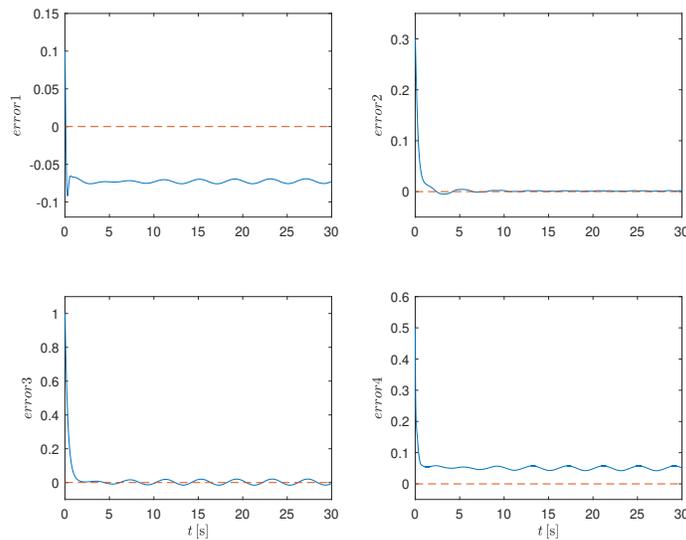


Figura 4.27: Control adaptativo basado en redes neuronales artificiales, seguimiento:
 e_d (---), e (—).

Para el análisis de regresión de los datos obtenidos de las simulaciones, se de-

terminó el error cuadrático medio (RMSE, del inglés: *Root Mean Square Error*), comparando en la Tabla 4.4 los resultados de los controladores PID, adaptable convencional y adaptativo basado en redes neuronales artificiales, bajo las pruebas de regulación y seguimiento de trayectoria.

ERROR CUADRÁTICO MEDIO							
		PID		ADAPTABLE		ANN	
		Regulación	Seguimiento	Regulación	Seguimiento	Regulación	Seguimiento
Sin perturbación	q1	0.0763m	0.0729m	0.0245m	0.0044m	0.0762m	0.0728m
	q2	0.0159°	0.0252°	0.0157°	0.0239°	0.0152°	0.0250°
	q3	0.0439°	0.0769°	0.0392°	0.0793°	0.0384°	0.0771°
	q4	0.4531°	0.4411°	0.00005395°	0.0.0245°	0.0514°	0.0567°
con perturbación	q1	0.0770m	0.0729m	0.0245m	0.0045m	0.0769m	0.0728m
	q2	0.0175°	0.0236°	0.0184°	0.0239°	0.0174°	0.0232°
	q3	0.0466°	0.0773°	0.0439°	0.0793°	0.0424°	0.0774°
	q4	0.4647°	0.4378°	0.0069°	0.0283°	0.0535°	0.0566°

Tabla 4.4: Análisis del RMSE de los datos de simulación para cada control implementado.

Aunado a este trabajo, se realizaron experimentos suministrando señales de perturbación de estructura sinusoidal, de manera que se analizará el comportamiento bajo esta condición obteniendo una respuesta de carácter robusto, necesario para el sistema del robot manipulador.

4.3. Experimentos

Para esta etapa, se desarrolló una aplicación mediante ©Microsoft Visual Studio, y mediante el protocolo de comunicación de interfaz de periféricos en serie (SPI, del inglés: *Serial Peripheral Interface*), se permite la interacción con el robot manipulador SCARA, tanto para proveer información de sus variables articulares, así como para enviarle los torques mediante las técnicas de control estudiadas, que permiten el posicionamiento del robot.

La aplicación permite al usuario interactuar con el brazo robótico de manera que se pueda seleccionar el algoritmo de control que se utilizará, así como el tipo de movimiento: regulación y/o seguimiento de trayectoria; y condiciones. La aplicación se puede visualizar con mayor detalle en el anexo C.

Aunado a lo anterior, la etapa de electrónica se diseñó a partir de la necesidad de un protocolo de comunicación para el control de los motores de DC que serán los encargados de posicionar al robot manipulador, así como la obtención de la información de las posiciones. Se empleó el protocolo de comunicación SPI mediante un módulo maestro y cuatro esclavos (de acuerdo al número de motores / articulaciones). El esquema del protocolo de comunicación y las conexiones del robot manipulador SCARA se visualizan con mayor detalle en el anexo B.

Entonces, se implementaron los algoritmos de control en el entorno ©Visual Studio, y como es de interés se trabajaron las técnicas PID, adaptable convencional, y adaptativo basado en redes neuronales artificiales, analizando las trayectorias y errores de cada articulación y obteniendo sus respectivas gráficas de acuerdo a las especificaciones de cada ejercicio. Las ganancias que se utilizaron para la experimentación son las siguientes: Para el controlador PID son las visualizadas en la Figura 4.5.

Ganancias de sintonización			
Regulación		Seguimiento	
k_p	[225, 25, 100, 100]	k_p	[400, 25, 121, 100]
k_i	[10.2, 10.3, 10.6, 10.3]	k_i	[0.3, 0.3, 1.6, 1.3]
k_d	[15, 5, 10, 10]	k_d	[20, 5, 11, 10]

Tabla 4.5: Ganancias de sintonización para la experimentación con el control PID.

Continuando con las ganancias y parámetros de adaptación del controlador adaptable convencional son los visualizados en la Figura 4.6.

Ganancias de sintonización y de adaptación		
Regulación		
k_v	[15, 5, 10, 10]	$k_v = k_d$
λ	[15, 5, 10, 10]	$\lambda = \frac{k_p}{k_2}$
γ	[0.00065, 0.00001, 0.00001, 0.00001, 0.00001, 0.0, 0.00001, 0.1, 0.1, 0.01]	
Seguimiento		
k_v	[20, 5, 11, 10]	$k_v = k_d$
λ	[20, 5, 11, 10]	$\lambda = \frac{k_p}{k_2}$
γ	[0.002, 0.00001, 0.00001, 0.00001, 0.00001, 0.0, 0.00001, 0.1, 0.1, 0.1]	

Tabla 4.6: Ganancias y parámetros de adaptación para la experimentación con el control adaptable convencional.

Finalmente, se establecen las ganancias y parámetros de adaptación para el controlador adaptativo basado en redes neuronales artificiales en la Figura 4.7.

Ganancias de sintonización y parámetros ANN		
k_2	[2, 1, 0.2, 0.01]	$k_2 = k_v$
k_1	[4.5, 2.5, 0.4, 0.7]	$k_1 = \lambda$
γ	[0.55 * (ones(1,32))]	
γ_τ	[0.55 * (ones(1,32))]	
σ	0.02	
η	100	

Tabla 4.7: Ganancias y parámetros de adaptación para la experimentación con el control adaptativo basado en redes neuronales artificiales.

4.3.1. Posiciones sin compensador de fricción

En la Figura 4.28, se observan las variables de posición y el valor de referencia para regulación, resultado del control PID.

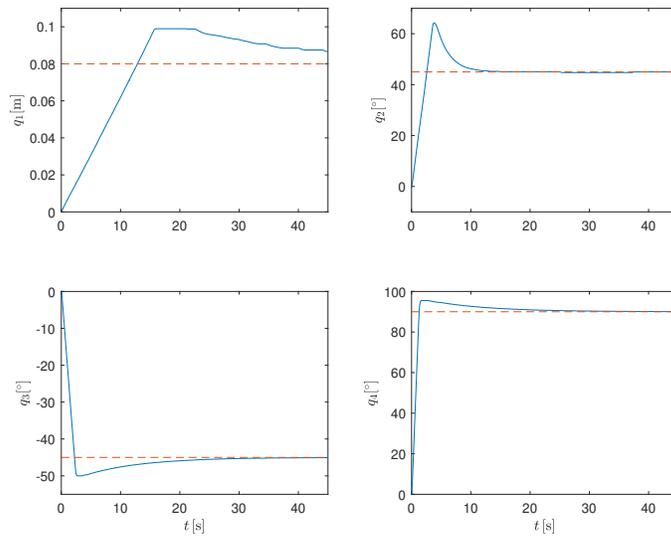


Figura 4.28: Control PID, regulación: q_d (---), q (—).

En la Figura 4.29, se observan las variables de posición y la trayectoria para seguimiento, resultado del control PID.

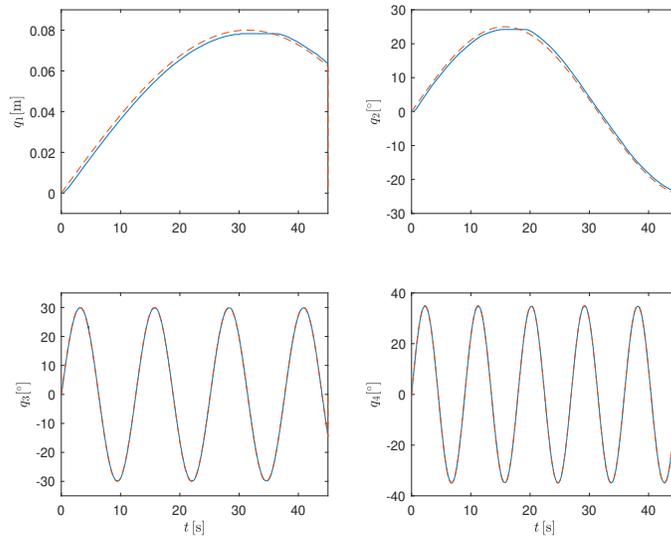


Figura 4.29: Control PID, seguimiento: q_d (---), q (—).

En la Figura 4.30, se observan las variables de posición y el valor de referencia para regulación, resultado del control adaptable convencional.

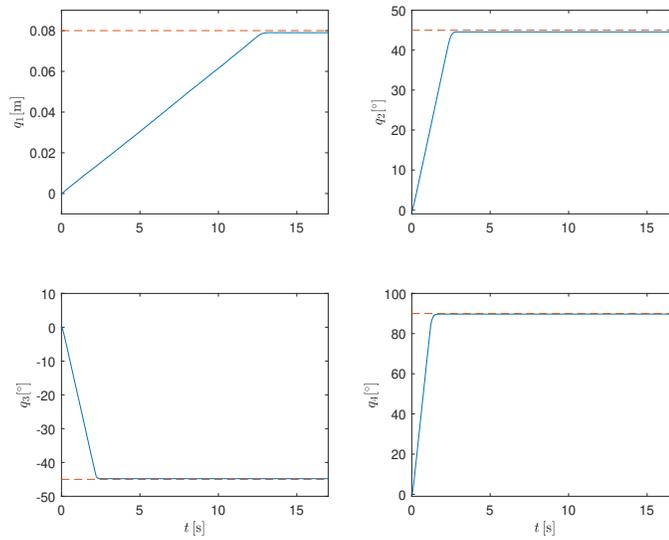


Figura 4.30: Control adaptable convencional, regulación: q_d (---), q (— — —).

En la Figura 4.31, se observan las variables de posición y la trayectoria para seguimiento, resultado del control adaptable convencional.

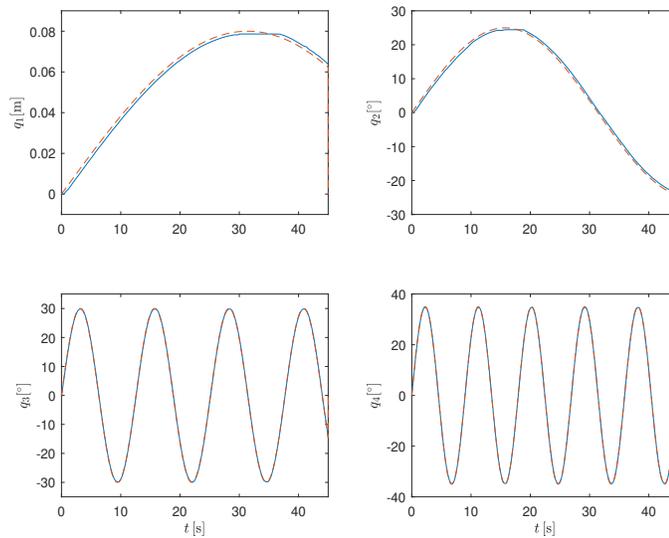


Figura 4.31: Control adaptable convencional, seguimiento: q_d (---), q (— — —).

En la Figura 4.32, se observan las variables de posición y el valor de referencia para regulación, resultado del control adaptativo basado en redes neuronales artificiales.

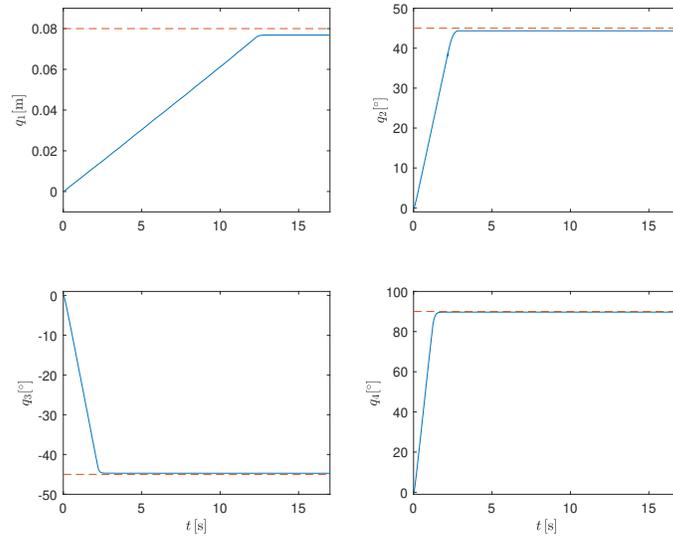


Figura 4.32: Control adaptativo basado en redes neuronales artificiales, regulación:
 q_d (---), q (—).

En la Figura 4.33, se observan las variables de posición y la trayectoria para seguimiento, resultado del control adaptativo basado en redes neuronales artificiales.

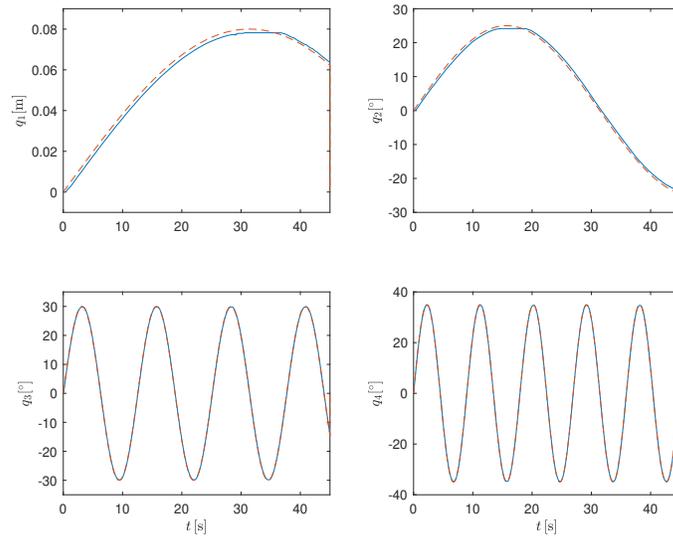


Figura 4.33: Control adaptativo basado en redes neuronales artificiales, seguimiento:
 q_d (---), q (—).

4.3.2. Errores sin compensador de fricción

En la Figura 4.34, se observan los errores de posición, para regulación, resultado del control PID.

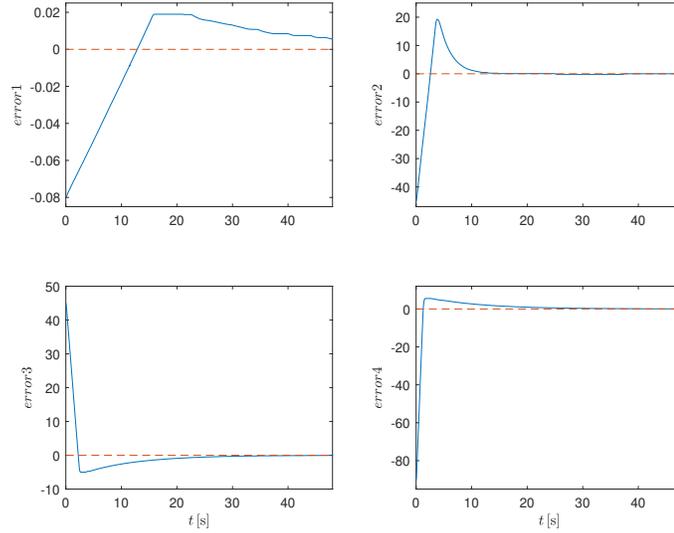


Figura 4.34: Control PID, regulación: e_d (---), e (- - -).

En la Figura 4.35, se observan los errores de posición, para seguimiento, resultado del control PID.

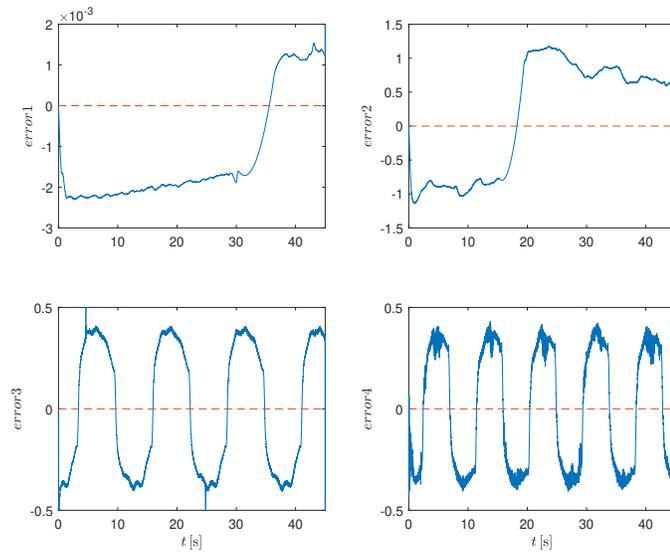


Figura 4.35: Control PID, seguimiento: e_d (---), e (- - -).

En la Figura 4.36, se observan los errores de posición, para regulación, resultado del control adaptable convencional.

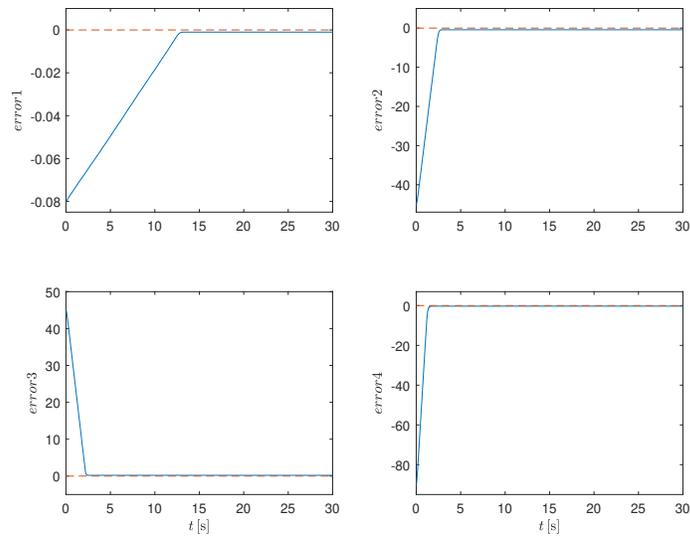


Figura 4.36: Control adaptable convencional, regulación: e_d (---), e (— — —).

En la Figura 4.37, se observan los errores de posición, para seguimiento, resultado del control adaptable convencional.

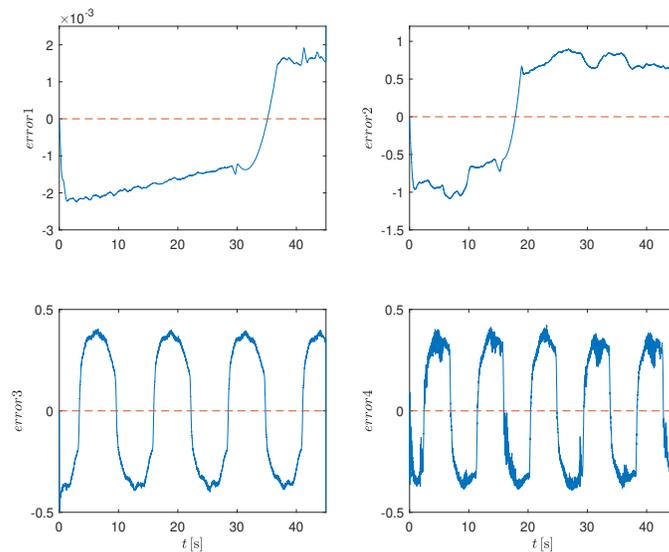


Figura 4.37: Control adaptable convencional, seguimiento: e_d (---), e (— — —).

En la Figura 4.38, se observan los errores de posición, para regulación, resultado

del control adaptativo basado en redes neuronales artificiales.

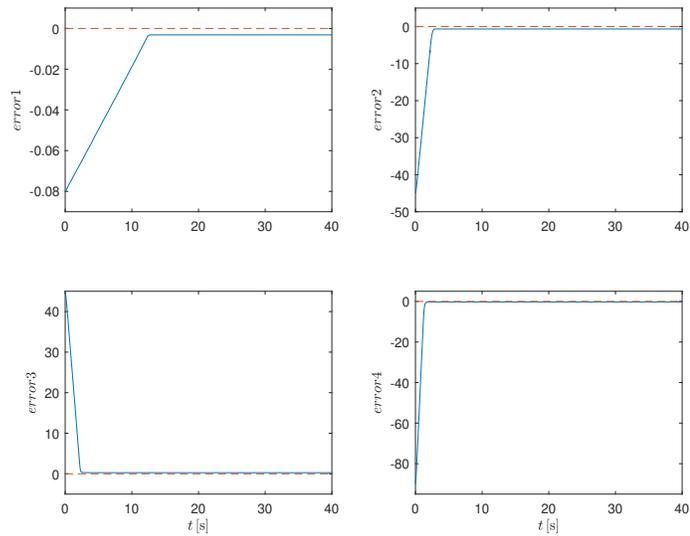


Figura 4.38: Control adaptativo basado en redes neuronales artificiales, regulación:
 e_d (—), e (- - -).

En la Figura 4.39, se observan los errores de posición, para seguimiento, resultado del control adaptativo basado en redes neuronales artificiales.

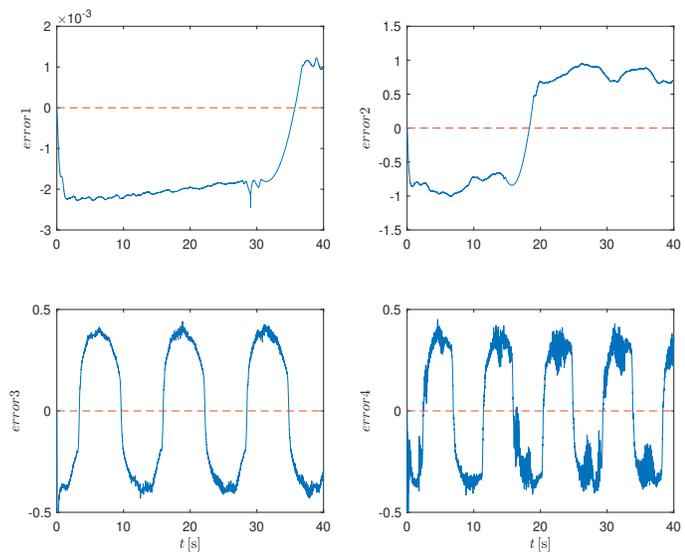


Figura 4.39: Control adaptativo basado en redes neuronales artificiales, seguimiento:
 e_d (—), e (- - -).

4.3.3. Posiciones con compensador de fricción

En la Figura 4.40, se observan las variables de posición y el valor de referencia para regulación, resultado del control PID.

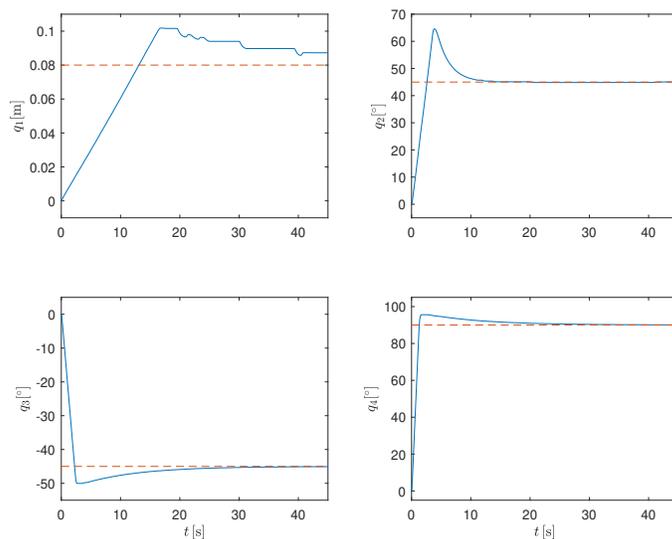


Figura 4.40: Control PID, regulación: q_d (- -), q (- - -).

En la Figura 4.41, se observan las variables de posición y la trayectoria para seguimiento, resultado del control PID.

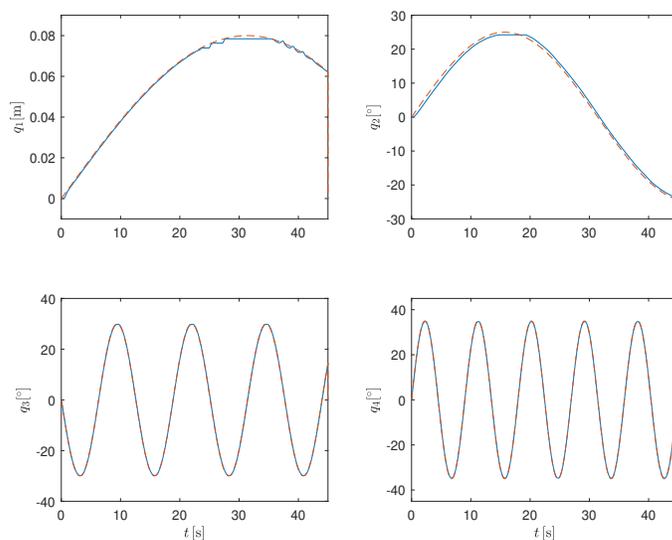


Figura 4.41: Control PID, seguimiento: q_d (- -), q (- - -).

En la Figura 4.42, se observan las variables de posición y el valor de referencia para regulación, resultado del control adaptable convencional.

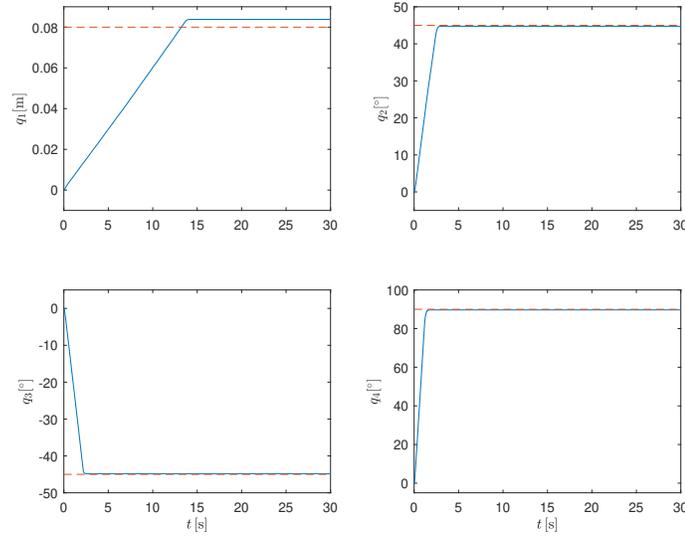


Figura 4.42: Control adaptable convencional, regulación: q_d (---), q (— — —).

En la Figura 4.43, se observan las variables de posición y la trayectoria para seguimiento, resultado del control adaptable convencional.

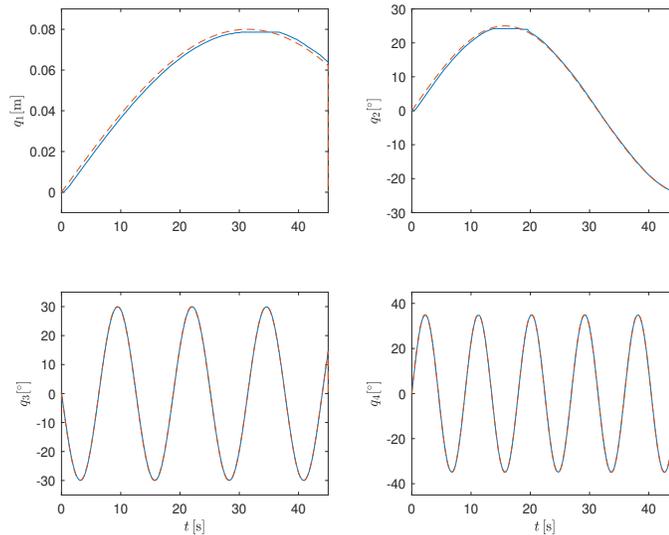


Figura 4.43: Control adaptable convencional, seguimiento: q_d (---), q (— — —).

En la Figura 4.44, se observan las variables de posición y el valor de refe-

rencia para regulación, resultado del control adaptativo basado en redes neuronales artificiales.

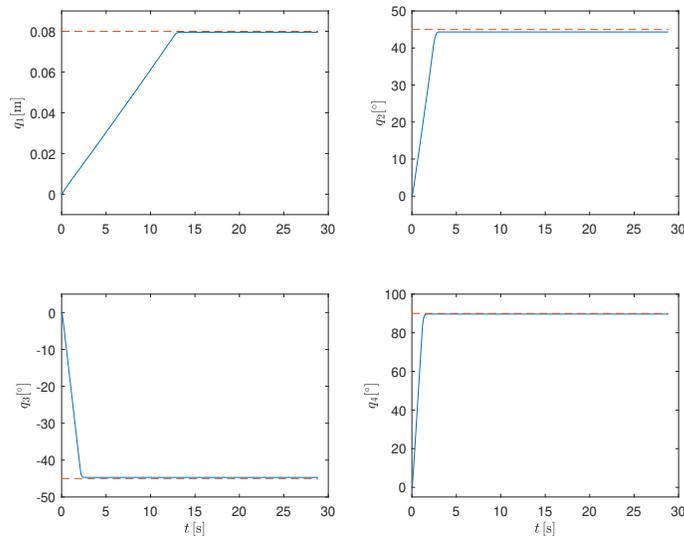


Figura 4.44: Control adaptativo basado en redes neuronales artificiales, regulación: q_d (---), q (—).

En la Figura 4.45, se observan las variables de posición y la trayectoria para seguimiento, resultado del control adaptativo basado en redes neuronales artificiales.

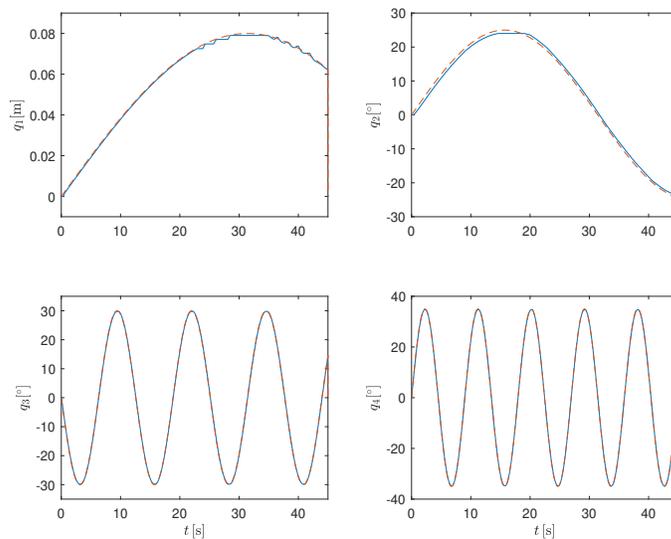


Figura 4.45: Control adaptativo basado en redes neuronales artificiales, seguimiento: q_d (---), q (—).

4.3.4. Errores con compensador de fricción

En la Figura 4.46, se observan los errores de posición, para regulación, resultado del control PID.

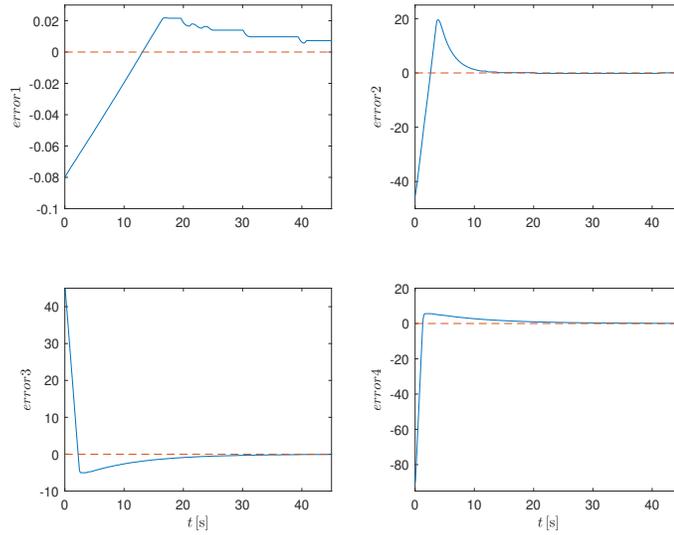


Figura 4.46: Control PID, regulación: e_d (- - -), e (- - - -).

En la Figura 4.47, se observan los errores de posición, para seguimiento, resultado del control PID.

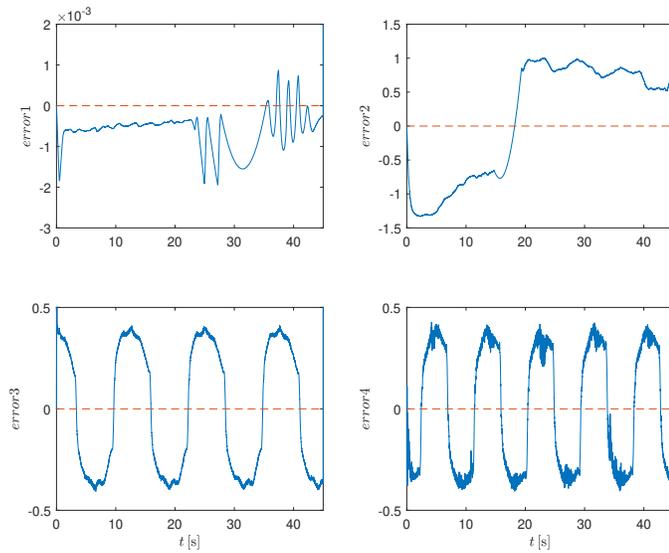


Figura 4.47: Control PID, seguimiento: e_d (- - -), e (- - - -).

En la Figura 4.48, se observan los errores de posición, para regulación, resultado del control adaptable convencional.

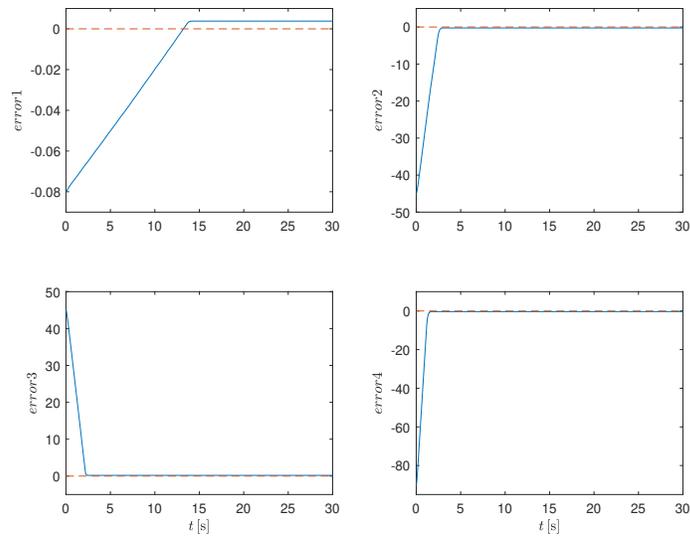


Figura 4.48: Control adaptable convencional, regulación: e_d (---), e (- - -).

En la Figura 4.49, se observan los errores de posición para seguimiento, resultado del control adaptable convencional.

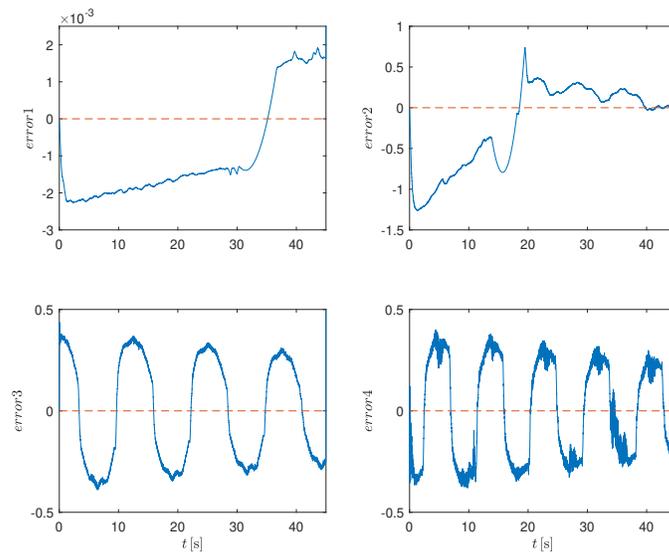


Figura 4.49: Control adaptable convencional, seguimiento: e_d (---), e (- - -).

En la Figura 4.50, se observan los errores de posición, para regulación, resultado

del control adaptativo basado en redes neuronales artificiales.

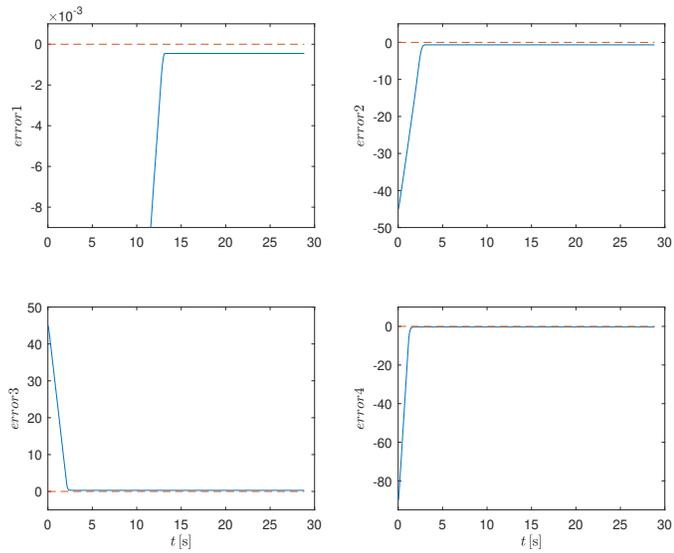


Figura 4.50: Control adaptativo basado en redes neuronales artificiales, regulación:
 e_d (---), e (— — —).

En la Figura 4.51, se observan los errores de posición, para seguimiento, resultado del control adaptativo basado en redes neuronales artificiales.

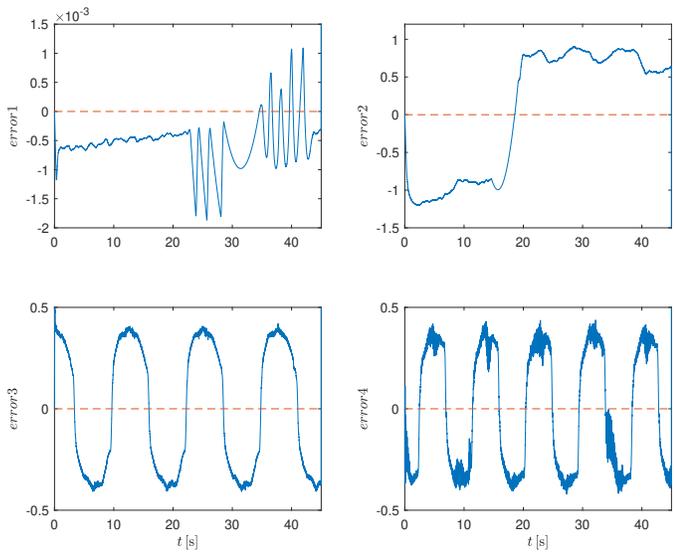


Figura 4.51: Control adaptativo basado en redes neuronales artificiales, seguimiento:
 e_d (---), e (— — —).

4.4. Análisis de regresión

Para el análisis de los datos obtenidos a partir de la experimentación se determinó el RMSE, comparando los datos de la Tabla 4.8 los resultados de las experimentaciones de los controladores PID, adaptable convencional y adaptativo basado en redes neuronales artificiales, bajo las pruebas de regulación y seguimiento de trayectoria.

ERROR CUADRÁTICO MEDIO							
		PID		ADAPTABLE		ANN	
		Regulación	Seguimiento	Regulación	Seguimiento	Regulación	Seguimiento
Sin compensador de fricción	q1	0.0306m	0.0138m	0.0331m	0.0135m	0.0293m	0.0140m
	q2	8.7847°	2.3934°	8.9775°	2.3169°	8.0271°	2.4133°
	q3	7.4336°	1.1249°	8.5510°	1.1008°	7.7186°	1.1438°
	q4	11.2250°	0.3117°	13.1281°	0.3088°	11.9102°	0.3009°
Con compensador de fricción	q1	0.0313m	0.0126m	0.0294m	0.0123m	0.0312m	0.0114m
	q2	8.9349°	2.2708°	7.3295°	2.1064°	8.2266°	2.0723°
	q3	7.4815°	1.0728°	6.9286°	0.9932°	7.4104°	0.9772°
	q4	11.3197°	0.3159°	10.5565°	0.2378°	11.3621°	0.2826°

Tabla 4.8: Análisis de RMSE de los datos de experimentación para cada control implementado.

Así mismo, se realizaron experimentos bajo el criterio de compensación de fricción seca que tiene presencia en el robot manipulador real, se consideró debido a la necesidad de mejorar la señal de salida del controlador τ para lograr las posiciones y/o trayectorias deseadas con una mejor evaluación para la mayoría de los experimentos, especialmente en el controlador adaptativo basado en redes neuronales artificiales de acuerdo a la Tabla 4.8, misma que afecta a la variable prismática, ya que se encuentra en contacto con el tornillo sin fin.

Capítulo 5

Conclusiones y observaciones

5.1. Conclusiones generales

A partir del análisis mediante gráficas y tablas comparativas del error cuadrático medio se analizaron las diversas pruebas y se observó que:

Para los casos de simulación:

Se contemplaron las pruebas de regulación y seguimiento a partir de condiciones sin perturbación externa, obteniendo que la técnica de control adaptativa basado en redes neuronales artificiales tiene una respuesta favorable con respecto al control PID y adaptable convencional para la mayoría de las articulaciones del robot manipulador SCARA, exceptuando la variable articular q_3 y q_4 en las pruebas de seguimiento de trayectoria, donde el desempeño es ligeramente menor en comparación con las demás técnicas de control presentadas, sin embargo, es comparable el nivel de error. Así mismo, se agrega que en las articulaciones q_2 y q_3 de las pruebas de regulación, no solo se obtuvo un mejor desempeño en comparación al control PID, sino que fue notoria la mejora en comparación a la técnica de control adaptable convencional. Por ende, la técnica de control adaptativa basado en redes neuronales artificiales

mejoró el nivel de error en estado transitorio y estacionario de la señal de posición del robot. Esto se debe a la sintonización de las ganancias γ relacionadas con la ley de adaptación y actualización de parámetros $\hat{\theta}$.

Aunado a los casos de simulación, también se analizaron los casos agregando perturbaciones externas en términos de señales senoidales para las tres técnicas de control analizadas en esta Tesis, fue notorio el incremento del error en el desempeño de las tres técnicas de control, sin embargo, para la técnica de control adaptativa basado en redes neuronales artificiales mantuvo una mayor robustez ajustando la red neuronal para mantener la señal de referencia con un nivel de error menor que el control PID y adaptable convencional.

Para los casos experimentales:

Se realizaron experimentos de regulación y seguimiento de trayectoria, obteniendo resultados de regulación donde se comparó la respuesta del control PID con respecto al adaptativo basado en redes neuronales artificiales, se obtuvo que para regulación, las articulaciones q_3 y q_4 el control PID tuvo un mejor rendimiento; así mismo, en el caso de seguimiento de trayectorias, la mayoría de las articulaciones obtuvieron un RMSE alto para la técnica de control adaptativa basado en redes neuronales.

De acuerdo a estas pruebas se identificó la presencia de fricción seca suministrada por el contacto de elementos metálicos en el brazo robótico experimental, por lo que se recurrió al ajuste de los modelos de control compensando dicha característica. Entonces, se realizaron las pruebas pertinentes de regulación y seguimiento de trayectoria.

Para los casos de regulación, el nivel de error disminuyó considerablemente en la técnica de control adaptativa basado en redes neuronales artificiales. Esto brindando una mejor comparativa con respecto a las otras dos estrategias de control. Sin embargo, donde fue más notoria la mejora es en los casos donde se requería

seguimiento de trayectorias, analizando que la variable articular q_4 (de la técnica de control adaptativa basado en redes neuronales), obtuvo un mejor desempeño que en el caso del control PID, pero ligeramente menor en comparación al control adaptable convencional. Para el resto de variables articulares (q_1 , q_2 y q_3), el nivel de error era menor tanto en comparación con el control PID, así como también con el control adaptable convencional.

Si bien, se observó que en comparativa del control adaptativo basado en redes neuronales artificiales y el control adaptable convencional, el rendimiento en algunas variables no es totalmente mejor, sin embargo, cabe destacar que el modelo basado en inteligencia computacional nos permite aproximar los parámetros del modelo, situación que en el control adaptable convencional ya se cuenta con ellos y de mayor información especificada.

Entonces la técnica de control adaptativa basado en redes neuronales artificiales nos brinda un acercamiento de dichos parámetros del robot manipulador para poder aproximar y mejorar el rendimiento del sistema en comparación a otras técnicas de control lineales y no lineales que requieran de mayor información.

5.2. Trabajo a futuro

- Propuesta de algoritmo de control que no requiera del modelo dinámico y parámetros del sistema, utilizando aproximaciones para generar la nueva ley de control.
- Simulación de modelo de control propuesto y su experimentación a partir de la implementación en el robot manipulador SCARA de cuatro grados de libertad.
- Mejora de la interfaz de comunicación del robot manipulador SCARA realizando ensayos para tareas de contacto y manipulación de objetos mediante

pruebas de regulación y seguimiento de trayectorias del efector final.

- Comparación de la técnica de control propuesta con respecto a los modelos revisados en este trabajo de Tesis, ampliando el análisis de técnicas de control para sistemas no lineales.

Apéndice A

Convención de Denavit Hartenberg

Para calcular la cinemática directa de un manipulador de cadena abierta, es necesario desarrollar un método sistemático y general que permita definir la posición relativa y la orientación de cuatro eslabones consecutivos. Aunque los marcos pueden elegirse de manera arbitraria, es conveniente establecer reglas para definir los marcos de enlace. La convención Denavit–Hartenberg (D–H) se utiliza comúnmente para definir el marco de enlace i [1].

El siguiente algoritmo asigna los marcos de coordenadas de tal manera que satisface la convención de Denavith–Hartenberg [60]:

1. Localizar y enumerar las uniones de los ejes z_0, \dots, z_{n-1} . Alinear los ejes z_i con los ejes de movimiento rotacional o lineal de la $(i + 1)$ unión.
2. Asignar un marco diestro ortonormal o_{x_0, y_0, z_0} para la base del robot, con el eje z_0 a lo largo del eje de movimiento de la unión 1. Localizar el origen o_0 en un punto conveniente sobre el eje z_0 . Los ejes x_0 y y_0 pueden localizarse convenientemente siempre que permanecen ortogonal al eje z_0 y formando un marco de coordenada diestro.

3. Localizar el origen o_i , para $i = 1, \dots, n - 1$, en la intersección del eje z_i y normal común a z_i y z_{i-1} . Si z_i intersecciona a z_{i-1} , se coloca el origen o_i en la intersección. Si z_i y z_{i-1} son paralelos, coloca el origen o_i en un punto conveniente sobre z_i .
4. Colocar el eje x_i , para $i = 1, \dots, n - 1$, a lo largo de la normal común a z_i y z_{i-1} pasando a través del origen o_i , o a lo largo de la dirección normal al plano $z_i - z_{i-1}$ si z_i intersecciona z_{i-1} .
5. Colocar el eje y_i , para $i = 1, \dots, n - 1$, en forma al sistema coordenado de acuerdo a la regla de la mano derecha.
6. Establecer el sistema coordenado del efector final o_{x_n, y_n, z_n} . Localizar el origen o_n en el punto mas importante del efector final, por ejemplo: el centro de la pinza, la punta de la herramienta, etc. Colocar el eje z_n paralelo a z_{n-1} pasando a través de o_n . Colocar y_n convenientemente, por ejemplo: en la dirección en la que la pinza cierra. Finalmente, colocar x_n en forma al sistema coordenado de acuerdo a la regla de la mano derecha.

Apéndice B

Robot físico de cuatro grados de libertad

Dada la finalidad del presente trabajo, se requirió de la implementación de algoritmos de control aplicados a un robot manipulador de tipo SCARA de cuatro grados de libertad para realizar los diferentes experimentos inherentes al diseño del algoritmo de control, por lo tanto, se llevaron a cabo procesos para la fabricación de las piezas mecánicas del mismo.

El brazo robótico empleado parte de diseños conocidos por lo que a lo largo de este trabajo se ha utilizado un robot manipulador que cuenta con una variable articular de tipo prismática y tres variables articulares de tipo rotativa, la primera está dedicada para establecer la altura del efector final con respecto a la base del robot manipulador con respecto al eje "z", y las ultimas tres se articulaciones se emplean en referencia a un plano x y y para los giros del robot, donde su estructura es similar a la de un robot planar, visualizado en la Figura B.1.

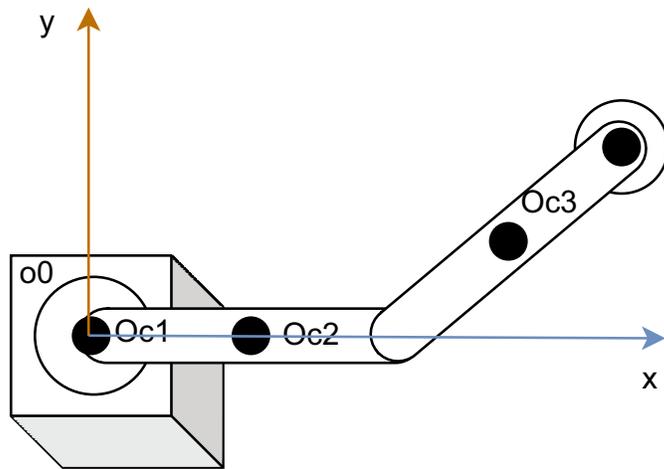


Figura B.1: Robot manipulador SCARA visualizado desde la parte superior.

Por otro lado, se empleó el diseño en 3D mediante el software ©Solid Works para diseñar las piezas necesarias y posteriormente fabricarlas.

Algunos de los diseños empleados para este brazo robótico se muestran a continuación en la Figura B.2.



Figura B.2: Diseño de piezas en 3D para el robot manipulador SCARA.

Estos diseños se fabricaron mediante el maquinado por control numérico, ha-

ciendo uso de filamento a base de ácido Poliláctico (PLA, del inglés: *Polylactic Acid*), adquiriendo las piezas de la Figura B.3.

Si bien, en el diseño de las piezas iniciales se consideraron huellas de motores paso a paso nema 17, sin embargo, para este trabajo se emplearon motores de corriente directa con caja de engranaje reductora.



Figura B.3: Piezas fabricadas para el robot manipulador SCARA, a base de filamento PLA.

Por otra parte, se establecieron un conjunto de elementos tanto mecánicos como electrónicos, visualizados en la Tabla B.1, para la estructura y circuitería del robot manipulador SCARA, respectivamente.

Artículo	Cantidad
Motor DC 25GA310	4
Módulo puente H de dos canales	2
Microcontrolador PIC18F4550	1
Microcontrolador PIC18f2331	4
Servo motor MG996R	1
Micro interruptor NA/NC	4
Fuente de alimentación DC 12v a 5A, 600W	1
Barra riel (10mm x 400mm)	4
Tornillo sin fin (8mm x 400mm)	1
Rodamientos lineales (10mm)	4
Rodamientos horizontales (40x60x13mm)	4
Rodamientos horizontales (35x52x12mm)	4
Rodamientos radiales (8x22x7mm)	4
Adaptador para varilla guía (4mm a 8mm)	1

Tabla B.1: Elementos mecánicos, eléctricos y electrónicos del robot manipulador SCARA para cuatro grados de libertad.

Para realizar la comunicación entre los microcontroladores (maestro y esclavos) se emplea el protocolo SPI, donde su diagrama de conexión entre módulos se presenta en la Figura B.4.

Este protocolo parte de un esquema en arreglo directo con respecto a los puertos de entradas y salidas de cada uno de los módulos. La interacción de cada esclavo con el maestro se realiza a partir de una señal de selección "ss".

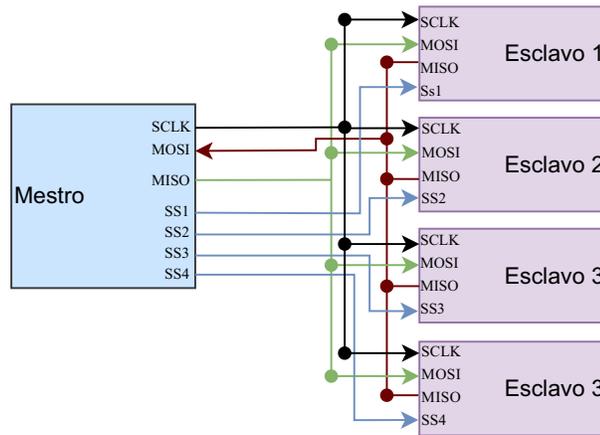


Figura B.4: Diagrama de conexión del protocolo de comunicación SPI.

Por otra parte, el diagrama de conexión electrónica se observa de manera general a partir de la Figura B.5, de la cual destacan los elementos tales como los módulos en placa de circuito impreso (PCB, siglas del inglés: *Printed Circuit Board*), donde van colocados los microcontroladores pic18f4550 y pic18f2331, los cuales interactúan mediante el protocolo de comunicación SPI, en función maestro–esclavo, respectivamente.

Una serie de "encoders" agregados a los motores de DC se conectan a los módulos de los esclavos para el intercambio de la información de las cuentas y el término de torque. Así mismo, se cuenta con un par de módulos de potencia para los cuatro motores de DC, estos controlados por las terminales de modulación por ancho de pulso (PWM, siglas del inglés: *Pulse Width Modulation*), y de dirección, señales brindadas por los módulos esclavo.

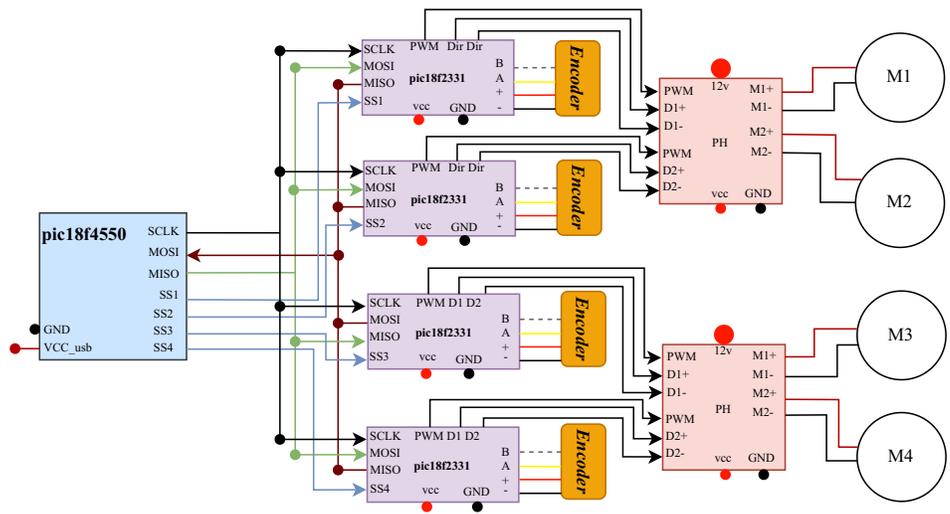


Figura B.5: Esquema electrónico general para el robot manipulador SCARA de cuatro grados de libertad.

Las conexiones del sistema electrónico del robot manipulador se observa en la Figura B.6.

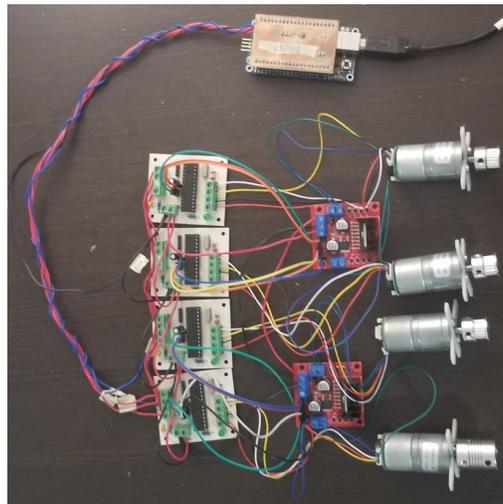


Figura B.6: Conexiones eléctricas del robot manipulador.

Con este conjunto de circuitos se realizaron pruebas de funcionamiento de cada

módulo antes de la instalación en el robot SCARA. Finalmente, el armado del robot manipulador SCARA se visualiza en la Figura B.7.

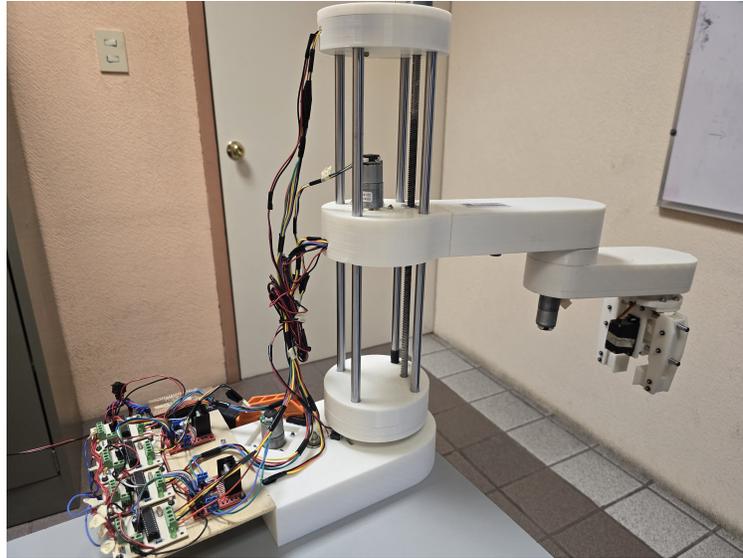


Figura B.7: Robot manipulador SCARA de cuatro grados de libertad para pruebas experimentales.

Apéndice C

Entorno de programación

Para la experimentación se requirió del uso del ID ©Visual Studio para llevar a cabo una aplicación mediante programación orientada a objetos, con la finalidad de poder interactuar con el robot manipulador SCARA de cuatro grados de libertad, de manera se puedan realizar pruebas de funcionamiento mecánico y eléctrico del sistema, así como también llevar a cabo la implementación de los algoritmos de control.

Para esto se realizó la interfaz de la Figura C.1



Figura C.1: Aplicación para la dirección del robot manipulador SCARA, desarrollada en ©Visual Studio.

La aplicación se enlaza mediante el botón de "Connect", al establecer comu-

nicación se habilita el botón de "Toggle LED" el cual es un indicador de encendido del sistema, con esto se confirma la conexión entre la aplicación de ©Visual Studio y los módulos propios de la interfaz de comunicación SPI.

La interfaz cuenta con una sección de prueba donde se puede ingresar el torque especificado por el usuario y enviar de manera directa a partir del botón "*Set Voltage*", probando así el funcionamiento de los motores de DC, de manera que se posicione el robot en lazo abierto. De la misma manera, se cuenta con etiquetas que desplegarán los valores de posición de las articulaciones rotativas (en grados sexagesimales) y prismática (en metros).

En la sección de control, se establecen un conjunto de "*Combo box*" que permiten la selección de la técnica de control y el tipo de movimiento deseado ya sea de regulación o seguimiento de trayectoria.

Se cuenta con un conjunto de "*Box*" para ajustar el "*Set Point*" de las articulaciones, así como los valores deseados de posición para el movimiento de regulación.

Una última sección permite obtener información de los estados del error de las articulaciones para observar el rendimiento.

El botón "*Control*" ejecuta el sistema de control con las configuraciones previas. Y finalmente, el botón "*Stop*" tiene la función de detener la ejecución del sistema.

Bibliografía

- [1] S. Bruno, S. Lorenzo, V. Luigi, and O. Giuseppe, “Robotics: modelling, planning and control, 2010,” *Cited on*, vol. 1, 1994.
- [2] K. S. Fu, R. C. González, and C. G. Lee, “Robótica: Control, detección, visión e inteligencia,” 1988.
- [3] F. R. Cortés, *Robótica: control de robots manipuladores*. Marcombo, 2020.
- [4] W. S. Barbosa, M. M. Gioia, V. G. Natividade, R. F. Wanderley, M. R. Chaves, F. C. Gouvea, and F. M. Gonçalves, “Industry 4.0: examples of the use of the robotic arm for digital manufacturing processes,” *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 14, pp. 1569–1575, 2020.
- [5] M. S. Ko, G.-N. Wang, H. S. Shin, and S. C. Park, “Machine control level simulation of an as/rs in the automotive industry,” in *Proceedings of the 2010 Winter Simulation Conference*, pp. 1727–1738, IEEE, 2010.
- [6] J. Barnett, M. Duke, C. K. Au, and S. H. Lim, “Work distribution of multiple cartesian robot arms for kiwifruit harvesting,” *Computers and Electronics in Agriculture*, vol. 169, p. 105202, 2020.
- [7] T. Pham, “Position control of cartesian robot,” 2019.

- [8] B. Singh, N. Sellappan, and P. Kumaradhas, “Evolution of industrial robots and their applications,” *International Journal of emerging technology and advanced engineering*, vol. 3, no. 5, pp. 763–768, 2013.
- [9] R. Chaturvedi, A. Islam, and K. Sharma, “Anticipated investigation of a cylindrical robot arm by means of compound materials,” *Eur J Mol Clin Med*, vol. 7, no. 4, pp. 736–745, 2020.
- [10] F. Collins and M. Yim, “Design of a spherical robot arm with the spiral zipper prismatic joint,” in *2016 IEEE international conference on robotics and automation (ICRA)*, pp. 2137–2143, IEEE, 2016.
- [11] Q. Zhang, Q. Jia, H. Sun, and Z. Gong, “Application of a genetic algorithm-based pi controller in a spherical robot,” in *2009 IEEE International Conference on Control and Automation*, pp. 180–184, IEEE, 2009.
- [12] A. A. Kracht *et al.*, “A linear base articulated robot arm for surgical endoscopy,” 2006.
- [13] R. Jain, M. N. Zafar, and J. Mohanta, “Modeling and analysis of articulated robotic arm for material handling applications,” in *IOP Conference Series: Materials Science and Engineering*, vol. 691, p. 012010, IOP Publishing, 2019.
- [14] E. Akdoğan and M. A. Adli, “The design and control of a therapeutic exercise robot for lower limb rehabilitation: Physiotherobot,” *Mechatronics*, vol. 21, no. 3, pp. 509–522, 2011.
- [15] C. Hua, X. Guan, and G. Duan, “Variable structure adaptive fuzzy control for a class of nonlinear time-delay systems,” *Fuzzy sets and systems*, vol. 148, no. 3, pp. 453–468, 2004.
- [16] W. Yu and J. Rosen, “A novel linear pid controller for an upper limb exoskeleton,” in *49th IEEE Conference on Decision and Control (CDC)*, pp. 3548–3553, IEEE, 2010.

- [17] A. J. Critchlow, "Introduction to robotics," 1985.
- [18] W. S. Pambudi and J. P. B. Pelawi, "Simulasi folding machine dengan pid, p, pi, pd dan fuzzyâ€pd (proportional differential)," *Jurnal Sains dan Informatika*, vol. 1, no. 1, pp. 25–33, 2015.
- [19] W. M. Elsrogy, M. Fkirin, and M. M. Hassan, "Speed control of dc motor using pid controller based on artificial intelligence techniques," in *2013 International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 196–201, IEEE, 2013.
- [20] M. A. Al-Khedher and M. S. Alshamasin, "Scara robot control using neural networks," in *2012 4th International Conference on Intelligent and Advanced Systems (ICIAS2012)*, vol. 1, pp. 126–130, IEEE, 2012.
- [21] O. Shahnazari, A. Chatraei, K. Shojaei, and M. Pourrahim, "Fuzzy gain scheduling saturated pid controller: design and implementation on robot manipulator," in *2017 Iranian Conference on Electrical Engineering (ICEE)*, pp. 597–602, IEEE, 2017.
- [22] J. Coyle-Byrne and R. D. Klafter, "Real-time zone-adaptive control of scara-type robot arm," in *Proceedings., IEEE International Conference on Robotics and Automation*, pp. 2089–2094, IEEE, 1990.
- [23] S. Suvilath, K. Khongsomboun, T. Benjanarasuth, and N. Komine, "Imc-based pid controllers design for a two-links scara robot," in *TENCON 2011-2011 IEEE Region 10 Conference*, pp. 1030–1034, IEEE, 2011.
- [24] L.-X. Wang, "A course in fuzzy systems," 1999.
- [25] S.-J. Huang and D.-N. Yang, "Fuzzy logic controller for a scara robot with synchronous network," *International Journal of Computer Applications in Technology*, vol. 10, no. 1-2, pp. 15–26, 1997.

- [26] Y. Oktarina, F. Septiarini, T. Dewi, P. Risma, and M. Nawawi, “Fuzzy-pid controller design of 4 dof industrial arm robot manipulator,” *Computer Engineering and Applications Journal*, vol. 8, no. 2, pp. 123–136, 2019.
- [27] G. Feng, “A survey on analysis and design of model-based fuzzy control systems,” *IEEE Transactions on Fuzzy systems*, vol. 14, no. 5, pp. 676–697, 2006.
- [28] Y. He, X. Mai, C. Cui, J. Gao, Z. Yang, K. Zhang, X. Chen, Y. Chen, and H. Tang, “Dynamic modeling, simulation, and experimental verification of a wafer handling scara robot with decoupling servo control,” *IEEE Access*, vol. 7, pp. 47143–47153, 2019.
- [29] A. Visioli and G. Legnani, “On the trajectory tracking control of industrial scara robot manipulators,” *IEEE transactions on industrial electronics*, vol. 49, no. 1, pp. 224–232, 2002.
- [30] D. Camilleri and T. Prescott, “Analysing the limitations of deep learning for developmental robotics,” in *Biomimetic and Biohybrid Systems: 6th International Conference, Living Machines 2017, Stanford, CA, USA, July 26–28, 2017, Proceedings 6*, pp. 86–94, Springer, 2017.
- [31] M. Daş and L. Dülger, “Control of a scara robot: Pso-pid approach.,” *Control & Intelligent Systems*, vol. 38, no. 1, 2010.
- [32] M. T. Daş, L. C. Dülger, and G. S. Daş, “Robotic applications with particle swarm optimization (pso),” in *2013 International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 160–165, IEEE, 2013.
- [33] A. Coronel-Escamilla, F. Torres, J. Gómez-Aguilar, R. Escobar-Jiménez, and G. Guerrero-Ramírez, “On the trajectory tracking control for an scara robot manipulator in a fractional model driven by induction motors with pso tuning,” *Multibody System Dynamics*, vol. 43, pp. 257–277, 2018.

- [34] G. Mester, “Design of the fuzzy control systems based on genetic algorithm for intelligent robots,” *Interdisciplinary Description of Complex Systems: INDECS*, vol. 12, no. 3, pp. 245–254, 2014.
- [35] D. P. Garg, “Adaptive control of nonlinear dynamic scara type of manipulators,” *Robotica*, vol. 9, no. 3, pp. 319–326, 1991.
- [36] J. A. Sarapura, F. Roberti, and R. Carelli, “Adaptive 3d visual servoing of a scara robot manipulator with unknown dynamic and vision system parameters,” *Automation*, vol. 2, no. 3, pp. 127–140, 2021.
- [37] L.-A. Dessaint, M. Saad, B. Hébert, and C. Gargour, “An adaptive controller for a direct-drive scara robot: Analysis and simulation,” in *[Proceedings] IECON’90: 16th Annual Conference of IEEE Industrial Electronics Society*, pp. 414–420, IEEE, 1990.
- [38] Z. Li, J. Li, and Y. Kang, “Adaptive robust coordinated control of multiple mobile manipulators interacting with rigid environments,” *Automatica*, vol. 46, no. 12, pp. 2028–2034, 2010.
- [39] S. I. Han and J. M. Lee, “Output-tracking-error-constrained robust positioning control for a nonsmooth nonlinear dynamic system,” *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 6882–6891, 2014.
- [40] F. R. C. Talledo, *Control de trayectoria de Robots Manipuladores*. PhD thesis, University of Groningen, 2023.
- [41] J. R. Flores, L. S. Muyulema, J. C. Cruz, and A. M. Caiza, “Mejora de la precisión del seguimiento de trayectoria de un brazo robótico mediante un algoritmo de red neuronal inicializado con lyapunov,” *Polo del Conocimiento: Revista científico-profesional*, vol. 8, no. 10, pp. 443–466, 2023.

- [42] W. Li, X. Chang, F. M. Wahl, and J. Farrell, “Tracking control of a manipulator under uncertainty by fuzzy p+ id controller,” *Fuzzy Sets and Systems*, vol. 122, no. 1, pp. 125–137, 2001.
- [43] I. Cervantes and J. Alvarez-Ramirez, “On the pid tracking control of robot manipulators,” *Systems & control letters*, vol. 42, no. 1, pp. 37–46, 2001.
- [44] K. Gkountas and A. Tzes, “Leader/follower force control of aerial manipulators,” *IEEE Access*, vol. 9, pp. 17584–17595, 2021.
- [45] B. Hu, Z.-H. Guan, F. L. Lewis, and C. P. Chen, “Adaptive tracking control of cooperative robot manipulators with markovian switched couplings,” *IEEE Transactions on Industrial Electronics*, vol. 68, no. 3, pp. 2427–2436, 2020.
- [46] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. John Wiley & Sons, 2020.
- [47] J. Cornejo, V. Cruz, F. Carrillo, R. Cerda, and E. R. S. Penadillo, “Mechatronics design and kinematic simulation of scara robot to improve safety and time processing of covid-19 rapid test,” in *2022 First International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*, pp. 1–6, IEEE, 2022.
- [48] P. I. Corke, “A simple and systematic approach to assigning denavit–hartenberg parameters,” *IEEE transactions on robotics*, vol. 23, no. 3, pp. 590–594, 2007.
- [49] A. Fernández and S. A. Marrero-Osorio, “Modelos cinemático y dinámico de un robot de cuatro grados de libertad,” *Ingeniería Electrónica, Automática y Comunicaciones*, vol. 38, no. 3, pp. 56–75, 2017.
- [50] A. O. Baturone, *Robótica: manipuladores y robots móviles*. Marcombo, 2005.

- [51] W. He and Y. Dong, “Adaptive fuzzy neural network control for a constrained robot using impedance learning,” *IEEE transactions on neural networks and learning systems*, vol. 29, no. 4, pp. 1174–1186, 2017.
- [52] Y. Lin, Z. Chen, and B. Yao, “Unified motion/force/impedance control for manipulators in unknown contact environments based on robust model-reaching approach,” *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 4, pp. 1905–1913, 2021.
- [53] T. I. Fossen and J. P. Strand, “Tutorial on nonlinear backstepping: Applications to ship control,” 1999.
- [54] H. Simon, “Neural networks and learning machines,” 2009.
- [55] P. N. Suganthan and R. Katuwal, “On the origins of randomization-based feed-forward neural networks,” *Applied Soft Computing*, vol. 105, p. 107239, 2021.
- [56] H. D. Bautista Vargas and J. Sosa Franco, “Tribología en robots manipuladores: fundamentos, efectos, modelado y simulación,” 2007.
- [57] V. Callejas Guerrero, “Control adaptable para robots manipuladores,” 2006.
- [58] W. He, A. O. David, Z. Yin, and C. Sun, “Neural network control of a robotic manipulator with input deadzone and output constraint,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 6, pp. 759–770, 2015.
- [59] A. Jarillo-Silva, O. A. Dominguez-Ramirez, V. Parra-Vega, and J. P. Ordaz-Oliver, “Haptic guidance based on sub-optimal passivity control,” in *2009 Electronics, Robotics and Automotive Mechanics Conference (CERMA)*, pp. 175–180, IEEE, 2009.
- [60] M. A. Arteaga, A. Gutiérrez-Giles, and J. Pliego-Jiménez, *Local stability and ultimate boundedness in the control of robot manipulators*. Springer, 2022.