BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA ELECTRÓNICA

MAESTRÍA EN CIENCIAS DE LA ELECTRÓNICA,
OPCIÓN EN AUTOMATIZACIÓN

"UN ENFOQUE LAGRANGIANO PARA EL CONTROL BASADO EN APRENDIZAJE DE ROBOTS MANIPULADORES"

T E S I S

Presentada para obtener el título de:

Maestro en Ciencias de la Electrónica, Opción en Automatización

Presenta:

Ing. Viridiana Luna Zempoalteca*

Directores:

Dr. José Fernando Reyes Cortés (FCE-BUAP)

Dr. Rodolfo Reyes Báez (UG)

Puebla, México **Agosto 2022**

* Becario CONACYT

** University of Groningen.

BUAP®

Agradecimientos

Agradezco a mi papá Jose Alberto Luna López quien siempre me guió e inspiró para la realización y término de mis estudios de maestría. Así como, a mi mamá Rocío Zempoalteca Pérez quien siempre creyó en mí y me alento a siempre dar lo mejor de mí y más. A mis hermanas Anayantzi y Xochitl por su compañía, consejos y cariño.

A la Benemérita Universidad Autónoma de Puebla y en especial a todos los profesores de la Maestría en Ciencias de la Electrónica, Opción en Automatización quienes me brindaron la oportunidad de realizar mis estudios de maestría y por todos los conocimientos aportados a mi desarrollo en este trabajo de tesis.

A mis asesores, el Dr. Fernando Reyes y el Dr. Rodolfo quienes guiaron mi tema de tesis, compartieron sus conocimientos y me inspiraron a trabajar arduamente para lograr los objetivos planteados.

A CONACYT por el apoyo económico brindado.

Dedicatoria

Este trabajo lo dedico a mis padres, Jose Alberto y Rocío.
Y a mis hermanas, Anayantzi y Xochitl.

Resumen

En el presente trabajo de tesis se realiza el estudio de aprendizaje de máquina e iterativo para diseñar un esquema de control que resuelva problemas de servo-control (simplicidad estructural y seguimiento perfecto) en robots manipuladores. Dicho esquema posee una clara interpretación en sus ecuaciones físicas constituidas por el formalismo de Euler-Lagrange y la teoría de pasividad de sistemas de control en robots manipuladores. Con los conocimientos adquiridos se realizan simulaciones en MATLAB de ambos tipos de aprendizaje, en el caso del aprendizaje de máquina se trabaja con una red neuronal de una sola capa aplicada al lazo cerrado de control cartesiano y articular para el control de posición del robot manipulador de dos grados de libertad. Por otro lado para el aprendizaje iterativo recursivo y el aprendizaje por tareas repetitivas se realiza una serie de simulaciones con sistemas de primer y segundo orden en MATLAB y Simulink donde se trabaja con el algoritmo de control $u_k = u_{k-1} + \beta\widetilde{y}_k$, característico de este tipo de aprendizaje, en conjunto con el control proporcional derivativo (PD), y así, finalmente se obtienen las simulaciones del control de posición del robot manipulador de un grado de libertad y de dos grados de libertad, con estos resultados se obtienen sus normas euclidianas $\mathcal{L}_2$ que definen el índice de desempeño en los algoritmos de control.

Índice general

Resumen VII

Introducción XVI

Objetivos XVII

Justificación XVIII

1. Estado del arte 1
1.1. Control de Aprendizaje Iterativo . . . 2
1.2. Aprendizaje de máquina . . . 3

2. Marco teórico 5
2.1. Robot manipulador . . . 5
2.2. Cinemática analítica de Euler . . . 6
2.2.1. Sistema de referencia inercial . . . 6
2.2.2. Ángulos de Euler . . . 7
2.2.3. Movimiento de traslación y rotación . . . 8
2.2.4. Matrices de transformación homogénea . . . 9
2.3. Cinemática directa . . . 9
2.3.1. Convención Denavit-Hartenberg . . . 10
2.3.2. Modelo dinámico . . . 12
2.4. Pasividad de robots manipuladores . . . 13
2.5. Aprendizaje de máquina (Machine Learning) . . . 14
2.5.1. Perceptron . . . 16
2.6. Control de Aprendizaje Iterativo . . . 16
Feedback . . . 17
Feedforward . . . 17
2.6.1. Procesos repetitivos . . . 18
2.6.2. Enfoque original del Control de Aprendizaje Iterativo . . . 19
2.6.3. Control de Aprendizaje Iterativo con Control de Retroalimentación . . . 22
2.7. Algoritmos Canónicos de control de Aprendizaje Iterativo . . . 23
2.7.1. Embebidas . . . 23
Ciclo previo de aprendizaje . . . 23
Ciclo actual de aprendizaje . . . 24
Ciclo previo y actual de aprendizaje . . . 25
2.7.2. Cascada . . . 25
Control de aprendizaje iterativo en cascada . . . 26
Control de aprendizaje iterativo en cascada incremental . . . 26

3. Modelo dinámico **28**
3.1. Robot manipulador de un grado de libertad (péndulo simple) . . . 28
3.1.1. Cinemática analítica de Euler para el péndulo simple . . . 28
3.1.2. Cinemática diferencial . . . 32
3.1.3. Propiedades de la matriz $S(\dot{q}_1\hat{\mathbf{k}})$ . . . 32
Conmutación del producto de la matriz $S(\dot{q}_1\hat{\mathbf{k}})$ y $S(\dot{q}_1\hat{\mathbf{k}})^T$ . . . 33
Valores propios de la matriz $S(\dot{q}_1\hat{\mathbf{k}})$ . . . 33
Norma espectral de $S(\dot{q}_1\hat{\mathbf{k}})$ . . . 33
3.1.4. Rapidez de traslación . . . 34
3.1.5. Modelo de energía del péndulo simple . . . 34
3.1.6. Pasividad del péndulo simple . . . 36
3.2. Robot manipulador de 2 gdl . . . 37
3.2.1. Cinemática analítica de Euler para el robot manipulador de 2 gdl . . . 37
3.2.2. Cinemática analítica de Euler (Representación Denavit-Hartenberg) . . . 38
3.2.3. Tabla de Denavit-Hartenberg . . . 39
3.2.4. Transformaciones homogéneas Denavit-Hartenberg . . . 39
3.2.5. Cinemática diferencial vectorial . . . 40
3.2.6. Rápidez de traslación . . . 41
3.2.7. Modelo dinámico de un robot manipulador de 2 gdl . . . 42

4. Perceptrón aplicado a robot manipulador de 2gdl **45**

5. Ciclo previo de aprendizaje **54**
5.1. Sistema dinámico lineal escalar . . . 54
5.2. Ejemplo: Sistema dinámico lineal de primer orden . . . 56
5.3. Sistema dinámico vectorial . . . 59
Ejemplo: Sistema dinámico lineal de segundo orden . . . 60
Ejemplo: Robot manipulador de un grado de libertad . . . 62

6. Proceso repetitivo **64**
6.1. Simulaciones con sistemas lineales de segundo orden . . . 65
6.2. Simulación del robot manipulador de un grado de libertad . . . 77
6.3. Simulación del robot manipulador de dos grados de libertad . . . 81
6.4. Control PD+ de aprendizaje iterativo . . . 86

Conclusiones **90**

Participación en congreso **91**

Certificación de idioma **92**

Bibliografía **93**

Índice de figuras

1.1. Linea del tiempo de la evolución del aprendizaje por máquina. 1
1.2. Control PD de aprendizaje iterativo (Arimoto, 1986). 2
1.3. Aplicación de aprendizaje de máquina en el año 2006 (Kober, 2019). 3

2.1. Elementos de un robot manipuldor de 3 grados de libertad (gdl). 5
2.2. Cadena cinemática abierta. 6
2.3. Sistema de referencia inercial. 7
2.4. Ángulos de Euler. 7
2.5. Representación general de ejes de rotación y los nuevos sub-sistemas de referencia que se tienen. 8
2.6. Movimientos de traslación y orientación. 9
2.7. Diagrama de bloques del control de aprendizaje de máquina (Duriez, Brunton y Noack, 2017). 15
2.8. Red neuronal de una sola capa. 16
2.9. Sistema de retroalimentación frente a sistema de alimentación. En ambos sistemas tenemos la planta o proceso y un controlador C. El controlador de retroalimentación a) mide la salida y para determinar el efecto de la perturbación v, mientras que el controlador de avance b) mide la perturbación directamente, pero no mide directamente la salida del proceso. 18
2.10. Diagrama de bloques del proceso repetitivo (de tareas repetitivas) (Oomen, 2018). 18
2.11. Ejemplo de error de convergencia para el aprendizaje a partir de tareas pasadas en un sistema de impresión, la cual tiene una rápida convergencia (Oomen, 2018). 19
2.12. Diagrama a bloques del control de aprendizaje iterativo y el progreso de aprendizaje de las señales relacionadas. 19
2.13. Diagrama de bloque Control de Aprendizaje Iterativo PD. 21
2.14. Diagrama de bloques del control de aprendizaje iterativo del tipo P (proporcional). 21
2.15. Diagrama de bloques del controlador híbrido. 22
2.16. Diagrama a bloques del ciclo previo de aprendizaje (Jian-Xin Xu y Lee, 2009). 23
2.17. Diagrama de bloques del ciclo actual de aprendizaje (Jian-Xin Xu y Lee, 2009). 24
2.18. Diagrama a bloques de PCCL (Previous cycle-Current cycle) ciclo previo y ciclo actual. . 25
2.19. Diagrama de bloques del control de aprendizaje en cascada (Jian-Xin Xu y Lee, 2009). . . 26
2.20. Diagrama a bloques del control de aprendizaje iterativo en cascada incremental (Jian-Xin Xu y Lee, 2009). 26

3.1. Vista lateral del péndulo simple y vista superior, con $q_1 = 0$ (posición de casa). 28
3.2. Vista frontal del péndulo con $q_1 \neq 0$. 30
3.3. Análisis de energía potencial, donde l_{c1} se indica con los puntos rojos. 35
3.4. Servomecanismo péndulo. 36
3.5. Vista lateral del péndulo simple y vista superior, con $q_1 = q_2 = 0$ (posición de casa). 37
3.6. Vista frontal del péndulo con $q_{1,2} \neq 0$. 37
3.7. Robot manipulador de 2 GDL. 39
3.8. Robot manipulador de 2gdl. 42

4.1. Cinemática inversa del robot manipulador de 2gdl. 46
4.2. Diagrama de bloques del lazo cerrado de control para el robot manipulador de 2gdl. 46
4.3. Diagrama de red neuronal. 47
4.4. Aplicación de la red neuronal en el lazo cerrado de control del robot manipulador de 2gdl. 48
4.5. Posiciones articulares deseadas, línea verde q_{d_1} y línea azul q_{d_2}. 48
4.6. Comparativa de posiciones deseadas VS posiciones obtenidas por el regresor y finalmente el nuevo modelo de regresor lineal implementado en el lazo cerrado de control PD. 49
4.7. Diagrama de bloques del lazo cerrado de control cartesiano. 50
4.8. Diagrama de red neuronal. 51
4.9. Diagrama de bloques 51
4.10. Posición cartesiana deseada en metros $x = 0.55m$ y $y = 0.45m$. 52
4.11. Comparativa de la posición cartesiana obtenida entre la cinemática directa, el regresor propuesto y el control propuesto en conjunto con el regresor. 53

5.1. Diagrama de bloques del ciclo previo de aprendizaje. 54
5.2. Posición de salida para diferentes iteraciones. 57
5.3. Error de posición para cada una de las salidas k-ésimas. 58
5.4. Posición de salida k-ésima. 61
5.5. Posición de salida k-ésima y error de posición para 10 iteraciones. 62
5.6. Salida k-ésima del sistema de segundo orden no lineal y error de posición para el caso en el que $k = 1,000$ iteraciones y cuando $k = 10,000$ iteraciones. 63

6.1. Función pulso, con un periodo de 50 segundos y un ancho de pulso del 90 %. 64
6.2. Polos del sistema dinámico lineal. 65
6.3. Respuesta transitoria de la ecuación 1.1 al escalón unitario. 66
6.4. Simulación en simulink, diagrama de bloques del sistema en lazo cerrado de control PD. . 66
6.5. Respuesta del sistema de 2do orden en el lazo cerrado de control PD, donde la tarea repetida se establece a partir de la función pulso. 66
6.6. Gráfica del error respecto al número de iteraciones y el tiempo. Para el sistema con control PD en el lazo cerrado de control. 67
6.7. Simulación en simulink del sistema de segundo orden con el control de aprendizaje iterativo proporcional. 68
6.8. Salida del sistema en la k-ésima iteración, al implementar el sistem dinámico de segundo orden con el control de aprendizaje iterativo. 68
6.9. Error de posición respecto al tiempo y número de iteraciones. 69
6.10. Simulación en simulink del control de aprendizaje iterativo proporcional, implementado en el lazo cerrado de control PD con un sistema de 2do orden. 69
6.11. Señal de salida del control de aprendiaje iterativo con tarea repetitiva, implementado en el lazo cerrado de control PD, con sistema de segundo orden. 70
6.12. Error de posición respecto al tiempo, número de iteraciones y magnitud del error. 70
6.13. Ubicación de los polos del sistema dinámico lineal. 71
6.14. Repuesta del sistema al escalón unitario. 72
6.15. Diagrama a bloques del sistema en lazo cerrado con el ILC-P. 72
6.16. Respuesta del sistema en la primera iteración del ILC-P. 73
6.17. Salida del sistema con ILC-P en lazo cerrado de control. 73
6.18. Gráfica del error respecto al número de iteraciones y el tiempo. 74
6.19. Diagrama de bloques implementado en Simulink MATLAB, para el lazo cerrado de control PD. 74
6.20. Señales de salida del sistema y el error, para el lazo cerrado de control PD. 75

6.21. Diagrama a bloques implementado en Simulink MATLAB. 75
6.22. Gráfica de la señal de salida y el error del sistema con ILC-P en paralelo con el cotnrol PD del lazo cerrado de control. 76
6.23. Gráfica del error del sistema respecto al número de iteraciones y tiempo. 76
6.24. Diagrama de bloques del péndulo simple en lazo cerrado de control PD. 77
6.25. Señal de salida del sistema y error. 78
6.26. Posición de salida q_k en 10 iteraciones. 78
6.27. Simulación en MatLab simulink del péndulo simple. 79
6.28. Señal de salida del péndulo simple y error de posición 80
6.29. Error de posición del lazo cerrado de control respecto al número de iteraciones y al tiempo. 80
6.30. Simulación en simulink del lazo cerrado de control PD para el robot manipulador de dos grados de libertad. 81
6.31. Posición de salida del primer eslabón q_1 y del segundo eslabón q_2 para $k = 1$. 82
6.32. Señales de salida del robot manipulador de 2gdl y errores de posición 82
6.33. Simulación en MatLab simulink del robot de 2gdl. 83
6.34. Posiciones de salida q_1 y q_2 del robot manipulador de dos grados de libertad. 84
6.35. Señales de salida del robot manipulador de 2gdl y errores de posición 84
6.36. Gráficas comparativas de la posición de salida q_1 ante perturbaciones del control PD vs el control PD-ILC, primero se añade una ganancia = 4, y después una ganancia= 0.5. 85
6.37. Gráficas comparativas de la posición de salida q_2 ante perturbaciones del control PD vs el control PD-ILC, primero se añade una ganancia = 4, y después una ganancia= 0.5. 86
6.38. Diagrama de bloques del controlador PD+ de aprendizaje iterativo. 87

Índice de tablas

2.1. Parámetros Denavit-Hartenberg. 11
2.2. Tipos de algoritmos de control de aprendizaje iterativo. 21

3.1. Parámetros del robot de 2gdl. 43

4.1. Parámetros de simulación para el robot de 2gdl. 45
4.2. Comparativa de $\mathcal{L}_2$ para el control PD hiperbólico-exponencial y la red neuronal propuesta. 50
4.3. Comparativa de $\mathcal{L}_2$ para el control PD hiperbólico-exponencial y la red neuronal propuesta. 53

5.1. Ganancias proporcionales. 58

6.1. Comparativa de $\mathcal{L}_i$ para el sistema de la ecuación (6.6). 71
6.2. Comparativa de $\mathcal{L}_i$ para el sistema de la ecuación (6.17). 77
6.3. Comparativa de $\mathcal{L}_i$ para el sistema de la ecuación (6.30). 81
6.4. Comparativa de $\mathcal{L}_i$ para el sistema de la ecuación (6.30). 85

Introducción

La regulación o control de posición de robots manipuladores radica en colocar el extremo final del robot manipulador en una posición deseada $\mathbf{q}_d(t)$ y que es designada por el usuario. Esto implica diseñar una ley de control $\boldsymbol{\tau}$ que suministre el par (pares) aplicados en los servomotores del robot manipulador de n grados de libertad (gdl) para cualquier condición inicial $\begin{bmatrix}\mathbf{q}(0) & \dot{\mathbf{q}}(0)\end{bmatrix}^T \in \mathbb{R}^{2n}$ y se cumple cuando $\lim_{t\to\infty}\begin{bmatrix}\dot{\mathbf{q}}(t) & \widetilde{\mathbf{q}}(t)\end{bmatrix}^T = \begin{bmatrix}0 & 0\end{bmatrix}^T$. El diseño de controladores esta basado en el principio de retroalimentación (control clásico), donde los errores de salida se devuelven, a través de un controlador, a la entrada, permitiendo precisión en la señal de salida y una reducción en los efectos de perturbaciones no deseadas e incertidumbre de la planta. Sin embargo, surge la posibilidad de provocar inestabilidad, ya que las entradas limitadas al sistema pueden ocasionar salidas no limitadas aún para sistemas estables.

A partir de los trabajos realizados por Alan Turing sobre inteligencia artificial se desarrollaron los conceptos sobre aprendizaje de máquinas, los cuales imitan la capacidad de aprendizaje del ser humano y dieron lugar a la invención del perceptrón por Frank Rosenblatt, quien desarrolló la red neuronal artificial más antigua de la historia capaz de aprender patrones y reconocerlos. Esto condujo al desarrollo de otras estrategias de control con diferentes objetivos y plantas, en el caso del control de aprendizaje iterativo se tenía como objetivo controlar sistemas como fuentes de alimentación y discos duros, ya que sus perturbaciones y señales de referencia son periódicas, sin embargo, diversos procesos de producción industrial tienen esta característica periódica, como el montaje de piezas, pintado de autos, ensamblado de chasis, etc., que requiere establecer posiciones articulares deseadas repetitivas.

En este tema de tesis se aplica el control de aprendizaje iterativo mediante simulaciones (MATLAB y Simulink) para diversos sistemas dinámicos lineales, y así finalmente trabajar con el robot manipulador de un grado de libertad y de dos grados de libertad. Se definen las características cualitativas de cada uno de los sistemas dinámicos, su respuesta transitoria al escalón unitario y se realiza una comparación de las posiciones de salida cuando a cada uno de los sistemas se le aplica únicamente el control de aprendizaje iterativo, el control proporcional derivativo (PD) y cuando se aplican ambos tipos de controladores en el lazo cerrado de control. Con los resultados obtenidos se aplica la norma euclidiana $\mathcal{L}_2$ la cual define el índice de desempeño en cada uno de los controladores simulados.

De igual manera, se aplica el modelo del perceptrón en el lazo cerrado de control articular y cartesiano para el robot manipulador de 2gdl, se desarrolla en código fuente de MATLAB y se proponen funciones de activación que en conjunto con la identificación paramétrica obtiene los pesos correspondientes de las funciones de activación que definen la red neuronal propuesta y con la cual es posible sustituir el modelo de cinemática directa e inversa del robot manipulador de 2gdl, como se muestra en el capítulo 4. Parte de los conocimientos adquiridos para el desarrollo de la red neuronal fueron obtenidos en el curso extracurricular de Introducción a las Redes Neuronales Artificiales (IRNA) impartida por el Dr. Humberto Sossa del IPN, además, durante el desarrollo del lazo cerrado de control para el robot manipulador de 2gdl se propuso un algoritmo de control hiperbólico-exponencial con el que se trabaja para el desarrollo de la red neuronal y el cual fue presentado en el Congreso Internacional de Ingeniería Electrónica ELECTRO 2021.

Objetivos

Los objetivos general y específicos de este trabajo de tesis se enuncian a continuación.

Objetivo general

Diseñar un controlador basado en técnicas de aprendizaje (iterativo o de máquina) para robots manipuladores, con propiedades físicas.

Objetivos particulares

1. Estudiar de los algoritmos canónicos de aprendizaje de máquina e iterativo en un péndulo robot en simulaciones numéricas y de manera experimental.
2. Obtener el modelo dinámico de robots manipuladores usando mecánica Lagrangiana.
3. Estudiar el control basado en pasividad de robots manipuladores con experimentos.
4. Diseñar del algoritmo de control basado en aprendizaje con propiedades físicas.
5. Publicar resultados en congresos arbitrados y/o revista indexada.

Justificación

Los recientes avances tecnológicos en la Industria 4.0, ingeniería biomédica, instrumentación avanzada, energías renovables, entre otras; requieren que robots u otros sistemas autónomos se desempeñen en terrenos complejos e inciertos. Debido a esto, no es posible prever cada circunstancia con la que el robot se va a enfrentar. En su lugar, el robot debe ser capaz de adaptarse y aprender. Específicamente, en el caso de la manufactura inteligente, esto significa dotar a los robots de habilidades de aprendizaje, los cuales tradicionalmente realizan tareas repetitivas mediante lazos de servo-control basado en modelos.

Cuando la tarea que el robot debe llevar a cabo es conocida con antelación y el ambiente de trabajo no cambia de manera significativa, existen una basta variedad de métodos de diseño que resuelven el problema de control, ver por ejemplo las referencias en (Reyes, 2011). Sin embargo, si el robot también debe de afrontar circunstancias complejas y desconocidas que no pueden ser modeladas o preprogramadas, es necesario que su sistema de control sea capaz de adaptarse y aprender de las mediciones y experiencia durante su operación (Arimoto, 1986), (Oomen, 2018), (Fisac y col., 2019). El proceso de aprendizaje en este contexto consiste en un problema dinámico (Zhang, Chu y Shu, 2019), pues el robot debe aprender de la información que generó a través de sus acciones y sistemas de percepción

Por tal motivo, nace la necesidad de desarrollar en este trabajo de tesis una combinación ventajosa de la versatilidad en los algoritmos de aprendizaje (de máquina e iterativo) con las propiedades físicas del modelo dinámico de robots manipuladores, y en particular la propiedad de pasividad (van der Schaft, 2017). Esto permitirá dotar a los esquemas de control basados en aprendizaje de una clara interpretación física, y al mismo tiempo de bases teóricas sólidas para garantizar la convergencia asintótica del punto de equilibrio en el lazo cerrado de control de posición.

Capítulo 1

Estado del arte

El aprendizaje de máquina e iterativo no sería conocido como hoy en día sino fuera por las contribuciones intelectuales de los científicos que surgieron en el siglo XX, quienes a partir de un interés común de conocer y entender el funcionamiento neuronal de nuestro cerebro, desarrollaron un modelo base de una red neuronal. Esto desencadenaría una serie de investigaciones y aplicaciones tecnológicas, que al día de hoy continua en desarrollo. En la figura 1.1 se muestra una línea del tiempo de los avances tecnológicos que han surgido a partir el modelo matemático de redes neuronales humanas.

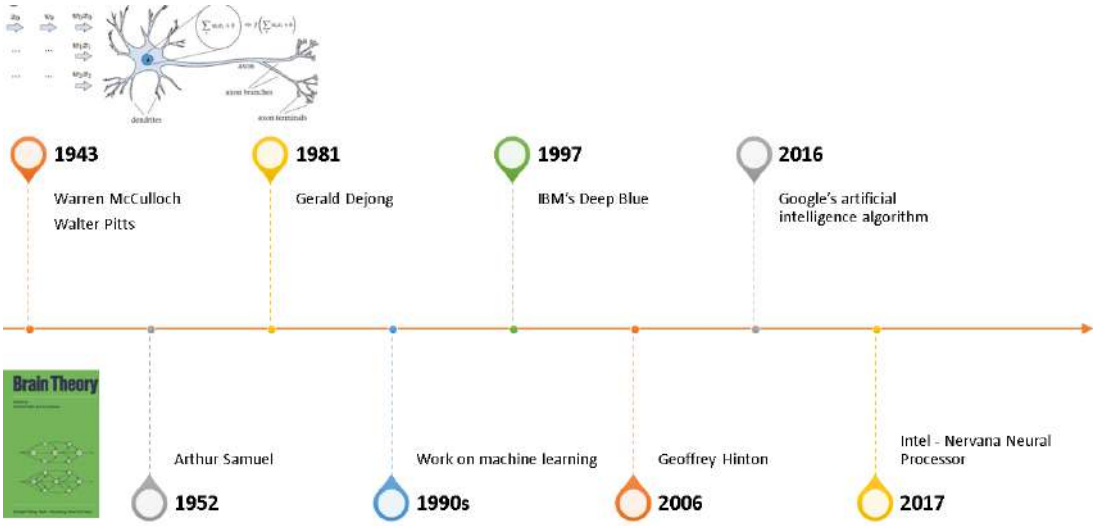


Figura 1.1: Linea del tiempo de la evolución del aprendizaje por máquina.

1. 1943: El neurofisiólogo Warren McCulloch y el matemático Walter Pitts escriben el primer artículo sobre redes neuronales y como funcionan. Crearon un modelo del mismo con un circuito eléctrico, y así nació el estudio de la red neuronal.
2. 1950: Alan Turing creó la prueba de Turing de fama mundial. Esta prueba es bastante simple: para que una computadora pase, debe ser capaz de convencer a un humano de que es un humano y no una computadora.
3. 1952: Arthur Samuel creo el primer programa de computadora que podía aprender mientras se ejecutaba, empleado en un juego de damas.
4. 1984: El primer paso hacia tratamientos rigurosos de control de aprendizaje iterativo lo realizaron simultánea e independientemente Arimoto, Casalino y Bartolini y Craig.
5. 1990: Durante esta década el trabajo en el aprendizaje automático pasa de un enfoque basado en el conocimiento a un enfoque basado en datos. Los científicos comienzan a crear programas para computadoras para analizar grandes cantidades de datos y sacar conclusiones, o .ªprender", de los resultados.
6. 1997: Deep Blue de IBM vence al campeón mundial de ajedrez.
7. 2006: Geoffrey Hinton acuña el término .ªprendizaje profundo"para explicar nuevos algoritmos que permiten a las computadoras "verz distinguir objetos y texto en imágenes y vídeos.
8. 2016: el algoritmo de inteligencia artificial de Google supera a un jugador profesional en el juego de mesa chino Go, que se considera el juego de mesa más complejo del mundo y es mucho más difícil que el ajedrez. El algoritmo AlphaGo desarrollado por Google DeepMind logró ganar cinco juegos de cinco en la competencia Go.

9. 2017: Intel - Procesador neuronal Nervana
   Desarrollaron un chip personalizado dedicado a acelerar los cálculos realizados en redes neuronales, llamado Procesador Neural Nervana. Esto es muy similar a la TPU (unidad de procesamiento tensorial) de Google. Las multiplicaciones y convoluciones de matrices son las dos operaciones principales realizadas por el procesador.

A continuación se presentan y discuten un par de trabajos de investigación respecto al control basado en aprendizaje iterativo y aprendizaje de máquina, aplicado a robots manipuladores, que otorgan sustentabilidad a la presente tesis.

1.1. Control de Aprendizaje Iterativo

La primera idea novedosa respecto a Control de Aprendizaje Iterativo se remonta a (Edwards, 1974), desarrollada a pesar de las limitaciones existentes con la aplicación del control clásico. Posteriormente, (Arimoto, 1990) propone un nuevo concepto de control de aprendizaje para la mejora de los movimientos de los robots manipuladores, que puede referirse a una modelación matemática del aprendizaje y la generación de programas motores en el sistema nervioso central.

Se diferencia de las técnicas de control clásicas y modernas convencionales. Representa la repetibilidad del funcionamiento de un determinado sistema robótico y la posibilidad de mejorar la entrada de comandos sobre la base de los datos de medición reales adquiridos en la operación anterior. Por lo tanto, se asumen condiciones adecuadas sobre la repetibilidad y la invariabilidad de la dinámica del sistema, pero no se requiere una descripción precisa de la dinámica para la construcción de los algoritmos de aprendizaje. Se proponen dos tipos de algoritmos de aprendizaje iterativo: uno utiliza una actualización de tipo PD (proporcional y diferencial) de los comandos de entrada y el otro una actualización de tipo PI (proporcional e integral) en la que las señales de velocidad se consideran salidas, en la figura 1.2 se muestra el diagrama a bloques propuesto por Arimoto para el controlador PD.

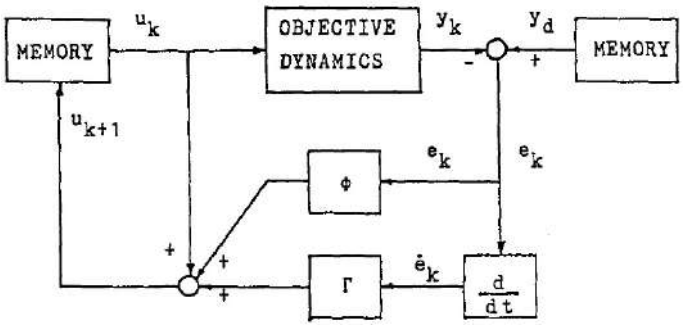


Figura 1.2: Control PD de aprendizaje iterativo (Arimoto, 1986).

Se demuestra que en ambos tipos de control se obtiene un mejor rendimiento en cada prueba de funcionamiento, siempre que se dé un movimiento deseado a priori y se pueda medir el movimiento real (señales de velocidad) en cada operación. Además, se analiza en detalle la robustez de estos algoritmos de control de aprendizaje con respecto a la existencia de errores perturbados de inicialización del robot, perturbaciones y ruido de medición durante la operación (Arimoto, 1990).

Algunas ecuaciones importantes en el desarrollo teórico de convergencia del PD que Arimoto presenta:

- Señal de error

$$\mathbf{e}_k(t) \quad = \quad \mathbf{y}_d - \mathbf{y}_k \tag{1.1}$$

- La siguiente entrada de comandos:

$$\mathbf{u}_{k+1} = F(\mathbf{u}_k(t), \mathbf{e}_k(t)) \tag{1.2}$$

- Señal de control en la k-*ésima* iteración:

$$\mathbf{u}_k = \left[J + H(\mathbf{q}_k)\right] \ddot{\mathbf{q}}_k + \left[B + \dot{H}(\mathbf{q}_)\right] \dot{\mathbf{q}}_k - \frac{\partial T}{\partial \mathbf{q}_k + A(\mathbf{q}_k - \mathbf{q}_d)} \tag{1.3}$$

A partir de estas ecuaciones el autor desarrolla un conjunto de cotas que demuestran la convergencia asintótica del punto de equilibrio, sin embargo, en el capítulo 4 se desarrolla la demostración de convergencia asintótica del punto de equilibrio usando la función estricta de Lyapunov. También se presenta un nuevo diagrama a bloques, representando el lazo cerrado de control que incluye el modelo dinámico del robot manipulador y el controlador PD+, se demuestra convergencia asintótica del punto de equilibrio, con base en lo realizado por Arimoto.

Existen diversas problemáticas entorno al control de robots manipuladores. Una de ellas es desarrollada por (Norouzi, 2019), en la cual se implementa la simulación para un robot manipulador planar de un solo brazo con una plataforma móvil (figura 1.1), donde se realiza una comparación entre las simulaciones obtenidas con los controladores de aprendizaje iterativo difuso del tipo PD (Proporcional-Derivativo), P (Proporcional), las versiones originales PD y P. En este caso se agrega el control difuso para la obtención de las ganancias Proporcional-Derivativa.

1.2. Aprendizaje de máquina

El aprendizaje de máquina es un área particular que surge a partir del concepto y desarrollo de redes neuronales, sin embargo, en éste mismo existen diversas clasificaciones que surgen a partir del aprendizaje de máquina. Y que han sido implementadas para la solución de problemas de control de robots manipuladores.

La figura 1.3 muestra un ejemplo de modelos de aprendizaje: el modelo aprendido se emplea posteriormente para aprender el vuelo en helicóptero invertido.

Figura 1.3: Aplicación de aprendizaje de máquina en el año 2006 (Kober, 2019).

En (Michael Lutter y Peters, 2019) se deriva una topología de red denominada Redes Lagrangianas Profundas (DeLaN por sus siglas en inglés) que codifica la ecuación de Euler-Lagrange originada en la Mecánica Lagrangiana. Esta topología puede ser entrenada utilizando técnicas estándar de optimización de extremo a extremo (end-to-end), manteniendo las características físicas. Por lo tanto, el modelo obtenido debe cumplir con la física del sistema. A diferencia de los enfoques anteriores para el aprendizaje de la física de (Christofer G. Atkeson y Hollerbach, 1986) y (Ledezma y Haddadin, 2017), que diseñaron características fijas a partir de suposiciones físicas que requerían el conocimiento de la encarnación física específica, nosotros ″solo″ hacemos cumplir la física a una red profunda genérica.

En el caso de DeLaN, solo el estado del sistema y la señal de control son específicos del sistema físico, pero no la estructura de la red propuesta ni el procedimiento de entrenamiento. En segundo lugar, se evalúa ampliamente el enfoque propuesto utilizando el modelo para controlar un robot simulado de 2 grados de libertad (gdl) y el robot físico de 7 gdl BarrettWAM en tiempo real. Se demuestra el rendimiento del control de DeLaN, que aprende el modelo dinámico en línea partiendo de una inicialización aleatoria. En comparación con los modelos analíticos y otros modelos aprendidos, DeLaN produce un mejor rendimiento de control y al mismo tiempo extrapola a nuevas trayectorias deseadas (Michael Lutter y Peters, 2019).

En (Mohammad Jafari, 2021) se desarrolla un controlador de retroalimentación en línea basado en el aprendizaje automático (ML) para lograr una respuesta espacio-temporal precisa de los sistemas biológicos mediante la bioelectrónica, impulsada por un algoritmo de aprendizaje adaptativo externo de "detección y respuesta". El controlador de retroalimentación basado en ML propuesto aprovecha la capacidad de aprendizaje y la baja complejidad computacional de una clase de redes neuronales artificiales llamada red de función de base radial (RBF por sus siglas en inglés). Se proporciona un análisis de Lyapunov para para demostrar la estabilidad del controlador propuesto.

El artículo "Learning in Machines"(Oomen, 2018), emplea el algoritmo de control de aprendizaje iterativo definidos por Arimoto, sin embargo, se muestran de forma explicita los diagramas de bloques que representan el aprendizaje iterativo de máquina, el cual en este caso implica aprender de las tareas pasadas. Esto significa que a partir de la primera iteración o la primer tarea realizada se implementa una memoria que tiene acceso a la información anterior y que permite mejorar la señal de salida (reducir el error de salida), además, queda expuesto que el control de aprendizaje iterativo es un mecanismo de retroalimentación.

En (Muhammad Arif e Inooka, 1999) se implementa el control de aprendizaje iterativo utilizando el error como método de predicción. Por lo tanto, el algoritmo garantiza un rendimiento de seguimiento uniforme cuando se aplica al sistema de forma iterativa. Esto implica que el controlador aprende recordando la eficacia de las entradas anteriores y aplica esa información para mejorar la entrada siguiente.

Mientras que en los libros (Jian-Xin Xu y Lee, 2009), (Chen y Moore, 2000), (Ahn, Moore y Chen, 2007) y (Wang, Ye y Zhang, 2014) se tienen diferentes definiciones del control de aprendizaje iterativo con sus respectivos diagramas de bloques y diferentes ejemplos de simulación e implementación práctica que resaltan la importancia de este tipo de control que al compararlas con los esquemas de control clásico hacen evidente una ventaja sustancial, la cual es, que no se requiere de un controlador en lazo cerrado complejo, así como, la demostración correspondiente de convergencia asintótica del punto de equilibrio del sistema implementado.

El control de aprendizaje iterativo y el aprendizaje de máquina son dos métodos diferentes que tienen características comunes, sin embargo, la relación entre ellos no se ha explorado en profundidad. Como es notable en los trabajos mencionados, los métodos aplicados en cada caso van acompañados de controladores conocidos (PD y P). En este tema de tesis se implementará un control para un robot manipulador de 2 grados de libertad que cumpla con una interpretación clara, y se seleccionará el método que a conveniencia otorgue mayor robustez al lazo cerrado de control y que permita una implementación en línea a futuro.

Capítulo 2

Marco teórico

En esta sección se plantean los conceptos básicos teóricos de los robots manipuladores, desde su composición, la definición de sistemas de referencia para obtener la cinemática analítica de Euler hasta la obtención del modelo dinámico a partir del formalismo de Euler-Lagrange y la demostración de pasividad, de igual forma se introducen los conceptos teóricos del aprendizaje de máquina (Machine Learning (ML)) y del control de aprendizaje iterativo (Iterative Learning Control (ILC)) junto a sus correspondientes diagramas de bloques.

2.1. Robot manipulador

Un robot manipulador de 3 grados de libertad esta constituido por las siguientes partes y su analogía con la anatomía humana se muestran en la figura 2.1.

Figura 2.1: Elementos de un robot manipuldor de 3 grados de libertad (gdl).

Cada servomotor representa una articulación (joint en inglés) que interconecta dos eslabones, estas articulaciones permiten el movimiento rotacional (articulaciones rotacionales) o lineal (articulaciones prismáticas) de traslación.

Un eslabón (link en inglés) es una barra metálica (usualmente de aluminio) la cual se encuentra acoplada al rotor y estátor de la articulación siguiente, como se muestra en la figura 2.1.

El robot manipulador de 3gdl que se muestra en la figura 2.1 representa un tipo de robot formado por cadena cinemática abierta, lo que significa que se puede tener un acoplamiento de eslabones uno tras otro de manera acotada, como se muestra en la figura 2.2.

Figura 2.2: Cadena cinemática abierta.

2.2. Cinemática analítica de Euler

- **Cinemática** es la parte de la física que estudia el movimiento de sistemas mecánicos, sin tomar en cuenta las fuerzas que lo originan, lo cual implica que no se involucran ecuaciones diferenciales como es el caso de modelos dinámicos.
- Al estudio de la cinemática aplicado a la robótica se denomina cinemática directa.
- **Cinemática directa** es el estudio analítico del movimiento del robot con respecto a un sistema de referencia cartesiano fijo denominado $\Sigma_0(x_0, y_0, z_0)$ relacionando la dependencia que existe entre las coordenadas articulares o generalizadas $\mathbf{q} \in \mathbb{R}^n$ con las coordenadas cartesianas $\left[x, y, z\right]^T \in \mathbb{R}^3$ y la orientación $\left[\psi, \theta, \phi\right]^T \in \mathbb{R}^3$ del extremo final del robot (en el cual suele incluirse la herramienta de trabajo).
- Esta dependencia se realiza por medio de una función vectorial f_R continua y diferenciable en la variable de estado articular $\mathbf{q}$.

2.2.1. Sistema de referencia inercial

El sistema de referencia inercial define los cuatro cuadrantes del sistema cartesiano, no se deforma ante el movimiento y es fijo o estático; aunque también puede tener movimiento con velocidad constante. Este sistema de referencia permite conocer las mediciones de variables como posición, velocidad y aceleración de una partícula en el espacio con respecto al origen del sistema cartesiano, como se muestra en la figura 2.3.

De acuerdo con la figura 2.3, el vector $\mathbf{r}$ representa la posición de la partícula p, con origen relativo a $\Sigma_0(x_0, y_0, z_0)$ y termina con la partícula p.

El vector $\mathbf{r}$ está dado por:

$$\mathbf{r} = x_0\mathbf{i}_0 + y_0\mathbf{j}_0 + z_0\mathbf{k}_0 \tag{2.1}$$

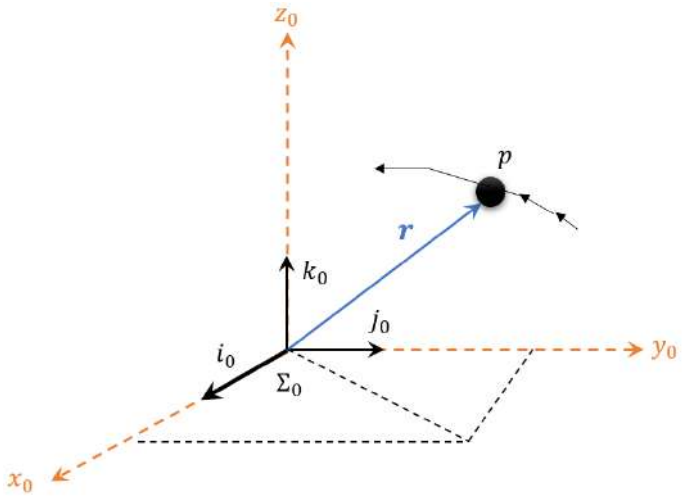


Figura 2.3: Sistema de referencia inercial.

De igual forma se obtiene la norma euclideana del vector **r**:

$$\| \mathbf{r} \| \quad = \quad \sqrt{\mathbf{r}^T \mathbf{r}} = \sqrt{x_0^2 + y_0^2 + z_0^2} \tag{2.2}$$

Visto geométricamente la norma euclidiana es la línea diagonal que une el origen de $\Sigma_0(x_0, y_0, z_0)$ con la partícula p.

2.2.2. Ángulos de Euler

Ángulos de Euler es una regla de composición de rotaciones usando 3 matrices rotacionales:

$$R(\phi, \theta, \psi) \quad = \quad R_z(\phi) R_x(\theta) R_y(\psi) \tag{2.3}$$

donde $R(\phi, \theta, \psi) \in \mathbb{R}^{3\times 3}$.

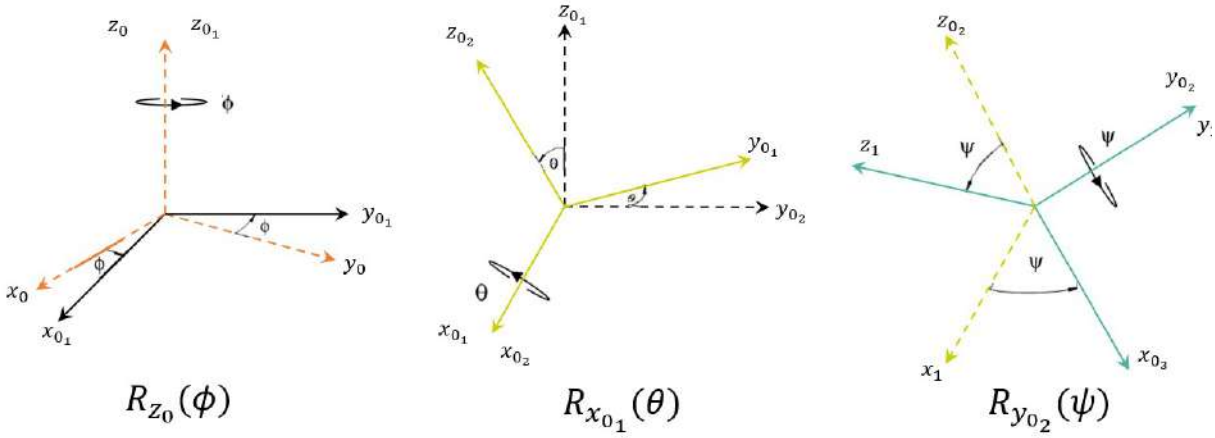


Figura 2.4: Ángulos de Euler.

Como se observa en la figura 2.4, esta composición de tres rotaciones parte de un sistema de referencia fijo y acoplado, que al cabo de tres rotaciones nos lleva al siguiente sistema de referencia fijo y acoplado.

$$\Sigma_0(x_0, y_0, z_0) \rightarrow \Sigma_1(x_1, y_1, z_1) \tag{2.4}$$

$$R_0^1(\phi, \theta, \psi) = R_{z_0}(\phi) R_{x_{0_1}}(\theta) R_{y_{0_2}}(\psi) \tag{2.5}$$

donde:

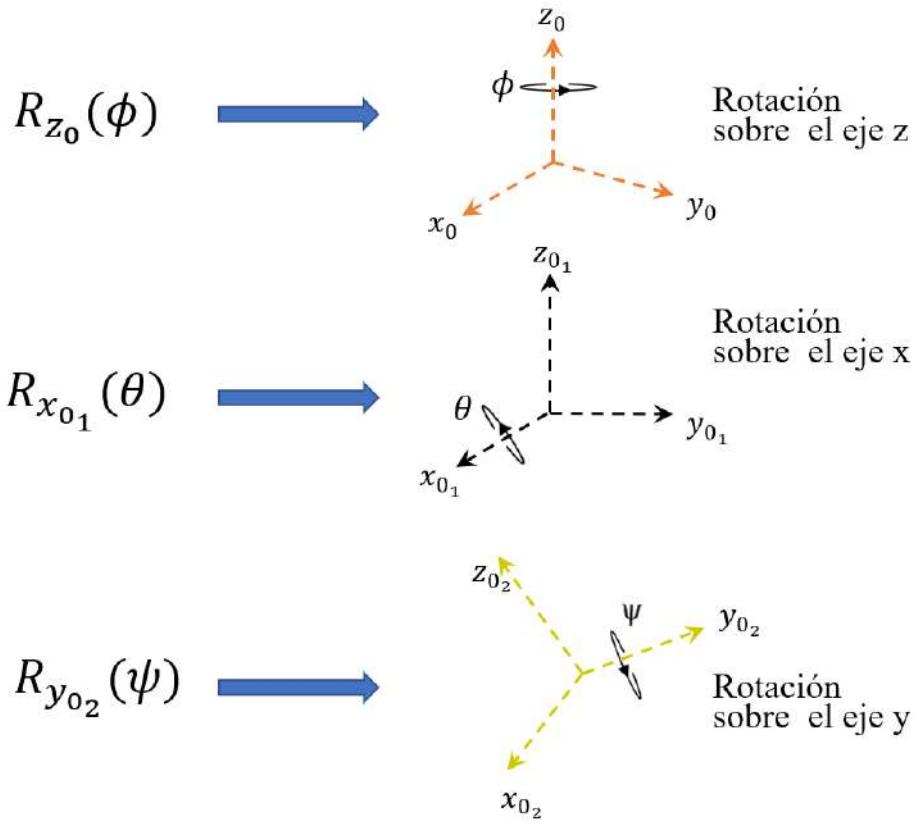


Figura 2.5: Representación general de ejes de rotación y los nuevos sub-sistemas de referencia que se tienen.

En la figura 2.5 se muestra la representación rotacional respecto a cada uno de los ejes que conforman la composición de rotaciones de Euler.

2.2.3. Movimiento de traslación y rotación

En la figura 2.6 se aprecian dos sistemas de referencia $\Sigma_0(x_0, y_0, z_0)$ y $\Sigma_1(x_1, y_1, z_1)$, con orígenes no coincidentes. El origen del sistema de referencia $\Sigma_1(x_1, y_1, z_1)$ se encuentra desplazado una distancia $\mathbf{d}_0^1$ con respecto al origen del sistema $\Sigma_0(x_0, y_0, z_0)$. Dado que el vector $\mathbf{d}_0^1$ esta expresado en coordenadas del sistema Σ_0, es decir: ($\mathbf{d} = \begin{bmatrix} d_{0_x}^1 & d_{0_y}^1 & d_{0_z}^1 \end{bmatrix}^T$), de tal forma, que cualquier punto $\mathbf{p}$ tendrá representaciones $\mathbf{p}_0$ y $\mathbf{p}_1$. La relación general entre los sistemas de referencias $\Sigma_0(x_0, y_0, z_0)$ y $\Sigma_1(x_1, y_1, z_1)$ incluyendo la matriz de rotación $R_{z,\theta}$ y el vector de traslación $\mathbf{d}_0^1$ es:

$$\mathbf{p}_0 = \mathbf{d}_0^1 + R_0^1(\phi, \theta, \psi) = \mathbf{d}_0^1 + R_{z_0}(q_1) R_{x_0}(\alpha_1) R_{y_0}(0) \mathbf{p}_1 \tag{2.6}$$

$$\begin{bmatrix} p_{0_x} \\ p_{0_y} \\ p_{0_z} \end{bmatrix} = \begin{bmatrix} d_{0_x}^1 \\ d_{0_y}^1 \\ d_{0_z}^1 \end{bmatrix} + R_{z_0}(q_1) R_{x_0}(\alpha_1) \mathtt{I} \begin{bmatrix} p_{1_x} \\ p_{1_y} \\ p_{1_z} \end{bmatrix} \tag{2.7}$$

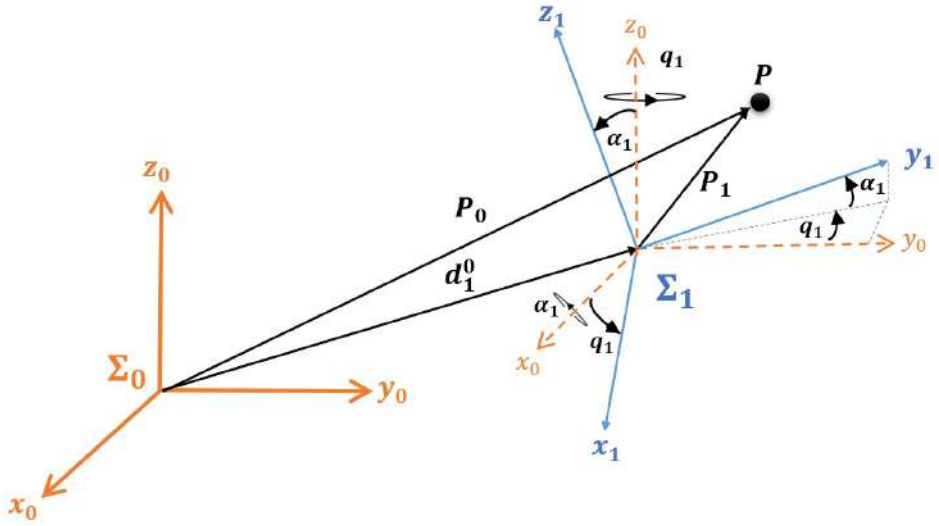


Figura 2.6: Movimientos de traslación y orientación.

2.2.4. Matrices de transformación homogénea

De manera general, un robot de n grados de libertad esta conformado por eslabones unidos por n articulaciones (cadena cinemática abierta), de tal forma que a cada par articulación-eslabón le corresponde un grado de libertad y por consiguiente al eslabón se le asigna un sistema de referencia inercial fijo. Que como ya se ha comentado anteriormente involucran sistemas coordenados cartesianos que especifican posición y orientación del extremo final del robot.

La matriz de transformación homogénea es una herramienta matemática que involucra operaciones de rotación y traslación estructurando el modelo de cinemática directa. Además de que brinda una notación más compacta de las matrices de rotación y el vector de traslación. Su expresión se muestra a continuación:

$$\begin{aligned} H &= \begin{bmatrix} R_{z,\theta} & \mathbf{d}_0^1 \\ \mathbf{0}^T & 1 \end{bmatrix} \\ &= \begin{bmatrix} \texttt{Matriz de rotación} & \vdots & \texttt{Vector de traslación} \\ \cdots & \vdots & \cdots \\ \mathbf{0}^T & & 1 \end{bmatrix} \end{aligned} \tag{2.8}$$

donde $R_{z,\theta} \in SO(3)$ (grupo de rotación tridimensional) es una matriz ortogonal y $\mathbf{d}_0^1 \in \mathbb{R}^3$. El vector $\mathbf{0}^T$ y el número 1 aparecen en el último renglón para propósitos de acoplamiento de dimensiones.

2.3. Cinemática directa

- **Cinemática directa** es una función vectorial $f_R(l_i, \mathbf{q})$ que relaciona las coordenadas articulares $\mathbf{q} \in \mathbb{R}^n$ y propiedades geométricas del sistema mecánico l_i con las coordenadas cartesianas $[x, y, x]^T \in \mathbb{R}^3$ del robot y la orientación $[\psi, \theta, \phi]^T \in \mathbb{R}^3$ de la herramienta de trabajo colocada en el extremo final (Reyes, 2011).

- La cinemática directa es un mapeo vectorial $\mathbf{f}_R : \mathbb{R}^n \rightarrow \mathbb{R}^m$ tal que:

$$\begin{bmatrix} x \\ y \\ z \\ \psi \\ \theta \\ \phi \end{bmatrix} = \mathbf{f}_R(l_i, \mathbf{q}) \tag{2.9}$$

- La posición del extremo final del robot manipulador en el espacio tridimencional (*pose*) precisa de 6 coordenadas ($m = 6$); 3 coordenadas que representan la posición cartesiana y 3 coordenadas que representan la orientación de la herramienta de trabajo.

- El empleo de la cinemática directa resulta de utilidad en la planificación de trayectorias y en el control cartesiano. El papel fundamental de la cinemática directa, es establecer la posición y orientación del extremo final del robot manipulador como una función de las variables articulares.

donde n representa el número de grados de libertad y la dimensión del vector de coordenadas articulares $\mathbf{q}$, m es la dimensión conjunta de las coordenadas cartesianas y la orientación de la herramienta de trabajo.

2.3.1. Convención Denavit-Hartenberg

El método de Denavit-Hartenberg es una herramienta ampliamente conocida en el área de ingeniería, ya que ofrece un procedimiento sencillo para obtener el modelo cinemático directo cuya estructura queda en términos de las transformaciones homogéneas (Reyes, 2011).

El algoritmo a seguir se describe a continuación:

1. Observar y localizar la dirección de colocación de cada tipo de motor para definir los ejes $z_0, z_1, \cdots, z_{n-1}$.

2. Establecer en la base del robot manipulador el sistema de referencia cartesiano fijo $\Sigma_0(x_0, y_0, z_0)$ cuyo origen es colocado sobre el sistema de referencia en la base del robot. Los ejes x_0, y_0 son determinados de acuerdo con la regla de la mano derecha.

3. Definir el sistema de refencia $\Sigma_i(x_i, y_i, z_i)$ que conlleva a un proceso iterativo usando el sistema de referencia $\Sigma_{i-1}(x_{i-1}, y_{i-1}, z_{i-1})$, iniciando con el sistema de referencia $\Sigma_1(x_1, y_1, z_1)$. A continuacion, se llevan a cabo los pasos 3 a 5 para las articulaciones $i = 1, \cdots, n-1$.

4. Definir el origen o_i en la intersección de la normal común, la cual, une al eje z_i con el eje z_{i-1}. Si el eje z_i intercepta al eje z_{i-1} colocar o_i en la intercepción. Para el caso en que los ejes z_i y z_{i-1} son paralelos:

   - Si la articulación i-ésima es rotacional, colocar el origen o_i sobre la articulación i-ésima, tal que $d_i = 0$.
   - Si la articulación i-ésima es prismática, colocar el origen o_i en un punto límite físico de la articulación i-ésima, por ejemplo en un punto extremo.

5. Seleccionar el eje x_i en toda la longitud de la norma común que une a los ejes z_{i-1} y z_i, en dirección de la articulación $i-1$ hacia la articulación i.

6. Determinar y_i empleando la regla de la mano derecha.
7. Establecer el sistema de referencia del extremo final $\Sigma_n(x_n, y_n, z_n)$.
   - Si la articulación n-ésima es rotatoria, entonces alinear el eje z_n con el eje z_{n-1}.
   - Si la articulación n-ésima es prismática, entonces seleccionar el eje z_n de forma arbitraria. El eje x_n debe cumplir el paso 5.
8. Obtener la tabla 2.1 de parámetros para cada uno de los eslabones.
9. Establecer las matrices de transformación homogéneas: H_{i-1}^{i} para $i = 1, 2, \cdots, n-1$.

Tabla de Denavit-Hartenberg

Una vez que se han establecido las características anteriores, se puede obtener la siguiente tabla representativa de Denavit-Hartenberg (Reyes, 2011):

Características de eslabones	
l_i	Longitud del eslabón i-ésimo
d_i	Articulaciones lineales o prismáticas. también representa el espesor del servomotor (β_i)
α_i	Ángulo entre los ejes z_{i-1} y z_i medido con respecto al eje x_i.
θ_i	articulaciones rotacionales; representa el ángulo entre los ejes x_{i-1} y x_i medido alrededor del eje z_{i-1}.

Tabla 2.1: Parámetros Denavit-Hartenberg.

donde:

1. l_i representa la longitud del i-ésimo eslabón, distancia del eje z_{i-1} hacia el eje z_i medida sobre el eje x_{i-1}.
2. α_i representa el ángulo de torsión entre los ejes z_{i-1} a z_i medido en el sentido de las manecillas del reloj sobre el eje x_i.
3. d_i se emplea en articulaciones lineales o prismáticas y representa el desplazamiento lineal. Cuando la articulación es rotacional, entonces representa el $offset$ o espesor del servomotor (la distancia de x_{i-1} a x_i medido sobre el eje z_{i-1}), se denota por β_i.
4. θ_i representa el desplazamiento rotacional de x_{i-1} a x_i medido alrededor del eje z_{i-1}. El signo positivo de θ_i es el sentido contrario a las manecillas del reloj.

Representación Denavit-Hartenberg con transformaciones homogéneas H_{i-1}^{i}

La representación de Denavit-Hartenberg a través del producto de cuatro transformaciones básicas es:

$$H_{i-1}^{i} = H_{R_{z_{i-1}}}(\theta_i) H_{T_{z_{i-1}}}(d_i;\beta_i) H_{T_{x_{i-1}}}(l_i) H_{R_{x_{i-1}}}(\alpha_i) \tag{2.10}$$

$$= \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i(\beta_i) \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & l_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.11}$$

$$= \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & l_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) & l_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i(\beta_i) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.12}$$

La transformación homogénea total se obtiene como $H_0^n = H_0^1 H_1^2 \cdots H_{n-2}^{n-1} H_{n-1}^n$. La cinemática directa es la forma general de transformaciones homogéneas que concatena los sistemas de referencia cartesianos asociados a los eslabones del robot, todos relativos al sistema de referencia fijo Σ_0.

2.3.2. Modelo dinámico

El modelo dinámico de un robot manipulador de n grados de libertad, formado con eslabones rígidos conectados por articulaciones libres de elasticidad en cadena cinemática abierta esta representado por (Reyes, 2011):

$$\boldsymbol{\tau} = M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{f}_f(\dot{\mathbf{q}},\boldsymbol{f}_e) \tag{2.13}$$

donde:

- $\mathbf{q} \in \mathbb{R}^n$, es el vector de coordenadas generalizadas o posiciones articulares.
- $\dot{\mathbf{q}} \in \mathbb{R}^n$ es el vector de velocidades articulares.
- $\ddot{\mathbf{q}} \in \mathbb{R}^n$ es el vector de aceleraciones articulares.
- $M(\mathbf{q}) \in \mathbb{R}^{n\times n}$ es la matriz de inercia, la cual es simétrica y definida positiva.
- $C(\mathbf{q},\dot{\mathbf{q}}) \in \mathbb{R}^{n\times n}$ es la matriz de fuerzas centrípetas y de Coriolis.
- $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ es el vector de fuerzas o pares gravitacionales obtenido como el gradiente de la energía potencial, es decir:

$$\mathbf{g}(\mathbf{q}) = \frac{\partial U(\mathbf{q})}{\partial \mathbf{q}} \tag{2.14}$$

- $\boldsymbol{f}_f(\dot{\mathbf{q}},\boldsymbol{f}_e,\tau) \in \mathbb{R}^n$ es el vector de pares de fricción que incluye la fricción viscosa, de Coulomb y estática $(\boldsymbol{f}_e)$ de cada articulación del servomecanismo.

El efecto inercial representado por $M(\mathbf{q})$ significa el cambio de estado de movimiento del robot manipulador (Reyes, 2011).

Metodología para obtener el modelo dinámico

1. Determinar la cinemática directa del centro de masa de cada uno de los eslabones. Tomar en cuenta las longitudes anteriores (l_{i-1}, q_1 y q_{i-1})en el i-ésimo eslabón (Reyes, 2011):

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \boldsymbol{f}_R(l_i, l_{i-1}, l_{ci}, q_{i-1}, q_i) \tag{2.15}$$

2. Calcular la cinemática diferencial del i-ésimo eslabón y deducir la rapidez lineal:

$$\mathbf{v}_i = \frac{d}{dt}\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \tag{2.16}$$

   La ecuación (2.17) representa la rapidez lineal del centro de masa de cada eslabón:

$$\mathbf{v}_i^T \mathbf{v}_i = \dot{x}_i^2 + \dot{y}_i^2 + \dot{z}_i^2 \tag{2.17}$$

3. Obtener el modelo de energía:

$$\mathcal{H}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) + \mathcal{U}(\mathbf{q}) \tag{2.18}$$

   - El movimiento de rotación y traslación son representadas por la energía cinética $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$:

$$\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} m_i \mathbf{v}_i^T \mathbf{v}_i + \frac{1}{2} I_i \left[\sum_i^n \dot{q}_i\right]^2 \tag{2.19}$$

     donde I_i es el momento de inercia del i-ésimo eslabón.
   - La energía potencial $\mathcal{U}(\mathbf{q})$ es un caso diferente al de la energía cinética, depende de la geometría del servomecanismo en general $\mathcal{U}_i(\mathbf{q}) = m_i g l_{ci} h_i(\mathbf{q})$, siendo $h_i(\mathbf{q})$ una función que representa la altura del eslabón con respecto al origen del sistema de referencia del servomecanismo (Reyes, 2011).
   - Obtener el lagrangiano: $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{U}(\mathbf{q})$.

4. Ecuaciones de movimiento de Euler-Lagrange. Se aplican las ecuaciones de movimiento de Euler-Lagrange y se incluye el modelo de fricción.

$$\tau_i = \frac{d}{dt}\left[\frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{q}_i}\right] - \frac{\partial \mathcal{L}\mathbf{q}, \dot{\mathbf{q}}}{\partial q_i} + b_i \dot{q}_i + f_{ci} signo(\dot{q}_i) + f_{ei}\left[1 - |signo(\dot{q}_i)|\right] \tag{2.20}$$

2.4. Pasividad de robots manipuladores

El análisis y estudio de sistemas dinámicos requiere del concepto de disipación de energía, ya que, al representar el sistema dinámico en variables de estados las propiedades de almacenar energía y disiparla son clave en la teoría de estabilidad de Lyapunov. Un sistema pasivo no puede almacenar más energía que la energía aplicada con respecto a la energía disipada (Reyes, 2011; Tahirovic y Magnani, 2013).

Estabilidad $\mathcal{L}_2$:

Se considera la siguiente notación para representar un sistema dinámico:

$$\Sigma : \begin{cases} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, u),\ \mathbf{x}(0) \in \mathbb{R}^n \\ y &= h(\mathbf{x}, u) \end{cases} \tag{2.21}$$

donde $\mathbf{x} \in \mathbb{R}^n$ = vector de estado
$\dot{\mathbf{x}} \in \mathbb{R}^n$ = variacion temporal del estado $\dot{\mathrm{x}}(\mathrm{t})$
$\mathbf{f} \in \mathbb{R}^n$ = mapa (vectorial que define el sistema dinámico)
$y \in \mathbb{R}$ = respuesta del sistema
$u \in \mathbb{R}$ = entrada del sistema

El sistema Σ es pasivo, si existe una función continuamente diferenciable semidefinida positiva (función de almacenamiento de energía) $V(x)$ tal que:

$$uy \geq \dot{V}(x) \tag{2.22}$$

$$\int_0^t u(\sigma)y(\sigma)d\sigma = V(x(t)) - V(x(0)) \geq -\beta \tag{2.23}$$

donde $\beta \in \mathbb{R}_+$ es alguna constante positiva, es estable $\mathcal{L}_2$.

La pasividad de robots manipuladores cumple (Reyes, 2011):

$$\underbrace{\int_0^t \dot{\mathbf{q}}^T \boldsymbol{\tau}}_{\texttt{Energía total}} = \underbrace{H(\mathbf{q}, \dot{\mathbf{q}})}_{\mathcal{K}+\mathcal{U}} - \underbrace{H(\mathbf{0}, \dot{\mathbf{0}})}_{\mathcal{K}(0)+\mathcal{U}(0)} + \underbrace{\int_0^t B\dot{\mathbf{q}}^2(\sigma)d\sigma + \int_0^t F_c|\dot{\mathbf{q}}(\sigma)|d\sigma}_{\texttt{Energía } disipativa} \tag{2.24}$$

donde la energía aplicada $\int_0^t \dot{\mathbf{q}}^T \boldsymbol{\tau}$ es la energía total del péndulo siendo igual a la función de almacenamiento de energía $\mathcal{H}(t)$, que representa la suma de la energía cinética y potencial, sumado a la energía disipada. La proporción de suministro $\dot{\mathbf{q}}^T \boldsymbol{\tau}$ es la potencia entregada, mientra que la potencia disipada es $B\dot{\mathbf{q}}^2 + F_c|\dot{\mathbf{q}}|$. Se considera que la entrada u es $\boldsymbol{\tau}(t)$ y que la salida y es $\dot{\mathbf{q}}(t)$. Por lo que el sistema Σ: $u \to y$ es pasivo y $\mathcal{L}_2$ estable.

2.5. Aprendizaje de máquina (Machine Learning)

Debido a la creciente complejidad de los sistemas robóticos modernos y el requerimiento de que los robots trabajen en ambientes no estructurados, se ha incrementado notablemente el interés de la aplicación de métodos basados en aprendizaje iterativo y/o de máquina e inteligencia artificial en los problemas de control de robots manipuladores (Kober, 2019).

Arquitectura básica

La arquitectura de control del aprendizaje de máquina se muestra en la figura 2.7. El procedimiento implica tener una tarea de control bien definida que se formula en términos de ***minimizar/maximizar*** una función de costo J que puede evaluarse en función de los resultados medidos del sistema, z. A continuación, el responsable del tratamiento debe tener una representación general y flexible para que se pueda implementar un algoritmo de aprendizaje de máquina en el espacio de los posibles controladores. Finalmente, se

elegirá el algoritmo de aprendizaje automático para descubrir la ley de control más adecuada. a través de algún procedimiento de entrenamiento que involucre datos de experimentos o simulaciones (Duriez, Brunton y Noack, 2017).

De forma específica el diagrama de retroalimentación mostrado en la figura 2.7 esta compuesto por:

- Planta. Sistema dinámico (Physical system) lineal o no lineal.
- Señales de perturbación (w), que afectan o cambian la dinámica del sistema.
- La función de costo (J), donde se maximiza o minimiza lo requerido, es la parte final de la optimización deseada.
- Controlador (Machine learning control), la idea es diseñar un controlador utilizando métodos basados en datos, mediciones dadas del sistema por sensores (s), que pueda impulsar el sistema a través de la actuación (b) y en función de su control.

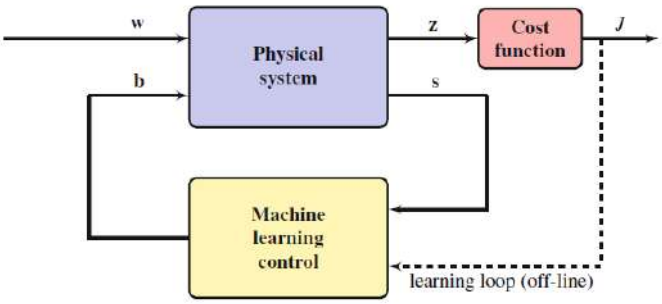


Figura 2.7: Diagrama de bloques del control de aprendizaje de máquina (Duriez, Brunton y Noack, 2017).

Tipos de aprendizaje de máquina (Machine Learning)

El aprendizaje de robots manipuladores nace de la aplicación conjunta entre el aprendizaje de máquina y técnicas de inteligencia artificial para los sistemas y problemas de control en robótica. Éste se enfoca en control y planificación de bajo nivel y alto nivel, percepción y fusión de sensores, así como modelos del robot y su entorno (Kober, 2019).

Existen diferentes tipos de aprendizaje de máquina, los cuales son:

Aprendizaje supervisado: Se asume que el aprendiz tiene acceso a muestras de combinaciones de entrada y salida, deben ser reproducidas y para generalizar el mapeo se necesita interpolar y extrapolar las muestras. La parte complicada aquí es encontrar una representación compacta del mapeo que solo modele partes relevantes y no detalles innecesarios. Por lo tanto, el criterio de optimización es una combinación de ajustar bien los datos observados y fomentar modelos simples.

Aprendizaje sin supervisión: El aprendiz solo tiene acceso a los datos de entrada. La tarea es descubrir estructuras en los datos que permitan representaciones de menor dimensión, donde el objetivo es aprender modelos que generen datos falsos para distribuirlos lo más similar posible a los datos reales.

Aprendizaje reforzado: No opera en un conjunto de datos fijo, sino que aprende el mapeo óptimo de entradas y salidas a través de interacciones con el sistema. Este mapeo es un controlador, llamado norma, mapeando desde estados a acciones óptimas. Considera que la dinámica del sistema subyacente y las funciones de recompensa son desconocidas y, por lo tanto, el agente necesita aprenderlas implícitamente a través de interacciones con el sistema.

Combinaciones: Aprendizaje semi-supervisado donde solo algunos de los puntos de datos están etiquetados y las etiquetas de los otros puntos deben inferirse de la estructura de los datos.

2.5.1. Perceptron

La red feed-forward de una sola capa consta de una o más neuronas de salida, cada una de las cuales está conectada con un factor de ponderación w_{io} a todas las entradas i. En el caso más sencillo, la red tiene sólo dos entradas y una única salida, como se dibuja en la figura 2.8.

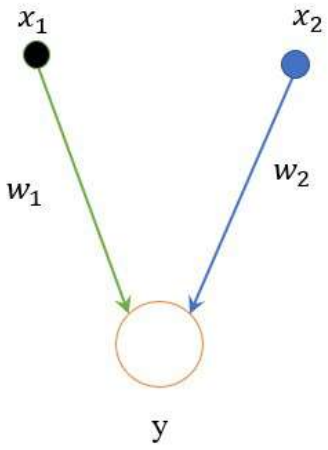


Figura 2.8: Red neuronal de una sola capa.

La entrada de la neurona es la suma ponderada de las entradas. La salida de la red está formada por la activación de la neurona de salida, que es una función de la entrada (Kröse y col., 1993):

$$y = \mathcal{F}\left(\sum_{i=1}^{2} w_i x_i\right) \tag{2.25}$$

La función de activación $\mathcal{F}$ puede ser lineal o no lineal, por lo que seria una red lineal o no lineal respectivamente.

Identificación paramétrica La identificación paramétrica es una de las herramientas más usadas para determinar los parámetros dinámicos de los robots manipuladores, sin embargo, ésta es expresada como el producto de una matriz de regresión compuesta la mayoría de las veces de funciones no lineales (dependientes de posición, velocidad y aceleración) y un vector de parámetros constantes (como masas, longitudes, momentos de inercia, etc) (Armstrong, Khatib y Burdick, 1986; Goodwin y Sin, 2009).

Es por tal motivo que uno de los métodos para determinar los pesos w_{io} del Perceptron, será a partir de la identificación paramétrica. Un ejemplo, se muestra a continuación:

$$y = x_1 w_1 + x_2 w_2 = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \tag{2.26}$$

donde $\theta_1 = w_1$ y $\theta_2 = w_2$, son los pesos a encontrar.

2.6. Control de Aprendizaje Iterativo

Diversos autores han aportado una definición al concepto de ILC, como el descrito en el capítulo 1 por Suguru Arimoto. A continuación se citan un conjunto de definiciones para ILC de diversos autores:

- Es un método de control recursivo en línea que se basa en menos cálculos y requiere menos conocimiento a priori sobre la dinámica del sistema. La idea es aplicar un algoritmo simple

repetidamente a una planta desconocida, hasta que se logre un seguimiento perfecto (Bien y Huh, 1989).

- El control de aprendizaje iterativo es un enfoque para mejorar la respuesta transitoria del sistema que opera repetidamente durante un intervalo de tiempo fijo (Moore, 1993).
- El control por aprendizaje iterativo considera sistemas que realizan repetidamente la misma tarea con el fin de mejorar secuencialmente la precisión (N. Amann y Rogers, 1996).
- El controlador que aprende a producir un error de seguimiento cero durante las repeticiones de un comando o aprende a eliminar los efectos de una perturbación repetitiva en la salida de un sistema de control (M. Q. Phan y Moore, 2000).

Se han propuesto varios métodos de diseño de ILC. A grandes rasgos, pueden clasificarse en dos categorías, el diseño de ILC sin modelo y el diseño de ILC basado en modelo. El diseño de ILC sin modelo aprende utilizando solo la entrada y el error anteriores, ejemplo de ILC de tipo Proporcional-Integral-Derivativo (PID) impulsado por datos (Bolder y T., 2018).

El diseño de ILC basado en modelos, utiliza alguna información explícita del modelo (no necesariamente precisa). En general, debido al uso de la información del modelo, el ILC basado en el modelo tiende a tener un mejor rendimiento de convergencia en comparación con el diseño sin modelo. Más adelante se dan los detalles de ambos tipos de controles.

Para entender el control de aprendizaje iterativo (ILC) es importante conocer las siguientes definiciones del control clásico ():

Feedback

El término retroalimentación (feedback en inglés) se refiere a una situación en la que dos (o más) sistemas dinámicos están conectados entre sí de tal manera que cada sistema influye en el otro y su dinámica está fuertemente acoplada. Todos los ejemplos citados hasta ahora se refieren a la retroalimentación negativa, en la que intentamos reaccionar a las perturbaciones de manera que sus efectos disminuyan. La retroalimentación positiva es lo contrario, cuando el aumento de alguna variable o señal conduce a una situación en la que en la que esa cantidad se incrementa aún más a través de la retroalimentación, como se muestra en la figura 2.9a).

Feedforward

La retroalimentación es reactiva: debe haber un error antes de que se tomen medidas correctivas. Sin embargo, en algunas circunstancias es posible medir una perturbación antes de que entre en el sistema, y esta información puede utilizarse para tomar medidas correctoras antes de que la perturbación haya influido en el sistema. El efecto de la perturbación se reduce así midiéndola y generando una señal de control que la contrarreste. Esta forma de controlar un sistema se denomina prealimentación (feedforward en inglés), como se muestra en la figura 2.9b).

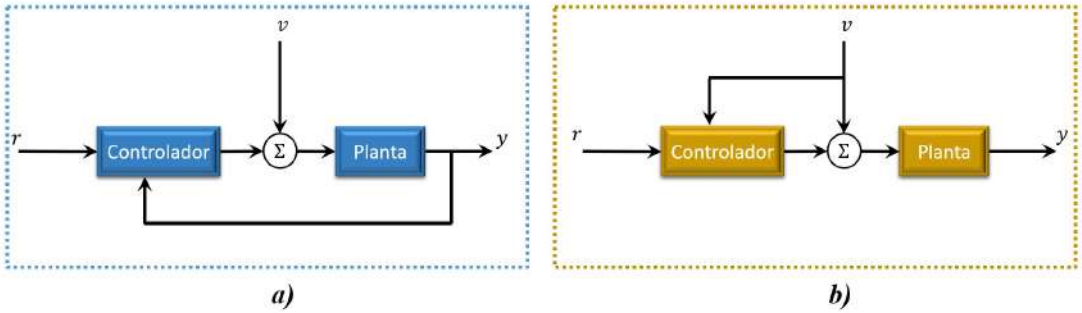


Figura 2.9: Sistema de retroalimentación frente a sistema de alimentación. En ambos sistemas tenemos la planta o proceso y un controlador C. El controlador de retroalimentación $a)$ mide la salida y para determinar el efecto de la perturbación v, mientras que el controlador de avance $b)$ mide la perturbación directamente, pero no mide directamente la salida del proceso.

2.6.1. Procesos repetitivos

También denominados procesos multipasos/multitareas en la literatura temprana, que se caracterizan por una serie de barridos, denominados tareas, a través del conjunto de dinámicas donde la duración, o longitud, de cada tarea es finita. En cada prueba se produce una salida que actúa como función de forzamiento y contribuye a la siguiente salida. El concepto de proceso repetitivo se introdujo por primera vez a principios de los años 70 para obtener el modelo de control en las operaciones de corte de carbón y laminación de metales, que son dos aplicaciones típicas en la industria.

En la figura 2.10 se muestra un ejemplo en diagrama de bloques de la aplicación de aprendizaje iterativo a partir de tareas pasadas, donde a partir de la primera tarea la señal de salida se ve afectada.

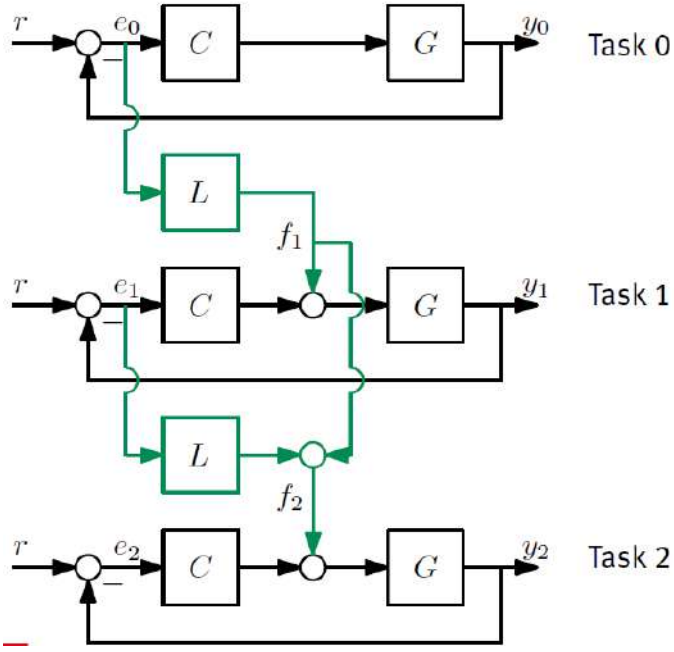


Figura 2.10: Diagrama de bloques del proceso repetitivo (de tareas repetitivas) (Oomen, 2018).

Se aplica en un sistema de impresión, que contiene un encoder, el cual nos permite conocer la posición del carril de impresión, obtener el error y a partir del proceso repetitivo reducir el error de posición, como se muestra en la figura 2.11.

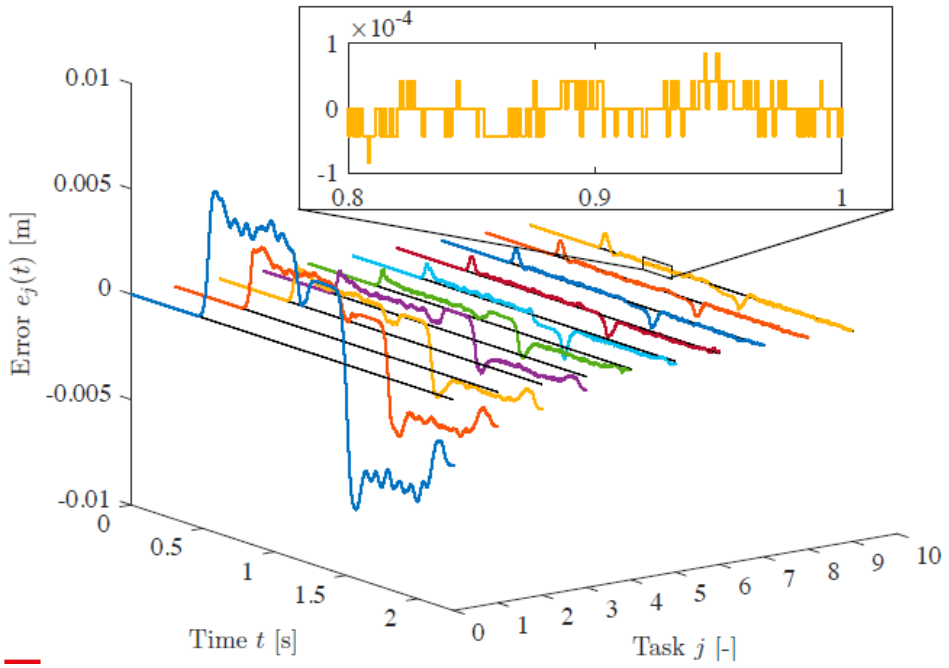


Figura 2.11: Ejemplo de error de convergencia para el aprendizaje a partir de tareas pasadas en un sistema de impresión, la cual tiene una rápida convergencia (Oomen, 2018).

2.6.2. Enfoque original del Control de Aprendizaje Iterativo

Retomando la representación del lazo cerrado de control de Arimoto, figura 1.2, se colocan los bloques como se muestra en la figura 2.12, la cual, representa el enfoque original del control de aprendizaje iterativo implementado únicamente con la señal de prealimentación (feedforward), con entradas de control actualizadas por el error anterior, mientras que la salida de control puede ser modificada por un algoritmo de control del tipo P o D (Cai, 2009; Saab y col., 2021).

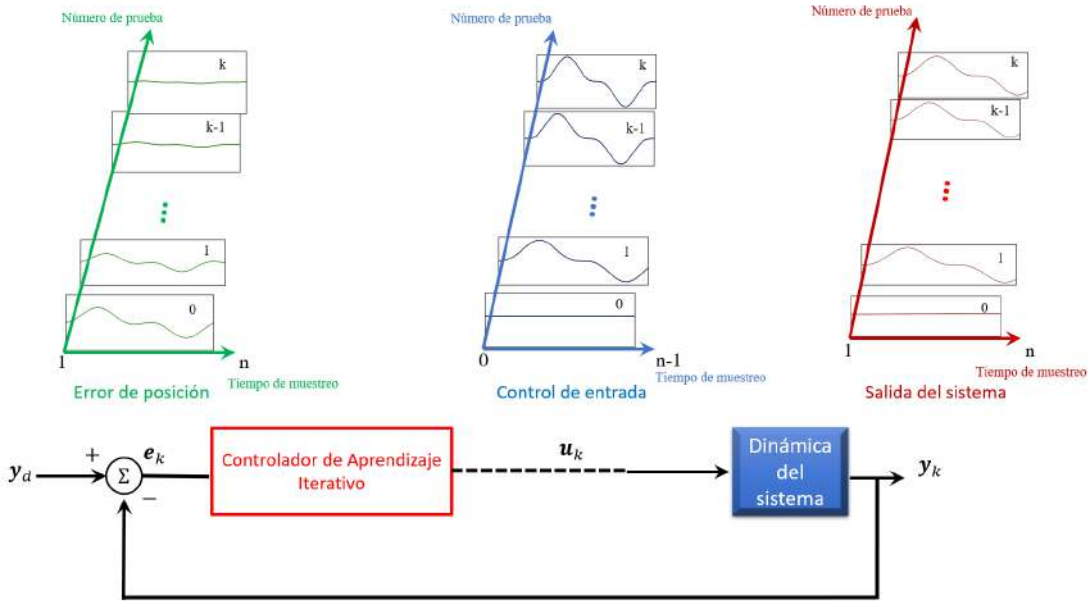


Figura 2.12: Diagrama a bloques del control de aprendizaje iterativo y el progreso de aprendizaje de las señales relacionadas.

Se definen los siguientes conceptos (Arimoto, 1986):

- Cada operación termina en un tiempo fijo acotado $0 \leq t \leq t_f$.
- Señal deseada (posición deseada) para un tiempo finito.

$$\mathbf{y}_d \tag{2.27}$$

- Se satisface la repetibilidad de ajuste inicial, el estado inicial $\mathbf{x}_k(0)$ del sistema objetivo puede fijarse igual al principio de cada operación de la siguiente manera

$$\mathbf{x}_k(0) \quad = \quad \mathbf{x}^0 \texttt{ for } k = 1, 2, \ldots \tag{2.28}$$

  donde k denota el número de iteración de la operación.

- Señal de error

$$e_k(t) \quad = \quad y_d - y_k \tag{2.29}$$

- Derivada de la señal de error

$$\frac{d}{dt}e_k(t) \quad = \quad \frac{d}{dt}\left(y_d - y_k\right) \tag{2.30}$$

$$\tag{2.31}$$

- Algoritmo de control:

$$u_{k+1} \quad = \quad F(u_k(t), e_k(t)) \tag{2.32}$$

- Control PD de aprendizaje iterativo:

$$u_{k+1}(t) \quad = \quad u_k(t) + \left(\Gamma\tfrac{d}{dt} + \Phi\right)\left[y_d(t) - y_k(t)\right] \tag{2.33}$$

- Dinámica de los robot manipuladores:

$$\left[J + H(\mathbf{q})\right]\ddot{\mathbf{q}} + \left[B_0 + \dot{H}(\mathbf{q})\right]\dot{\mathbf{q}} - \frac{\partial T}{\partial \mathbf{q}} + \mathbf{g}(\mathbf{q}) \quad = \quad K\mathbf{v} \tag{2.34}$$

  donde el autor (Arimoto, 1986) define $J + H(\mathbf{q})$ como las matrices de inercia de los actuadores y $\mathbf{v}$ como la entrada de control.

En la figura 2.13 se muestra el diagrama a bloques del control de aprendizaje iterativo que Arimoto propone (Arimoto, 1986).

Este puede ser llevado a una forma más simple donde se usa el error de la señal con una ganancia proporcional, mejor conocido como control de aprendizaje iterativo del tipo P (proporcional) como se muestra en la figura 2.14.

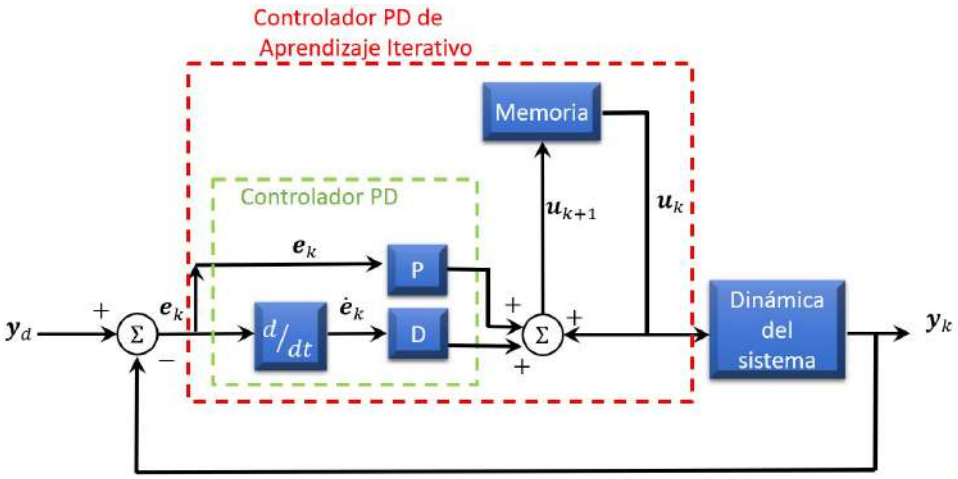


Figura 2.13: Diagrama de bloque Control de Aprendizaje Iterativo PD.

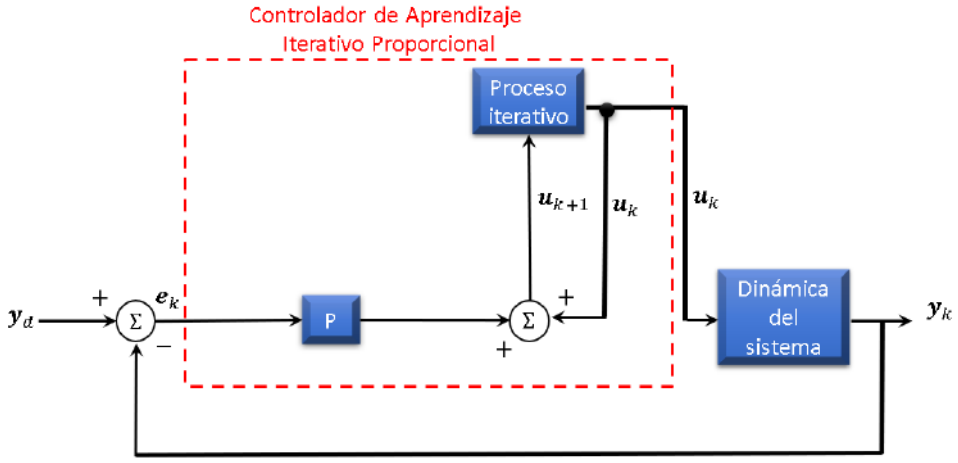


Figura 2.14: Diagrama de bloques del control de aprendizaje iterativo del tipo P (proporcional).

El algoritmo de control queda definido como:

$$u_{k+1} = u_k + Pe_k \tag{2.35}$$

En la TABLA 2.2 se muestran ejemplos sobre tipos de control de aprendizaje iterativo.

Control de aprendizaje iterativo	
Algoritmo	Ecuaciones
Tipo-D	$u_{k+1}(t) = u_k(t) + L\dot{e}_k(t)$
Tipo-P	$u_{k+1}(t) = u_k(t) + Le_k(t)$
Tipo-PID	$u_{k+1} = u_k(t) + K_p e_k(t) + K_i \int e_k(t)dt + K_d \dot{e}_k(t)$

Tabla 2.2: Tipos de algoritmos de control de aprendizaje iterativo.

2.6.3. Control de Aprendizaje Iterativo con Control de Retroalimentación

En la figura 2.15 se muestra la estructura del controlador que utiliza enfoques hibrídos. La disposición paralela de un controlador de retroalimentación PID (en este ejemplo) y un controlador de aprendizaje iterativo, presentando una mejora de compensamiento a los cambios repentinos en la dinámica del sistema (planta). Además, permite un mejor rendimiento de seguimiento para las pruebas iniciales. Cuando se activa el PID, se implementa un esquema de retroalimentación/adelanto.

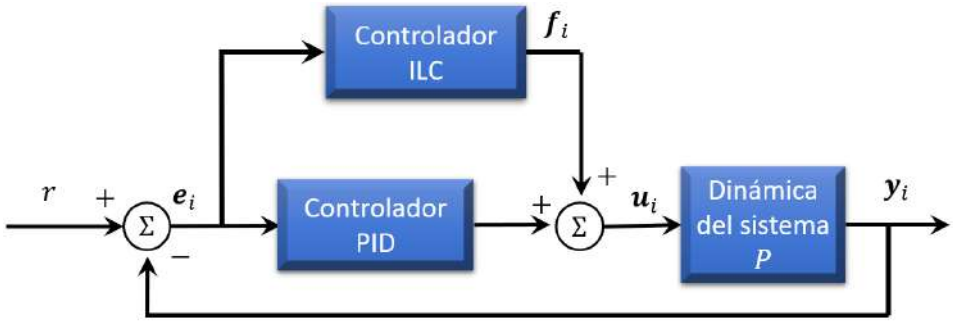


Figura 2.15: Diagrama de bloques del controlador híbrido.

Para f_k la cual es la señal de adelanto del control de aprendizaje iterativo, se tiene que el algoritmo tipo-D esta dado por:

$$f_{k+1}(t) = f_k(t) + K_k\left[e_k(t+1) - e_k(t)\right] \tag{2.36}$$

y para el control del tipo-P se tiene:

$$f_{k+1}(t) = f_k(t) + K_k e_k(t+1) \tag{2.37}$$

De tal forma que el algoritmo de control es:

$$u_k(t) = f_k(t) + \left[K_p e_k(t) + K_i \int_0^t e_k(\sigma)d\sigma + K_d \frac{de_k(t)}{dt}\right] \tag{2.38}$$

Definiendo el operador $PID[e_k(t)] \triangleq K_p e_k(t) + K_i \int e_k(t)dt + K_d \frac{de_k(t)}{dt}$ y sustituyendo en la ecuación (2.64).

$$u_k(t) = f_k(t) + PID[e_k(t)] \tag{2.39}$$
$$u_{k+1}(t) = f_{k+1}(t) + PID[e_{k+1}(t)] \tag{2.40}$$
$$= u_k(t) + PID[e_{k+1}(t) - e_k(t)] + K_k[e_k(t+1) - e_k(t)] \tag{2.41}$$

Con un controlador de retroalimentación PID, la actualización de la entrada es, en consecuencia, la suma de la actualización del control de aprendizaje iterativo más la respuesta del controlador PID a la diferencia entre las salidas de la tarea actual y anterior. Tras el muestreo con periodo t_s se convierte en:

$$u_k(t) = f_k(t) + \left[K_p e_k(t) + K_i \sum_{\sigma=1}^{t} \frac{\left(e_k(\sigma) + e_k(\sigma-1)\right)t_s}{2} + K_d \frac{e_k(t) - e_k(t-1)}{t_s}\right] \quad t \in [0, t_f] \tag{2.42}$$

2.7. Algoritmos Canónicos de control de Aprendizaje Iterativo

Los controladores de aprendizaje iterativo pueden ser construidos de diversas formas. En esta sección se muestran las configuraciones más comunes, clasificadas en dos categorías principales: embebidas y de cascada.

2.7.1. Embebidas

Ciclo previo de aprendizaje

La configuración del ciclo previo de aprendizaje se muestra en la figura 2.16, donde i denota la iteración y se tiene que $y_{r,i}(s)$, $y_i(s)$, $u_i(s)$ y $e_i(s)$ denotan las señales de referencia, de salida, de control y el error, respectivamente en la i-ésima iteración. Así como, $P(s)$ y $C_l(s)$ representan las funciones de la planta y el compensador de retroalimentación, respectivamente. Cuando el sistema realiza la misma tarea de control $y_{r,i+1}(s) = y_{r,i}$. Los datos de memoria etiquetados con y, y_r y u son matrices de memoria, que almacenan las señales del sistema del ciclo actual, es decir, la iteración $(i+1)$, que se utilizaran en el siguiente ciclo de aprendizaje (iteración) (Jian-Xin Xu y Lee, 2009).

Supongamos que la tarea de control se repite para todas las iteraciones, es decir, $y_r = y_{r,i} = y_{r,i+1}$. Según la configuración del PCL mostrada en la figura 2.16.

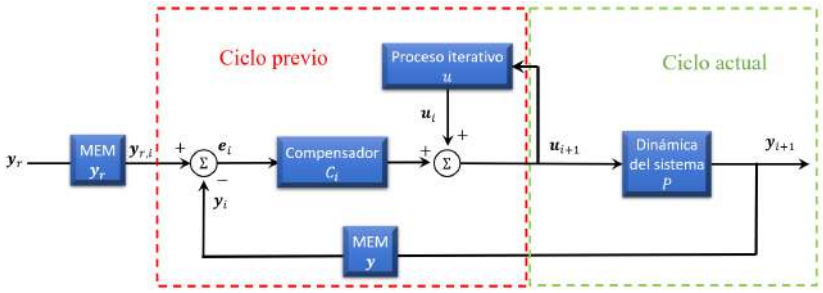


Figura 2.16: Diagrama a bloques del ciclo previo de aprendizaje (Jian-Xin Xu y Lee, 2009).

$$y_i = Pu_i \tag{2.43}$$
$$e_i = y_r - y_i \tag{2.44}$$
$$u_{i+1} = u_i + C_l e_i \tag{2.45}$$

La ecuación (2.40) es la ley de actualización del ciclo previo de aprendizaje. Se denomina aprendizaje del ciclo anterior simplemente porque sólo se utilizan las señales de control del ciclo anterior u_i y las señales de error anteriores e_i para formar la entrada de control del ciclo actual $u_i + 1$. Es un control de lazo abierto en el dominio del tiempo, pero un control de lazo cerrado en el dominio de la iteración.

La condición de convergencia de aprendizaje para el PCL puede derivarse como (Jian-Xin Xu y Lee, 2009):

$$e_{i+1} = y_r - y_{i+1} \tag{2.46}$$
$$= y_r - Pu_{i+1} \tag{2.47}$$
$$= y_r - P\left(u_i + C_l e_i\right) \tag{2.48}$$
$$= y_r - Pu_i + PC_l e_i \tag{2.49}$$

$$= (1 - PC_l)e_i \tag{2.50}$$

$$\Rightarrow \frac{e_{i+1}}{e_i} = 1 - PC_l \tag{2.51}$$

$$\Rightarrow \| \frac{e_{i+1}}{e_i} = \| 1 - PC_l \| \leq \gamma < 1 \tag{2.52}$$

donde la norma es la norma del infinito para todas las frecuencias $w \in \Omega$, $\Omega = [w_a, w_b]$, $w_b > w_a \geq 0$. Ω denota la banda de frecuencias de interés, o la banda de frecuencias que coincide con el ancho de banda del controlador. Claramente, en medida que las señales de error de seguimiento de la 1^{ra} iteración, e_0, es finita, entonces $\| e_i \leq \gamma^i \| e_0 \| \to 0$ conforme $i \to \infty$. También se puede ver que, para un proceso P, una elección adecuada de C_l hará que γ sea menor y, por tanto, acelerar el proceso de aprendizaje.

Ciclo actual de aprendizaje

Debido a la naturaleza de bucle abierto en el ciclo actual, el PCL podría ser sensible a perturbaciones pequeñas y no repetibles. Esto puede mejorarse mediante un aprendizaje basado en la retroalimentación si el bucle puede cerrarse adecuadamente en el dominio del tiempo, dando lugar al control del ciclo actual (CCL). La configuración del esquema CCL se muestra en la figura 2.17.

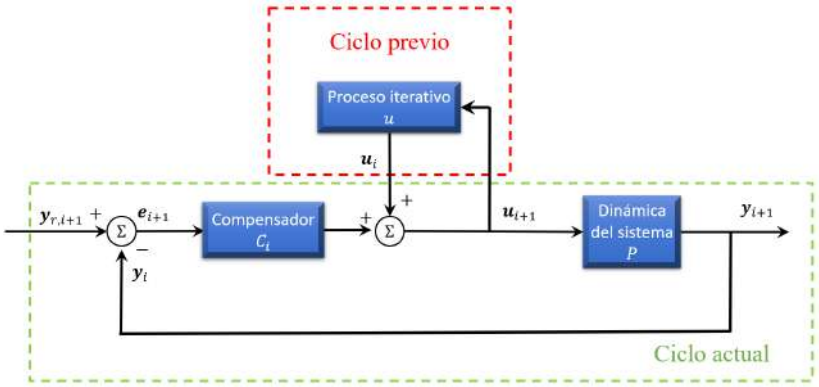


Figura 2.17: Diagrama de bloques del ciclo actual de aprendizaje (Jian-Xin Xu y Lee, 2009).

En consecuencia, la ley de actualización del régimen de la CCL es:

$$u_{i+1} = u_i + Ce_{i+1} \tag{2.53}$$

donde C es la función de transferencia del compensador, que es en realidad un controlador de retroalimentación. Se llama aprendizaje del ciclo actual porque el error de seguimiento del ciclo actual, e_{i+1}, está involucrado en el aprendizaje.

Para la tarea de control repetida y_r , la condición de convergencia para CCL se deduce de la siguiente manera (Jian-Xin Xu y Lee, 2009):

$$e_{i+1} = y_r - y_{i+1} \tag{2.54}$$

$$= y_r - Pu_{i+1} \tag{2.55}$$

$$= y_r - P\left(u_i + Ce_{i+1}\right) \tag{2.56}$$

$$= y_r - Pu_i - PCe_{i+1} \tag{2.57}$$

$$\left(1 + PC\right) e_{i+1} = e_i \tag{2.58}$$

$$\Rightarrow \frac{e_{i+1}}{e_i} = \frac{1}{1+PC} \tag{2.59}$$

$$\Rightarrow \| \frac{e_{i+1}}{e_i} \| = \| \frac{1}{1+PC} \| \leq \gamma < 1 \tag{2.60}$$

Se puede observar que el PCL y el CCL funcionan de forma complementaria. PCL requiere que el factor $1 - PC_l$ sea inferior a 1 para todas las frecuencias dentro de la banda Ω , y a menudo conduce a una ganancia baja C_l . CCL requiere que el factor $1 + PC$ sea superior a 1 para todas las frecuencias dentro de la banda Ω , y a menudo se prefiere una ganancia C alta en la medida en que se garantice la estabilidad del bucle cerrado. Obsérvese que las matrices de memoria requeridas para CCL son la mitad de las de PCL, porque no se requiere datos de memoria para e_i. El tamaño de la memoria es un factor importante a la hora de implementar un esquema ILC.

Ciclo previo y actual de aprendizaje

La combinación de ambos tipos de ciclos, dificulta encontrar una C_1 que permita satisfacer $\| 1 - PC_1 \| < 1$ para todas las frecuencias Ω. Del mismo modo, puede ser un trabajo difícil encontrar un C adecuada tal que el denominador en $\| \frac{1}{1+PC} \|$ sea estrictamente mayor que 1 para todas las frecuencias Ω. Esto puede lograrse si las condiciones de convergencia (2.47) y (2.55) pueden complementarse entre sí en las frecuencias en las que se viola una de las condiciones de convergencia. Para la tarea de control repetida y_r, puede derivarse fácilmente que la condición de convergencia es (Jian-Xin Xu y Lee, 2009):

$$\frac{\| 1 - PC_1 \|}{\| 1 + PC \|} \tag{2.61}$$

La ecuación (2.56) es la multiplicación de ambas condiciones. Este ejemplo se muestra en la figura 2.18.

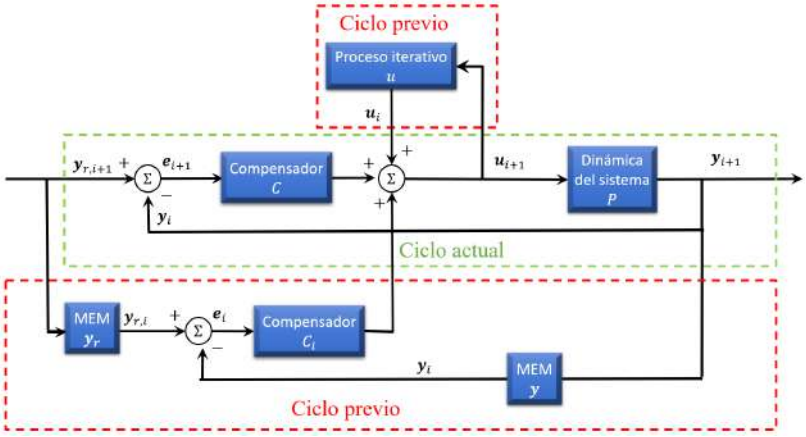


Figura 2.18: Diagrama a bloques de PCCL (Previous cycle-Current cycle) ciclo previo y ciclo actual.

2.7.2. Cascada

En los tres esquemas de control de aprendizaje iterativo anteriores, se puede observar que el sistema de control ha sido rediseñado, con un nuevo componente feedforward añadido al canal de entrada del sistema. A partir de las config uraciones mostradas en las figuras 2.9, 2.16 y 2.17 se incorpora un nuevo bloque de control en el lazo cerrado de control. Esta estructura incrustada es habitual en la mayoría de los esquemas de control de aprendizaje iterativo en tiempo real existentes. Por lo tanto, cuando se quiere incorporar un mecanismo de aprendizaje iterativo a un sistema de control existente, o bien, hay que reescribir el programa

de ejecución central o bien hay que sustituir el chip del microcontrolador. En muchas aplicaciones reales, esta reconfiguración de un controlador comercial no es aceptable debido a las limitaciones de coste, seguridad y propiedad intelectual. Por ejemplo, un dispositivo de termoprocesamiento rápido en la industria de las obleas cuesta millones de dólares, y la única parte sintonizable es un número de puntos de ajuste. En estas circunstancias, el método de aprendizaje en cascada es adecuado, ya que sólo modifica la trayectoria de referencia de forma iterativa para mejorar el rendimiento del control.

Control de aprendizaje iterativo en cascada

El control de aprendizaje iterativo con la estructura en cascada utilizará las señales de referencia modificadas y la salida real del sistema del ciclo anterior para generar las nuevas señales de referencia para el ciclo actual. No es necesario integrar el control de aprendizaje iterativo en el bucle de control existente, debido a la estructura en cascada, evitando la reconfiguración del hardware del sistema o del programa de ejecución del núcleo. En este caso lo que se requiere es una reprogramación de las señales de referencia, que puede ser implementado fácilmente en aplicaciones reales, como se muesta en la figura 2.19.

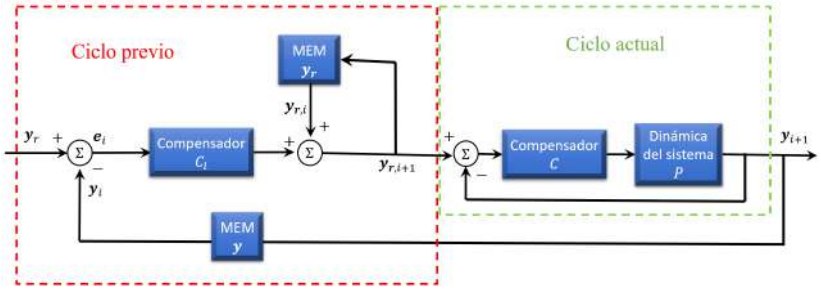


Figura 2.19: Diagrama de bloques del control de aprendizaje en cascada (Jian-Xin Xu y Lee, 2009).

Control de aprendizaje iterativo en cascada incremental

En la ecuación (2.62) se expresa la relación entre la trayectoria de referencia original, y_r, y la trayectoria de referencia revisada iterativamente, $y_{r,i+1}$ (Jian-Xin Xu y Lee, 2009).

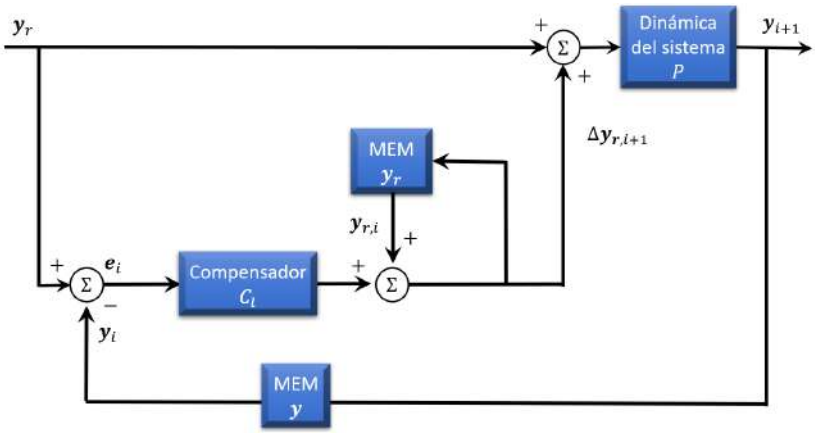


Figura 2.20: Diagrama a bloques del control de aprendizaje iterativo en cascada incremental (Jian-Xin Xu y Lee, 2009).

$$
\begin{aligned}
y_{r,i+1} &= y_{r,i} + C_1 e_i && (2.62)\\
&= y_{r,i-1} + \sum_{j=0}^{1} C_1 e_{i-j} && (2.63)\\
&= \cdots = && (2.64)\\
&= y_{r,0} + \sum_{j=0}^{i} C_1 e_{i-j} && (2.65)\\
&= y_r + \sum_{j=0}^{i} C_1 e_{i-j} && (2.66)
\end{aligned}
$$

La referencia original se modifica con los errores $\sum_{j=0}^{i} C_1 e_{i-j}$. Si el error $e_0, \cdots, e_i$ son cero, entonces $y_{r,i+1} = y_r$. De tal forma, que si el bucle de retroalimentación es capz de lograr un seguimiento perfecto con respecto a la referencia original y_r, entonces el control de aprendizaje iterativo en cascada incremental conservará la trayectoria de referencia original, como se observa en la figura 2.20. Sin embargo, un problema potencial para este caso para aplicaciones en tiempo real es el filtrado. Una práctica común en las aplicaciones de control de aprendizaje iterativo es diseñar un filtro paso bajo o un filtro paso banda.

Capítulo 3

Modelo dinámico

La obtención del modelo dinámico para robots manipuladores de n grados de libertad en cadena cinemática abierta es de gran importancia cuando se quiere implementar el sistema dinámico en un lazo cerrado de control, ya que, éste permite representarlo matemáticamente para realizar un análisis y estudio a profundidad de los fenómenos físicos característicos del sistema. A continuación se emplean los conceptos teóricos para la obtención del modelo dinámico en el robot manipulador de un grado de libertad y en el robot manipulador de dos grados de libertad.

3.1. Robot manipulador de un grado de libertad (péndulo simple)

Para el desarrollo de esta tesis es elemental el estudio del modelo dinámico de robots manipuladores, el cual, ya fue definido en el capítulo anterior. Inicialmente se aplica el desarrollo del modelo dinámico para el robot manipulador de un grado de libertad, ya que es la representación básica de los robots manipuladores, y permite establecer la metodología puntual que se lleva a cabo para obtener dicho modelo dinámico, así como, se aplica la teoría de pasividad en el modelo obtenido.

3.1.1. Cinemática analítica de Euler para el péndulo simple

En la figura 3.1 se muestra un péndulo simple compuesto por un servomotor, el cual tiene acoplado una barra metálica de longitud l_1 y esta sometido a la acción de la gravedad.

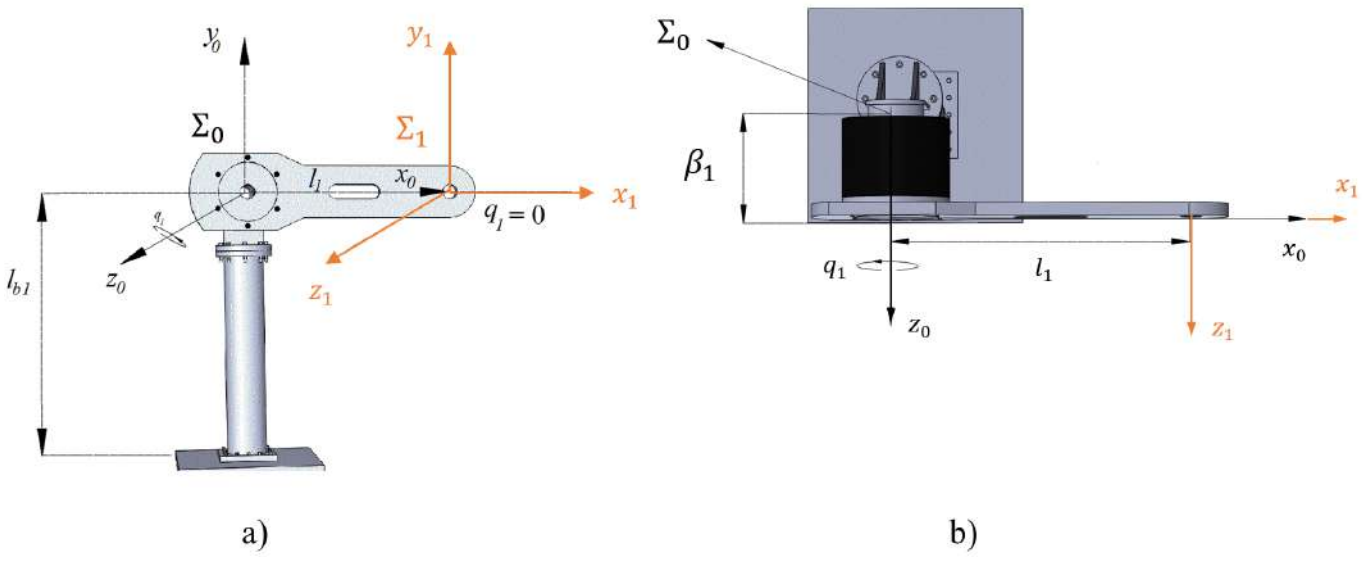


Figura 3.1: Vista lateral del péndulo simple y vista superior, con $q_1 = 0$ (posición de casa).

En la figura 3.1a) se muestra una vista lateral del péndulo simple, el cual consiste de una base anclada al suelo, una base para el servomotor, el servomotor y una barra metálica (eslabón) anclada al rotor y al estator del servomotor para generar el movimiento rotacional (positiva en sentido contrario a las manecillas del reloj).

En la figura 3.1b) se tiene la vista superior del péndulo simple, donde se muestran los parámetros geométricos del péndulo, como son; el espesor del servomotor y barra metálica β_1 (donde se fija la referencia al centro geométrico del estator Σ_0), la distancia l_1 la cual inicia en el centro geométrico del estator y termina en el centro geométrico del efector final (donde se ubica la referencia Σ_1 debido al efector final o extremo final) y finalmente se muestra q_1 la rotación efectuada sobre el centro geométrico del estator.

Debido a que la localización de un cuerpo rígido en el espacio precisa de especificar tanto su posición como su orientación. Ambas deben estar relacionadas a un sistema de referencia definido. El sistema de referencia Σ_0 es el sistema de referencia definido y Σ_1 es el sistema de referencia del efector final o extremo final el cual agrupa la traslación y rotación del eslabón respecto a Σ_0.

De acuerdo a la figura 3.1a), la posición de casa se ubica colocando la barra del péndulo sobre el eje x_0, es decir cuando $q_1 = 0$ y dado que los sistemas de referencia Σ_0 y Σ_1 están rigídamente acoplados al péndulo no se deforman. Como ya se comento, Σ_0 está alineado de forma concéntrica al eje o rotor del servomotor por medio del eje z_0 y su punto de origen (0,0,0) está anclado a la parte final del estator como se muestra en la figura 3.1b). Por lo tanto se tiene una distancia β_1 hasta la superficie del eslabón. El sistema de referencia Σ_0 es fijo, no gira respecto a ningún eje. Mientras que Σ_1 está localizado en el efector final o extremo final del eslabón, y se mueve rígidamente con respecto al mismo eslabón.

En la ecuación (3.1) se tiene la representación del vector traslación y matrices de rotación de Σ_1 respecto a Σ_0.

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = Rz_0(q_1 = 0) \begin{bmatrix} l_1 \\ 0 \\ \beta_1 \end{bmatrix} + Rz_0(q_1 = 0) R_{x0}(\alpha_1) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{3.1}$$

En la ecuación (3.2) se muestra la matriz de rotación en z_0 con $q_1 = 0$, que representa la posición de casa.

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \begin{bmatrix} \cos(0) & -\sin(0) & 0 \\ \sin(0) & \cos(0) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} l_1 \\ 0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \cos(0) & -\sin(0) & 0 \\ \sin(0) & \cos(0) & 0 \\ 0 & 0 & 1 \end{bmatrix} R_{x0}(\alpha_1) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{3.2}$$

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} l_1 \\ 0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} R_{x0}(\alpha_1) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{3.3}$$

Se obtiene la forma de Denavit-Hartenberg

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = H_{R_{z0}}(q_1)H_{T_{z0}}(\beta_1)H_{T_{x0}}(l_1)H_{R_{x0}}(0) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \tag{3.4}$$

Para el caso de $q_1 = 0$, posición de casa, se tiene la representación de matriz de transformación homogénea (Denavit-Hartenberg) representada como:

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & l_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \beta_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{pmatrix} \tag{3.5}$$

La tabla de Denavit-Hartenberg en este caso es:

DH del péndulo				
Eslabón	l_i	α_1	β_i	θ_1
1	l_1	0	β_1	$q_1 = 0$

(3.6)

Ahora se presenta el caso para cualquier q_1, de tal forma que la barra del péndulo no se encuentra sobre el eje x_0, por lo tanto $q_1 \neq 0$ como se muestra en la figura 3.2. Se conservan los mismos sistemas de referencia Σ_0 y Σ_1 descritos para la figura 3.1.

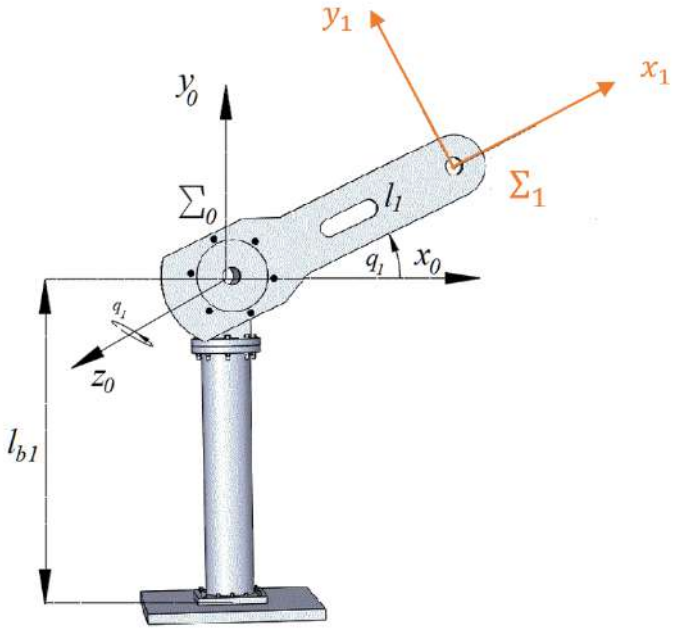


Figura 3.2: Vista frontal del péndulo con $q_1 \neq 0$.

De acuerdo a la figura 3.2, se tiene el péndulo para cualquier posición ($q_1 \neq 0$) y dado que los sistemas de referencia Σ_0 y Σ_1 son rígidos al péndulo y no se deforman. Σ_0 está alineado de forma concéntrica al eje o rotor del servomotor por medio del eje z_0 y su punto de origen (0,0,0) está anclado a la parte final del estator como se muestra en la figura 3.1b. Por lo tanto se tiene una distancia β_1 hasta la superficie

del eslabón. Y este sistema de referencia es fijo, no gira respecto a ningún eje. Mientras que Σ_1 está localizado en el efector final o extremo final del eslabón, y se mueve rígidamente con respecto al mismo eslabón.

En la ecuación (3.7) se tiene la representación del vector traslación y las matrices de rotación de Σ_1 respecto a Σ_0.

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = Rz_0(q_1) \begin{bmatrix} l_1 \\ 0 \\ \beta_1 \end{bmatrix} + R_{z0}(q_1)R_{x0}(\alpha_1) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{3.7}$$

Se tiene el vector de traslación de Σ_1 respecto a Σ_0, más el producto de las matrices de rotación $\mathrm{R}z_0(q_1)$ (orientación de z_1 respecto a z_0, en este caso los ejes son paralelos) para obtener las coordenadas (x_0, y_0) de l_1 debido a la rotación, multiplicado por la matriz de rotación $\mathrm{R}x_0(\alpha_1)$ (orientación de z_1 con respecto a z_0 sobre el eje x_0).

En la ecuación (3.8) se muestra la matriz de rotación respecto a z_0 para cualquier q_1 con $q_1 \neq 0$.

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 \\ \sin(q_1) & \cos(q_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} l_1 \\ 0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 \\ \sin(q_1) & \cos(q_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} R_{x0}(\alpha_1) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{3.8}$$

Se llega a la forma de Denavit-Hartenberg

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = H_{R_{z0}}(q_1)H_{T_{z0}}(\beta_1)H_{T_{x0}}(l_1)H_{R_{x0}}(0) \begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{pmatrix} \tag{3.9}$$

Para el caso de $q_1 \neq 0$, se tiene la representación de matriz de transformación homogénea (Denavit-Hartenberg) representada como:

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 & l_1\cos(q_1) \\ \sin(q_1) & \cos(q_1) & 0 & l_1\sin(q_1) \\ 0 & 0 & 1 & \beta_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{pmatrix} \tag{3.10}$$

La tabla de Denavit-Hartenberg en este caso es:

DH del péndulo				
Eslabón	l_i	α_1	β_i	θ_1
1	l_1	0	β_1	q_1

(3.11)

3.1.2. Cinemática diferencial

Dada la matriz de rotación $R_{z_0}(q_1)$ tomada de la ecuación (3.8) la cual es igual a:

$$R_{z_0}(q_1) = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 \\ \sin(q_1) & \cos(q_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.12}$$

Se aplica la derivada respecto a la coordenada generalizada q_1:

$$S = \frac{dR_{z_0}(q_1)}{dq_1} R_{z_0}^T(q_1) \tag{3.13}$$

$$S = S(\hat{\mathbf{k}}) \tag{3.14}$$

Y por tanto de (2.2) se tiene:

$$\frac{dR_{z_0}(q_1)}{dq_1} = S(\hat{\mathbf{k}}) R_{z_0}(q_1) \tag{3.15}$$

Teniendo estos resultados aplicamos la derivada respecto al tiempo de la matriz de rotación $R_{z_0}(q_1)$ de la ecuación (2.1):

$$\frac{dR_{z_0}(q_1)}{dt} = \frac{dR_{z_0}(q_1)}{dq_1} \frac{d(q_1)}{dq_1} \tag{3.16}$$

Donde se encuentra la igualdad con el Jacobiano de R_{z_0}, se tiene:

$$\frac{dR_{z_0}(q_1)}{dt} = \frac{dR_{z_0}(q_1)}{dq_1} \frac{d(q_1)}{dq_1} = \frac{\partial R_{z_0}(q_1)}{\partial q_1} \dot{q}_1 \tag{3.17}$$

Y sustituyendo la derivada respecto de q_1 (Jacobiano) de acuerdo con la ecuación (2.4) y dado que $\dot{q}_1$ es un escalar se puede conmutar y asociar en la matriz S, se tiene:

$$\frac{dR_{z_0}(q_1)}{dt} = S(\hat{\mathbf{k}}) R_{z_0}(q_1) \dot{q}_1 = S(\dot{q}_1 \hat{\mathbf{k}}) R_{z_0}(q_1) \tag{3.18}$$

Sustituyendo las matrices (antisimétrica y de rotación):

$$\frac{dR_{z_0}(q_1)}{dt} = \begin{bmatrix} 0 & -\dot{q}_1 & 0 \\ \dot{q}_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 \\ \sin(q_1) & \cos(q_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.19}$$

3.1.3. Propiedades de la matriz $S(\dot{q}_1 \hat{\mathbf{k}})$

Dada la matriz obtenida de la ecuación (2.10) se cumplen las siguientes propiedades:

Conmutación del producto de la matriz $S(\dot{q}_1\hat{\mathbf{k}})$ y $S(\dot{q}_1\hat{\mathbf{k}})^T$

$$S(\dot{q}_1\hat{\mathbf{k}})S(\dot{q}_1\hat{\mathbf{k}})^T = S(\dot{q}_1\hat{\mathbf{k}})^T S(\dot{q}_1\hat{\mathbf{k}}) \tag{3.20}$$

$$\begin{bmatrix} \dot{q}_1^2 & 0 & 0 \\ 0 & \dot{q}_1^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \dot{q}_1^2 & 0 & 0 \\ 0 & \dot{q}_1^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.21}$$

Se comprueba que la igualdad se satisface.

Valores propios de la matriz $S(\dot{q}_1\hat{\mathbf{k}})$

Se obtienen los valores propios de la matriz$S(\dot{q}_1\hat{\mathbf{k}})$:

$$det\left[\lambda I - S(\dot{q}_1\hat{\mathbf{k}})\right] = det\left[\begin{pmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{pmatrix} - \begin{pmatrix} 0 & -\dot{q}_1 & 0 \\ \dot{q}_1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}\right] = \lambda^2(\lambda + \dot{q}_1^2) \tag{3.22}$$

Se obtienen los valores propios:

$$\lambda_1 = -\dot{q}_1^2 \tag{3.23}$$

$$\lambda_{2,3} = 0 \tag{3.24}$$

Norma espectral de $S(\dot{q}_1\hat{\mathbf{k}})$

La norma espectral de la matriz S esta determinada como:

$$\| S(\dot{q}_1\hat{\mathbf{k}}) \| = \sqrt{\lambda_{S^TS}^{máx}} \tag{3.25}$$

Y dado que ya se conoce la matriz S^TS de la ecuación (2.12), se obtienen sus valores propios:

$$det\left[\lambda I - S(\dot{q}_1\hat{\mathbf{k}})^T S(\dot{q}_1\hat{\mathbf{k}})\right] = det\left[\begin{pmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{pmatrix} - \begin{pmatrix} \dot{q}_1^2 & 0 & 0 \\ 0 & \dot{q}_1^2 & 0 \\ 0 & 0 & 0 \end{pmatrix}\right] = \lambda(\lambda - \dot{q}_1^2)^2 \tag{3.26}$$

Se obtienen los valores propios:

$$\lambda_{1,2} = \dot{q}_1^2 \tag{3.27}$$

$$\lambda_3 = 0 \tag{3.28}$$

En este caso se tiene $\lambda_{S^TS}^{máx} = \dot{q}_1^2$, se tiene la norma espectral:

$$\| S(\dot{q}_1\hat{\mathbf{k}}) \| = \sqrt{\dot{q}_1^2} = \dot{q}_1 \tag{3.29}$$

Se aplica la derivada respecto al tiempo para la matriz de transformación homogénea (1.15). Se realiza el cambio de la longitud l_1 por el centro de masa l_{c1} para realizar el análisis de energía y fuerza.

$$\begin{bmatrix} \dot{x_0} \\ \dot{y_0} \\ \dot{z_0} \\ 0 \end{bmatrix} = \begin{bmatrix} -\sin(q_1)\dot{q}_1 & -\cos(q_1)\dot{q}_1 & 0 & -l_{c1}\sin(q_1)\dot{q}_1 \\ \cos(q_1)\dot{q}_1 & -\sin(q_1)\dot{q}_1 & 0 & l_{c1}\cos(q_1)\dot{q}_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} \dot{x_0} \\ \dot{y_0} \\ \dot{z_0} \end{bmatrix} = S(\dot{q}_1\hat{\mathbf{k}})R_{z0}(q_1) \begin{bmatrix} l_{c1} \\ 0 \\ \beta_1 \end{bmatrix} + S(\dot{q}_1\hat{\mathbf{k}})R_{z0}(q_1) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{3.30}$$

Donde $R_{x_0}(\alpha_1)$ es la matriz identidad.

3.1.4. Rapidez de traslación

Se obtiene la rapidez para el péndulo simple a partir de la cinemática diferencial obtenida en la ecuación (2.30), donde la rapidez esta dada como:

$$\boldsymbol{v}^T\boldsymbol{v} = \| \boldsymbol{v} \|^2 \tag{3.31}$$

$$\boldsymbol{v} = \begin{bmatrix} \dot{x_0} \\ \dot{y_0} \\ \dot{z_0} \end{bmatrix} = S(\dot{q}_1\hat{\mathbf{k}})R_{z0}(q_1) \begin{bmatrix} l_{c1} \\ 0 \\ \beta_1 \end{bmatrix} + S(\dot{q}_1\hat{\mathbf{k}})R_{z0}(q_1) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{3.32}$$

Se obtiene la rapidez aplicando la ecuación (3.1):

$$\| \boldsymbol{v} \|^2 = \left\| \begin{matrix} \dot{x_0} \\ \dot{y_0} \\ \dot{z_0} \end{matrix} \right\|^2 = \left\| S(\dot{q}_1\hat{\mathbf{k}})R_{z0}(q_1) \begin{bmatrix} l_{c1} \\ 0 \\ \beta_1 \end{bmatrix} + S(\dot{q}_1\hat{\mathbf{k}})R_{z0}(q_1) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \right\|^2 \tag{3.33}$$

Dado que las coordenadas x_1 y y_1 representan el origen de Σ_1, sus valores son igual a 0, entonces se tiene:

$$\| \boldsymbol{v} \|^2 = l_{c1}^2\dot{q}_1^2 \tag{3.34}$$

3.1.5. Modelo de energía del péndulo simple

La energía del servomecanismo se compone de la energía cinética $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$ y de la energía potencial $U(q)$. La energía cinética $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$ incluye el movimiento de traslación y rotación:

$$\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) = \underbrace{\frac{1}{2}m_i\mathbf{v}_i^T\mathbf{v}_i}_{\textit{momento de inercia de la barra}} + \underbrace{\frac{1}{2}I_i\left[\textstyle\sum_i^n \dot{\mathbf{q}}_i\right]^2}_{\textit{momento de inercia del motor}} \tag{3.35}$$

donde I_i es el momento de inercia del i-ésimo servomotor.

Para el péndulo simple la energía cinética es:

$$\mathcal{K}(q_1, \dot{q}_1) = \frac{1}{2}m \| \boldsymbol{v} \|^2 + \frac{1}{2}I\dot{q}_1^2 = \frac{1}{2}ml_{c1}^2\dot{q}_1^2 + \frac{1}{2}I\dot{q}_1^2 = \frac{1}{2}(ml_{c1}^2 + I)\dot{q}_1^2 \tag{3.36}$$

Se obtiene la suma de las energías cinéticas, lo que representa que se tienen eje paralelos en su movimiento de traslación y rotación. La energía potencial $\mathcal{U}(\mathbf{q})$ no tiene forma especifica como en el caso de la energía cinética, depende de la geometría del servomecanismo en general $\mathcal{U}_i(\mathbf{q}) = m_i g l_{ci} h_i(\mathbf{q})$, siendo $h_i(\mathbf{q})$ una función que indica la altura del eslabón con respecto al origen del sistema de referencia del servomecanismo (Reyes, 2011).

Para este ejemplo se tiene el siguiente análisis de energía potencial.

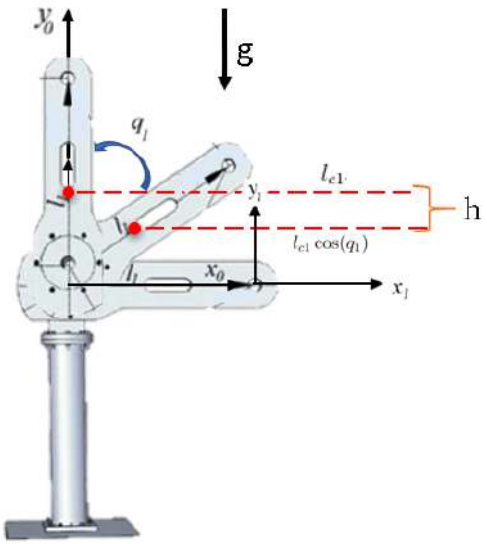


Figura 3.3: Análisis de energía potencial, donde l_{c1} se indica con los puntos rojos.

$$\mathcal{U}(q_1) = mgh(q_1) = mgl_{c1}(1-\sin(q_1)) \tag{3.37}$$

Se considera h_1 como el centro de masa más cercano al origen.

El lagrangiano para el caso del péndulo simple está dado por:

$$\mathcal{L}(q_1,\dot{q}_1) = \mathcal{K}(q_1,\dot{q}_1) - \mathcal{U}(q_1) = \frac{1}{2}(ml_{c1}^2 + I)\dot{q}_1^2 - mgl_{c1}(1-\sin(q_1)) \tag{3.38}$$

Se aplican las ecuaciones de movimiento de Euler-Lagrange y se incluye el modelo de fricción para el péndulo simple:

$$\tau = \frac{d}{dt}\left[\frac{\partial \mathcal{L}(q_1,\dot{q}_1)}{\partial \dot{q}_1}\right] - \frac{\partial \mathcal{L}(q_1,\dot{q}_1)}{\partial q_1} + b\dot{q}_1 + f_c signo(\dot{q}_1) + f_e\left[1 - |signo(\dot{q}_1)|\right] \tag{3.39}$$

Se obtienen las derivadas parciales:

$$\frac{\partial \mathcal{L}(q_1,\dot{q}_1)}{\partial q_1} = \frac{\partial}{\partial q_1}\left[\tfrac{1}{2}(ml_{c1}^2 + I)\dot{q}_1^2 - mgl_{c1}(1-\sin(q_1))\right] = mgl_{c1}\cos(q_1) \tag{3.40}$$

$$\frac{\partial \mathcal{L}(q_1,\dot{q}_1)}{\partial \dot{q}_1} = \frac{\partial}{\partial \dot{q}_1}\left[\tfrac{1}{2}(ml_{c1}^2 + I)\dot{q}_1^2 - mgl_{c1}(1-\sin(q_1))\right] = (ml_{c1}^2 + I)\dot{q}_1 \tag{3.41}$$

$$\frac{d}{dt}\left[\frac{\partial \mathcal{L}(q_1,\dot{q}_1)}{\partial \dot{q}_1}\right] = \frac{d}{dt}\left[(ml_{c1}^2 + I)\dot{q}_1\right] = (ml_{c1}^2 + I)\ddot{q}_1 \tag{3.42}$$

Finalmente se obtiene el modelo dinámico de un péndulo simple incluyendo los fenómenos de fricción.

$$\tau = (ml_{c1}^2 + I)\ddot{q}_1 - mgl_{c1}\cos(q_1) + b\dot{q}_1 + f_c signo(\dot{q}_1) + f_e\left[1 - |signo(\dot{q}_1)|\right] \tag{3.43}$$

3.1.6. Pasividad del péndulo simple

El modelo dinámico del péndulo simple es un sistema pasivo.

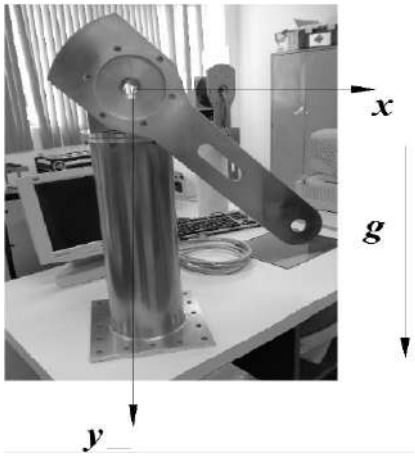


Figura 3.4: Servomecanismo péndulo.

El modelo dinámico del péndulo esta dado por:

$$\tau = i_p\ddot{q} + b\dot{q} + f_c signo(\dot{q}) + mgl_c\sin(q) \tag{3.44}$$

Multiplicando por la velocidad de movimiento articular $\dot{q}$ ambos lados de la ecuacion (4.1) se obtiene:

$$\dot{q}\tau = i_p\dot{q}\ddot{q} + b\dot{q}^2 + f_c\dot{q}signo(\dot{q}) + \dot{q}mgl_c\sin(q) \tag{3.45}$$

$$\Rightarrow \dot{q}\tau = \frac{d}{dt}[\frac{1}{2}\dot{q}^2] + b\dot{q}^2 + f_c\dot{q}signo(\dot{q}) + \dot{q}\frac{\partial}{\partial q}mgl_c\left[1 - \cos(q)\right] \tag{3.46}$$

$$\Rightarrow \int_0^t \dot{q}(\sigma)\tau(\sigma)d\sigma = \mathcal{K}(\dot{q}(t)) - \mathcal{K}(\dot{q}(0)) + \mathcal{U}(q(t)) - \mathcal{U}(q(0)) + \int_0^t b\dot{q}^2(\sigma)d\sigma + \int_0^t f_c|\dot{q}(\sigma)|d\sigma \tag{3.47}$$

Se obtiene la siguiente función de energía $\mathcal{H}(t)$:

$$\underbrace{\int_0^t \dot{q}(\sigma)\tau(\sigma)d\sigma}_{energa\ aplicada} = \underbrace{\mathcal{H}(t)}_{hamiltoniano} - \underbrace{\mathcal{H}(0)}_{energa\ inicial} + \underbrace{\int_0^t b\dot{q}^2(\sigma)d\sigma + \int_0^t f_c|\dot{q}(\sigma)|d\sigma}_{energa\ disipada} \tag{3.48}$$

Donde la energía aplicada $\int_0^t \dot{q}(\sigma)\tau(\sigma)d\sigma$ es la energía total del péndulo siendo igual a la suma de la energía representada por el hamiltoniano $\mathcal{H}(t)$ en el péndulo mas la energía disipada; la proporción de su suministro $\dot{q}(t)\tau(t)$ es la potencia entregada, mientras que la potencia disipada es $b\dot{q}^2 + f_c|\dot{q}|$. Se puede considerar que la entrada u es $\tau(t)$ y la salida y es $\dot{q}(t)$. Por lo que, el sistema $\Sigma : u \to y$ es pasivo.

3.2. Robot manipulador de 2 gdl

A continuación, se aplica la teoría vista en el capítulo anterior para la obtención del modelo dinámico del robot manipulador de dos grados de libertad (2 gdl), que como se ha visto para el caso del péndulo simple se sigue una metodología bien definida que consiste en reconocer los sistemas de referencia de cada uno de los servomotores que representan los grados de libertad que tiene el robot manipulador, sin embargo, para este caso se emplea la representación de Denavit-Hartenberg dada la naturaleza vectorial del modelo dinámico a obtener.

3.2.1. Cinemática analítica de Euler para el robot manipulador de 2 gdl

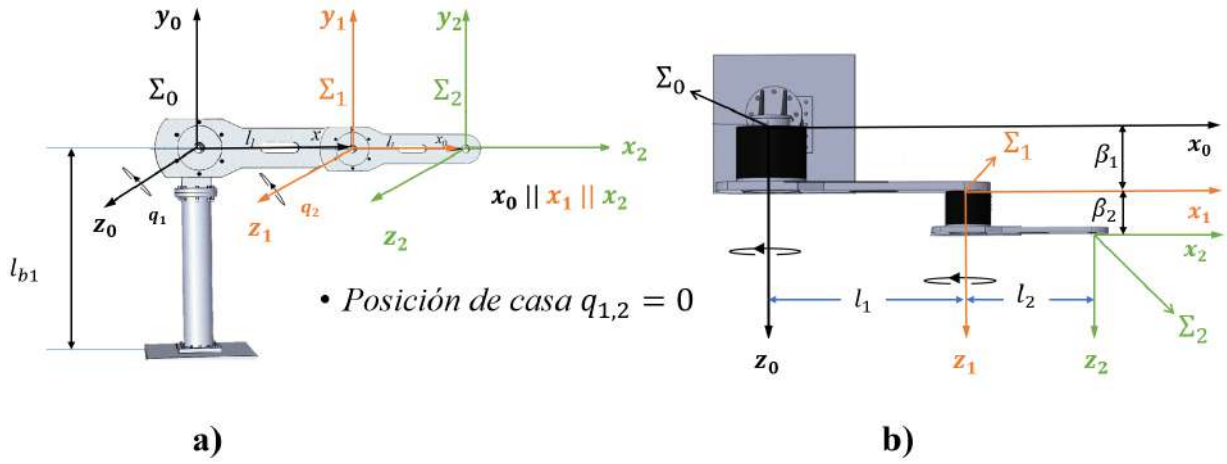


Figura 3.5: Vista lateral del péndulo simple y vista superior, con $q_1 = q_2 = 0$ (posición de casa).

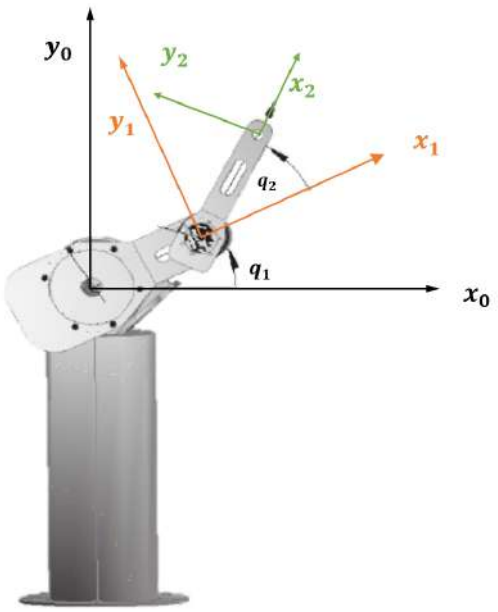


Figura 3.6: Vista frontal del péndulo con $q_{1,2} \neq 0$.

Traslación y rotación Las ecuaciones de traslación y rotación entre sistemas de referencia, de acuerdo a la figura 3.6 son:

$$
\begin{aligned}
\boldsymbol{p}_0 &= \boldsymbol{d}_0^1 + R(\phi,\theta,\psi)\boldsymbol{p}_1 \text{ con } \mathbf{p}_0 \in \mathbb{R}^n \\
&= \boldsymbol{d}_0^1 + R_{z_0}(q_1)R_{x_{0_1}}(\alpha_1)R_{y_{0_2}}(0)\boldsymbol{p}_1 \qquad (3.49)\\
\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} &= \mathbf{d}_0^1 \; + \; R_{z0}(q_1)R_{x0}(\alpha_1)R_{y0}(0) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \qquad (3.50)\\
&= \begin{bmatrix} l_1\cos(q_1) \\ l_1\sin(q_1) \\ \beta_1 \end{bmatrix} \; + \; R_{z0}(q_1)R_{x0}(\alpha_1)\mathrm{I} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \qquad (3.51)\\
&= R_{z_0}(q_1)\left[\left[\begin{pmatrix} 0 \\ 0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} l_1 \\ 0 \\ 0 \end{pmatrix}\right] + R_{x0}(\alpha_1)\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}\right] \qquad (3.52)\\
\boldsymbol{p}_1 &= \boldsymbol{d}_1^2 + R_{z_1}(q_2)R_{x_1}(\alpha_2)R_{y_1}(0)\boldsymbol{p}_2 \qquad (3.53)\\
&= \begin{bmatrix} l_2\cos(q_2) \\ l_2\sin(q_2) \\ \beta_2 \end{bmatrix} \; + \; R_{z1}(q_2)R_{x1}(\alpha_2)\mathrm{I} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} \qquad (3.54)
\end{aligned}
$$

$$
= R_{z_1}(q_2)\left[\left[\begin{pmatrix} 0 \\ 0 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} l_2 \\ 0 \\ 0 \end{pmatrix}\right] + R_{x1}(\alpha_2)\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix}\right] \qquad (3.55)
$$

Sustituyendo el vector $\boldsymbol{p}_1$:

$$
\begin{aligned}
\boldsymbol{p}_0 &= \boldsymbol{d}_0^1 + R_{z_0}(q_1)R_{x_0}(\alpha_1)R_{y_0}(0)\boldsymbol{p}_1 \qquad (3.56)\\
\boldsymbol{p}_0 &= \boldsymbol{d}_0^1 + R_{z_0}(q_1)R_{x_0}(\alpha_1)R_{y_0}(0)\left[\boldsymbol{d}_1^2 + R_{z_1}(q_2)R_{x_1}(\alpha_2)R_{y_1}(0)\boldsymbol{p}_2\right] \qquad (3.57)\\
&= R_{z_0}(q_1)\left[\begin{pmatrix} 0 \\ 0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} l_1 \\ 0 \\ 0 \end{pmatrix} + R_{x0}(\alpha_1)R_{z_1}(q_2)\left[\begin{pmatrix} 0 \\ 0 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} l_2 \\ 0 \\ 0 \end{pmatrix} + R_{x1}(\alpha_2)\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix}\right]\right] \qquad (3.58)
\end{aligned}
$$

3.2.2. Cinemática analítica de Euler (Representación Denavit-Hartenberg)

A partir de las ecuaciones obtenidas por la cinemática analítica de Euler se puede llegar a la representación de Denavit-Hartenberg. En las ecuaciones (3.60) y (3.62) se muestra la representación individual de cada articulación del robot de 2gdl, mientras que la ecuación (3.64) corresponde a la cinemática del segundo eslabón el cual esta sujeto a los movimientos de traslación y rotación del primer eslabón.

$$
\begin{aligned}
\boldsymbol{p}_0 &= R_{z_0}(q_1)\left[\begin{pmatrix} 0 \\ 0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} l_1 \\ 0 \\ 0 \end{pmatrix}\right] + R_{z0}(q_1)R_{x0}(\alpha_1)\boldsymbol{p}_1 \qquad (3.59)\\
\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} &= \begin{bmatrix} R_{z_0}(q_1) & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \begin{bmatrix} 0 \\ 0 \\ \beta_1 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \begin{bmatrix} l_1 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} R_{x_0}(\alpha_1) & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} \qquad (3.60)\\
\boldsymbol{p}_1 &= R_{z_1}(q_2)\left[\begin{pmatrix} 0 \\ 0 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} l_2 \\ 0 \\ 0 \end{pmatrix}\right] + R_{z1}(q_2)R_{x1}(\alpha_2)\boldsymbol{p}_2 \qquad (3.61)
\end{aligned}
$$

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} R_{z_1}(q_2) & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \begin{bmatrix} 0 \\ 0 \\ \beta_2 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \begin{bmatrix} l_2 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} R_{x_1}(\alpha_2) & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} \tag{3.62}$$

$$\boldsymbol{p}_0 = R_{z_0}(q_1) \left[\begin{pmatrix} 0 \\ 0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} l_1 \\ 0 \\ 0 \end{pmatrix} + R_{x0}(\alpha_1) R_{z_1}(q_2) \left[\begin{pmatrix} 0 \\ 0 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} l_2 \\ 0 \\ 0 \end{pmatrix} + R_{x1}(\alpha_2) \boldsymbol{p}_2 \right] \right] \tag{3.63}$$

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = \begin{bmatrix} R_{z_0}(q_1) & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \begin{bmatrix} 0 \\ 0 \\ \beta_1 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \begin{bmatrix} l_1 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} R_{x_0}(\alpha_1) & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix}$$

$$\begin{bmatrix} R_{z_1}(q_2) & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \begin{bmatrix} 0 \\ 0 \\ \beta_2 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \begin{bmatrix} l_2 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} R_{x_1}(\alpha_2) & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ [0\ 0\ 0] & 1 \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} \tag{3.64}$$

3.2.3. Tabla de Denavit-Hartenberg

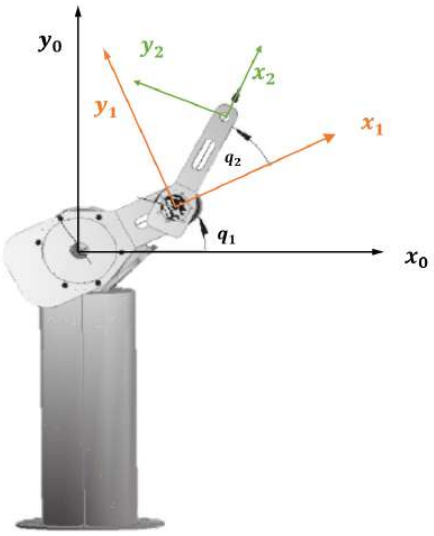


Figura 3.7: Robot manipulador de 2 GDL.

La tabla de Denavit-Hartenberg para el robot manipulador de 2 gdl de la figura 3.7 es:

DH del péndulo				
Eslabón	l_i	α_1	$\beta_i; d_i$	θ_1
1	l_1	0	β_1	q_1
2	l_2	0	β_2	q_2

3.2.4. Transformaciones homogéneas Denavit-Hartenberg

Para el robot de 2gdl se tiene:

$$H_0^1 = H_{R_{z_0}}(q_1) H_{T_{z_0}}(\beta_1) H_{T_{x_0}}(l_1) H_{R_{x_0}}(\alpha_1) \tag{3.65}$$

$$= \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 & 0 \\ \sin(q_1) & \cos(q_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \beta_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & l_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_1) & -\sin(\alpha_1) & 0 \\ 0 & \sin(\alpha_1) & \cos(\alpha_1) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos(q_1) & -\sin(q_1)\cos(\alpha_1) & \sin(q_1)\sin(\alpha_1) & l_1\cos(q_1) \\ \sin(q_1) & \cos(q_1)\cos(\alpha_1) & -\cos(q_1)\sin(\alpha_1) & l_1\sin(q_1) \\ 0 & \sin(\alpha_1) & \cos(\alpha_1) & \beta_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.66}$$

$$= \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 & l_1\cos(q_1) \\ \sin(q_1) & \cos(q_1) & 0 & l_1\sin(q_1) \\ 0 & 0 & 1 & \beta_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.67}$$

$$H_1^2 = H_{R_{z_1}}(q_2)H_{T_{z_1}}(\beta_2)H_{T_{x_1}}(l_2)H_{R_{x_1}}(\alpha_2) \tag{3.68}$$

$$= \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & 0 \\ \sin(q_2) & \cos(q_2) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \beta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & l_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_2) & -\sin(\alpha_2) & 0 \\ 0 & \sin(\alpha_2) & \cos(\alpha_2) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos(q_2) & -\sin(q_2)\cos(\alpha_2) & \sin(q_2)\sin(\alpha_2) & l_2\cos(q_2) \\ \sin(q_2) & \cos(q_2)\cos(\alpha_2) & -\cos(q_2)\sin(\alpha_2) & l_2\sin(q_2) \\ 0 & \sin(\alpha_2) & \cos(\alpha_2) & \beta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.69}$$

$$= \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & l_2\cos(q_2) \\ \sin(q_2) & \cos(q_2) & 0 & l_2\sin(q_2) \\ 0 & 0 & 1 & \beta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.70}$$

La transformación homogénea total se obtiene como $H_0^2 = H_0^1 H_1^2$:

$$H_0^2 = \begin{bmatrix} \cos(q_1+q_2) & -\sin(q_1+q_2) & 0 & l_1\cos(q_1)+l_2\cos(q_1+q_2) \\ \sin(q_1+q_2) & \cos(q_1+q_2) & 0 & l_1\sin(q_1)+l_2\sin(q_1+q_2) \\ 0 & 0 & 1 & \beta_1+\beta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.71}$$

La cinemática directa es:

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \boldsymbol{f}_R(\boldsymbol{q}) = \begin{bmatrix} l_1\cos(q_1)+l_2\cos(q_1+q_2) \\ l_1\sin(q_1)+l_2\sin(q_1+q_2) \\ \beta_1+\beta_2 \end{bmatrix} \tag{3.72}$$

3.2.5. Cinemática diferencial vectorial

Para el primer eslabón, se toma la ecuación (3.59) y se deriva respecto al tiempo:

$$\frac{d}{dt}\boldsymbol{p}_0 = \frac{d}{dt}R_{z_0}(q_1)\left[\begin{pmatrix} 0 \\ 0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} l_1 \\ 0 \\ 0 \end{pmatrix}\right] + R_{z0}(q_1)R_{x0}(\alpha_1)\boldsymbol{p}_1 \tag{3.73}$$

$$= \frac{d}{dt}\begin{bmatrix} l_1\cos(q_1) \\ l_1\,\mathrm{sen}(q_1) \end{bmatrix} = \begin{bmatrix} -l_1\,\mathrm{sen}(q_1)\dot{q}_1 \\ l_1\cos(q_1)\dot{q}_1 \end{bmatrix} \tag{3.74}$$

$$= \begin{bmatrix} -l1\,\mathrm{sen}(q_1) & 0 \\ 0 & l_1\cos(q_1) \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_1 \end{bmatrix} \tag{3.75}$$

Lo cual representa la velocidad $\boldsymbol{v}_1$ del primer eslabón.

$$\boldsymbol{v}_1 = \begin{bmatrix} -l_1 \operatorname{sen}(q_1)\dot{q}_1 \\ l_1 \cos(q_1)\dot{q}_1 \end{bmatrix} \tag{3.76}$$

$$\boldsymbol{p}_0 = R_{z_0}(q_1)\left[\begin{pmatrix}0\\0\\\beta_1\end{pmatrix} + \begin{pmatrix}l_1\\0\\0\end{pmatrix} + R_{x0}(\alpha_1)R_{z_1}(q_2)\left[\begin{pmatrix}0\\0\\\beta_2\end{pmatrix} + \begin{pmatrix}l_2\\0\\0\end{pmatrix} + R_{x1}(\alpha_2)\boldsymbol{p}_2\right]\right] \tag{3.77}$$

donde $\boldsymbol{p}_2 = \mathbf{0}$, representa el origen del sistema de referencia Σ_2.

$$\boldsymbol{p}_0 = R_{z_0}(q_1)\left[\begin{pmatrix}l_1\\0\\\beta_1\end{pmatrix} + R_{x0}(\alpha_1)R_{z_1}(q_2)\left[\begin{pmatrix}l_2\\0\\\beta_2\end{pmatrix}\right]\right] \tag{3.78}$$

$$\dot{\boldsymbol{p}}_0 = \dot{R}_{z_0}(q_1)\left[\begin{pmatrix}l_1\\0\\\beta_1\end{pmatrix} + R_{x0}(\alpha_1)R_{z_1}(q_2)\left[\begin{pmatrix}l_2\\0\\\beta_2\end{pmatrix}\right]\right] + R_{z_0}(q_1)R_{x_0}(\alpha_1)\dot{R}_{z_1}(q_2)\begin{bmatrix}l_2\\0\\\beta_2\end{bmatrix} \tag{3.79}$$

$$= \begin{bmatrix} \left(-l_1 \sin(q_1) - l_2 \sin(q_1+q_2)\right)\dot{q}_1 - l_2 \sin(q_1+q_2)\dot{q}_2 \\ \left(l_1 \cos(q_1) + l_2 \cos(q_1+q_2)\right)\dot{q}_1 + l_2 \cos(q_1+q_2)\dot{q}_2 \end{bmatrix} \tag{3.80}$$

$$\boldsymbol{v}_2 = \begin{bmatrix} -l_1 \sin(q_1) - l_2 \sin(q_1+q_2) & -l_2 \sin(q_1+q_2) \\ l_1 \cos(q_1) + l_2 \cos(q_1+q_2) & l_2 \cos(q_1+q_2) \end{bmatrix}\begin{bmatrix}\dot{q}_1\\\dot{q}_2\end{bmatrix} \tag{3.81}$$

Derivando respecto al tiempo la cinemática directa de la ecuación 3.72:

$$\frac{d}{dt}\begin{bmatrix}x_0\\y_0\\z_0\end{bmatrix} = \frac{d}{dt}\begin{bmatrix} l_1 \cos(q_1) + l_2 \cos(q_1+q_2) \\ l_1 \sin(q_1) + l_2 \sin(q_1+q_2) \\ \beta_1 + \beta_2 \end{bmatrix} = \frac{\partial \boldsymbol{f}(\boldsymbol{q})}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} \tag{3.82}$$

El Jacobiano se deduce de:

$$J(\boldsymbol{q}) = \frac{\partial \boldsymbol{f}(\boldsymbol{q})}{\partial \boldsymbol{q}} = \begin{bmatrix} -l_1 \sin(q_1) - l_2 \sin(q_1+q_2) & -l_2 \sin(q_1+q_2) \\ l_1 \cos(q_1) + l_2 \cos(q_1+q_2) & l_2 \cos(q_1+q_2) \end{bmatrix} \in \mathbb{R}^{2\times 2}$$

$$\begin{bmatrix}\dot{x}_0\\\dot{y}_0\\\dot{z}_0\end{bmatrix} = \begin{bmatrix} -l_1 \sin(q_1) - l_2 \sin(q_1+q_2) & -l_2 \sin(q_1+q_2) \\ l_1 \cos(q_1) + l_2 \cos(q_1+q_2) & l_2 \cos(q_1+q_2) \end{bmatrix}\begin{bmatrix}\dot{q}_1\\\dot{q}_2\end{bmatrix} \tag{3.83}$$

3.2.6. Rápidez de traslación

$$||\boldsymbol{v}_1||^2 = \left\| \begin{matrix} -l_1 \operatorname{sen}(q_1)\dot{q}_1 \\ l_1 \cos(q_1)\dot{q}_1 \end{matrix} \right\|^2 = v_1^T v_1 = l_1^2 \dot{q}_1^2 \tag{3.84}$$

$$||v_2||^2 = \left\| \begin{pmatrix} -l_1 \operatorname{sen}(q_1) - l_2 \operatorname{sen}(q_1+q_2) \end{pmatrix} \dot{q}_1 - l_2 \operatorname{sen}(q_1+q_2)\dot{q}_2 \\ \begin{pmatrix} l_1 \cos(q_1) + l_2 \cos(q_1+q_2) \end{pmatrix} \dot{q}_1 + l_2 \cos(q_1+q_2)\dot{q}_2 \right\|^2 \quad (3.85)$$

$$= v_2^T v_2$$

$$= \left[l_1^2 + l_2^2 + 2 l_1 l_2 \operatorname{sen}(q_2) \right] \dot{q}_1^2 + l_2^2 \dot{q}_2^2 + 2 \left[l_1 l_2 \operatorname{sen}(q_2) + \tfrac{2}{2} \right] \dot{q}_1 \dot{q}_2 \quad (3.86)$$

3.2.7. Modelo dinámico de un robot manipulador de 2 gdl

Para el robot manipulador de 2gdl que se muestra en la figura 3.8, y los parámetros de la TABLA 3.1 se aplica la siguiente metodología para obtener el modelo dinámico de un robot manipulador de n grados de libertad.

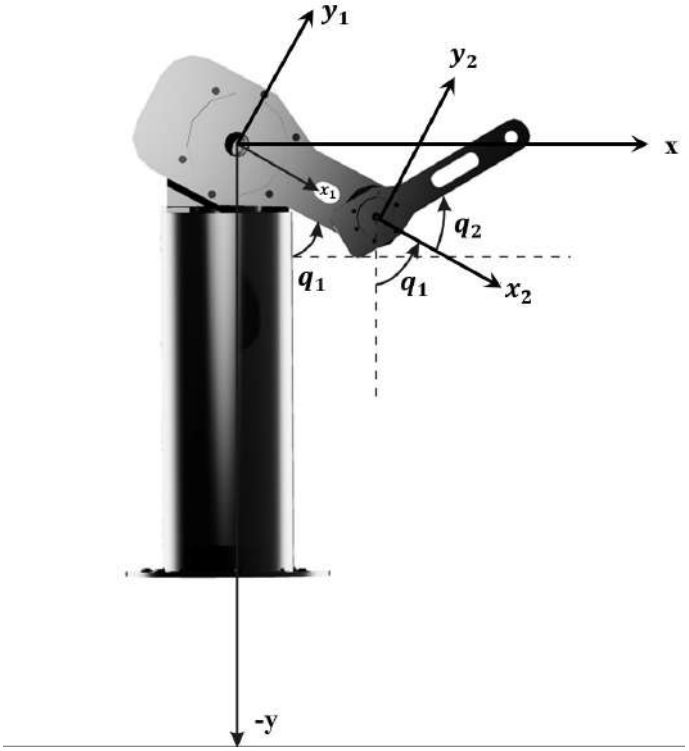


Figura 3.8: Robot manipulador de 2gdl.

Metodología para obtener el modelo dinámico de un robot de 2 grados de libertad en cadena cinemática abierta

1. Modelo de cinemática directa

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} l_{c1} \operatorname{sen}(q_1) \\ -l_{c1} \cos(q_1) \end{bmatrix} \quad (3.87)$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} l_1 \operatorname{sen}(q_1) + l_{c2} \operatorname{sen}(q_1+q_2) \\ -l_1 \cos(q_1) - l_{c2} \cos(q_1+q_2) \end{bmatrix} \quad (3.88)$$

Eslabón	Significado	Notación
	Masa del eslabón 1	m_1
	Longitud del eslabón 1	l_1
1 (Hombro)	inercia del eslabón 1	I_1
	Centro de masa del eslabón 1	l_{c1}
	Posición articular del estabón 1	q_1
	Masa del eslabón 2	m_2
	Longitud del eslabón 2	l_2
2 (Codo)	Inercia del eslabón 2	l_{c2}
	Centro de masa del eslabón 2	q_2
	Aceleración debida a la gravedad	g

Tabla 3.1: Parámetros del robot de 2gdl.

2. Cinemática diferencial

$$\boldsymbol{v}_1 = \frac{d}{dt}\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \frac{d}{dt}\begin{bmatrix} l_{c1}\operatorname{sen}(q_1) \\ -l_{c1}\cos(q_1) \end{bmatrix} = \begin{bmatrix} l_{c1}\cos(q_1) \\ l_{c1}\operatorname{sen}(q_1) \end{bmatrix}\dot{q}_1 \tag{3.89}$$

$$\boldsymbol{v}_2 = \frac{d}{dt}\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \frac{d}{dt}\begin{bmatrix} l_1\operatorname{sen}(q_1) + l_{c2}\operatorname{sen}(q_1+q_2) \\ -l_1\cos(q_1) - l_{c2}\cos(q_1+q_2) \end{bmatrix} \tag{3.90}$$

$$= \begin{bmatrix} l_1\cos(q_1) + l_{c2}\cos(q_1+q_2) & l_{c2}\cos(q_1+q_2) \\ l_1\operatorname{sen}(q_1) + l_{c2}\operatorname{sen}(q_1+q_2) & l_{c2}\operatorname{sen}(q_1+q_2) \end{bmatrix}\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \tag{3.91}$$

$$= \begin{bmatrix} \left[l_1\cos(q_1) + l_{c2}\cos(q_1+q_2)\right]\dot{q}_1 + \left[l_{c2}\cos(q_1+q_2)\right]\dot{q}_2 \\ \left[l_1\operatorname{sen}(q_1) + l_{c2}\operatorname{sen}(q_1+q_2)\right]\dot{q}_1 + l_{c2}\operatorname{sen}(q_1+q_2)\dot{q}_2 \end{bmatrix} \tag{3.92}$$

El cuadrado de la rapidez de cada eslabón se obtiene de la siguiente forma:

$$\| \boldsymbol{v}_1 \|^2 = \boldsymbol{v}_1^T\boldsymbol{v}_1 = \left[l_{c1}\cos(q_1)\dot{q}_1\right]^2 + \left[l_{c1}\operatorname{sen}(q_1)\dot{q}_1\right]^2 = l_{c1}^2\left[\cos^2(q_1) + \operatorname{sen}^2(q_1)\right]\dot{q}_1^2 \tag{3.93}$$

$$= l_{c1}\dot{q}_1^2 \tag{3.94}$$

$$\| \boldsymbol{v}_2 \|^2 = \boldsymbol{v}_2^T\boldsymbol{v}_2 = \left[\left[l_1\cos(q_1) + l_c2\cos(q_1+q_2)\right]\dot{q}_1 + l_{c2}\cos(q_1+q_2)\dot{q}_2\right]^2 + \left[\left[l_1\operatorname{sen}(q_1) + l_{c2}\operatorname{sen}(q_1+q_2)\right]\dot{q}_1 + l_{c2}\operatorname{sen}(q_1+q_2)\dot{q}_2\right]^2 \tag{3.95}$$

$$= \left[l_1\cos(q_1) + l_{c2}\cos(q_1+q_2)\right]^2\dot{q}_1^2 + l_{c2}^2\cos^2(q_1+q_2)\dot{q}_2^2 + 2\left[l_1\cos(q_1) + l_{c2}\operatorname{sen}(q_1+q_2)\right]\dot{q}_1\dot{q}_2 + \left[l_1\operatorname{sen}(q_1) + l_{c2}\operatorname{sen}(q_1+q_2)\right]^2\dot{q}_1^2 + l_{c2}^2\operatorname{sen}^2(q_1+q_2)\dot{q}_2^2 + 2\left[l_1\operatorname{sen}(q_1) + l_{c2}\operatorname{sen}(q_1+q_2)\right]l_{c2}\operatorname{sen}(q_1+q_2)\dot{q}_1\dot{q}_2 \tag{3.96}$$

$$= \left[l_1^2 + l_{c2}^2 + 2l_1l_{c2}\cos(q_2)\right]\dot{q}_1^2 + l_{c2}^2\dot{q}_2^2 + 2\left[l_1l_{c2}\cos(q_2) + l_{c2}^2\right]\dot{q}_1\dot{q}_2 \tag{3.97}$$

3. Modelo de energía:

La energía cinética $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$ del robot manipulador de 2gdl está dada por la siguiente expresión:

$$\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}_1) = \frac{1}{2}m_1\boldsymbol{v}_1^T\boldsymbol{v}_1 + \frac{1}{2}\mathtt{I}_1\dot{q}_1^2 + \frac{1}{2}m_2\boldsymbol{v}_2^T\boldsymbol{v}_2 + \frac{1}{2}\mathtt{I}_2\left[\dot{q}_1 + \dot{q}_2\right] \tag{3.98}$$

$$= \frac{1}{2}\left[m_1l_{c1}^2 + \mathtt{I}_1 + \mathtt{I}_2 + m_2l_1^2 + m_2l_{c2}^2 + 2m_2l_1l_{c2}\cos(q_2)\right]\dot{q}_1^2 + \frac{1}{2}\left[\mathtt{I}_2 + m_2l_{c2}^2\right]\dot{q}_2^2$$

$$+\left[m_2 l_1 l_{c2}\cos(q_2)+m_2 l_{c2}^2+\mathtt{I}_2\right]\dot{q}_1\dot{q}_2 \tag{3.99}$$

La energía potencial $\mathcal{U}(\mathbf{q})$ del centro de masa para ambos eslabones está dada como:

$$\mathcal{U}(\mathbf{q}) = m_1 g l_{c1}\left[1-\cos(q_1)\right]+m_2 g\left[\left[l_1+l_{c2}\right]+\left[l_1\cos(q_1)+l_{c2}\cos(q_1+q_2)\right]\right] \tag{3.100}$$

El Lagrangiano está definido como:

$$\begin{aligned}\mathcal{L}(\mathbf{q},\dot{\mathbf{q}}) &= \mathcal{K}(\mathbf{q},\dot{\mathbf{q}})-\mathcal{U}(\mathbf{q}) &(3.101)\\ &= \frac{1}{2}\left[m_1 l_{c1}^2+\mathtt{I}_1+\mathtt{I}_2+m_2 l_1^2+m_2 l_{c2}^2+2m_2 l_1 l_{c2}\cos(q_2)\right]\dot{q}_1^2+\frac{1}{2}\left[\mathtt{I}_2+m_2 l_{c2}^2\right]\dot{q}_2^2\\ &\quad+\left[m_2 l_1 l_{c2}\cos(q_2)+m_2 l_{c2}^2+\mathtt{I}_2\right]\dot{q}_1\dot{q}_2-m_1 g l_{c1}\left[1-\cos(q_1)\right]\\ &\quad-m_2 g\left[\left[l_1+l_{c2}\right]+\left[l_1\cos(q_1)+l_{c2}\cos(q_1+q_2)\right]\right] &(3.102)\end{aligned}$$

4. Ecuaciones de movimiento de Euler-Lagrange:

$$\begin{aligned}\frac{\partial\mathcal{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{q}_1} &= \left[m_1 l_{c1}^2+\mathtt{I}_1+\mathtt{I}_2+m_2 l_1^2+m_2 l_{c2}^2+2m_2 l_1 l_{c2}\cos(q_2)\right]\dot{q}_1+\left[m_2 l_1 l_{c2}\cos(q_2)+m_2 l_{c2}^2+\mathtt{I}_2\right]\dot{q}_2 &(3.103)\\ \frac{d}{dt}\left[\frac{\partial\mathcal{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{q}_1}\right] &= \left[m_1 l_{c1}^2+\mathtt{I}_1+\mathtt{I}_2+m_2 l_1^2+m_2 l_{c2}^2+2m_2 l_1 l_{c2}\cos(q_2)\right]\ddot{q}_1+\left[m_2 l_1 l_{c2}\cos(q_2)+m_2 l_{c2}^2+\mathtt{I}_2\right]\ddot{q}_2\\ &\quad-2m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1\dot{q}_2-m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_2^2 &(3.104)\\ -\frac{\partial\mathcal{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial q_1} &= m_1 g l_{c1}\,\mathrm{sen}(q_1)+m_2 g\left[l_1\,\mathrm{sen}(q_1)+l_{c2}\,\mathrm{sen}(q_1+q_2)\right] &(3.105)\\ \frac{\partial\mathcal{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{q}_2} &= \left[m_2 l_1 l_{c2}\cos(q_2)+m_2 l_{c2}^2+\mathtt{I}_2\right]\dot{q}_1+\left[\mathtt{I}_2+m_2 l_{c2}^2\right]\dot{q}_2 &(3.106)\\ \frac{d}{dt}\left[\frac{\partial\mathcal{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial\dot{q}_2}\right] &= \left[m_2 l_1 l_{c2}\cos(q_2)+m_2 l_{c2}^2+\mathtt{I}_2\right]\ddot{q}_1+\left[\mathtt{I}_2+m_2 l_{c2}^2\right]\ddot{q}_2-m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1\dot{q}_2 &(3.107)\\ -\frac{\partial\mathcal{L}(\mathbf{q},\dot{\mathbf{q}})}{\partial q_2} &= m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1+m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1\dot{q}_2+m_2 g l_{c2}\,\mathrm{sen}(q_1+q_2) &(3.108)\\ \tau_1 &= \left[m_1 l_{c1}^2+\mathtt{I}_1+\mathtt{I}_2+m_2 l_1^2+m_2 l_{c2}^2+2m_2 l_1 l_{c2}\cos(q_2)\right]\ddot{q}_1+\left[m_2 l_1 l_{c2}\cos(q_2)+m_2 l_{c2}^2+\mathtt{I}_2\right]\ddot{q}_2\\ &\quad-2m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1\dot{q}_2-m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_2^2+m_1 g l_{c1}\,\mathrm{sen}(q_1)+m_2 g\left[l_1\,\mathrm{sen}(q_1)+l_{c2}\,\mathrm{sen}(q_1+q_2)\right]\\ &\quad+b_1\dot{q}_1 &(3.109)\\ \tau_2 &= \left[m_2 l_1 l_{c2}\cos(q_2)+m_2 l_{c2}^2+\mathtt{I}_2\right]\ddot{q}_1+\left[\mathtt{I}_2+m_2 l_{c2}^2\right]\ddot{q}_2-\cancel{m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1\dot{q}_2}+m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1^2\\ &\quad+\cancel{m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1\dot{q}_2}+m_2 g l_{c2}\,\mathrm{sen}(q_1+q_2)+b_2\dot{q}_2 &(3.110)\end{aligned}$$

De forma matricial se tiene:

$$\begin{aligned}\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} &= \underbrace{\begin{bmatrix}m_1 l_{c1}^2+m_2 l_1^2+m_2 l_{c2}^2+2m_2 l_1 l_{c2}\cos(q_2)+\mathtt{I}_1+\mathtt{I}_2 & m_2 l_{c2}^2+m_2 l_1 l_{c2}\cos(q_2)+\mathtt{I}_2\\ m_2 l_{c2}^2+m_2 l_1 l_{c2}\cos(q_2)+\mathtt{I}_2 & m_2 l_{c2}^2+\mathtt{I}_2\end{bmatrix}}_{M(\mathbf{q})}\begin{bmatrix}\ddot{q}_1\\ \ddot{q}_2\end{bmatrix}\\ &\quad+\underbrace{\begin{bmatrix}-2m_2 l_1 l_{c}2\,\mathrm{sen}(q_2)\dot{q}_2 & -m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_2\\ m_2 l_1 l_{c2}\,\mathrm{sen}(q_2)\dot{q}_1 & \end{bmatrix}}_{C(\mathbf{q},\dot{\mathbf{q}})}\begin{bmatrix}\dot{q}_1\\ \dot{q}_2\end{bmatrix}\\ &\quad+\underbrace{g\begin{bmatrix}l_{c1}m_1\,\mathrm{sen}(q_1)+m_2 l_1\,\mathrm{sen}(q_1)+m_2 l_{c2}\,\mathrm{sen}(q_1+q_2)\\ l_{c2}m_2\,\mathrm{sen}(q_1+q_2)\end{bmatrix}}_{\mathbf{g}(\mathbf{q})}+\underbrace{\begin{bmatrix}b_1 & 0\\ 0 & b_2\end{bmatrix}\begin{bmatrix}\dot{q}_1\\ \dot{q}_2\end{bmatrix}}_{\mathbf{f}_f(\dot{\mathbf{q}})} &(3.111)\end{aligned}$$

Capítulo 4

Perceptrón aplicado a robot manipulador de 2gdl

Se implementa una red neuronal de una sola capa al lazo cerrado de control articular y cartesiano que ″aprenda″una base de datos determinada para obtener los parámetros correspondientes a los pesos w_i requeridos para la implementación de la misma, se aplica el método de identificación paramétrica la cual facilita la obtención de dichos pesos y se propone un control hiperbolico exponencial en ambos lazos de control. La TABLA 4.1 muestra los parámetros de simulación para el robot manipulador de 2gdl:

Eslabón	Significado	Notación	Valor
	Masa del eslabón 1	m_1	23.902 Kg
	Longitud del eslabón 1	l_1	0.45 m
1 (Hombro)	inercia del eslabón 1	I_1	1.266 Nmseg^2/rad
	Centro de masa del eslabón 1	l_{c1}	0.091 m
	Coef. de fricción viscosa	b_1	2.288 Nmseg/rad
	Masa del eslabón 2	m_2	3.88 Kg
	Longitud del eslabón 2	l_2	0.45 m
2 (Codo)	Inercia del eslabón 2	I_2	0.093 Nmseg^2/rad
	Centro de masa del eslabón 2	l_{c2}	0.048 m
	Coef. de fricción viscosa 2	b_2	0.175 Nmseg/rad
	Aceleración debida a la gravedad	g	9.81 m/seg^2

Tabla 4.1: Parámetros de simulación para el robot de 2gdl.

Y sea la cinemática inversa, que se muestra en la figura 4.1, del robot de 2gdl definida como:

$$q_2 = \texttt{acos}\left(\frac{x_0^2+y_0^2-l_1^2-l_2^2}{2l_1l_2}\right) \tag{4.1}$$

$$q_1 = \texttt{atan}\left(\frac{y_0}{x_0}\right) - \texttt{atan}\left(\frac{l_2\,\text{sen}(q_2)}{l_1+l_2\cos(q_2)}\right) \tag{4.2}$$

Y el modelo de cinemática directa del robot de 2gdl respecto al primer cuadrante:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} l_1\cos(q_1)+l_2\cos(q_1+q_2) \\ l_1\,\text{sen}(q_1)+l_2\,\text{sen}(q_1+q_2) \end{bmatrix} \tag{4.3}$$

Se puede mostrar la representación en lazo cerrado de control del robot manipulador de 2gdl de la figura 4.2. En la cual se observa que es necesario aplicar la cinemática inversa para obtener las posiciones articulares deseadas, a partir de las posiciones cartesianas deseadas dadas por el usuario, ya que, el lenguaje de programación de las posiciones articulares del robot manipulador esta dado en radianes, mientras que, para cualquier ser humano la posición esta en coordenadas cartesianas (metros). En este caso, se propone un

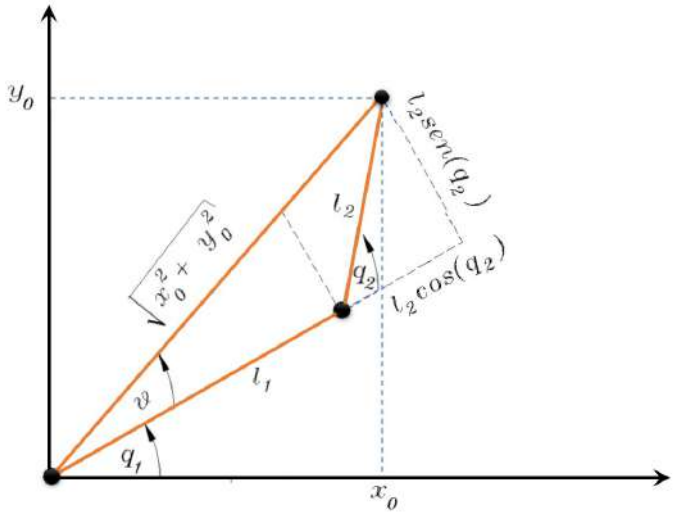


Figura 4.1: Cinemática inversa del robot manipulador de 2gdl.

control articular el cual posee características hiperbólicas-exponenciales, como se muestra en la ecuación (4.4):

$$\boldsymbol{\tau} = K_p(I - \alpha e^{-\alpha \cosh(\tilde{\mathbf{q}})}) \operatorname{senh}(\tilde{\mathbf{q}}) - K_v(I - \beta e^{-\beta \cosh(\dot{\mathbf{q}})}) \operatorname{senh}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) \tag{4.4}$$

donde:

- α y β son constantes definidas por el diseñador.
-

$$\underbrace{(I - \alpha e^{-\alpha \cosh(\tilde{\mathbf{q}})}) \operatorname{senh}(\tilde{\mathbf{q}})}_{\textit{Función de control exponencial}} = \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix} - \begin{bmatrix} \alpha e^{-\alpha \cosh(\tilde{q}_1)} & \cdots & 0 \\ \vdots & \ddots & 0 \\ 0 & \cdots & \alpha e^{-\alpha \cosh(\tilde{q}_n)} \end{bmatrix} \begin{bmatrix} \operatorname{senh}(\tilde{q}_1) \\ \operatorname{senh}(\tilde{q}_2) \\ \vdots \\ \operatorname{senh}(\tilde{q}_n) \end{bmatrix} \tag{4.5}$$

-

$$\underbrace{(I - \beta e^{-\beta \cosh(\dot{\mathbf{q}})}) \operatorname{senh}(\dot{\mathbf{q}})}_{\substack{\textit{Función de inyección} \\ \textit{de amortiguamiento}}} = \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix} - \begin{bmatrix} \beta e^{-\beta \cosh(\dot{q}_1)} & \cdots & 0 \\ \vdots & \ddots & 0 \\ 0 & \cdots & \beta e^{-\beta \cosh(\dot{q}_n)} \end{bmatrix} \begin{bmatrix} \operatorname{senh}(\dot{q}_1) \\ \operatorname{senh}(\dot{q}_2) \\ \vdots \\ \operatorname{senh}(\dot{q}_n) \end{bmatrix} \tag{4.6}$$

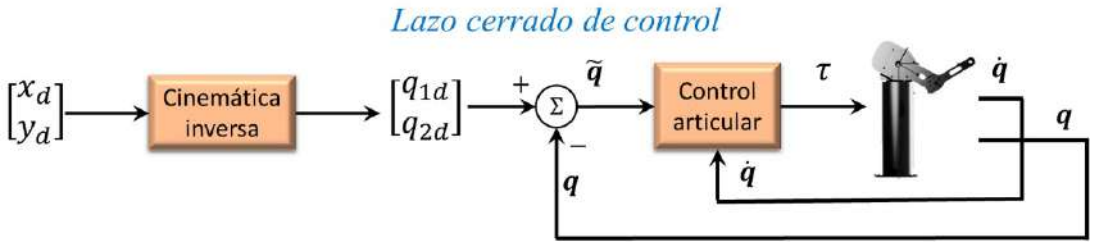


Figura 4.2: Diagrama de bloques del lazo cerrado de control para el robot manipulador de 2gdl.

Se diseña una red neuronal lineal de una capa, capaz de imitar las posiciones articulares q_1 y q_2 del robot manipulador, las cuales se interpretan como posiciones deseadas q_{d1} y q_{d2} a partir de conocer las posiciones

cartesianas deseadas x_d y y_d, en la figura 4.3 se muestra el diseño implementado.

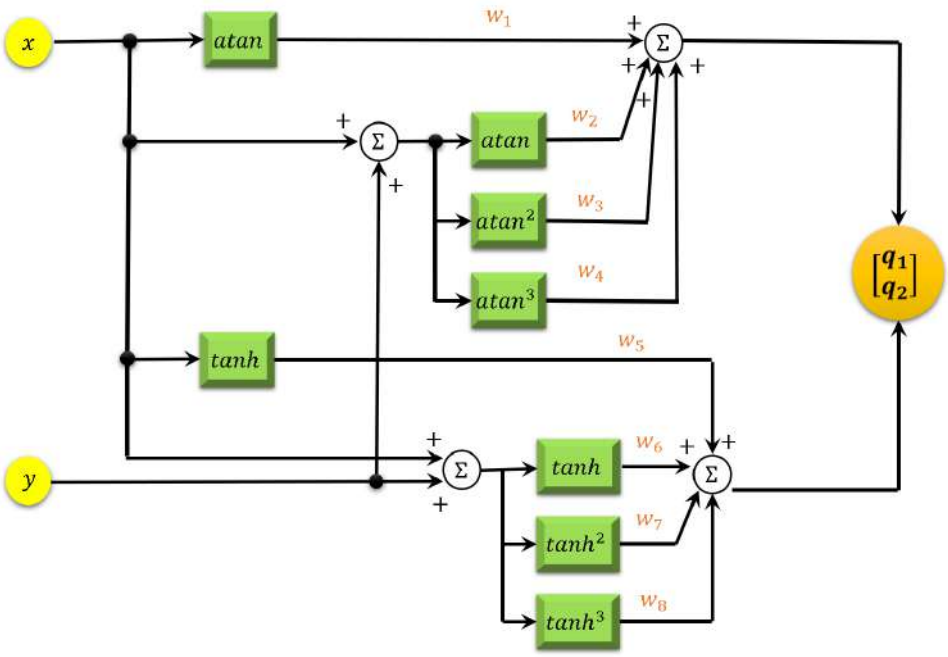


Figura 4.3: Diagrama de red neuronal.

Aplicando el método de identificación paramétrica se tiene el regresor vectorial lineal:

$$\begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} \texttt{atan}(x) & \texttt{atan}(\texttt{x}+\texttt{y}) & \texttt{atan}^2(\texttt{x}+\texttt{y}) & \texttt{atan}^3(\texttt{x}+\texttt{y}) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \tanh(x) & \tanh(x+y) & \tanh^2(x+y) & \tanh^3(x+y) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \\ \theta_8 \end{bmatrix} \tag{4.7}$$

Donde el vector $\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \vdots \\ \theta_8 \end{bmatrix}$ corresponde a los pesos $\begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ \vdots \\ w_8 \end{bmatrix}$.

Finalmente se tiene la representación en diagrama de bloques, figura 4.4 donde se muestra la aplicación de la red neuronal en el lazo cerrado de control articular.

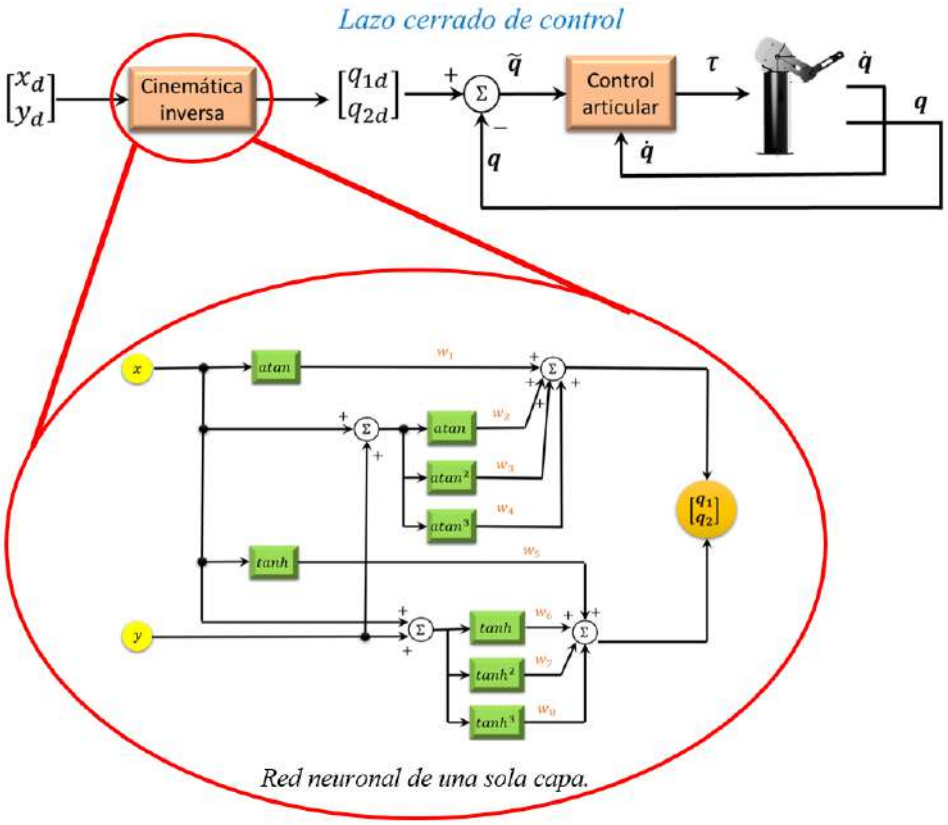


Figura 4.4: Aplicación de la red neuronal en el lazo cerrado de control del robot manipulador de 2gdl.

En la figura 4.5 se muestran las posiciones articulares deseadas las cuales son $q_{d_1} = 45°$ y $q_{d_2} = 90°$. De tal forma que al implementar la red neuronal propuesta en el lazo cerrado de control se lleguen a ambas posiciones con el mínimo error posible.

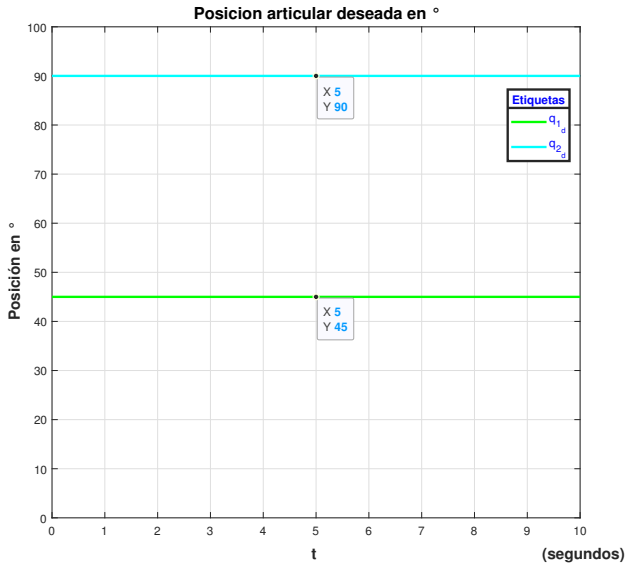
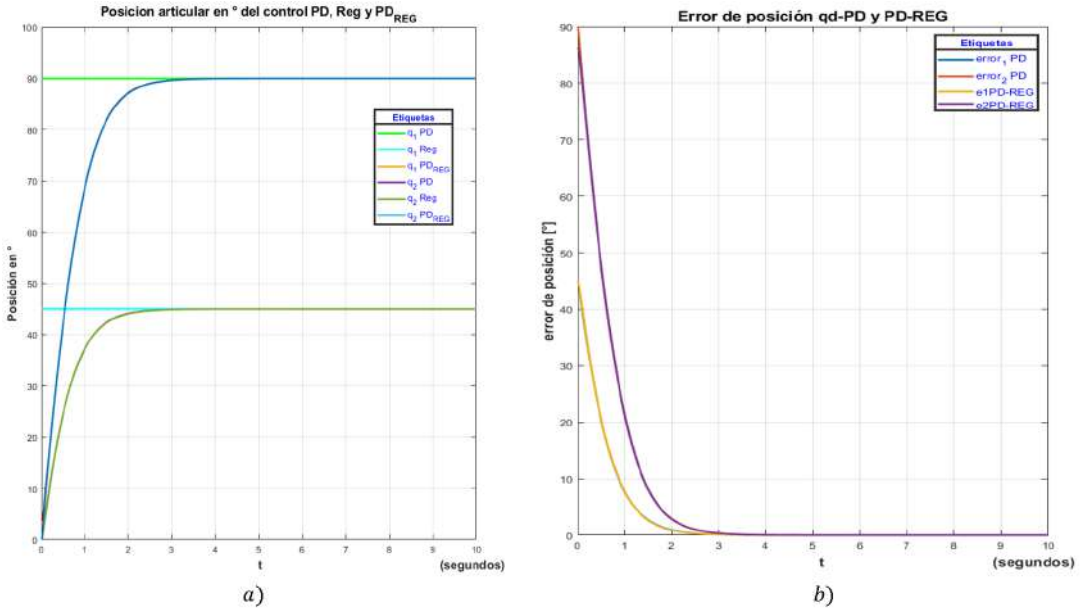


Figura 4.5: Posiciones articulares deseadas, línea verde q_{d_1} y línea azul q_{d_2}.

Los resultados obtenidos se muestran en las siguientes figuras:



a) *b)*

Figura 4.6: Comparativa de posiciones deseadas VS posiciones obtenidas por el regresor y finalmente el nuevo modelo de regresor lineal implementado en el lazo cerrado de control PD.

En la figura 4.6 se muestra la comparativa de posiciones articulares, donde q_1PD, q_2PD representan las posiciones articulares aplicando el lazo cerrado de control PD hiperbólico-exponencial propuesto y representa el sistema original, el cual, debe ser replicado por la nueva propuesta de red neuronal y donde se trabaja la base de datos de las posiciones articulares para obtener la cinemática directa, se observa que la señal de salida obtenida no presenta oscilaciones y es críticamente amortiguada dada la naturaleza del control PD, de tal forma que el control proporcional con la función de control exponencial es directamente proporcional al error de posición, mientras que la acción de control derivativa representada con una ganancia derivativa por la función de inyección de amortiguamiento elimina las oscilaciones del sistema por medio de la velocidad articular $\dot{\mathbf{q}}$ mejorando el desempeño del control proporcional.

Las señales q_1Reg y q_2Reg representan el vector de posiciones articulares estimadas con la propuesta de red neuronal a partir de la cinemática directa y finalmente q_1PD_{REG}, q_2PD_{REG} son las posiciones articulares obtenidas del lazo cerrado de control PD hiperbólico-exponencial sustituyendo la cinemática inversa por la propuesta de red neuronal para el robot manipulador de 2gdl.

Se obtiene el vector pesos:

$$\begin{bmatrix}\theta_1\\ \theta_2\\ \theta_3\\ \theta_4\\ \theta_5\\ \theta_6\\ \theta_7\\ \theta_8\end{bmatrix} = \begin{bmatrix}w_1\\ w_2\\ w_3\\ w_4\\ w_5\\ w_6\\ w_7\\ w_8\end{bmatrix} = \begin{bmatrix}-1.4441\\ 3.6166\\ -5.7279\\ 3.7208\\ -2.6529\\ 9.7419\\ -18.591\\ 12.139\end{bmatrix} \tag{4.8}$$

El error de posición articular en grados es:

$$\tilde{\mathbf{q}} = \begin{bmatrix} 0.0066962 \\ 0.0058275 \end{bmatrix} = \begin{bmatrix} 6.696 \times 10^{-3} \\ 5.827 \times 10^{-3} \end{bmatrix} \tag{4.9}$$

Y las normas euclidianas $\mathcal{L}_2$ para el control PD hiperbólico-exponencial en comparación con lo obtenido al sustituir la red neuronal se muestran en la tabla 4.2. La ecuación (4.9) nos muestra que el error de

$\mathcal{L}_i$	PD	RN
$\mathcal{L}_2$	0.3391	0.3390

Tabla 4.2: Comparativa de $\mathcal{L}_2$ para el control PD hiperbólico-exponencial y la red neuronal propuesta.

posición esta en milésimas, lo cual lo hace bastante adecuada para su implementación, ya que es un valor que es cercano a cero y su norma euclidiana $\mathcal{L}_2$ lo hace bastante competitivo respecto a otros controladores, como se observa en (Reyes-Cortes y Al-Hadithi, 2020; Suárez-Nuño, Cruz-Zavala y Nuño, 2019).

Se aplica la misma metodología para el caso cartesiano, de tal forma que se pueda sustituir el modelo de cinemática directa de la figura 4.7 por una red neuronal lineal de una capa, la cual es propuesta.

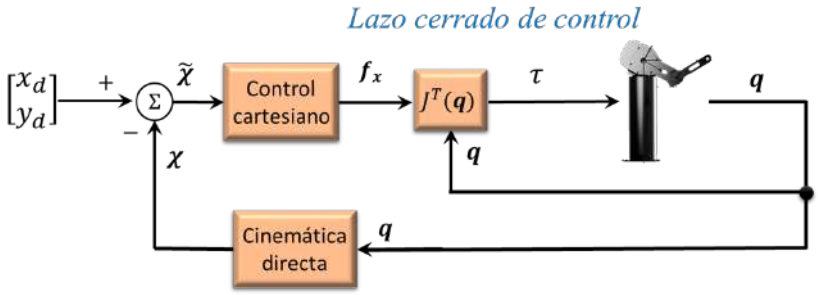


Figura 4.7: Diagrama de bloques del lazo cerrado de control cartesiano.

En la figura 4.8 se muestra el diagrama de red neuronal propuesto.

Aplicando el método de identificación paramétrica se tiene el regresor vectorial lineal de la ecuación 4.10.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \mathtt{atan}(q_1) & \mathtt{atan}(\mathtt{q}_1+\mathtt{q}_2) & \mathtt{atan}^2(\mathtt{q}_1+\mathtt{q}_2) & \mathtt{atan}^3(\mathtt{q}_1+\mathtt{q}_2) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \tanh(q_1) & \tanh(q_1+q_2) & \tanh^2(q_1+q_2) & \tanh^3(q_1+q_2) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \\ \theta_8 \end{bmatrix} \tag{4.10}$$

Finalmente se tiene la representación en diagrama de bloques, figura 4.9 donde se muestra la aplicación de la red neuronal en el lazo cerrado de control cartesiano.

Y para este caso se implementa la propuesta de control cartesiano hiperbólico-exponencial que se muestra en la ecuación (4.12).

$$\boldsymbol{\tau} = J^T(\mathbf{q})\mathbf{f}(\mathbf{x}) + \mathbf{g}(\boldsymbol{\chi}) \tag{4.11}$$

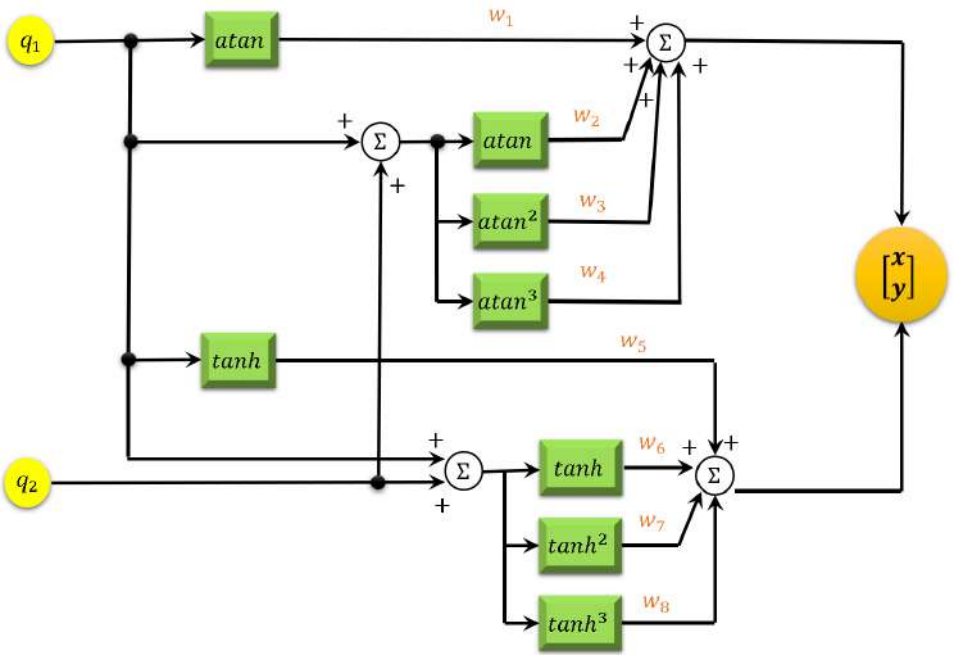


Figura 4.8: Diagrama de red neuronal.

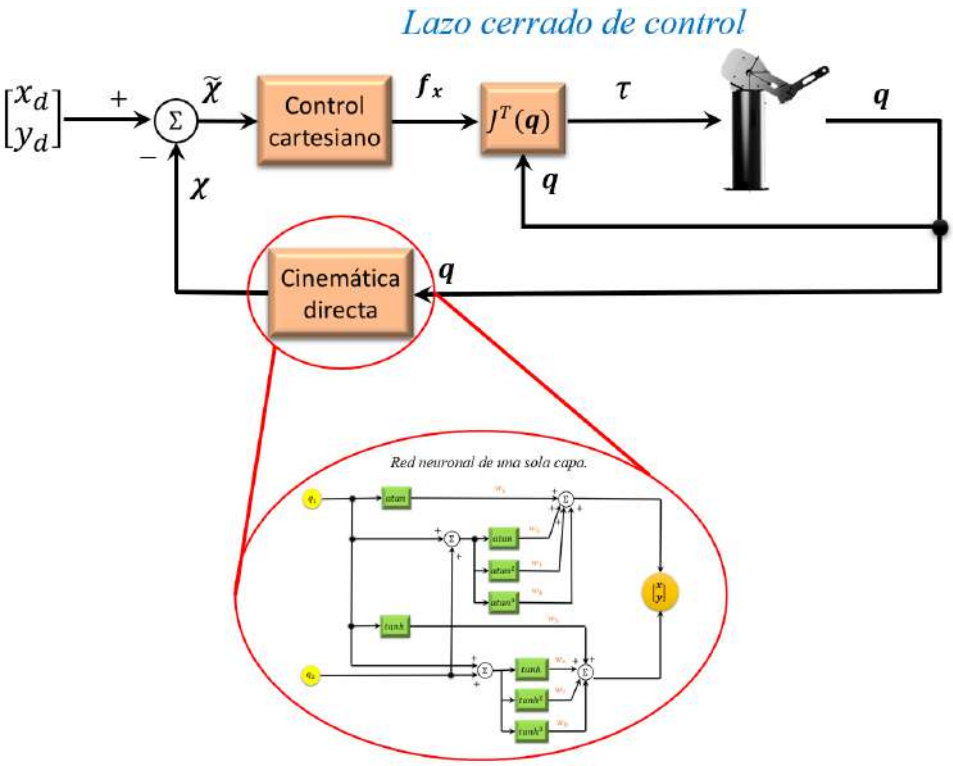


Figura 4.9: Diagrama de bloques

donde:

$$\mathbf{f}(\mathbf{x}) \quad = \quad K_p(I - \alpha e^{-\alpha \cosh(\tilde{\chi})}) \operatorname{senh}(\tilde{\chi}) - K_v(I - \beta e^{-\beta \cosh(\dot{\chi})}) \operatorname{senh}(\dot{\chi}) \tag{4.12}$$

- α y β son constantes definidas por el diseñador.

- $$\underbrace{(I-\alpha e^{-\alpha\cosh(\tilde{\chi})})\,\mathrm{senh}(\tilde{\chi})}_{\textit{Función de control exponencial}} = \left[\begin{bmatrix}1 & \cdots & 0\\ \vdots & \ddots & \vdots\\ 0 & \cdots & 1\end{bmatrix} - \begin{bmatrix}\alpha e^{-\alpha\cosh(\tilde{\chi}_1)} & \cdots & 0\\ \vdots & \ddots & 0\\ 0 & \cdots & \alpha e^{-\alpha\cosh(\tilde{\chi}_n)}\end{bmatrix}\right]\begin{bmatrix}\mathrm{senh}(\tilde{\chi}_1)\\ \mathrm{senh}(\tilde{\chi}_2)\\ \vdots\\ \mathrm{senh}(\tilde{\chi}_n)\end{bmatrix} \tag{4.13}$$

- $$\underbrace{(I-\beta e^{-\beta\cosh(\dot{\chi})})\,\mathrm{senh}(\dot{\chi})}_{\substack{\textit{Función de inyección}\\ \textit{de amortiguamiento}}} = \left[\begin{bmatrix}1 & \cdots & 0\\ \vdots & \ddots & \vdots\\ 0 & \cdots & 1\end{bmatrix} - \begin{bmatrix}\beta e^{-\beta\cosh(\dot{\chi}_1)} & \cdots & 0\\ \vdots & \ddots & 0\\ 0 & \cdots & \beta e^{-\beta\cosh(\dot{\chi}_n)}\end{bmatrix}\right]\begin{bmatrix}\mathrm{senh}(\dot{\chi}_1)\\ \mathrm{senh}(\dot{\chi}_2)\\ \vdots\\ \mathrm{senh}(\dot{\chi}_n)\end{bmatrix} \tag{4.14}$$

En la figura 4.10 se muestran las posiciones cartesianas deseadas que ingresan al lazo cerrado de control y definen la posición en la cual debe colocarse el extremo final del robot manipulador de dos grados de libertad, las cuales se encuentran ubicadas en el primer cuadrante $x = 0.55m$ y $y = 0.45m$.

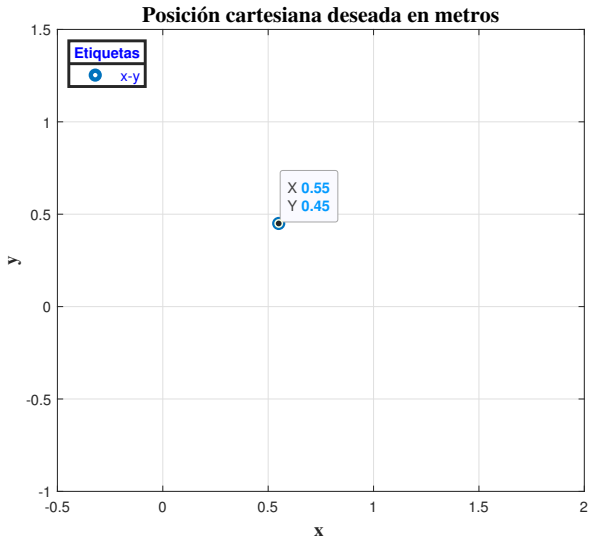


Figura 4.10: Posición cartesiana deseada en metros $x = 0.55m$ y $y = 0.45m$.

Se obtiene el vector de pesos correspondiente al regresor vectorial de la ecuación 4.10:

$$\begin{bmatrix}\theta_1\\ \theta_2\\ \theta_3\\ \theta_4\\ \theta_5\\ \theta_6\\ \theta_7\\ \theta_8\end{bmatrix} = \begin{bmatrix}w_1\\ w_2\\ w_3\\ w_4\\ w_5\\ w_6\\ w_7\\ w_8\end{bmatrix} = \begin{bmatrix}0.34898\\ -0.091626\\ 1.8515\\ -1.3297\\ 0.79024\\ -6.3798\\ 11.35\\ -5.2566\end{bmatrix} \tag{4.15}$$

El error de posición cartesiana en metros es:

$$\tilde{\chi} = \begin{bmatrix}0.00013546\\ 0.0013826\end{bmatrix} = \begin{bmatrix}13.546\times10^{-5}\\ 1.382\times10^{-3}\end{bmatrix} \tag{4.16}$$

Y las normas euclidianas $\mathcal{L}_2$ para el control cartesiano PD hiperbólico-exponencial en comparación con lo obtenido al sustituir la red neuronal se muestra en la tabla 4.3. La ecuación (4.16) nos muestra que el

$\mathcal{L}_i$	PD	RN
$\mathcal{L}_2$	0.3433	0.3229

Tabla 4.3: Comparativa de $\mathcal{L}_2$ para el control PD hiperbólico-exponencial y la red neuronal propuesta.

error de posición sobre el eje x esta en cien-milésimas mientras que el error de posición sobre el eje y esta en milésimas, ambos valores son cercanos a cero, y su normal euclidiana $\mathcal{L}_2$ lo hace bastante competitivos respecto a otros controladores, como se observa en (Reyes-Cortes y Al-Hadithi, 2020; Suárez-Nuño, Cruz-Zavala y Nuño, 2019).

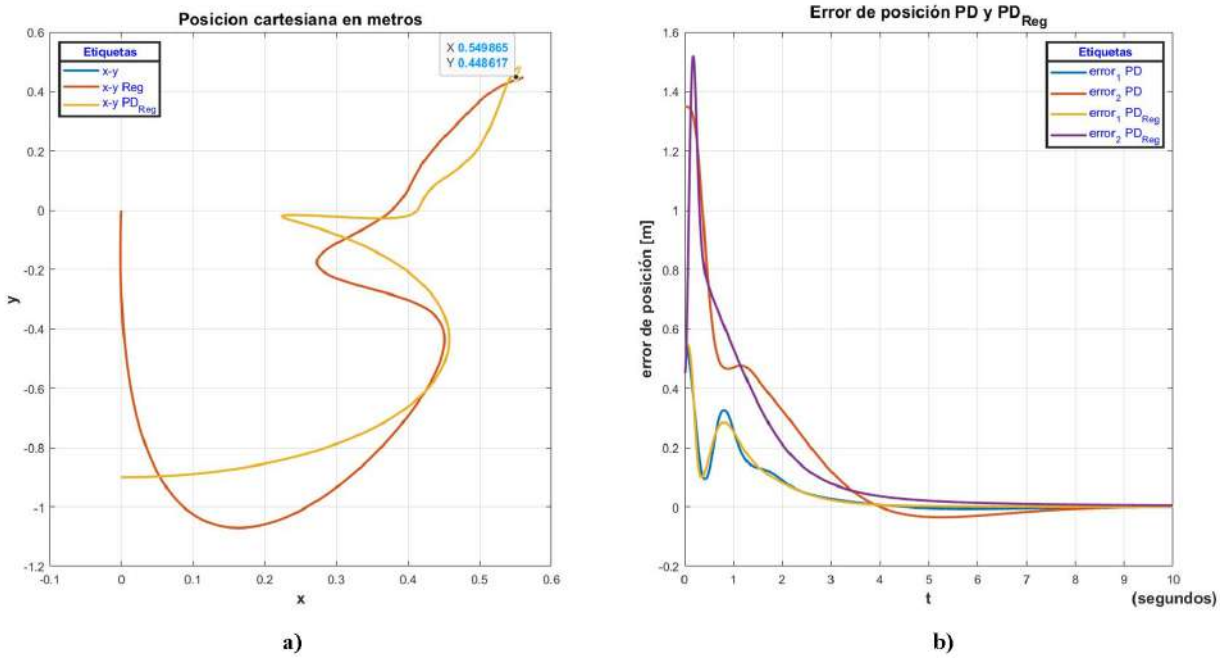


Figura 4.11: Comparativa de la posición cartesiana obtenida entre la cinemática directa, el regresor propuesto y el control propuesto en conjunto con el regresor.

En la figura 4.11a) se observan las gráficas comparativas de posición cartesiana, donde $x-y$ representan el lazo cerrado de control hiperbólico-exponencial cartesiano original, el cual debe ser replicado por la nueva propuesta de red neuronal, la señal de salida que se obtiene evidencia que se tiene una posición base alineada en el eje $-y$, por lo tanto, para llegar a la posición deseada el recorrido que el extremo final proyecta es del cuarto cuadrante al primer cuadrante, esto significa que la ubicación más cercana para el extremo final se encuentra sobre el eje x, por tal motivo, se tiene un error de posición en cienmilésimas, esto también queda expuesto en la gráfica del error de posición donde error$_1$PD$=\widetilde{x}$. Esto se debe a que la estabilidad asintótica en espacio cartesiano es local por el rango completo del jacobiano, por lo tanto, entre más cerca este el extremo final de la posición deseada el error será mucho menor.

La gráfica de la propuesta esta representada por la señal $x-y$ Reg se observa que sobrepasa la posición sobre el eje x. Sin embargo, una vez que es aplicado el lazo cerrado de control PD hiperbólico-exponencial con la red neuronal propuesta ($x-y$ PD$_{Reg}$), se llega de manera genuina a la posición cartesiana obtenida del control original tal y como se observa con el índice de desempeño $\mathcal{L}_2$, en la figura 4.11b) se observa que las gráficas del error son bastantes similares y tienden a cero.

Capítulo 5

Ciclo previo de aprendizaje

Dados los conocimientos aprendidos en los capítulos anteriores, se conocen diversos algoritmos canónicos de control de aprendizaje iterativo, sin embargo, nos enfocaremos únicamente a desarrollar el ciclo previo de aprendizaje en este tema de tesis. Se plantean sus respectivas demostraciones de forma recursiva y se realizan simulaciones para diversos sistemas de primer y segundo orden. La configuración del ciclo previo de aprendizaje se muestra en la figura 5.1, esta es implementada para la solución del sistema escalar y vectorial de forma recursiva.

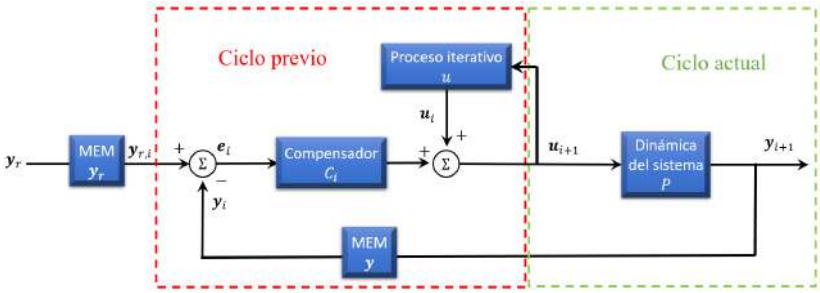


Figura 5.1: Diagrama de bloques del ciclo previo de aprendizaje.

5.1. Sistema dinámico lineal escalar

Sea el sistema dinámico escalar de la ecuación (5.1) y dada una posición deseada y_d. Demostrar que el proceso iterativo de un sistema escalar existe y que tiene estabilidad asintótica:

$$\dot{x} = -ax(t) + bu(t) \tag{5.1}$$
$$y(t) = cx(t) \tag{5.2}$$

donde:

- $x(t) \in \mathbb{R}$ representa la variable de estado del sistema.
- $u(t) \in \mathbb{R}$ es la entrada del sistema.
- $a, b \in \mathbb{R}_+$ son constantes definidas positivas.

La solución del sistema escalar de forma recursiva esta dada por:

$$
\begin{aligned}
x_k &= \phi x_{k-1} + \gamma u_{k-1} && (5.3)\\
y_k &= c x_k && (5.4)\\
\widetilde{y} &= y_d - y_k && (5.5)\\
u_k &= u_{k-1} + \beta \widetilde{y}_k && (5.6)\\
\widetilde{y}_{k+1} &= \widetilde{y}_k - \left(y_{k+1} - y_k\right) && (5.7)
\end{aligned}
$$

donde:

$$
\begin{aligned}
y_k &= c x_k \\
&= c\left(\phi x_{k-1} + \gamma u_{k-1}\right) && (5.8)\\
y_{k+1} &= c x_{k+1} \\
&= c\left(\phi x_k + \gamma u_k\right) && (5.9)
\end{aligned}
$$

Sustituyendo la ecuación (5.9) en la ecuación (5.7) se tiene:

$$
\begin{aligned}
\widetilde{y}_{k+1} &= \widetilde{y}_k - \left[c\phi x_k + c\gamma u_k - c\phi x_{k-1} - c\gamma u_{k-1}\right] \\
&= \widetilde{y}_k - c\left[\phi\left(x_k - x_{k-1}\right) + \gamma\left(u_k - u_{k-1}\right)\right] \\
&= \widetilde{y}_k - \left[\phi\left(c x_k - c x_{k-1}\right) + c\gamma\left(u_k - u_{k-1}\right)\right] \\
&= \widetilde{y}_k - \left[\phi\left(y_k - y_{k-1}\right) + c\gamma\beta\widetilde{y}_k\right] && (5.10)
\end{aligned}
$$

donde:

$$
\begin{aligned}
\widetilde{y}_{k+1-1} &= \widetilde{y}_{k-1} - \left(y_{k+1-1} - y_{k-1}\right) && (5.11)\\
\widetilde{y}_k &= \widetilde{y}_{k-1} - \left(y_k - y_{k-1}\right) && (5.12)\\
\widetilde{y}_k - \widetilde{y}_{k-1} &= -\left(y_k - y_{k-1}\right) && (5.13)\\
\widetilde{y}_{k-1} - \widetilde{y}_k &= y_k - y_{k-1} && (5.14)
\end{aligned}
$$

Sustituyendo en la ecuación (5.10) se tiene:

$$
\begin{aligned}
\widetilde{y}_{k+1} &= \widetilde{y}_k - \left[\phi\left(\widetilde{y}_{k-1} - \widetilde{y}_k\right) + c\gamma\beta\widetilde{y}_k\right] \\
&= \widetilde{y}_k - \phi\widetilde{y}_{k-1} + \phi\widetilde{y}_k - c\gamma\beta\widetilde{y}_k && (5.15)\\
\widetilde{y}_{k+1} &= \left[1 + \phi - c\gamma\beta\right]\widetilde{y}_k - \phi\widetilde{y}_{k-1} && (5.16)\\
\Rightarrow |\widetilde{y}_{k+1}| &\leq |1 + \phi - c\gamma\beta||\widetilde{y}_k| + |\phi||\widetilde{y}_{k-1}| && (5.17)
\end{aligned}
$$

Para ϕ se debe satisfacer que:

$$
e^{-ah} > 0 \ \therefore\ |\phi| > 0 \qquad (5.18)
$$

Aplicando la función pulso en la ecuación (5.19) se obtiene:

$$
q\widetilde{y}_k = \left[1 + \phi - c\gamma\beta - \phi q^{-1}\right]\widetilde{y}_k \qquad (5.19)
$$

De tal forma que se cumple la Hipotesis 1:

$$\begin{aligned} 1+\phi-c\gamma\beta-\phi q^{-1} &> 0 \\ 1+\phi-\phi q^{-1} &> c\gamma\beta \\ &\Rightarrow \quad 0<\beta<\frac{1+\phi-\phi q^{-1}}{c\gamma} \end{aligned} \tag{5.20}$$

β debe de ser un escalar positivo.

Salida del sistema

Sea la salida k-ésima del sistema escalar:

$$y_k = cx_k \tag{5.21}$$
$$\widetilde{y}_{k+1} = cx_{k+1} \tag{5.22}$$
$$x_{k+1} = \phi x_k + \gamma u_k \tag{5.23}$$

Hipotesis 2: La función pulso satisface.

$$x_k(q)\left[1-\phi q^{-1}\right] = \gamma q^{-1}u_k(q) \tag{5.24}$$
$$G(q)=\frac{cx_k(q)}{u_k(q)} = \frac{c\gamma q^{-1}}{1-\phi q^{-1}} \tag{5.25}$$
$$= \frac{c\gamma}{q-\phi} \tag{5.26}$$

donde γ, ϕ, c, $B \in \mathbb{R}_+$ y $y_d \in \mathbb{R}$.

Para que el sistema sea estable asintóticamente se debe satisfacer que:

$$q < \phi \tag{5.27}$$
$$q < e^{-ah} \tag{5.28}$$

De esta forma se asegura que los polos estén dentro del circulo unitario y sea un sistema lineal estable.

5.2. Ejemplo: Sistema dinámico lineal de primer orden

Sea el sistema dinámico escalar:

$$\dot{x}(t) = -ax(t)+bu(t) \tag{5.29}$$
$$y(t) = cx(t) \tag{5.30}$$

Y los parámetros de simulación para la solución del sistema:

$$a = 5 \tag{5.31}$$
$$b = 5 \tag{5.32}$$
$$c = 1 \tag{5.33}$$
$$\beta = 0.002 \tag{5.34}$$
$$y_d = 1 \tag{5.35}$$

donde:

- $x(t) \in \mathbb{R}$ representa la variable de estado del sistema.
- $u(t) \in \mathbb{R}$ es la entrada del sistema.
- $a, b \in \mathbb{R}_+$ son constantes definidas positivas.
- β es la ganancia proporcional del control iterativo

En la figura 5.2 se observa la posición de salida para $k = 2$, $k = 10$, $k = 20$, $k = 50$, $k = 100$ y $k = 10,000$. A partir de la iteración 10 la salida k-ésima comienza a aproximarse más a la posición deseada, en este caso a 1, dada una ganancia proporcional β ajustada en cada uno de los casos, la cual se muestra en la tabla 5.1. Se observa que despues de la iteración 100 la salida del sistema es bastante similar, tal es el caso que al tener $10,000$ iteraciones la diferencia entre ambas salidas es miníma en la respuesta transitoria y en estado estacionario reduce aún más la diferencia entre ambas.

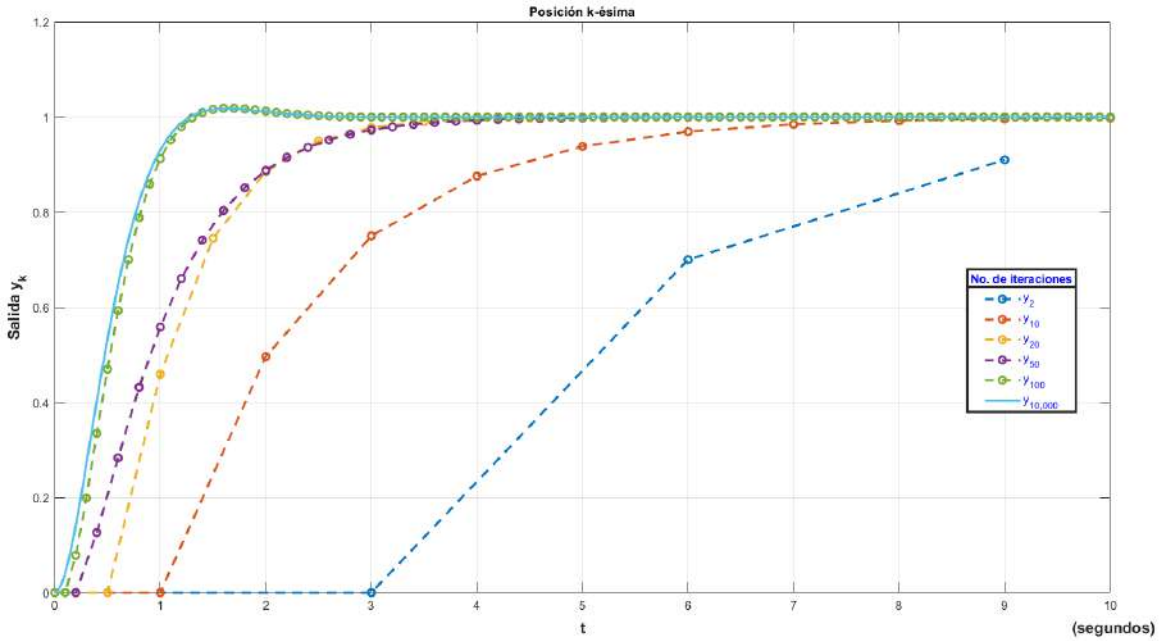


Figura 5.2: Posición de salida para diferentes iteraciones.

La figura (5.2) muestra la respuesta transitoria del sistema para una posición deseada $y_d = 1$, sin embargo, a pesar de que muestra una similitud a la respuesta transitoria para el escalón unitario en este ejemplo no se implementa ese tipo de entrada de control, el control de aprendizaje iterativo de la ecuación

(5.7) es el que da la energía para que el sistema llegue a la posición deseada, en este caso particular es notable que la recursividad del sistema esta relacionada con el tiempo de simulación, de tal forma, que no es posible obtener una curva cuando se tienen números de iteraciones menores a 20, ya que este número de iteraciones representa los puntos proyectados para generar la respuesta transitoria del sistema. En el caso contrario, se obtiene una curva bien definida para número de iteraciones mayores a 50, sin embargo, a pesar de que se realizaron diversas pruebas para determinar la ganancia β con los mejores resultados, aún se presenta una leve oscilación en la respuesta transitoria, definiendola como una señal subamortiguada.

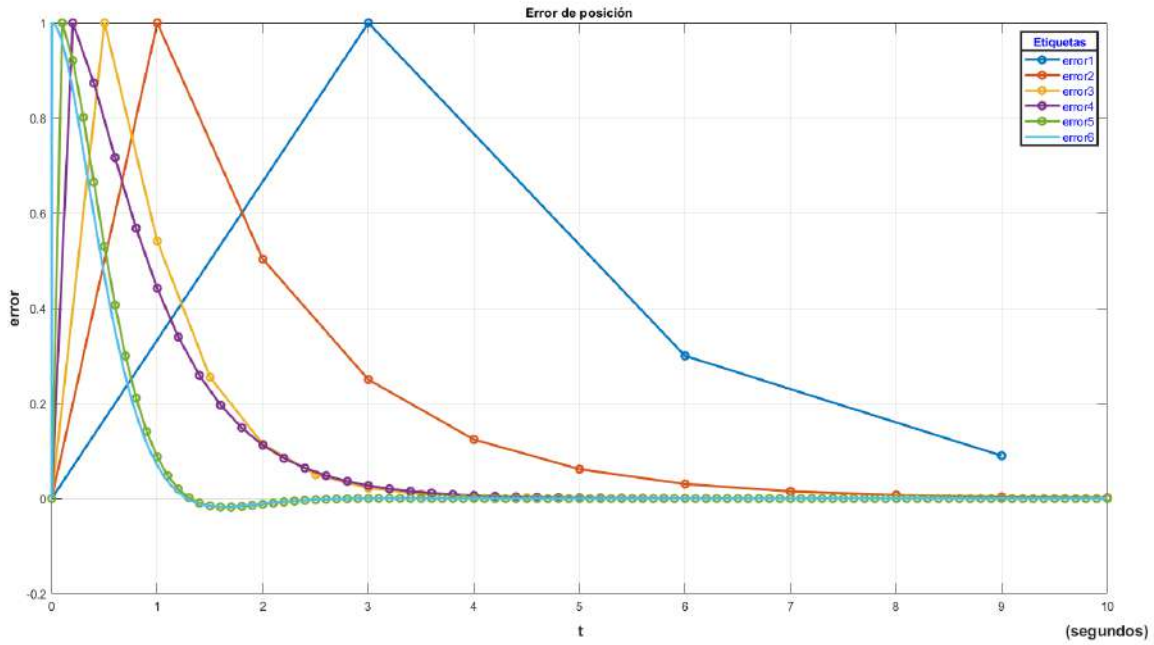


Figura 5.3: Error de posición para cada una de las salidas k-ésimas.

La figura 5.3 muestra el error de posición para cada uno de los casos ya mencionados, además refuerza los resultados obtenidos en la respuesta transitoria, así como, remarcan el hecho de que a pesar, de no tener una curva bien definida para un número de iteraciones menores a 20, el error de posición disminuye únicamente en la etapa transitoria para los primeros dos casos, mientras que para los siguientes casos el error de posición disminuye tanto para la etapa transitoria como en el estado estacionario.

En la tabla 5.1 se muestran los valores de las ganancias proporcionales β ajustadas a cada uno de los casos de iteración. Ya que se debe cumplir que $0 < \beta < 1$.

# de iteraciones	β	Tiempo (segundos)
3	0.7	10
10	0.5	10
20	0.5	10
50	0.2	10
100	0.2	10
10,000	0.002	10

Tabla 5.1: Ganancias proporcionales.

5.3. Sistema dinámico vectorial

Considere el sistema dinámico lineal de segundo orden establecido en espacio de estados de la ecuación (5.36):

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u} \tag{5.36}$$
$$\mathbf{y} = \mathbf{c}^T\mathbf{x} \tag{5.37}$$

donde $\mathbf{x} \in \mathbb{R}^n$ es la variable de estado y representa la solución de la ecuación diferencial, $\mathbf{c} \in \mathbb{R}^n$ es la velocidad, los parámetros del sistema son: $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^n$, $\mathbf{c} \in \mathbb{R}^n$, $u(t)$ es la señal de entrada y $\mathbf{y}(t)$ es la salida del sistema.

Sea la solución del sistema vectorial de forma recursiva:

$$\mathbf{x}_k = \Phi\mathbf{x}_{k-1} + \Gamma u_{k-1} \tag{5.38}$$
$$\mathbf{y}_k = \mathbf{c}^T\mathbf{x}_k \tag{5.39}$$
$$\widetilde{\mathbf{y}}_k = \mathbf{y}_d - \mathbf{y}_k \tag{5.40}$$
$$u_k = u_{k-1} + \Lambda\widetilde{\mathbf{y}}_k \tag{5.41}$$
$$\widetilde{\mathbf{y}}_{k+1} = \widetilde{\mathbf{y}}_k - \left(\mathbf{y}_{k+1} - \mathbf{y}_k\right) \tag{5.42}$$

donde:

$$x_k - x_{k-1} = \left(\Phi - \mathtt{I}\right) x_{k-1} + \Gamma u_{k-1} \tag{5.43}$$
$$\begin{aligned}\mathbf{y}_k &= \mathbf{c}^T\mathbf{x}_k \\ &= \mathbf{c}^T\left(\Phi\mathbf{x}_{k-1} + \Gamma u_{k-1}\right)\end{aligned} \tag{5.44}$$
$$\begin{aligned}\mathbf{y}_{k+1} &= \mathbf{c}^T\mathbf{x}_{k+1} \\ &= \mathbf{c}^T\left(\Phi\mathbf{x}_k + \Gamma u_k\right)\end{aligned} \tag{5.45}$$

Sustituyendo en la ecuación (5.42) se tiene:

$$\begin{aligned}\widetilde{\mathbf{y}}_{k+1} &= \widetilde{\mathbf{y}}_k - \left[\mathbf{c}^T\left(\Phi\mathbf{x}_k + \Gamma u_k\right) - \mathbf{c}^T\left(\Phi\mathbf{x}_{k-1} + \Gamma u_{k-1}\right)\right] \\ &= \widetilde{\mathbf{y}}_k - \left[\mathbf{c}^T\Phi\mathbf{x}_k + \mathbf{c}^T\Gamma u_k - \mathbf{c}^T\Phi\mathbf{x}_{k-1} - \mathbf{c}^T\Gamma u_{k-1}\right] \\ &= \widetilde{\mathbf{y}}_k - c^T\left[\Phi\mathbf{x}_k - \Phi\mathbf{x}_{k-1} + \Gamma u_k - \Gamma u_{k-1}\right] \\ &= \widetilde{\mathbf{y}}_k - c^T\left[\Phi\left(\mathbf{x}_k - \mathbf{x}_{k-1}\right) + \Gamma\left(u_k - u_{k-1}\right)\right]\end{aligned} \tag{5.46}$$

Finalmente se obtiene la ecuación (5.47):

$$\widetilde{\mathbf{y}}_{k+1} = \widetilde{\mathbf{y}}_k - \mathbf{c}^T\left[\Phi\left(\mathbf{x}_k - \mathbf{x}_{k-1}\right) + \Phi\Lambda\widetilde{\mathbf{y}}_k\right] \tag{5.47}$$

Aplicando la función pulso se tiene:

$$q\widetilde{\mathbf{y}}_k = \widetilde{\mathbf{y}}_k - \mathbf{c}^T\left[\Phi\left(\mathbf{x}_k - q^{-1}\mathbf{x}_k\right) + \Phi\Lambda\widetilde{\mathbf{y}}_k\right] \tag{5.48}$$
$$\left[q - \mathtt{I} - \Phi\Lambda\right]\widetilde{\mathbf{y}}_k = \mathbf{c}^T\Phi\left(q^{-1} - \mathtt{I}\right)x_k \tag{5.49}$$
$$\frac{\widetilde{\mathbf{y}}_k}{x_k} = \left[q - \mathtt{I} - \Phi\Lambda\right]^{-1}\mathbf{c}^T\Phi\left(q^{-1} - \mathtt{I}\right) \tag{5.50}$$

Hipotesis 3:

$$\begin{aligned}
\left[q - \mathtt{I} - \Phi\Lambda\right]^{-1} \mathbf{c}^T\Phi\left(q^{-1} - \mathtt{I}\right) &> \mathbf{0} \\
\left[q^{-1} - \mathtt{I} - \Phi^{-1}\Lambda^{-1}\right] \mathbf{c}^T\Phi\left(q^{-1} - \mathtt{I}\right) &> \mathbf{0} \\
q^{-2}c^T\Phi - q^{-1}c^T\Phi - Ic^T\Phi q^{-1} + Ic^T\Phi - \Lambda^{-1}\Phi^{-1}c^T\Phi q^{-1} + \Lambda^{-1}\Phi c^T\Phi &> \mathbf{0} \\
\Lambda^{-1}\left(\Phi c^T\Phi - \Phi^{-1}c^T\Phi q^{-1}\right) + \left(q^{-2} - 2q^{-1} + \mathtt{I}\right)c^T\Phi &> \mathbf{0} \\
\left(\Phi c^T\Phi - \Phi^{-1}c^T\Phi q^{-1}\right)\left[\left(2q^{-1} - q^{-2} - \mathtt{I}\right)c^T\Phi\right]^{-1} &> \Lambda > \mathbf{0}
\end{aligned} \tag{5.51}$$

De esta forma se asegura que los eigenvalores se encuentran dentro del círculo unitario en el plano z, y en el plano s estarían ubicados del lado izquierdo, existiendo estabilidad asintótica del punto de equilibrio.

Ejemplo: Sistema dinámico lineal de segundo orden

Considere el sistema dinámico lineal de segundo orden establecido en espacio de estados:

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u} = \begin{bmatrix} 0 & 1 \\ -w_n^2 & -2\rho w_n \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ w_n^2 \end{bmatrix} u(t) \tag{5.52}$$

Realizar la simulación del sistema dinámico (5.52) con condiciones iniciales $\mathbf{x}(0) = [0,0]^T$, $w_n = 1$ rad/seg, $\rho = 0.8$.

Dadas las ecuaciones (5.38) y (5.39) que representan la solución del sistema, se tienen los siguientes parámetros:

$$\begin{aligned}
\Phi &= e^{Ah} \\
&= \alpha_0(h)I + \alpha_1(h)A
\end{aligned} \tag{5.53}$$

$$\begin{aligned}
\Gamma &= \int_0^h e^{A\sigma}Bd\sigma \\
&= A^T\left[e^{Ah}B - e^0B\right] \\
&= A^T\left[e^{Ah}B - \mathtt{I}B\right] \\
&= A^T\left[\Phi B - \mathtt{I}B\right]
\end{aligned} \tag{5.54}$$

donde:

$$\alpha_0 = e^{ah}\cos(bh) - \alpha_1 a \tag{5.55}$$

$$\alpha_1 = \frac{e^{ah}\,\mathrm{sen}(bh)}{b} \tag{5.56}$$

a y b representan la parte real e imaginaria de λ_i los i-ésimos valores propios de la matriz $A \in \mathbb{R}^{n\times n}$.

$$a = -0.8 \tag{5.57}$$

$$b = -0.6 \tag{5.58}$$

Otros de los parámetros de salida del sistema es $\mathbf{c}^T = \begin{bmatrix} 1 & 0 \end{bmatrix}$. Y el vector de posiciones deseadas es $\mathbf{y}_d = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$.

En la figura 5.4 se muestra la gráfica para la salida k-ésima con $k = 2, k = 10, k = 20, k = 50, k = 100$ y $k = 10,000$. En este caso se implemento la ganancia proporcional como una función que depende de las mismas iteraciones.

$$u_k = u_{k-1} + \Lambda \tilde{\mathbf{y}}_k \tag{5.59}$$

$$\Lambda = \frac{e^{-1.6667k} + sinh(e^{-1.6667k})}{1 + e^{-1.6667k}} \tag{5.60}$$

De tal forma, que se puede diseñar una ganancia proporcional con funciones hiperbólicas respecto al número de iteraciones, que permita llegar a la posición deseada como se muestra en las siguientes figuras.

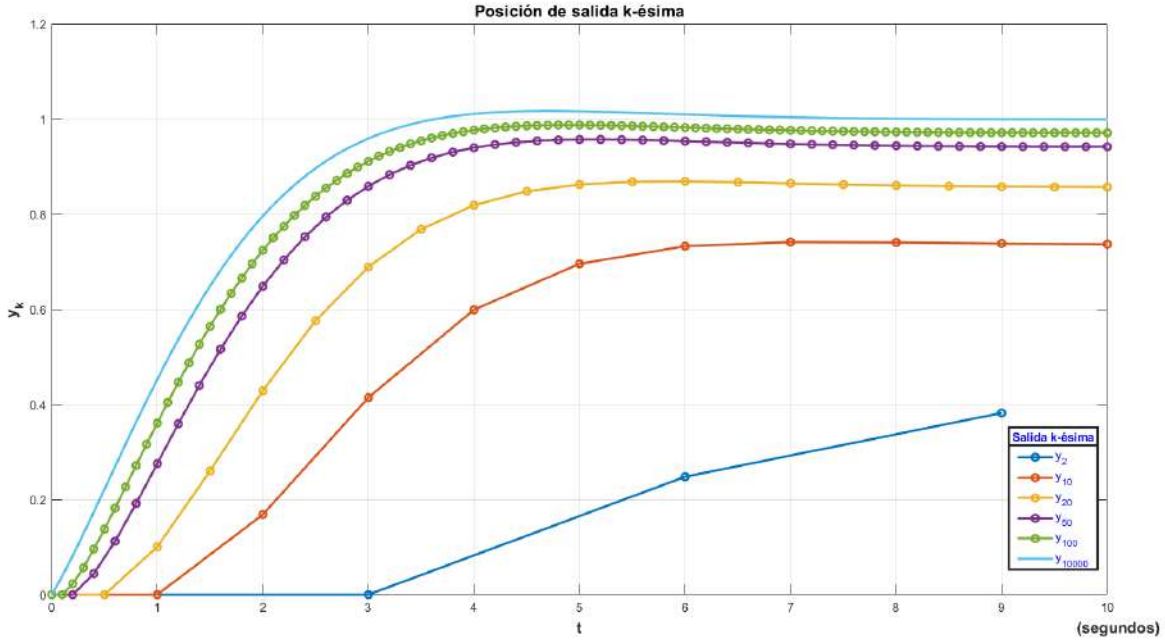


Figura 5.4: Posición de salida k-ésima.

En la figura 5.4 se muestra la respuesta transitoria para el sistema de segundo orden de la ecuación (5.52), de igual forma que en el caso anterior se tiene que para $k = 2$ no se forma la curva que representa la respuesta transitoria del lazo cerrado de control y en este caso, ni siquiera se puede argumentar que exista una etapa transitoria, sin embargo, para los casos cuando se tiene $k \geq 10$ ya se define una etapa transitoria y el estado estacionario. Es importante mencionar que la función que se define como la ganancia proporcional, produce un efecto de sobreamortiguamiento en la señal de salida del sistema dada la función exponencial en ésta, sin embargo, para sistemas de segundo orden es necesario proponer este tipo de funciones como ganancia proporcional.

En la figura 5.5 se muestra el error para la k-ésima salida y se evidencia que para el caso de $k = 2$ ni siquiera se obtiene un error por debajo de 0.6, mientras que, conforme se aumentan el número de iteraciones se va obteniendo una señal de salida mejor definida y de igual manera podemos observar que la respuesta tiene un sobreamortiguamiento ya que a la señal del error le toma más de dos segundos comenzar a aproximarse a cero.

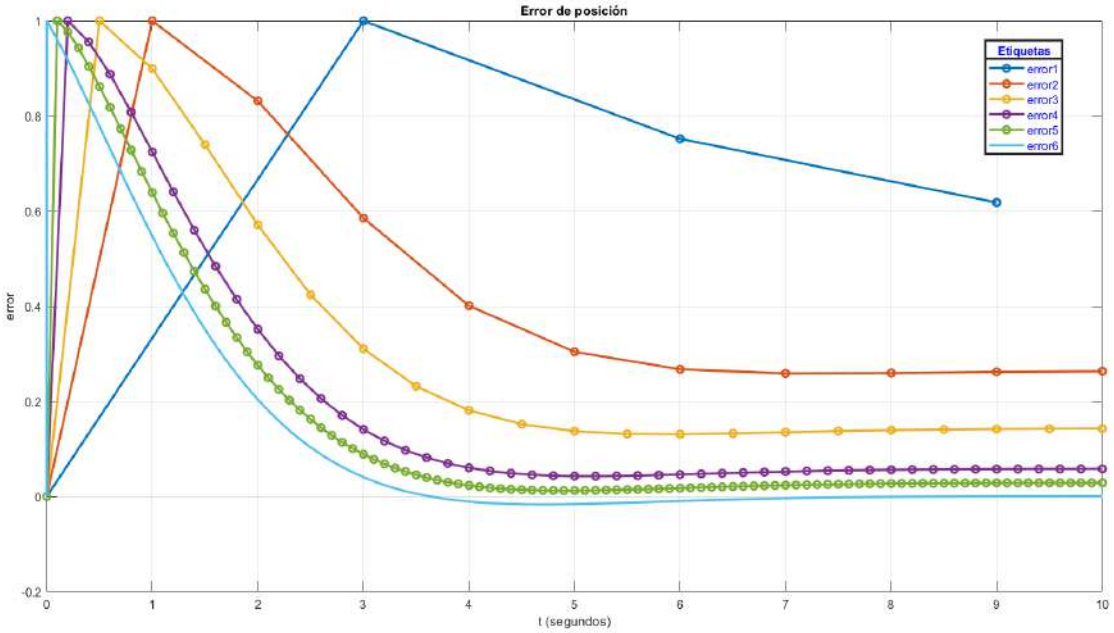


Figura 5.5: Posición de salida k-ésima y error de posición para 10 iteraciones.

Ejemplo: Robot manipulador de un grado de libertad

A continuación se realiza la implementación del control de aprendizaje iterativo para el robot manipulador de un grado de libertad (péndulo simple), el cual representa un sistema de segundo orden no lineal, como se muestra en la ecuación (5.61):

$$\tau = (ml_{c1}^2 + I)\ddot{q}_1 + mgl_{c1}\cos(q_1) + b\dot{q}_1 \tag{5.61}$$

Y expresado en diagrama de estados se obtiene la ecuación (5.62):

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \dot{\mathbf{q}} \\ \frac{\tau - b\dot{\mathbf{q}} - \mathtt{mgl}_c * \sin(q)}{\mathtt{ml}_c^2 + \mathtt{I}} \end{bmatrix} \tag{5.62}$$

Por lo que el algoritmo de aprendizaje iterativo esta dado por τ, de tal forma que:

$$\boldsymbol{\tau}_k = \boldsymbol{\tau}_{k-1} + \beta\widetilde{\mathbf{q}} \tag{5.63}$$

$$\widetilde{\mathbf{q}} = \mathbf{q}_d - \mathbf{q} \tag{5.64}$$

donde:

$$\beta = \frac{0.0001\tanh(\widetilde{q})^2}{(1+\tanh(\widetilde{q})^2)} \tag{5.65}$$

$$q_d = 95^\circ \tag{5.66}$$

Y la solución del sistema no lineal se realiza empleando la herramienta ODE45 en código fuente de MATLAB.

En la figura 5.6 se muestran las gráficas obtenidas con $k = 1,000$ iteraciones y $k = 10,000$ iteraciones del péndulo simple en el lazo cerrado de control iterativo recursivo.

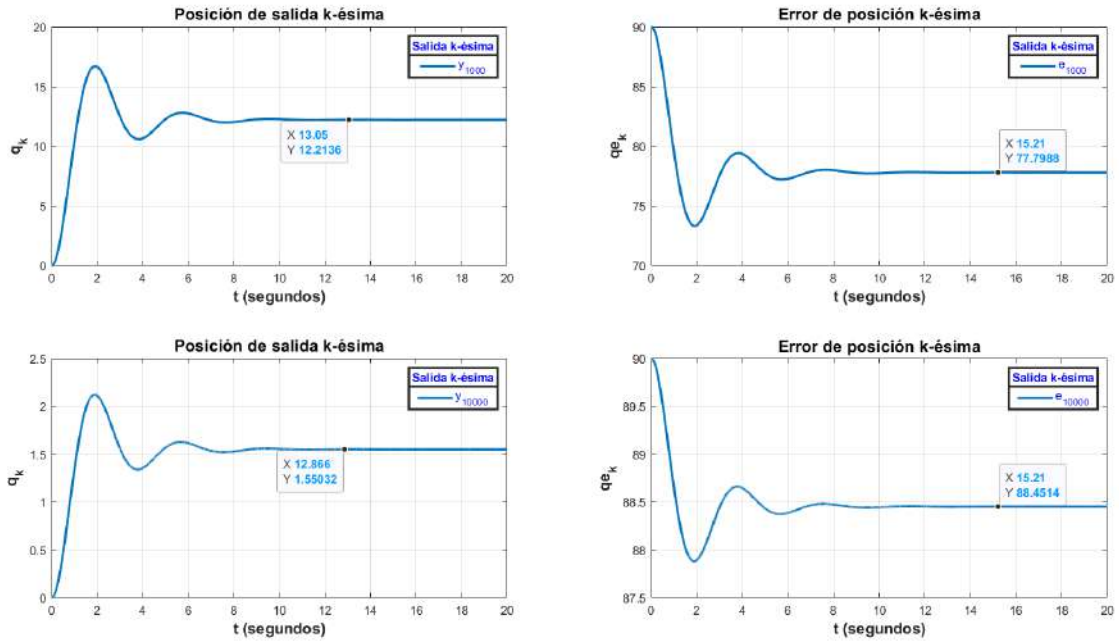


Figura 5.6: Salida k-ésima del sistema de segundo orden no lineal y error de posición para el caso en el que $k = 1,000$ iteraciones y cuando $k = 10,000$ iteraciones.

Se observa en la figura 5.6 que no se llega a la posición deseada con la función de la ecuación (5.65), lo que implica que el control de la ecuacion (5.63) no proporciona la energía suficiente para llegar a la posición deseada, además de que la respuesta transitoria da como resultado una señal subamortiguada. Se puede argumentar que al tener un sistema no lineal dado el par gravitacional que se obtiene del modelo dinámico del péndulo simple se tiene mayores dificultades para encontrar una ganancia que proporcione la suficiente energía de inyección.

Sin embargo este problema, puede ser solucionado, implementando el control de aprendizaje iterativo como proceso repetitivo en conjunto con el control de lazo cerrado, como más adelante se trabaja.

Capítulo 6

Proceso repetitivo

En este caso se define la tarea repetitiva, representada por la función pulso de la figura 6.1. Donde, no se implementa la solución del sistema de forma recursiva como en el caso anterior, sino que se trabaja en tiempo continuo t, y la recursividad recae únicamente en el control de aprendizaje iterativo. La figura 6.1 representa la tarea repetitiva, la cual es llegar a la posición deseada $y_d = 1$ cada 50 segundos, de tal forma que el control de aprendizaje iterativo almacena los datos de la primer tarea realizada, y con base al error almacenado y al valor de ganancia proporcional definido se aproxima a la posición de salida deseada. Como lo expresan las ecuaciones (6.3) y (6.4):

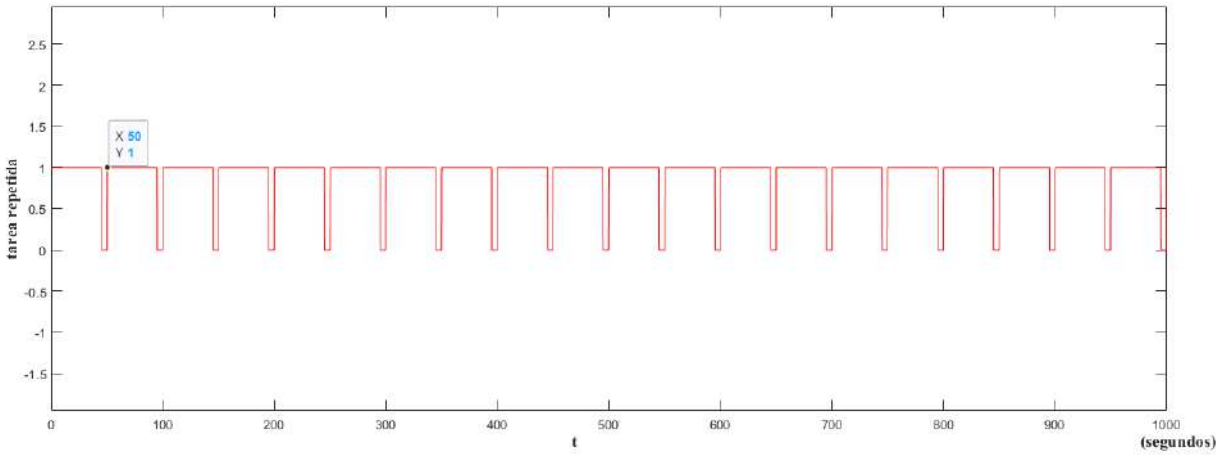


Figura 6.1: Función pulso, con un periodo de 50 segundos y un ancho de pulso del 90 %.

$$\dot{\mathbf{x}}_k(t) = A\mathbf{x}_k(t) + Bu_k(t) \tag{6.1}$$
$$\mathbf{y}_k(t) = \mathbf{c}^T\mathbf{x}_k(t) \tag{6.2}$$

El algoritmo de control esta dado por:

$$u_{k+1} = u_k + \beta e_k \tag{6.3}$$

donde $e_k(t) = y_d(t) - y_k(t)$, y β es una matriz de ganancia diagonal, la cual asegura que:

$$\lim_{k\to\infty} y_k(t) \to y_d(t) \tag{6.4}$$

para toda $t \in [0, t_f]$, si

$$\| \mathrm{I} - \mathbf{c}^T B\beta \|_i < 1 \tag{6.5}$$

Para cualquier norma vectorial $i = 1, 2, ..., \infty$. Que demuestra la estabilidad asintótica del punto de equilibrio.

6.1. Simulaciones con sistemas lineales de segundo orden

Sea el sistema lineal de segundo orden:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u} = \begin{bmatrix} -0.6667 & -1.8333 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) \\ \mathbf{y} &= \begin{bmatrix} 0 & 1.6667 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned} \tag{6.6}$$

Y su ecuación de transferencia definida por:

$$G(s) = \frac{1.667}{s^2 + 0.6667s + 1.833} \tag{6.7}$$

El cual es un sistema lineal estable, como se muestra en la figura 6.2, donde los polos definen el comportamiento del sistema y estos se ubican en el lado izquierdo del plano S: $p_1 = -0.333 + 1.31i$ y $p_2 = -0.333 - 1.31i$, por lo tanto, se demuestra que es un sistema lineal estable. El comando pzmap(G(s)) es el que permite obtener la representación de los polos en el plano S.

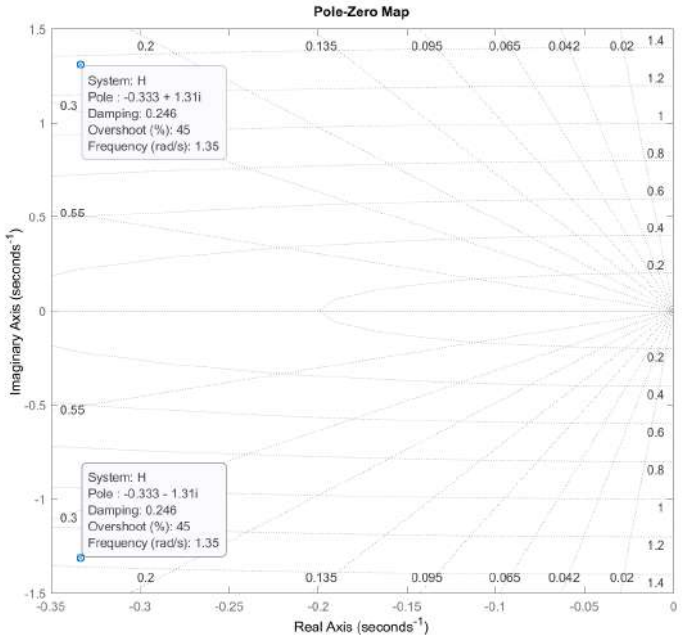


Figura 6.2: Polos del sistema dinámico lineal.

Dichos polos tienen parte real e imaginaria, por lo que la respuesta transitoria que se espera de este sistema es que posea sobreimpulsos, esto se comprueba a continuación.

La respuesta transitoria ante un escalón unitario se muestra en la figura 6.3, con la cual, se puede definir a este sistema como subamortiguado y en estado estacionario se tiene una amplitud igual a 0.91.

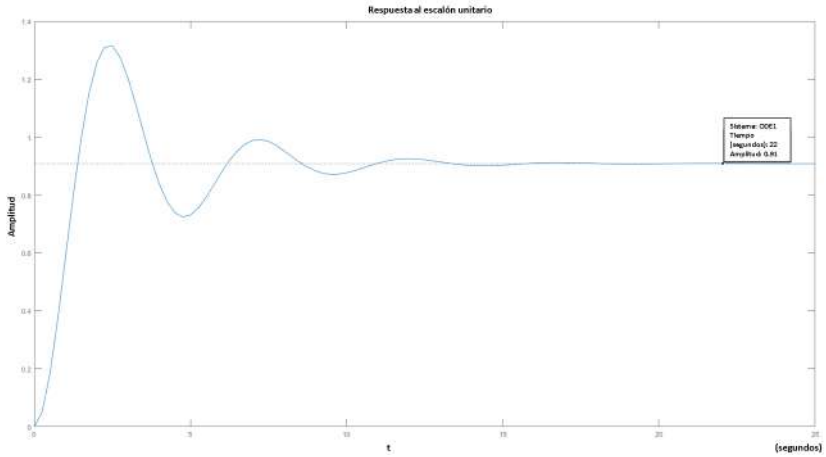


Figura 6.3: Respuesta transitoria de la ecuación 1.1 al escalón unitario.

Se implementa en Simulink la simulación del sistema de 2do orden, como se muestra en la figura 6.4, con el lazo cerrado de control PD exclusivamente, y así, conocer el comportamiento del sistema ante con tipo de control.

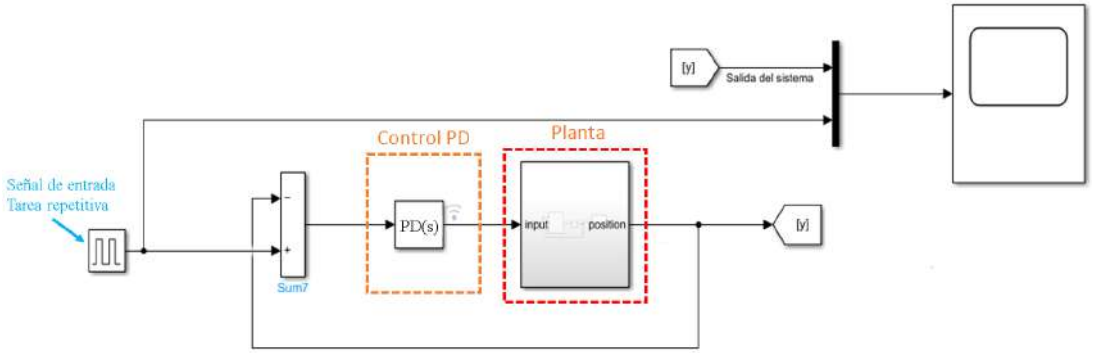


Figura 6.4: Simulación en simulink, diagrama de bloques del sistema en lazo cerrado de control PD.

En la figura 6.5 se observa la respuesta del sistema en lazo cerrado, donde la función pulso representa el proceso iterativo/repetitivo de la posición deseada,que en este caso $y_d = 1$, se puede observar que el sistema mantiene un valor constante en cada tarea repetida debido a que se esta aplicando el mismo control PD cada vez que esta tarea se repite, y en este tipo de control no se tiene la característica de aprender de tareas previas, como más adelante se muestra.

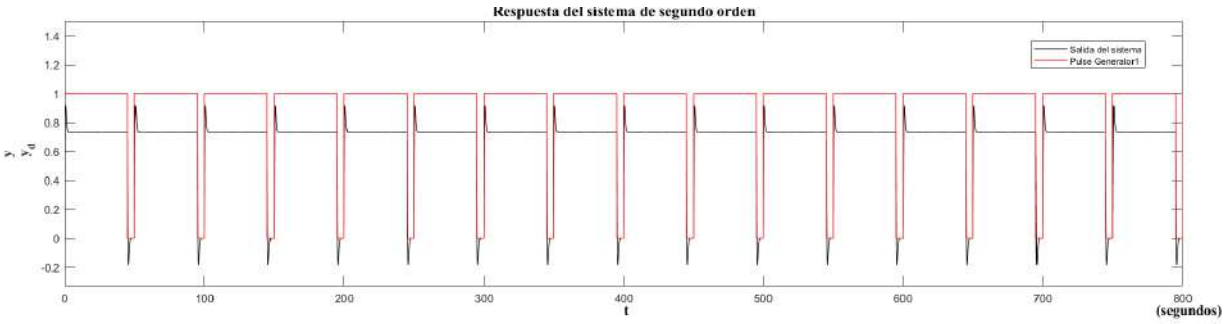


Figura 6.5: Respuesta del sistema de 2do orden en el lazo cerrado de control PD, donde la tarea repetida se establece a partir de la función pulso.

Con ganancia proporcional $k_p = 3$ y ganancia derivativa $k_v = 0.7k_p$ que representa el 70 % de la ganancia proporcional para evitar oscilaciones de amortiguamiento. Cabe resaltar que la sobreamortiguación de la figura 6.3 es reducida a un solo pico de amortiguación, debido a la ganancia derivativa establecida, sin embargo, no esta completamente atenuada, en este caso se buscaría aumentar la ganancia proporcional para reducir el error de posición y llegar a la posición deseada $y_d = 1$, sin embargo, tener una ganancia proporcional demasiado alta en la práctica no siempre es posible. De esta forma, se observa que la sintonización de las ganancias proporcional y derivativa es continua, por lo que el error que existe entre la señal deseada y la señal de salida no es corregida cuando $t \to \infty$.

El error respecto al número de iteraciones y el tiempo se muestra en la figura 6.6, y se representa de esta forma para facilitar el concepto de tarea repetida/k-ésima iteración respecto al tiempo en el que se lleva a cabo dicha tarea, este tipo de gráficas se aplican para todos los ejemplos siguientes. Donde se observa que el error no es reducido en las 10 iteraciones realizadas, debido al tipo de control que se implementa, el cual, no esta diseñado para reducir el error en cada tarea repetida, únicamente modifica la respuesta transitoria del sistema a partir de las ganancias proporcional k_p y derivativa k_v.

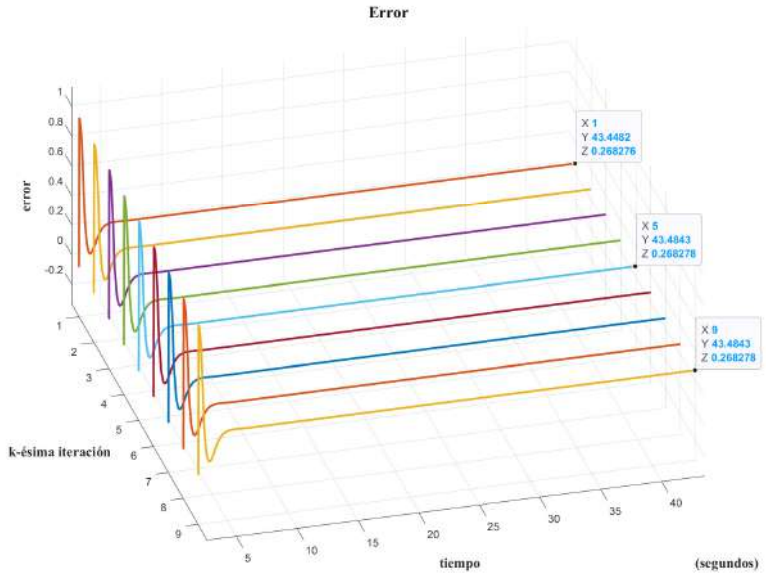


Figura 6.6: Gráfica del error respecto al número de iteraciones y el tiempo. Para el sistema con control PD en el lazo cerrado de control.

Las normas euclidianas $\mathcal{L}_i$ obtenidas para este caso son:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| e_k \|_1 = 0.5 \tag{6.8}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| e_k \|_2 = 0.2675 \tag{6.9}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| e_k \|_\infty = 0.5 \tag{6.10}$$

La norma $\mathcal{L}_2$ representa el índice de desempeño del algoritmo de control y es uno de los más citados en el área de control, cuanto menor sea el valor obtenido mucho mejor es el comportamiento del índice de desempeño del controlador, esto significa que se tiene una mejor exactitud en la respuesta del robot manipulador. Mientras que, $\mathcal{L}_\infty$ obtiene la cota superior más pequeña y $\mathcal{L}_1$ es la suma de los valores absolutos de los elementos del vector.

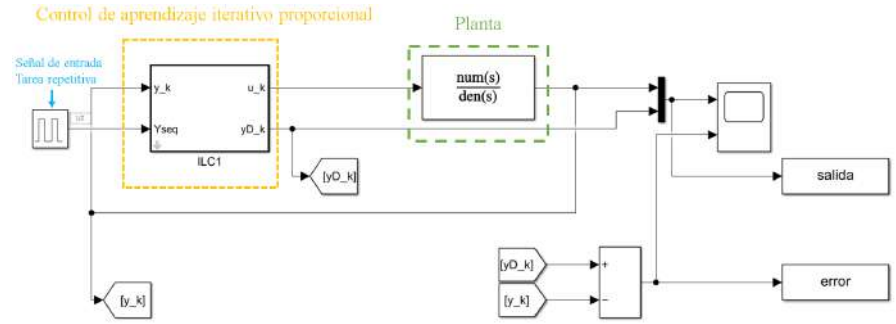


Figura 6.7: Simulación en simulink del sistema de segundo orden con el control de aprendizaje iterativo proporcional.

En la figura 6.7 se muestra la implementación del sistema dinámico de segundo orden de la ecuación (6.6) con el control de aprendizaje iterativo únicamente, con lo cual, se puede conocer ahora, la respuesta del sistema ante este tipo de control.

Y las señales de salida y el error de posición que se genera con la función pulso, se muestran en la figura 6.8, la cual representa nuestra tarea repetitiva.

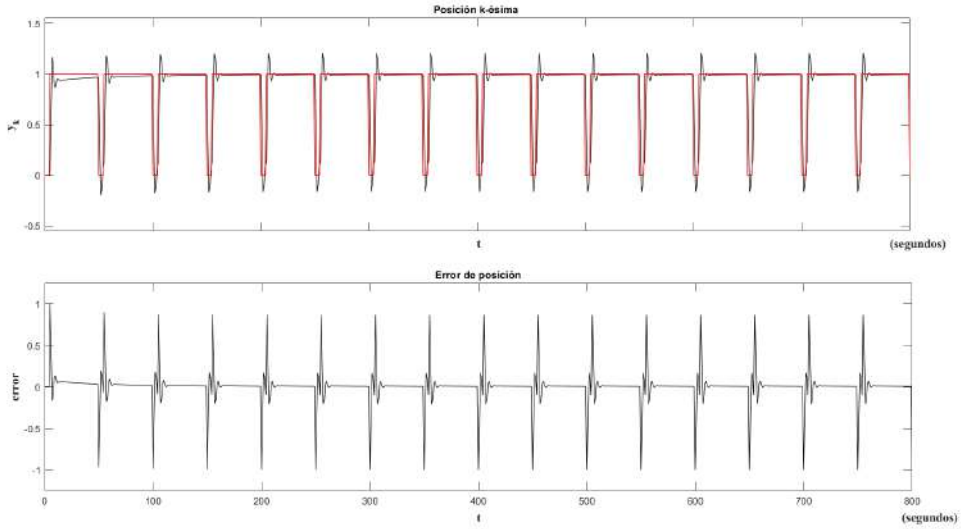


Figura 6.8: Salida del sistema en la k-ésima iteración, al implementar el sistem dinámico de segundo orden con el control de aprendizaje iterativo.

Se observa en la figura 6.8 que el control de aprendizaje iterativo mejora la señal de salida al llegar a la posición deseada conforme la tarea se va repitiendo, en comparación con el control PD, donde se mantiene constante la salida en cada tarea repetida. Sin embargo, el sobreamortiguamiento que existe debida la naturaleza del sistema dinámico de segundo orden se mantiene en cada tarea repetida, ya que, no se tiene una acción de control directa para la respuesta transitoria de la señal como en el caso anterior. En este caso el control de aprendizaje iterativo únicamente toma la señal anterior para reducir el error en estado estacionario como se muestra en las gráficas del error de la figura 6.8, sin embargo, una ganancia β mal atenuada para el ILC afectaría de manera negativa la etapa transitoria del sistema.

En la figura 6.9 se muestran las señales de error obtenidas en 10 iteraciones, en relación con el tiempo de cada tarea repetida. Donde se tiene un error inicial $\widetilde{y} = 0.0909$ con $k = 0$, para $k = 4$ se tiene $\widetilde{y} = 0.0069$

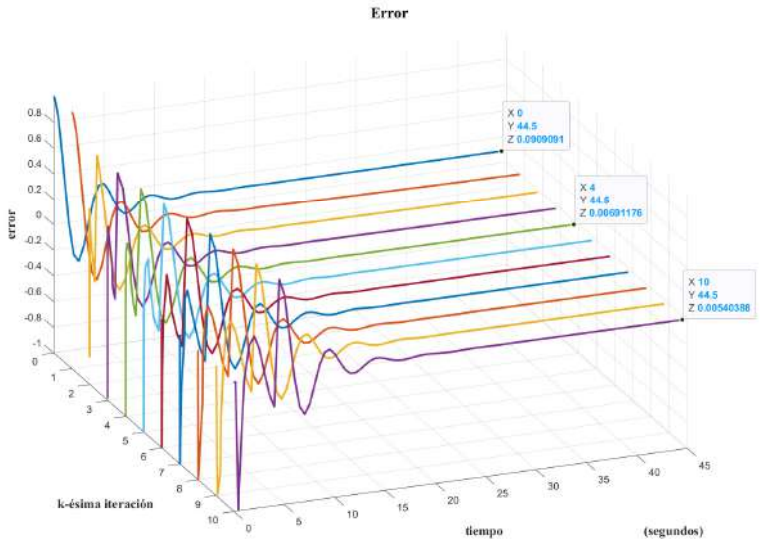


Figura 6.9: Error de posición respecto al tiempo y número de iteraciones.

y para $k = 10$ se tiene un error de posición $\widetilde{y} = 0.0054$.

De acuerdo con las ecuaciones (6.4) y (6.5) las normas respecto al error que se obtiene para el Control de Aprendizaje Iterativo Proporcional con el sistema de la ecuación (6.6) son:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| e_k \|_1 = 0.2795 \tag{6.11}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| e_k \|_2 = 0.1950 \tag{6.12}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| e_k \|_\infty = 0.2795 \tag{6.13}$$

Para este caso se tiene un índice de desempeño $\mathcal{L}_2$ mucho menor que en el caso del control PD, ya que, en comparación si se reduce el error de posición en el lazo cerrado de control iterativo para el estado estacionario, algo que no sucede para el caso del control PD.

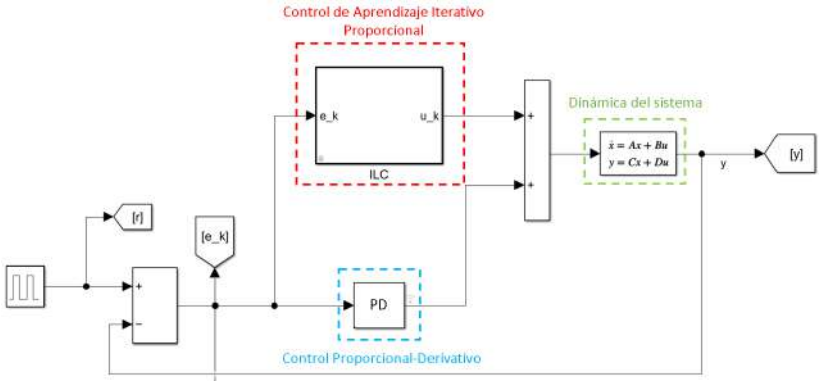


Figura 6.10: Simulación en simulink del control de aprendizaje iterativo proporcional, implementado en el lazo cerrado de control PD con un sistema de 2do orden.

Finalmente en la figura 6.10 se muestra la implementación del control de aprendizaje iterativo, con el control de retroalimentación PD y la planta (sistema de segundo orden).

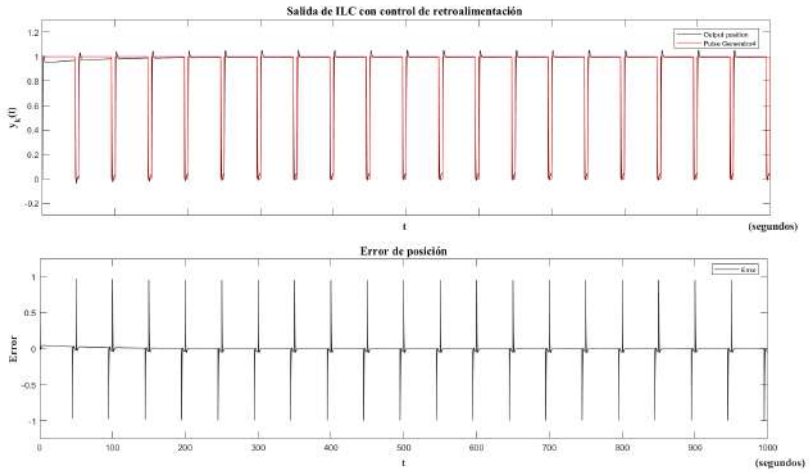


Figura 6.11: Señal de salida del control de aprendiaje iterativo con tarea repetitiva, implementado en el lazo cerrado de control PD, con sistema de segundo orden.

En la figura 6.11 se observa a la señal de salida aproximándose cada vez más a la posición deseada en cada tarea repetida, y se reducen las oscilaciones del sistema dinámico original, tal y como se obtiene en la figura 6.5,para el caso del lazo cerrado de control PD. La implementación para el sistema de segundo orden obtiene dos beneficios en uno solo, el primero es tener una respuesta transitoria regulada por el control PD, mientras que el control de aprendizaje iterativo, reduce el error de posición en cada una de las tareas repetidas, además, de que también reduce el tiempo en que los sobreimpulsos se mantienen en la etapa transitoria. Esta ventaja sobrepasa a los controladores clásicos que por si solos no son capaces de corregir errores de posición que puedan ocurrir inesperadamente. Así como esta respuesta en el lazo cerrado de control esta mejor adaptada a las necesidades industriales, en las cuales idealmente se requiere de sistemas de control fáciles de implementar y económicos en su producción.

El error de posición respecto a la k-ésima iteración se muestra en la figura 6.12.

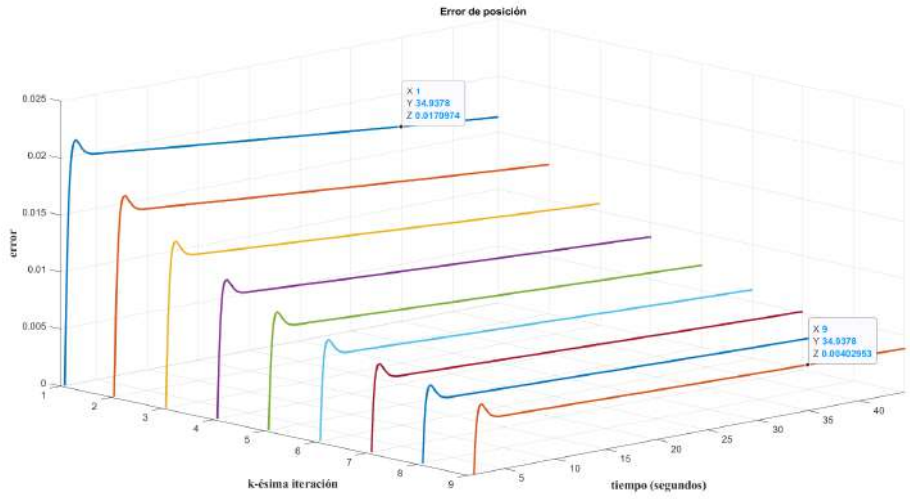


Figura 6.12: Error de posición respecto al tiempo, número de iteraciones y magnitud del error.

Donde se observa que desde la primera iteración se tiene una reducción del error significativa pasados los 30 segundos, la cual esta representada en centésimas $\widetilde{y} = 0.017$, y para la novena iteración pasados los

30 segundos se tiene un error en milésimas $\widetilde{y} = 0.004$. Las normas obtenidas para este caso son:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| e_k \|_1 = 0.1677 \tag{6.14}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| e_k \|_2 = 0.0894 \tag{6.15}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| e_k \|_\infty = 0.1677 \tag{6.16}$$

En la tabla 6.1 se muestra la comparativa de las normas $\mathcal{L}_i$ obtenidas:

$\mathcal{L}_i$	PD	ILC	PD-ILC
$\mathcal{L}_1$	0.5	0.2795	0.1677
$\mathcal{L}_2$	0.2675	0.1950	0.0894
$\mathcal{L}_\infty$	0.5	0.2795	0.1677

Tabla 6.1: Comparativa de $\mathcal{L}_i$ para el sistema de la ecuación (6.6).

La tabla 4.3 representa las normas euclidianas en cada uno de los casos de lazo cerrado de control en los que se implemento el sistema de segundo orden, de tal forma, que el índice de desempeño registrado en los tres casos es mucho mejor cuando se implementa en conjunto el el lazo cerrado de control PD con el control de aprendizaje iterativo, se obtiene $\mathcal{L}_2 = 0.0894$.

Ejemplo 2: Sistema lineal de segundo orden

Para el sistema dinámico lineal de segundo orden de la ecuación (6.17), se implementan las mismas simulaciones que se llevaron a cabo para el ejemplo anterior.

$$\dot{x} = \underbrace{\begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}_{\mathbf{x}(t)}}_{} + \underbrace{\begin{bmatrix} 0 \\ 9 \end{bmatrix}}_{B} u(t) \qquad y(t) = \underbrace{\begin{bmatrix} 3 & 1 \end{bmatrix}}_{c^T} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\mathbf{x}(t)} \tag{6.17}$$

Se comprueba que el sistema dinámico de la ecuación (6.17) es lineal, por lo que, se obtiene el plano S de la figura 6.13 donde se ubican los polos que describen el comportamiento del sistema.

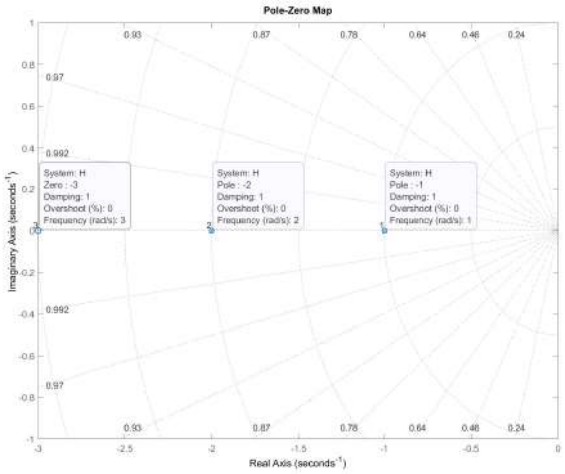


Figura 6.13: Ubicación de los polos del sistema dinámico lineal.

La ubicación de los polos es: $p_1 = -1$, $p_2 = -2$ y un cero ubicado en $z = -3$, lo que indica que el comportamiento del sistema no posee sobreimpulsos debido a que ningún polo tiene parte imaginaria y el cero esta ubicado más lejos del origen que los polos. A continuación se obtiene la respuesta transitoria ante un escalón unitario para visualizar el comportamietno que éste tiene.

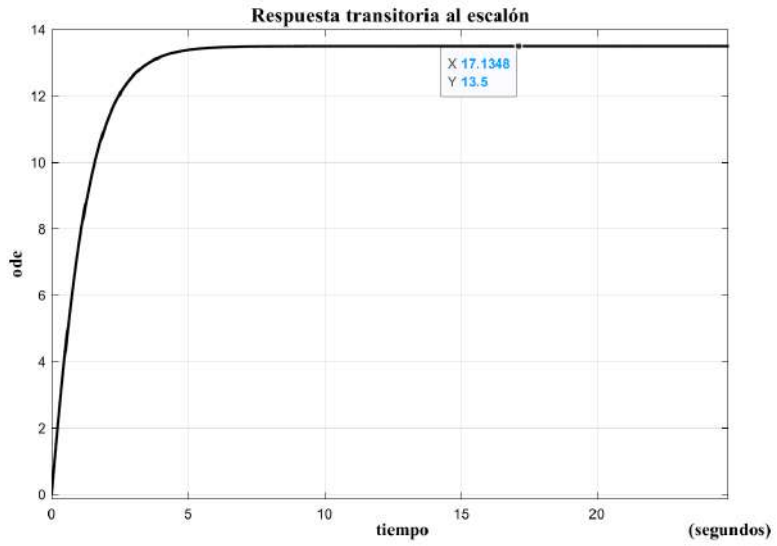


Figura 6.14: Repuesta del sistema al escalón unitario.

La respuesta transitoria ante un escalón unitario se muestra en la figura 6.14, donde se tiene una respuesta sin oscilaciones (sobreimpulsos) mejor denominada como respuesta críticamente amortiguada, pero, en estado estacionario tiene una amplitud igual a 13.5.

Se desarrolla la simulación en Simulink MATLAB, como se muestra en la figura 6.15.

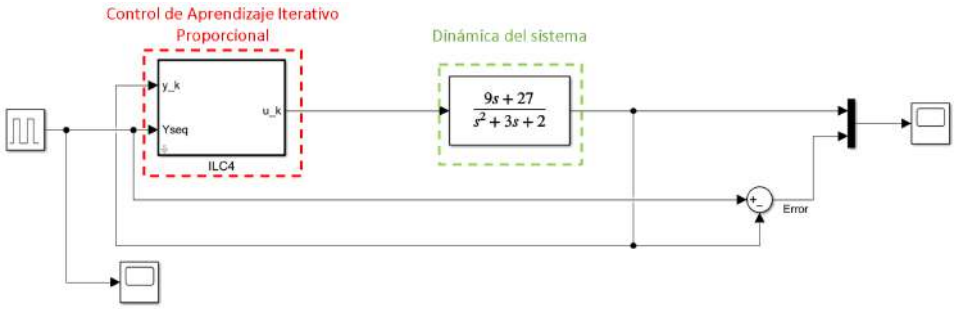


Figura 6.15: Diagrama a bloques del sistema en lazo cerrado con el ILC-P.

En este caso se tiene una $\beta = 0.02$ como ganancia proporcional del Control de Aprendizaje Iterativo Proporcional, la respuesta del sistema en la primer iteración se muestra en la figura 6.16, donde se observa que la respuesta transitoria se conserva, sin embargo, se aproxima al escalón unitario. La señal de color negro representa la respuesta del sistema, mientras que la señal de color rojo representa el error entre la referencia deseada y la referencia actual.

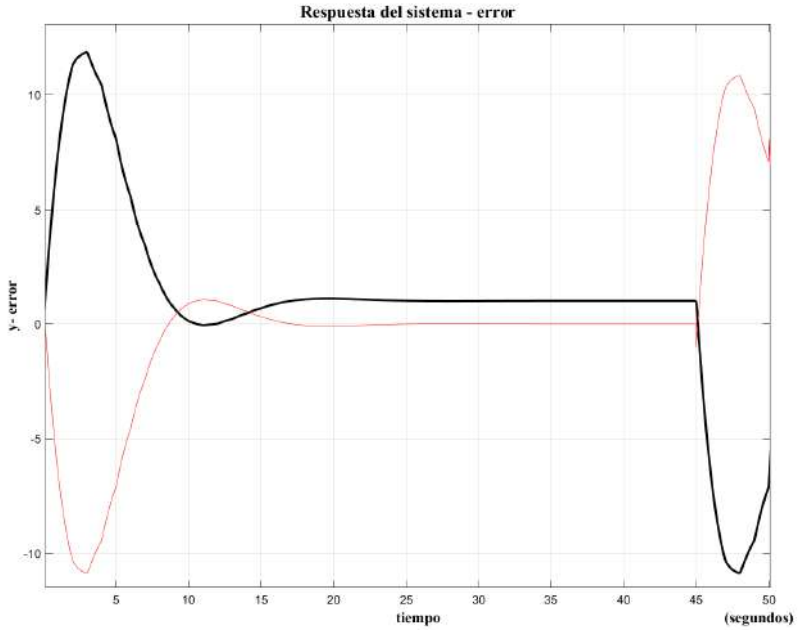


Figura 6.16: Respuesta del sistema en la primera iteración del ILC-P.

La respuesta que se obtiene en la figura 6.16 se debe a que el control de aprendizaje iterativo actúa tanto en la etapa transitoria (reduciendo la sobreamortiguación) como en el estado estacionario (reduciendo el error de posición). Por lo tanto, la oscilación presente en la señal de salida forma parte de la respuesta transitoria por el que pasa el sistema dinámico, que como se ha visto anteriormente el sistema tiene una amplitud igual a 13.5, y se reduce hasta cero en el momento que empieza el estado estacionario del sistema para finalmente llegar a la posición de salida deseada.

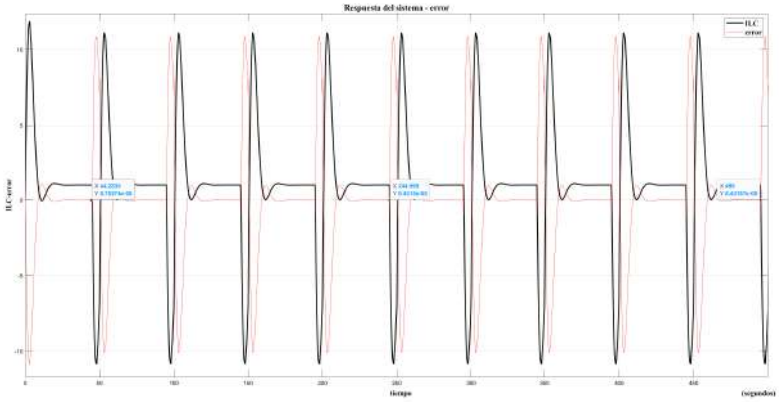


Figura 6.17: Salida del sistema con ILC-P en lazo cerrado de control.

Para la décima iteración se obtiene la figura 6.17, en la cual no se observa un cambio notable de la señal de salida, ya que desde la primera iteración se tiene un error de 10^{-5}, para este ejemplo la aproximación de la posición deseada es bastante rápida, dado que se trabaja con un sistema que tiene una amplitud mayor a 1.

En la figura 6.18 se muestran las señales de error obtenidas en 10 iteraciones con respecto al tiempo de simulación, en cada una de las iteraciones se tiene un error similar a 8.4102×10^{-5}.

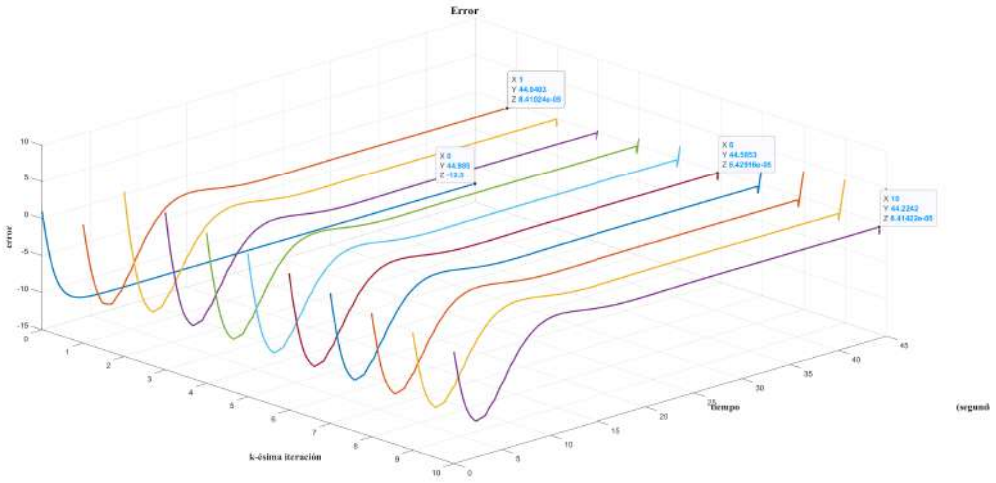


Figura 6.18: Gráfica del error respecto al número de iteraciones y el tiempo.

De acuerdo con las ecuaciones (6.4) y (6.5) las normas respecto al error que se obtiene para el Control de Aprendizaje Iterativo Proporcional con el sistema de la ecuación (6.17) son:

$$\| \mathrm{I} - \mathbf{c}^T \mathrm{B}\Gamma \|_1 = 0.3700 < 1 \tag{6.18}$$

$$\| \mathrm{I} - \mathbf{c}^T \mathrm{B}\Gamma \|_2 = 0.5500 < 1 \tag{6.19}$$

$$\| \mathrm{I} - \mathbf{c}^T \mathrm{B}\Gamma \|_\infty = 0.8200 < 1 \tag{6.20}$$

$$\mathcal{L}_1 = \lim_{k\to\infty} \| e_k \|_1 = 1.3891 \tag{6.21}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| e_k \|_2 = 3.8900 \tag{6.22}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| e_k \|_\infty = 1.3891 \tag{6.23}$$

Las ecuaciones (6.18), (6.19) y (6.20) indican que el sistema de la ecuación (6.17) tiene estabilidad asintótica en el punto de equilibrio. Y como se ha comentado anteriormente la norma $\mathcal{L}_2$ es el índice de desempeño del controlador, en este caso, el control de aprendizaje iterativo tiene una mala exactitud en la respuesta del sistema.

A continuación, se implementa el sistema de la ecuación (6.17) en el lazo cerrado de control Proporcional Derivativo (PD), como se muestra en la figura 6.19.

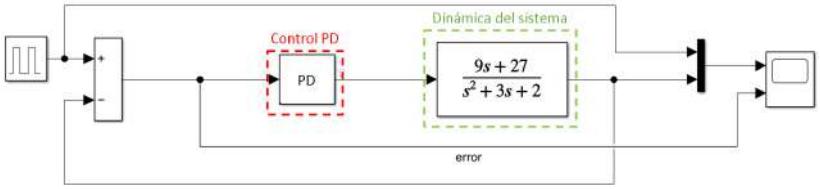


Figura 6.19: Diagrama de bloques implementado en Simulink MATLAB, para el lazo cerrado de control PD.

Con una ganancia proporcional $k_p = 1.5$ y una ganancia derivativa $k_v = 0.7k_p$. La señal que se obtiene se muestra en la figura 6.20.

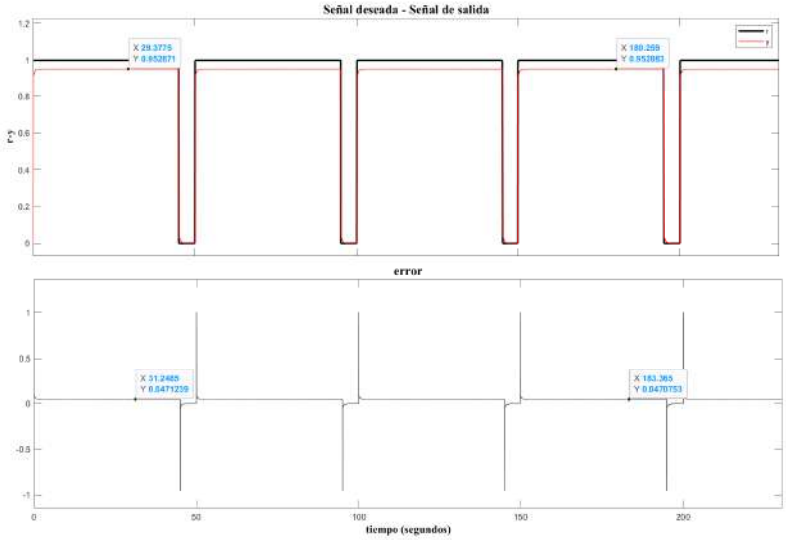


Figura 6.20: Señales de salida del sistema y el error, para el lazo cerrado de control PD.

En la figura 6.20 se observa que la señal de salida se aproxima a la señal deseada, sin embargo, en cada iteración se mantiene la señal, por lo que el error que existe entre la señal deseada y la señal de salida no es corregida cuando $t \to \infty$. Se pueden ocupar ganancias proporcionales mayores a la sintonizada que se aproximen a la posición deseada, sin embargo, tendrían que ser ganancias mayores a 10 que en la práctica no siempre es posible obtenerlas.

Las normas obtenidas para el error de posición en este caso son:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| e_k \|_1 = 0.2087 \tag{6.24}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| e_k \|_2 = 0.0473 \tag{6.25}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| e_k \|_\infty = 0.2087 \tag{6.26}$$

Se tiene un índice de desempeño $\mathcal{L}_2 = 0.0473$ que en comparación con el control de aprendizaje iterativo la exactitud en la respuesta del sistema es mucho mejor con el control PD.

Finalmente se implementa el ILC-P en paralelo con el control PD en el lazo cerrado de control, como se muestra en la figura 6.21.

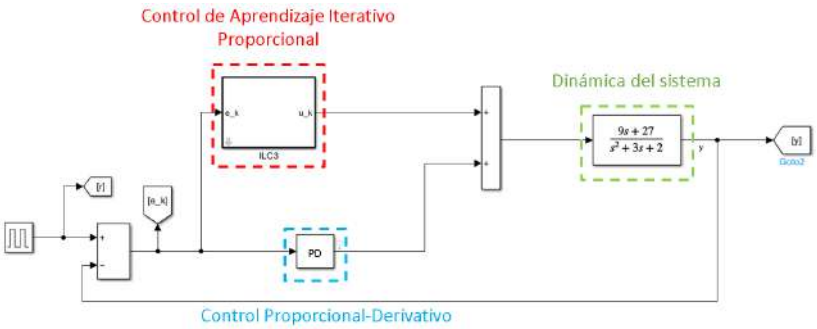


Figura 6.21: Diagrama a bloques implementado en Simulink MATLAB.

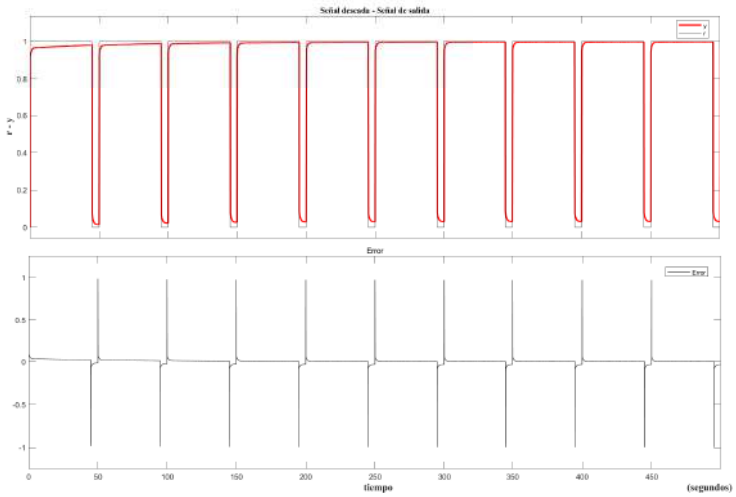


Figura 6.22: Gráfica de la señal de salida y el error del sistema con ILC-P en paralelo con el cotnrol PD del lazo cerrado de control.

La señal de salida que se obtiene se muestra en la figura 6.22, se observa la unión en paralelo del control PD y el control de aprendizaje iterativo proporcional, ya que, corrige el sobreamortiguamiento del sistema original debido al control PD y tras cada tarea repetida, disminuye el error debido al ILC-P.

El error del sistema respecto al número de iteraciones y el tiempo, se muestra en la figura 6.23. Con una ganancia proporcional del ILC $\beta = 0.1$ y la sintonización del control PD con ganancia proporcional $k_p = 1.5$ y ganancia derivativa $k_v = 0.7kp$.

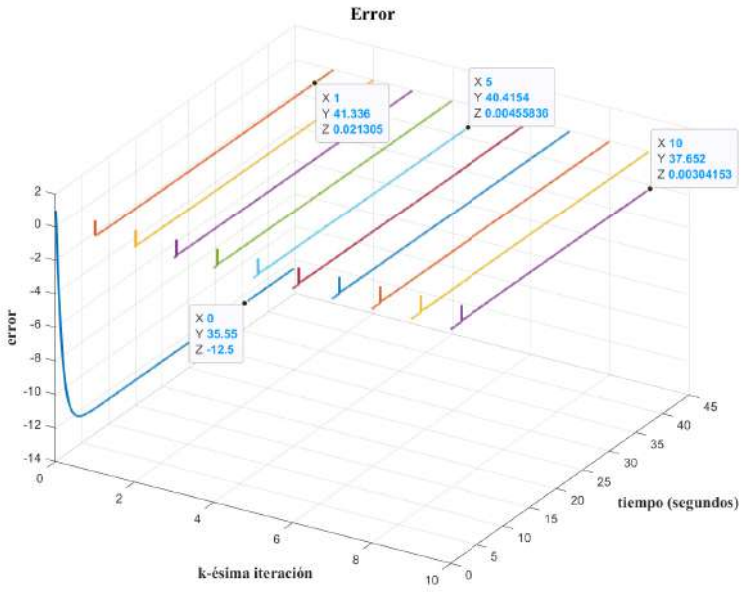


Figura 6.23: Gráfica del error del sistema respecto al número de iteraciones y tiempo.

Las normas obtenidas para este caso son:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| e_k \|_1 = 0.0990 \tag{6.27}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| e_k \|_2 = 0.0207 \tag{6.28}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| e_k \|_\infty = 0.0990 \tag{6.29}$$

El índice de desempeño que se obtiene es $\mathcal{L}_2 = 0.0207$, lo que indica que se tiene una buena exactitud en la respuesta del sistema dinámico, y que se ve reflejado en la figura 6.22 y 6.23. En la figura 6.23 se observa como el error de posición disminuye conforme el número de iteraciones va aumentando, para $k = 1$ se tiene $\widetilde{y}(41.336) = 0.0213$ y para $k = 10$ se tiene $\widetilde{y}(37.652) = 0.0030$.

La tabla 6.2 muestra la comparativa de las normas $\mathcal{L}_i$ en cada una de las simulaciones implementadas para el sistema de segundo orden de la ecuación (6.17). Se observa que el índice de desempeño cambia considerablemente cuando se tiene el control de aprendizaje iterativo y cuando se implementa el control PD junto con el control de aprendizaje iterativo, debido a que el control PD por si solo ya proporciona un desempeño en décimas $\mathcal{L}_2 = 0.0473$, cuando se implementa en conjunto con el control de aprendizaje iterativo se mantiene un buen desempeño y en este caso disminuye 0.0266 unidades debido a la corrección en la etapa transitoria y en el estado estacionario que proporciona el control de aprendizaje iterativo.

$\mathcal{L}_i$	PD	ILC	PD-ILC
$\mathcal{L}_1$	0.2087	1.3891	0.0990
$\mathcal{L}_2$	0.0473	3.8900	0.0207
$\mathcal{L}_\infty$	0.2087	1.3891	0.0990

Tabla 6.2: Comparativa de $\mathcal{L}_i$ para el sistema de la ecuación (6.17).

6.2. Simulación del robot manipulador de un grado de libertad

Sea el sistema dinámico no lineal de segundo orden de la ecuación (6.30) que define el lazo cerrado de control del robot manipulador de un grado de libertad (péndulo simple) se implementan las simulaciones anteriormente realizadas en los sistemas dinámicos de segundo orden.

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \dot{\mathbf{q}} \\ (\boldsymbol{\tau} - b\dot{\mathbf{q}} - mgl_c \sin(\mathbf{q})) \end{bmatrix} \tag{6.30}$$

Este sistema dinámico es no lineal dado el par gravitacional expresado como $\mathbf{g}(\mathbf{q}) = mgl_c sin(\mathbf{q})$. Se implementa la simulación en MATLAB del lazo cerrado de control PD como se muestra en la figura 6.24.

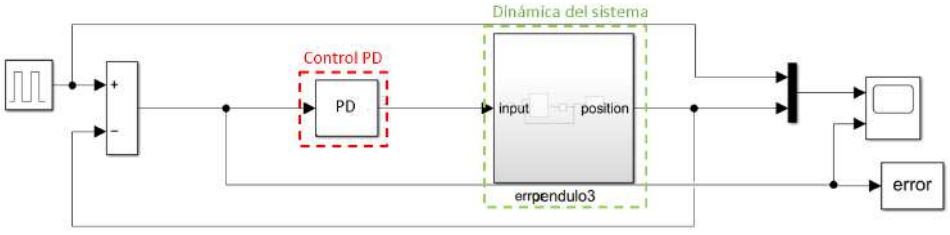


Figura 6.24: Diagrama de bloques del péndulo simple en lazo cerrado de control PD.

Con una ganancia proporcional $k_p = 5$ y una ganancia derivativa $k_v = 0.7k_p$ obtenidas experimentalmente después de probar con diferentes ganancias y determinar que la señal de salida no tiene un cambio significativo para ganancias proporcionales $k_p \gg 5$. La señal que se obtiene para la primera iteración se muestra en la figura 6.25.

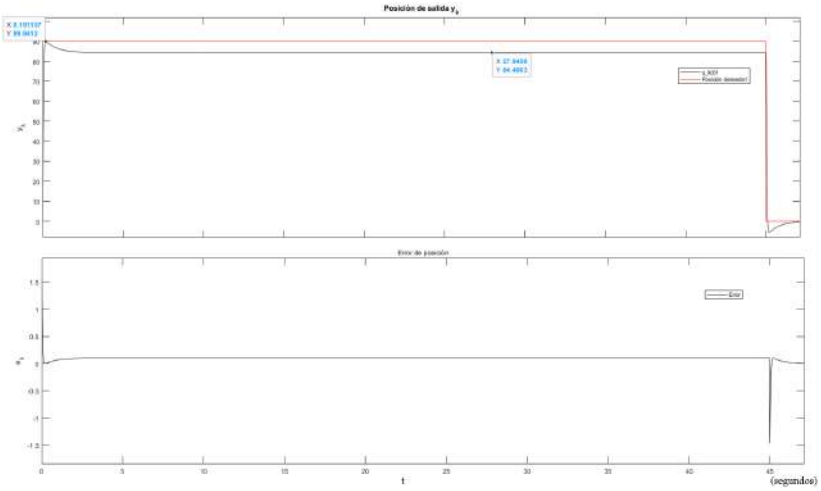


Figura 6.25: Señal de salida del sistema y error.

En la figura 6.25 se observa la respuesta del sistema en el lazo cerrado de control PD, el cual tiene un sobreimpulso en la etapa transitoria, sin embargo, la respuesta del sistema se aproxima a la posición deseada en el estado estacionario y la posición de salida es $q = 84.4063$. El error que se obtiene no se reduce durante 10 iteraciones, tal y como se muestra en la figura 6.26.

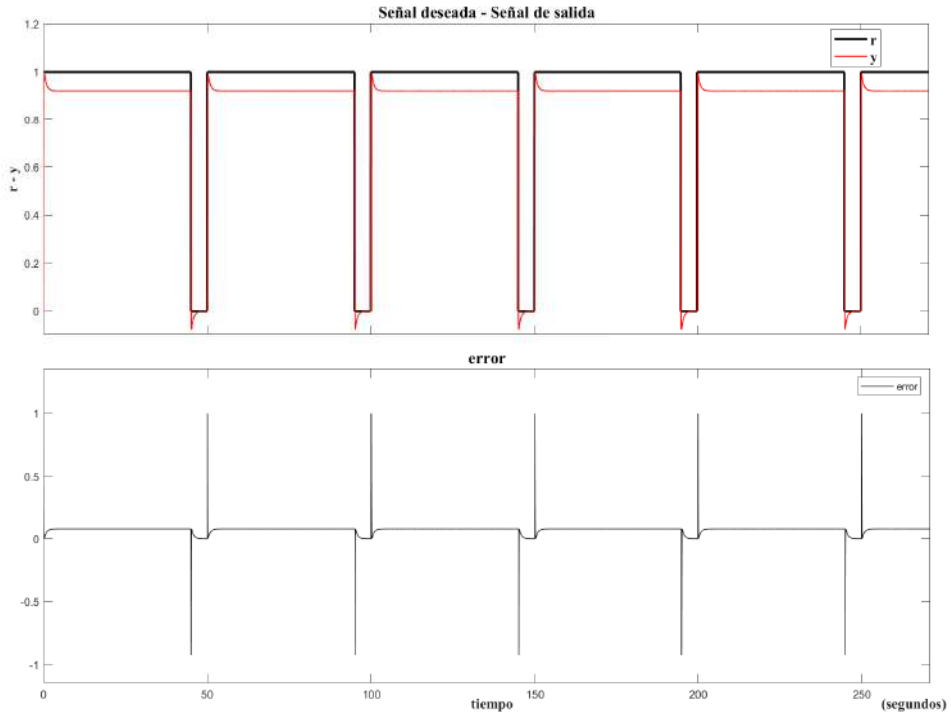


Figura 6.26: Posición de salida q_k en 10 iteraciones.

La salida del sistema que se obtiene esta en función de las ganancias atenuadas anteriormente, ya que, la ley de control es directamente proporcional al error de posición del péndulo debido al control proporcional

$k_p\tilde{q}$ y la acción de control derivativa $k_v\dot{q}$ tiene el efecto de amortiguamiento o freno mecánico y mejora el desempeño del control proporcional.

Las normas obtenidas para este caso son:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| e_k \|_1 = 0.31636 \tag{6.31}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| e_k \|_2 = 0.15546 \tag{6.32}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| e_k \|_\infty = 0.31636 \tag{6.33}$$

El índice de desempeño $\mathcal{L}_2 = 0.1554$ de la respuesta del control PD muestra una exactitud en décimas la cual puede ser bien implementada físicamente si no se busca un alto desempeño en la salida de posición, lo cual se logra cuando la exactitud en el error de posicionamiento es idealmente cero.

Finalmente en la figura 6.27 se implementa el robot manipulador de un grado de libertad (péndulo simple), y como se ha visto anteriormente, cuando solo se implementa el control PD no se llega a la posición deseada conforme se repiten las tareas repetidas, por ello, se implementa el lazo cerrado de control PD en conjunto con el control de aprendizaje iterativo, con una posición articular deseada es $q_d = 90°$.

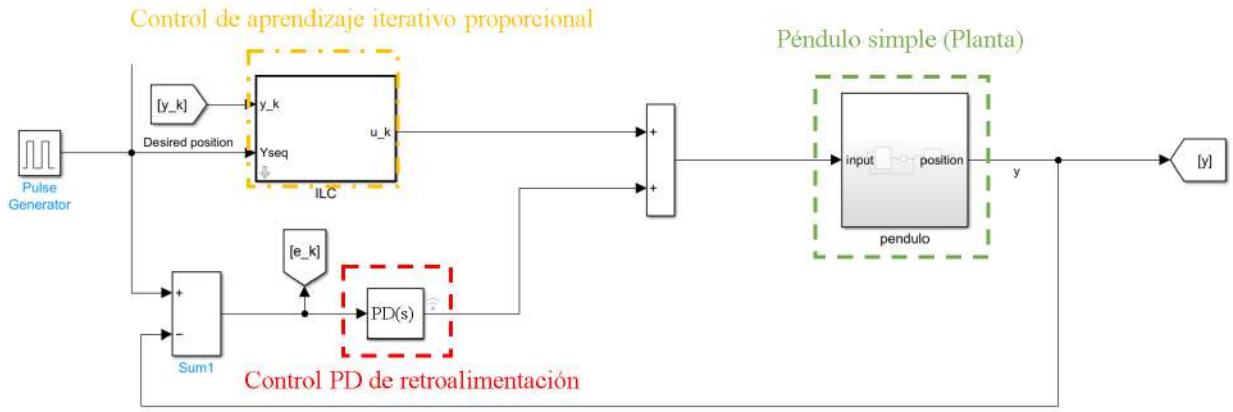


Figura 6.27: Simulación en MatLab simulink del péndulo simple.

La señal de salida y el error de posición se muestran en la figura 6.28, con $\beta = 0.1$, la ganancia proporcional y derivativa igual a $k_p = 5$ y $k_v = 0.7k_p$. Como se observa en la figura 6.28, el péndulo simple llega a la posición deseada conforme la tarea se va repitiendo, así como el error de posición va disminuyendo. Esto gracias a que el control PD de retroalimentación ayuda a que el sistema no lineal llegue al estado estacionario, y la implementación del control de aprendizaje iterativo disminuye el error de acuerdo a la posición deseada.

El error del sistema respecto al número de iteraciones y el tiempo, se muestra en la figura 6.28. Con una ganancia proporcional del ILC $\beta = 0.1$ y la sintonización del control PD con ganancia proporcional $k_p = 5$ y ganancia derivativa $k_v = 0.7kp$.

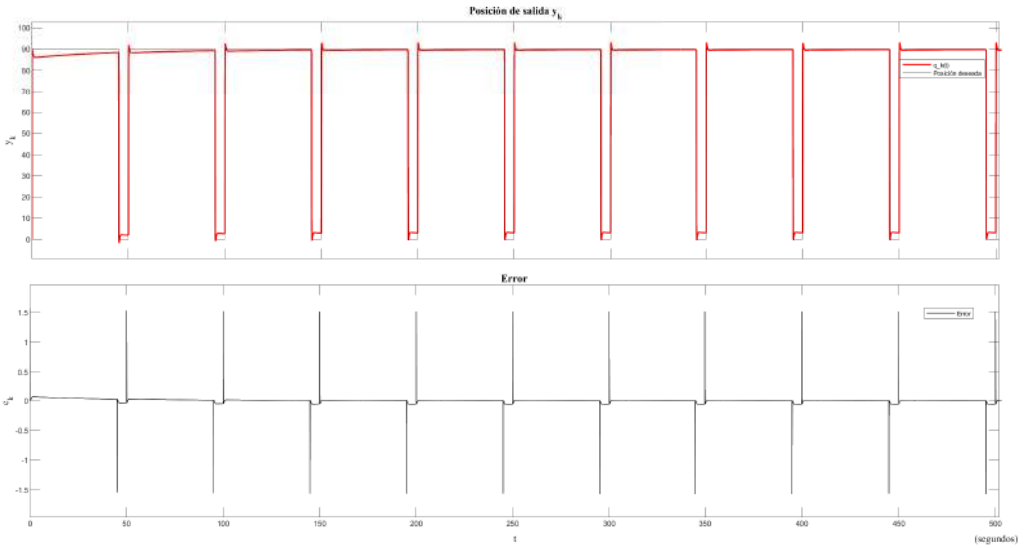


Figura 6.28: Señal de salida del péndulo simple y error de posición

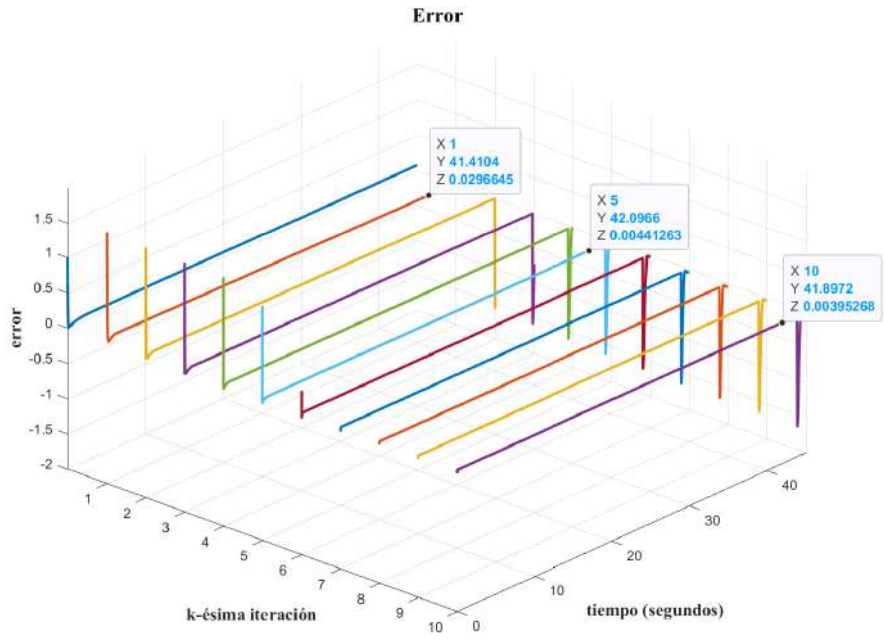


Figura 6.29: Error de posición del lazo cerrado de control respecto al número de iteraciones y al tiempo.

Las normas obtenidas para este caso son:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| e_k \|_1 = 0.16576 \tag{6.34}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| e_k \|_2 = 0.12983 \tag{6.35}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| e_k \|_\infty = 0.16576 \tag{6.36}$$

En la figura 6.29 se observa como disminuye el error de posición deseada en cada tarea repetida, para $k = 1$ se tiene $\widetilde{q}(41.41) = 0.0296$, mientras que para $k = 10$ se obtiene $\widetilde{q}(41.89) = 0.0039$.

En la tabla 6.3 se muestra la comparativa de las normas $\mathcal{L}_i$ obtenidas:

$\mathcal{L}_i$	PD	PD-ILC
$\mathcal{L}_1$	0.3136	0.1657
$\mathcal{L}_2$	0.1554	0.1298
$\mathcal{L}_\infty$	0.3136	0.1657

Tabla 6.3: Comparativa de $\mathcal{L}_i$ para el sistema de la ecuación (6.30).

El índice de desempeño que se obtiene en este caso es igual a $\mathcal{L}_2 = 0.1298$, en comparación con el resultado obtenido anteriormente se redujo el indice de desempeño 0.02563 unidades. Esto significa que el índice de desempeño esta en función de la respuesta transitoria del lazo cerrado de control, por lo tanto, el control PD es quien regula el índice de desempeño.

6.3. Simulación del robot manipulador de dos grados de libertad

La ecuación del lazo cerrado de control para el robot manipulador de 2gdl se muestra en la ecuación (6.37), donde $\boldsymbol{\tau}$ representa el par que inyecta la energía al robot manipulador.

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} -\dot{\mathbf{q}} \\ M^{-1}\mathbf{q}(\boldsymbol{\tau} - C\dot{\mathbf{q}} - B\dot{\mathbf{q}})) \end{bmatrix} \tag{6.37}$$

Anteriormente ya se han realizado las simulaciones correspondientes para el péndulo simple, sin embargo, en el robot manipulador de dos grados de libertad se tiene una matriz de inercias y se presentan fuerzas centrípetas y de coriolis generando fuerzas radiales y desviación del movimiento de traslación debido a su componente de rotación, esto implica que en la matriz de inercias y en la matriz de fuerzas centripetas y de coriolis existen elementos no lineales (sen y cos) expresados en la ecuación (3.111). Sin embargo, los pares gravitacionales se cancelan en el lazo cerrado de control PD.

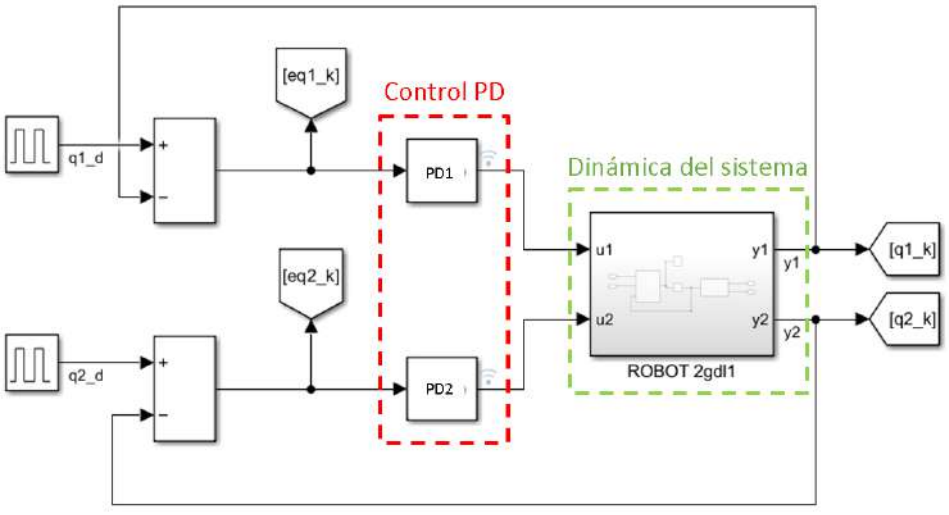


Figura 6.30: Simulación en simulink del lazo cerrado de control PD para el robot manipulador de dos grados de libertad.

Se implementan las simulaciones para el caso del robot manipulador de dos grados de libertad con el lazo cerrado de control PD, como se muestra en la figura 6.30.

En la figura 6.31 se tienen las posiciones de salida q_1 y q_2 correspondientes al extremo final del primer y segundo eslabón respectivamente. Con posiciones deseadas $q_{d_1} = 45°$ y $q_{d_2} = 90°$.

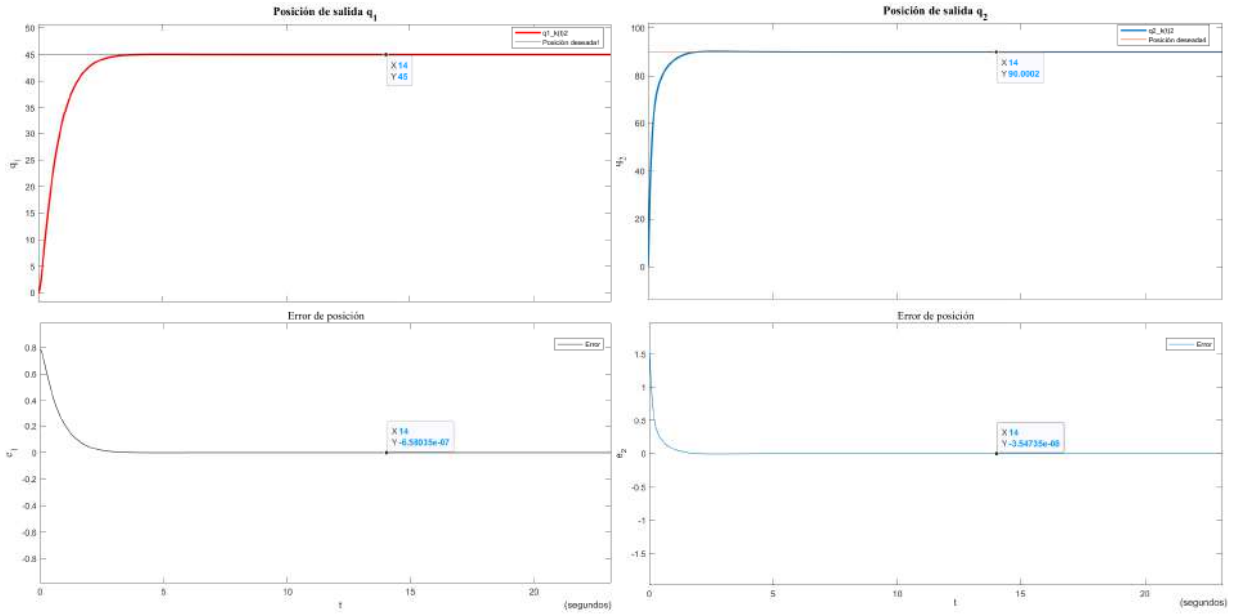


Figura 6.31: Posición de salida del primer eslabón q_1 y del segundo eslabón q_2 para $k = 1$.

Las matrices de ganancias proporcionales y derivativas que se implementaron para este ejemplo son: $K_p = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$ y $K_v = \begin{bmatrix} 0.7(4.5) & 0 \\ 0 & 0.7k_p \end{bmatrix}$, las cuales, fueron atenuadas proponiendo diferentes ganancias. En la figura 6.31 se observan las salidas de posición en la primera iteración o primer tarea realizada, analizando la etapa transitoria se observa un sistema críticamente amortiguado, debido a que se tienen matrices de ganancia derivativa que contrarrestan la energía al control proporcional de tal forma que la acción de control derivativo reduce los sobreimpulsos dada la naturaleza del robot manipulador expresada en su modelo dinámico, mientras que en estado estacionario se llega a la posición deseada, por lo que te tienen las posiciones $q_1(14) = 45°$ y $q_2(14) = 90°$, mientras que los errores de posición $\widetilde{q}_1(14) = -6.5803 \times 10^{-7}$ y $\widetilde{q}_2(14) = -3.5473 \times 10^{-8}$, lo que significa que se da un poco más de la energía requerida para colocar al robot manipulador en las posiciones deseadas, sin embargo, este error puede ser interpretado como un error nulo.

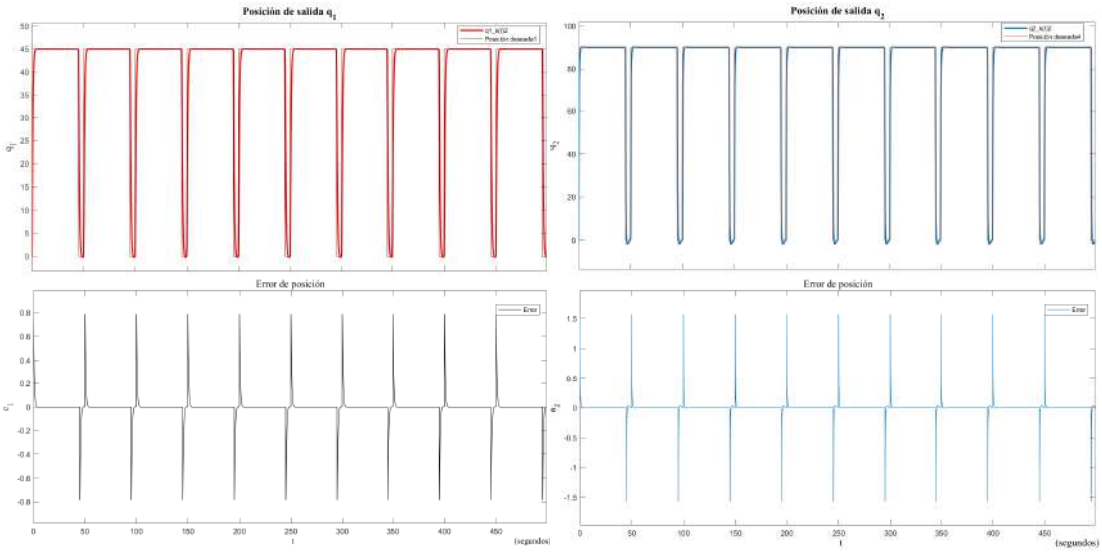


Figura 6.32: Señales de salida del robot manipulador de 2gdl y errores de posición

Las posiciones de salida cuando se repite 10 veces la tarea junto con sus errores de posición se muestran en la figura 6.32, que como ha sucedido en todos los ejemplos anteriores, se tiene la misma salida en cada una de las tareas repetidas.

Se obtienen las normas euclidianas para cada posición articular:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| \widetilde{q}_{1_k} \|_1 = 0.0465 \tag{6.38}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| \widetilde{q}_{1_k} \|_2 = 0.031601 \tag{6.39}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| \widetilde{q}_{1_k} \|_\infty = 0.0465 \tag{6.40}$$

$$\mathcal{L}_1 = \lim_{k\to\infty} \| \widetilde{q}_{2_k} \|_1 = 0.047215 \tag{6.41}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| \widetilde{q}_{2_k} \|_2 = 0.037834 \tag{6.42}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| \widetilde{q}_{2_k} \|_\infty = 0.047215 \tag{6.43}$$

Donde el índice de desempeño $\mathcal{L}_2$ indica un buen desempeño tanto para el primer eslabón como para el segundo, de tal forma que el control PD inyecta la energía suficiente para llevar los extremos finales a la posición deseada.

Finalmente en la figura 6.33 se implementa el robot manipulador de dos grados de libertad en conjunto con el control de aprendizaje iterativo, donde las posiciones articulares deseadas son: $q_{d_1} = 45°$. $q_{d_2} = 90°$.

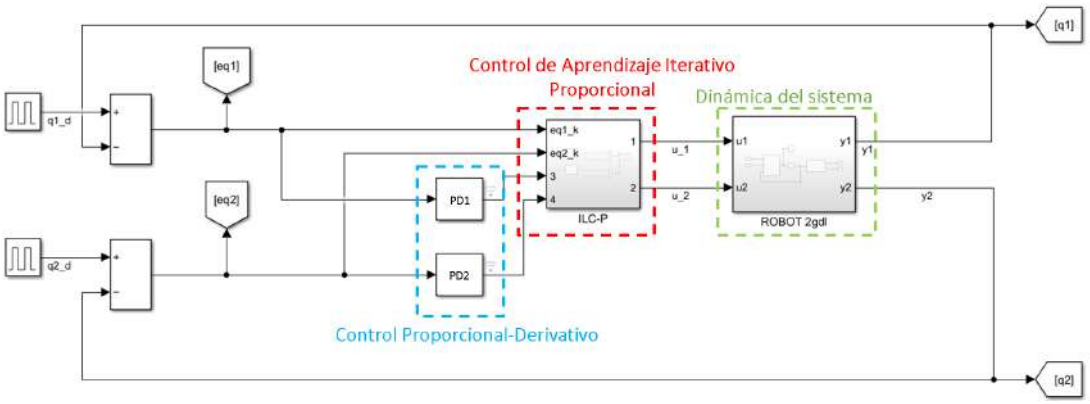


Figura 6.33: Simulación en MatLab simulink del robot de 2gdl.

Las posiciones de salida para $k = 1$ son $q_1(14) = 45.378$ y $q_2(14) = 91.514$ correspondientes al extremo final del primer y segundo eslabón respectivamente con posiciones deseadas $q_{d_1} = 45°$ y $q_{d_2} = 90°$ mostradas en la figura 6.34.

Las señales de salida y los errores de posición se muestran en la figura 6.35, con $\beta = 0.01$ y $\gamma = 0.1$, las matrices de ganancia proporcional y derivativa son $K_p = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$ y $K_v = \begin{bmatrix} 0.7(4.5) & 0 \\ 0 & 0.7k_p \end{bmatrix}$. De acuerdo con la figura 6.35 y las posiciones obtenidas, se tiene un leve incremento de la posición de salida producido por la integración del control de aprendizaje iterativo que aporta una ganancia adicional para corregir el error que en este caso particular es tan pequeño que se considera como cero y por esto mismo se ve reflejado en la etapa transitoria y en el estado estacionario.

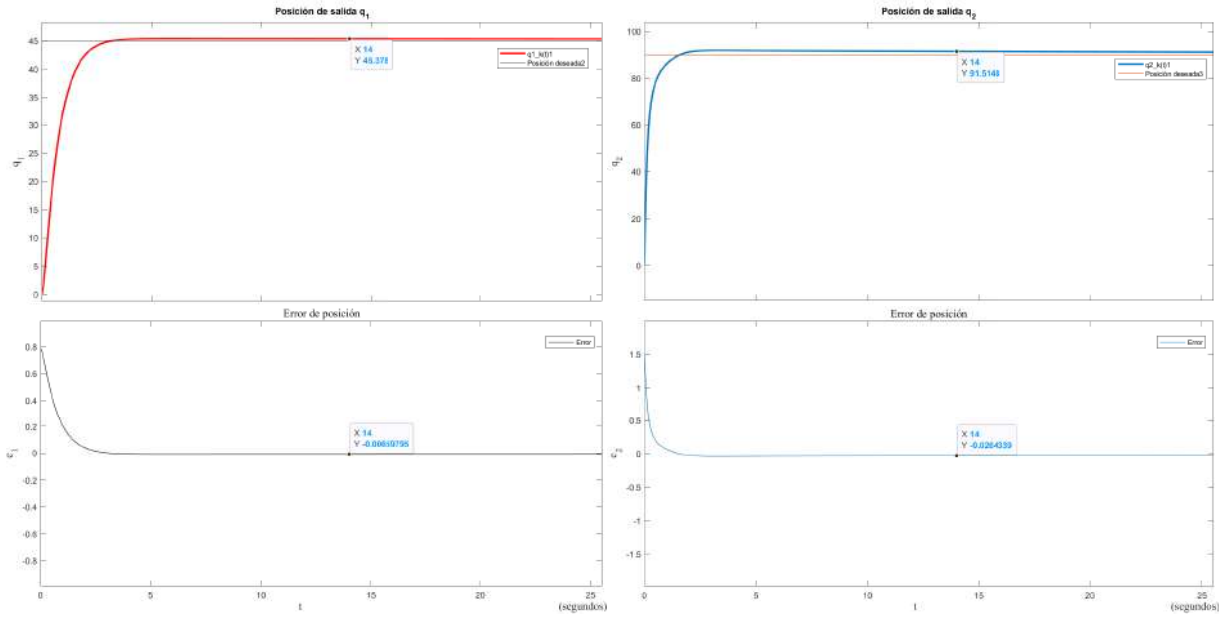


Figura 6.34: Posiciones de salida q_1 y q_2 del robot manipulador de dos grados de libertad.

Esta pequeña ganancia adicional disminuye lentamente en cada tarea realizada como se muestra en la figura 6.35, sin embargo, no se obtiene un error del mismo valor como en el caso anterior.

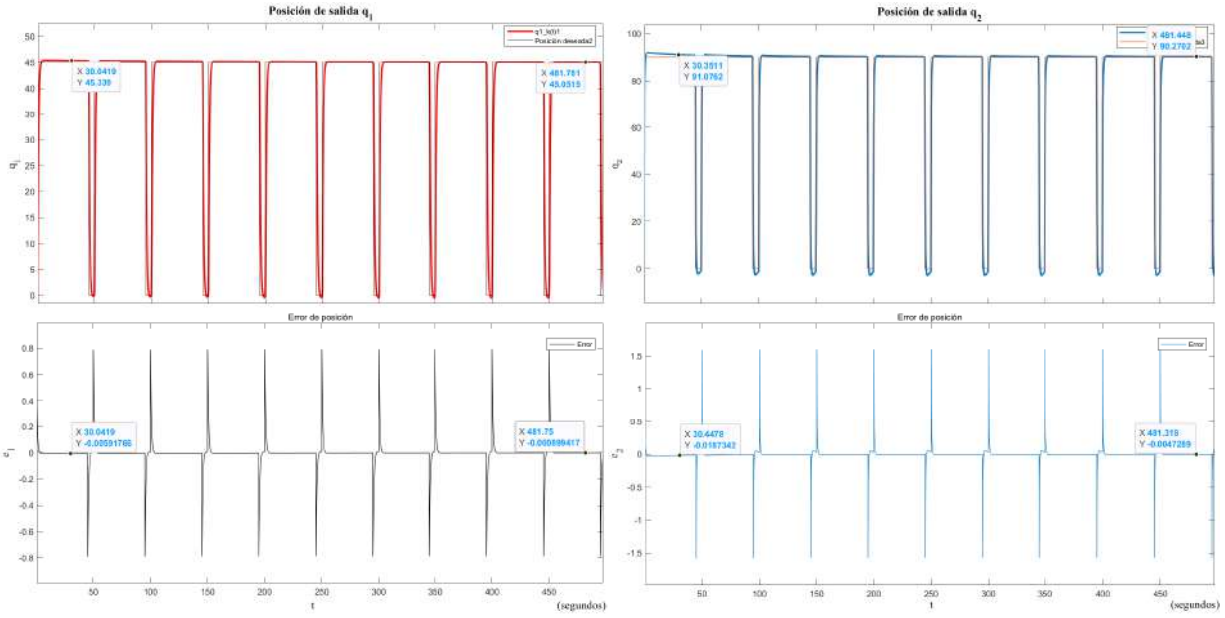


Figura 6.35: Señales de salida del robot manipulador de 2gdl y errores de posición

Las normas obtenidas para este caso son:

$$\mathcal{L}_1 = \lim_{k\to\infty} \| \widetilde{q}_{1_k} \|_1 = 0.1772 \tag{6.44}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| \widetilde{q}_{1_k} \|_2 = 0.1236 \tag{6.45}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| \widetilde{q}_{1_k} \|_\infty = 0.1772 \tag{6.46}$$

$$\mathcal{L}_1 = \lim_{k\to\infty} \| \widetilde{q}_{2_k} \|_1 = 0.1817 \tag{6.47}$$

$$\mathcal{L}_2 = \lim_{k\to\infty} \| \widetilde{q}_{2_k} \|_2 = 0.1558 \tag{6.48}$$

$$\mathcal{L}_\infty = \lim_{k\to\infty} \| \widetilde{q}_{2_k} \|_\infty = 0.1817 \tag{6.49}$$

El índice de desempeño aumenta en este caso debido a la ganancia extra que incorpora el control de aprendizaje iterativo.

La tabla 6.4 contiene las normas $\mathcal{L}_i$ donde se comparan los índices de desempeño $\mathcal{L}_2$ para los dos casos anteriormente simulados. El mejor índice de desempeño esta dado por el control PD, sin embargo, el índice de desempeño del control PD con el ILC es bueno, ya que solo se tiene una diferencia de 0.092 para q_1 y de 0.118 para q_2.

$\mathcal{L}_i$	PD (q_1)	PD-ILC (q_1)	PD (q_2)	PD-ILC (q_2)
$\mathcal{L}_1$	0.0465	0.1772	0.0472	0.1817
$\mathcal{L}_2$	0.0316	0.1236	0.0378	0.1558
$\mathcal{L}_\infty$	0.0472	0.1657	0.0472	0.1817

Tabla 6.4: Comparativa de $\mathcal{L}_i$ para el sistema de la ecuación (6.30).

A pesar de que en el ejemplo anterior no es posible ver el aporte del control de aprendizaje iterativo, es importante resaltar una de las ventajas de éste: "se adapta a situaciones desconocidas". Por lo tanto, se realiza una última prueba que consiste en ingresar una ganancia en las posiciones actuales $\mathbf{q}(t)$ y ver como reaccionan los controles en esta situación. Esa ganancia ingresada provoca una perturbación en las posiciones de salida $\mathbf{q}(t)$.

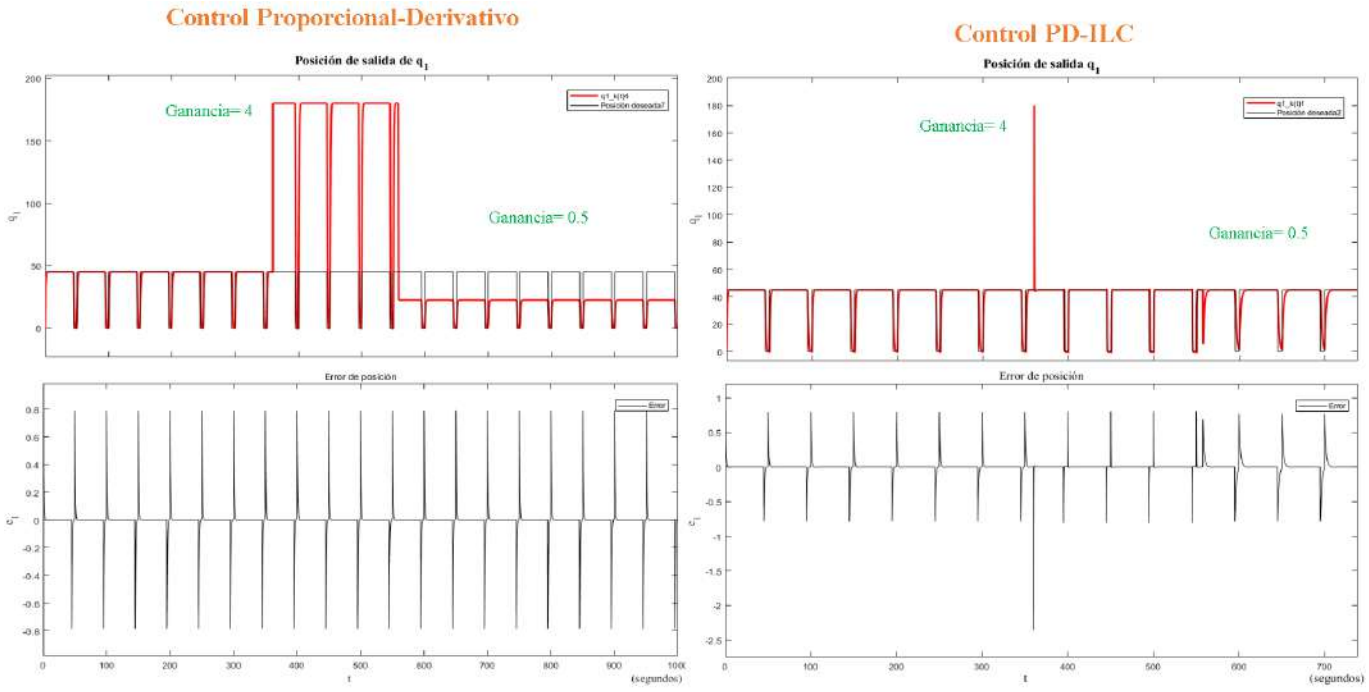


Figura 6.36: Gráficas comparativas de la posición de salida q_1 ante perturbaciones del control PD vs el control PD-ILC, primero se añade una ganancia = 4, y después una ganancia= 0.5.

En la figura 6.36 se muestran las posiciones de salida del control PD y del control PD-ILC para el primer eslabón q_1. Durante la simulación se ingresa una ganancia de 4 unidades a la posición de salida, la cual provoca la perturbación mostrada en la posición de salida de ambos controles, sin embargo, en el caso del control PD la perturbación generada en q_1 se mantiene y no se corrige, mientras que para el control PD-ILC se corrige de manera inmediata dicha perturbación. Ocurre lo mismo cuando se agrega una ganancia de 0.5 unidades para a la posición de salida, la perturbación generada no es corregida en el control PD, pero si es corregida en el control PD-ILC.

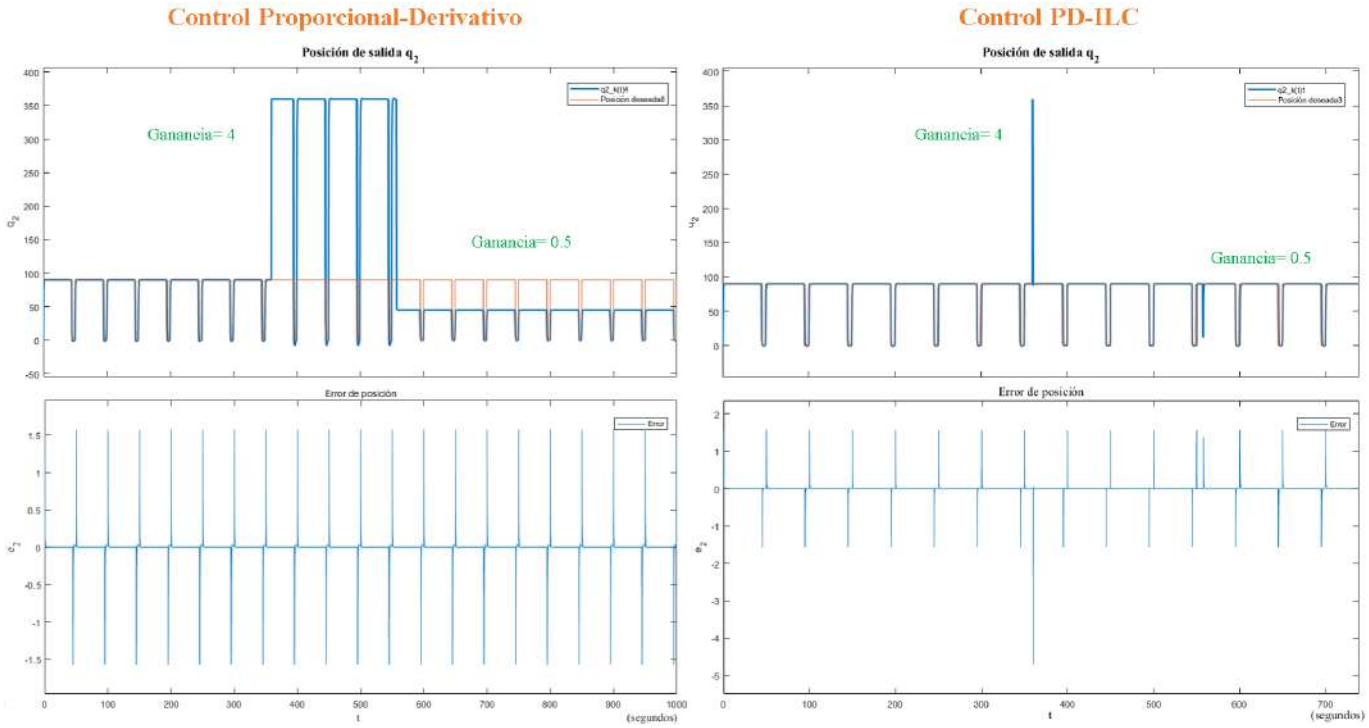


Figura 6.37: Gráficas comparativas de la posición de salida q_2 ante perturbaciones del control PD vs el control PD-ILC, primero se añade una ganancia = 4, y después una ganancia= 0.5.

Para la posición de salida q_2 de la figura 6.37 se tiene la misma situación que en q_1. Esto ocurre debido a que el control de aprendizaje iterativo almacena el vector de posiciones anteriores, lo que permite detectar un cambio en la posición de salida. La cual es corregida por la ganancia proporcional β en q_1 y α en q_2.

6.4. Control PD+ de aprendizaje iterativo

A continuación se presenta una propuesta teórica para el control PD+ siguiendo el desarrollo de (Arimoto, 1986), de tal forma, que se tenga un ejemplo teórico que defina el proceso iterativo recursivo del lazo cerrado de control, como se muestra en la figura 6.38, donde se considera un robot manipulador de 3gdl también llamado rotradi. Es importante resaltar que esta propuesta se realiza con la finalidad de entender los conceptos teóricos que implementa (Arimoto, 1986) respecto al control de aprendizaje iterativo de trayectorias, sin embargo, esta propuesta deja en claro, que así sea para un control PD, PD+ o par calculado es posible realizar este tipo de propuesta en conjunto con el control de aprendizaje iterativo para trabajos a futuro.

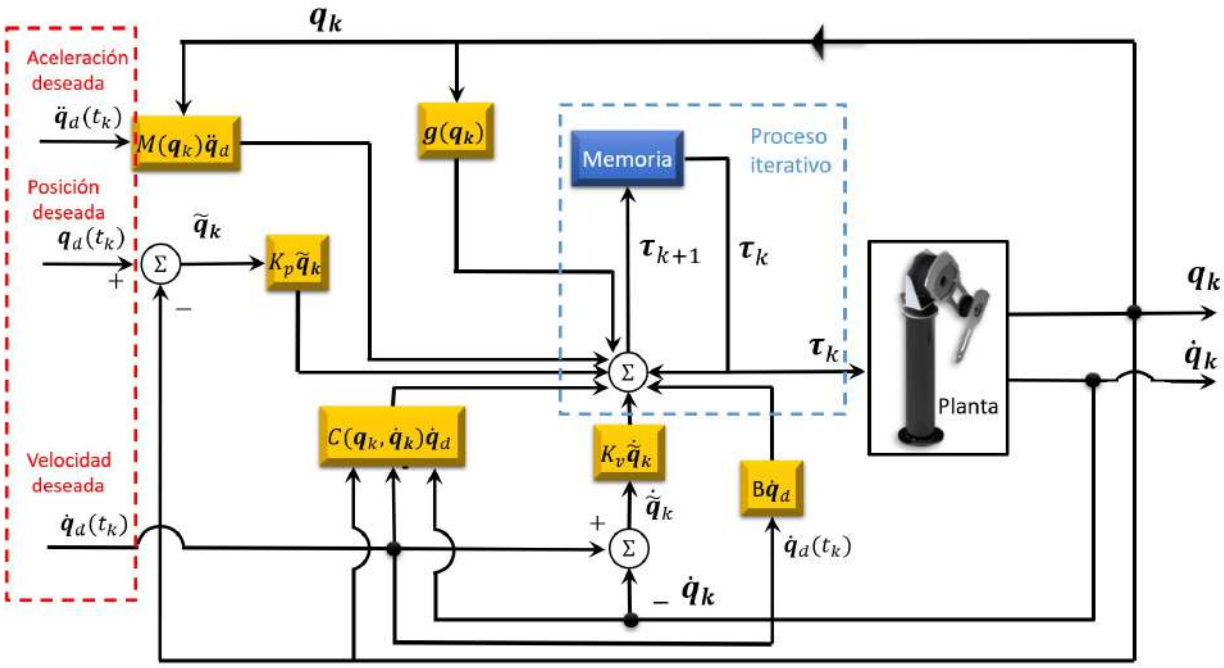


Figura 6.38: Diagrama de bloques del controlador PD+ de aprendizaje iterativo.

Se desarrolla la demostración de convergencia del punto de equilibrio mediante la función estricta de Lyapunov. Sea el modelo dinámico del robot manipulador:

$$\boldsymbol{\tau}_k = M(\mathbf{q}_k)\ddot{\mathbf{q}}_k + C(\mathbf{q}_k, \dot{\mathbf{q}}_k)\dot{\mathbf{q}}_k + \mathbf{g}(\mathbf{q}_k) + B\dot{\mathbf{q}}_k \tag{6.50}$$

Y el control PD+:

$$\begin{aligned}
\boldsymbol{\tau}_{PD} &= K_p\tilde{\mathbf{q}}_k + K_v\dot{\tilde{\mathbf{q}}}_k + M(\mathbf{q}_k)\ddot{\mathbf{q}}_d + C(\mathbf{q}_k, \dot{\mathbf{q}}_k)\dot{\mathbf{q}}_d + B\dot{\mathbf{q}}_d + \mathbf{g}(\mathbf{q}_k) && (6.51)\\
\boldsymbol{\tau}_k &= \boldsymbol{\tau}_{PD} && (6.52)\\
\boldsymbol{\tau}_k &= M(\mathbf{q}_k)\ddot{\tilde{\mathbf{q}}}_d + K_p\tilde{\mathbf{q}}_k + K_v\dot{\tilde{\mathbf{q}}}_k + C(\mathbf{q}_k, \dot{\mathbf{q}}_k)\dot{\tilde{\mathbf{q}}}_k + B\dot{\tilde{\mathbf{q}}}_k && (6.53)\\
\boldsymbol{\tau}_{k+1} &= \boldsymbol{\tau}_k + K_p\tilde{\mathbf{q}}_k + K_v\dot{\tilde{\mathbf{q}}}_k && (6.54)
\end{aligned}$$

La ecuación (6.55) de lazo cerrado que involucra el modelo dinámico del robot y el esquema de control PD+, expresado en variables de estado que define el problema de control de trayectoria, queda expresado como:

$$\frac{d}{dt}\begin{bmatrix}\tilde{\mathbf{q}}_k\\ \dot{\tilde{\mathbf{q}}}_k\end{bmatrix} = \begin{bmatrix}\dot{\tilde{\mathbf{q}}}\\ -M^{-1}(\mathbf{q}_d - \tilde{\mathbf{q}}_k)\left[K_p\tilde{\mathbf{q}}_k + K_v\dot{\tilde{\mathbf{q}}}_k + C(\mathbf{q}_d(t) - \tilde{\mathbf{q}}_k, \dot{\mathbf{q}}_d - \dot{\tilde{\mathbf{q}}}_k)\dot{\tilde{\mathbf{q}}}_k + B\dot{\tilde{\mathbf{q}}}_k\right]\end{bmatrix} \tag{6.55}$$

Se propone la siguiente función estricta de Lyapunov:

$$V(\dot{\tilde{\mathbf{q}}}_k, \tilde{\mathbf{q}}_k) = \underbrace{\frac{1}{2}\dot{\tilde{\mathbf{q}}}_k^T M(\mathbf{q}_k)\dot{\tilde{\mathbf{q}}}_k}_{\substack{\text{Término cuadrático del}\\ \text{error de velocidad}}} + \underbrace{\frac{1}{2}\tilde{\mathbf{q}}_k^T K_p\tilde{\mathbf{q}}_k}_{\substack{\text{Término cuadrático del}\\ \text{error de la posición}}} + \underbrace{\frac{\epsilon_0}{1+\parallel\tilde{\mathbf{q}}_k\parallel}\tilde{\mathbf{q}}_k^T M(\mathbf{q}_k)\dot{\tilde{\mathbf{q}}}_k}_{\substack{\text{Término de}\\ \text{productos cruzados}}} \tag{6.56}$$

donde $V(\dot{\tilde{\mathbf{q}}}_k, \tilde{\mathbf{q}}_k) > 0$.

Se obtiene la derivada respecto al tiempo de la ecuación (6.56)

$$\dot{V}(\dot{\tilde{\mathbf{q}}}_k, \tilde{\mathbf{q}}_k) \leq -\begin{bmatrix} \| \tilde{\mathbf{q}}_k \| \\ \| \dot{\tilde{\mathbf{q}}}_k \| \end{bmatrix}^T \begin{bmatrix} \frac{\epsilon_0}{1+\|\tilde{\mathbf{q}}_k\|}\lambda_{K_p}^{min} & -\frac{1}{2}\frac{\epsilon_0}{1+\|\tilde{\mathbf{q}}_k\|}\left[2k_c + \lambda_{K_v}^{max}\right] \\ -\frac{1}{2}\frac{\epsilon_0}{1+\|\tilde{\mathbf{q}}_k\|}\left[2k_c + \lambda_{K_v}^{max}\right] & \left[\lambda_{K_v}^{min} - \frac{\epsilon_0}{1+\|\tilde{\mathbf{q}}_k\|}\left(\lambda_{M(\mathbf{q}_k)}^{max} - 1\right)\right] \end{bmatrix} \begin{bmatrix} \| \tilde{\mathbf{q}}_k \| \\ \| \dot{\tilde{\mathbf{q}}}_k \| \end{bmatrix} < 0$$

$$V(\dot{\tilde{\mathbf{q}}}_k, \tilde{\mathbf{q}}_k) < 0 \Rightarrow \lim_{t\to\infty} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} \to \mathbf{0} \in \mathbb{R}^{2n} \; \forall t \geq 0 \tag{6.57}$$

Lo cual demuestra estabilidad asintótica del punto de equilibrio $\begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}$

.

Esta demostración de estabilidad del punto de equilibrio cuando $\mathbf{u}_k = 0$ asegura la acotación de todos los componentes de $\dot{\tilde{\mathbf{q}}}$. Se asume entonces que en la primera iteración $(k = 0)$ la derivada de $\dot{\tilde{\mathbf{q}}}_0$ desde la posición deseada $\dot{\tilde{\mathbf{q}}}_d$ esta acotada desde arriba de la siguiente forma (Arimoto, 1986):

$$\| \dot{\tilde{\mathbf{q}}}_0 - \dot{\tilde{\mathbf{q}}}_d \| < \gamma_0 \tag{6.58}$$

Siempre que la primera entrada se establezca como $\mathbf{u}_0(t) = 0$ en la ecuación (6.59).

$$M(\mathbf{q}_k)\ddot{\mathbf{q}}_k + C(\mathbf{q}_k, \dot{\mathbf{q}}_k)\dot{\mathbf{q}}_k + \left[B + K_v\right]\dot{\mathbf{q}}_k - K_p\tilde{\mathbf{q}}_k = \mathbf{u}_k \tag{6.59}$$

Mientras que la norma del vector $\mathbf{q}$ esta definida como:

$$\| \mathbf{q} \| = max_{i=1,\ldots,n} \| \mathbf{q}_i \| \tag{6.60}$$

Se asume acotación uniforme del error de la velocidad, donde existe una $\gamma_1 > 0$ de tal forma que:

$$\| \dot{\tilde{\mathbf{q}}}_k - \dot{\tilde{\mathbf{q}}}_d \| < \gamma_1 \text{ Para toda } k = 1, 2, \ldots n \tag{6.61}$$

Y sea el control de aprendizaje iterativo proporcional:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \Lambda\tilde{\mathbf{q}}_k \tag{6.62}$$

$$\tilde{\mathbf{q}}_k = \mathbf{q}_d - \mathbf{q}_k \tag{6.63}$$

donde $\Lambda > 0$ representa la matriz diagonal de ganancias proporcionales del algoritmo de control iterativo.

Por el *Teorema 1* propuesto por Arimoto se asume que la dinámico del robot están gobernadas por la ecuación (6.59) y la siguiente condición:

$$\| I - \Lambda M^{-1}(\mathbf{q}) \| \leq \rho < 1 \tag{6.64}$$

De tal forma que esto se cumple cuando se obtiene la desigualdad siguiente:

$$\| M(\mathbf{q})\dot{\tilde{\mathbf{q}}} \|_\lambda \leq \rho_0^k \| M(\mathbf{q})\dot{\tilde{\mathbf{q}}} \|_\lambda \tag{6.65}$$

Lo cual implica:

$$\dot{\tilde{\mathbf{q}}}(t) \to 0 \text{ uniformemente en } t \in [0, t_f] \tag{6.66}$$

conforme $k \to \infty$. Finalmente, dado que $\widetilde{\mathbf{q}}(0) = 0$ y $\mathbf{q}_k(0) = 0$ para toda k, se concluye que:

$$\dot{\widetilde{\mathbf{q}}}_k(t) \;\; \to \;\; \dot{\widetilde{\mathbf{q}}}_d(t) \text{ y } \; \mathbf{q}_k(t) \to \mathbf{q}_d \text{ uniformemente en } t \; \in \; [0, t_f] \tag{6.67}$$

conforme $k \to \infty$. Quedando demostrada la desigualdad (Arimoto, 1986).

Conclusiones

Con los estudios realizados sobre robots manipuladores, pasividad y aprendizaje de máquina se implementaron las simulaciones de red neuronal y aprendizaje iterativo. Los resultados obtenidos con la propuesta de red neuronal, demuestran que con una sola capa neuronal es posible sustituir la cinemática directa e inversa en los controles de lazo cerrado cartesiano y articular, respectivamente, además de que el error de posición se encuentra en unidades de milésimas satisfaciendo $e \to 0$ cuando $t \to \infty$. Esto permitió aplicar el aprendizaje en robots manipuladores con una red neuronal y dio paso a continuar con el estudio de los algoritmos canónicos de control de aprendizaje iterativo, haciendo posible demostrar la estabilidad asintótica del punto de equilibrio en el control de aprendizaje iterativo escalar y vectorial (sistema dinámico de segundo orden), se realizó la simulación de control de aprendizaje iterativo para un sistema lineal escalar, vectorial y el péndulo simple (no lineal).

Los resultados obtenidos con ambos sistemas lineales en el caso escalar y vectorial, muestran que a partir de definir diferentes números de iteraciones para la señal de salida, que en ambos casos fue llegar a 1, se tuvo una mejor respuesta transitoria al aumentar las iteraciones, mientras que el tiempo de llegada disminuyó, ya que, el paso de integración esta ligado con el tiempo de simulación, sea $t_k = hk$ donde h es el paso de integración. Sin embargo, para el caso del péndulo simple, dada su naturaleza no lineal, se tiene un rango de iteraciones de $k = 1,000$ a $k = 10,000$ en el que la señal de salida intentó aproximarse a la posición deseada 90○, sin embargo, no logró acercarse a q_d, y por el contrario cuando $k = 10,000$ decreció la aproximación a q_d.

Finalmente las simulaciones realizadas para tareas repetitivas mostraron una excelente respuesta en la señal de salida, incluso con el robot manipulador de un grado de libertad y de 2gdl, ya que en estos casos se agregó la parte de control PD de retroalimentación, la cual asegura que el sistema llegue a estado estacionario y sea posible disminuir el sobreimpulso en la etapa transitoria de cada uno de los sistemas simulados. Así como, quedo establecida la importancia de este tipo de control híbrido con los resultados obtenidos cuando se coloca una perturbación durante las tareas repetidas, ya que, si no se tuviera el control de aprendizaje iterativo el sistema seguiría trabajando con la perturbación ingresada y en un caso aplicado podría ser un robot manipulador que realiza cortes o trabaja sobre un área de pintura, etc. este tipo de errores podrían retrasar la producción en masa que se tiene en una industria, lo que implicaría perdidas monetarias.

Participación en congreso

Se realizó una presentación en el Congreso ELECTRO 2021.

EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLOGICO NACIONAL DE MÉXICO

EL TECNOLÓGICO NACIONAL DE MÉXICO

A TRAVÉS DEL INSTITUTO TECNOLÓGICO DE CHIHUAHUA

OTORGA LA PRESENTE

CONSTANCIA

A

Viridiana Luna Zempoalteca

POR IMPARTIR LA CONFERENCIA

REGULACION DE ROBOTS MANIPULADORES

EN EL **CONGRESO INTERNACIONAL DE INGENIERÍA ELECTRÓNICA ELECTRO 2021**,
REALIZADO DEL 13 AL 15 DE OCTUBRE DEL PRESENTE AÑO.

43° CONGRESO INTERNACIONAL DE INGENIERIA ELECTRONICA

CHIHUAHUA, CHIH., A OCTUBRE DE 2021.

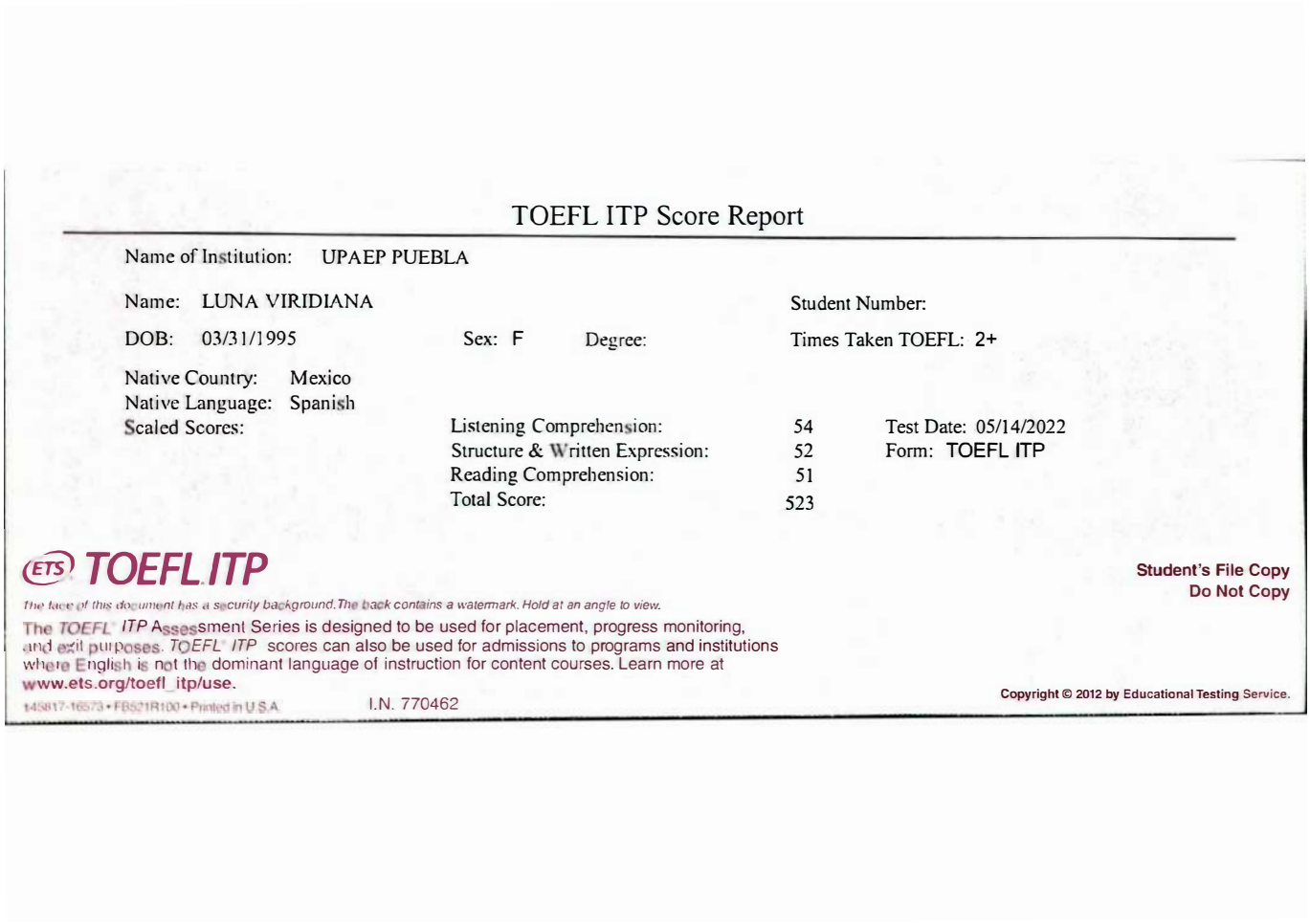
ROGELIO ENRIQUE BARAY ARANA
COORDINADOR GENERAL

INSTITUTO TECNOLÓGICO DE CHIHUAHUA

México 2021 Año de la Independencia

Certificación de idioma

Se realizó el examen de certificación TOEFL-ITP para la acreditación del idioma.



TOEFL ITP Score Report

Name of Institution: UPAEP PUEBLA

Name: LUNA VIRIDIANA

Student Number:

DOB: 03/31/1995 Sex: F Degree: Times Taken TOEFL: 2+

Native Country: Mexico
Native Language: Spanish
Scaled Scores:

Listening Comprehension:	54	Test Date: 05/14/2022
Structure & Written Expression:	52	Form: TOEFL ITP
Reading Comprehension:	51	
Total Score:	523	

TOEFL ITP

The face of this document has a security background. The back contains a watermark. Hold at an angle to view.

The TOEFL ITP Assessment Series is designed to be used for placement, progress monitoring, and exit purposes. TOEFL ITP scores can also be used for admissions to programs and institutions where English is not the dominant language of instruction for content courses. Learn more at www.ets.org/toefl_itp/use.

145817-16573 • FB521R100 • Printed in U.S.A. I.N. 770462

Student's File Copy
Do Not Copy

Copyright © 2012 by Educational Testing Service.

Bibliografía

Ahn, H.-S, K.L. Moore y YangQuan Chen (ene. de 2007). «Iterative learning control: Robustness and monotonic convergence for interval systems». En: págs. 1-228.

Arimoto, Suguru (1986). «Mathematical Theory of Learning with Applications to Robot Control». En: *Adaptive and Learning Systems*. USA: Springer, Boston, MA, págs. 379-388.

— (1990). «Learning control theory for robotic motion». En: *International Journal of Adaptive Conrtol and Signal Processing* 4, págs. 543-564.

Armstrong, B., O. Khatib y J. Burdick (1986). «The explicit dynamic model and inertial parameters of the PUMA 560 arm». En: *Proceedings. 1986 IEEE International Conference on Robotics and Automation*. Vol. 3, págs. 510-518.

Bien, Z. y K. M. Huh (1989). «Higher-order iterative control algorith». En: *IEEE Proc. Part D, Control Theory Appl.* 136, págs. 105-112.

Bolder J., Kleinendorst-S. y Oomen T. (2018). «Data-driven multivariabl ILC: enhanced performance by eliminating L and Q filters». En: *International Journal of Robust and Nonlinear Control* 28(12, págs. 3728-3751.

Cai, Zhonglun (abr. de 2009). «Iterative Learning Control: Algorithm Development and Experimental Benchmarking». En.

Chen, Y. Q. y K. L. Moore (2000). «Comments on United States patent 3,555,252-learning control of actuators in control systems». En: *2000 Int. Conf. Autom. Robot., Control, Singapur* N/A, N/A.

Christofer G. Atkeson, Chae H. An y John M. Hollerbach (1986). «Estimation of inertia parameters of manipulator loads and links». En: *The International Journal of Robotics Research* 5(3), págs. 101-119.

Duriez, Thomas, Steven L. Brunton y Bernd R. Noack (2017). *Machine Learning Control - Taming Nonlinear Dynamics and Turbulence*. 1st. Springer Publishing Company, Incorporated, pág. 229.

Edwards, J. B. (1974). «Stability problems inthe control of linear multipass processes». En: *Proc. Inst. Elect. Eng.* 121, págs. 1425-1431.

Fisac, J. F. y col. (2019). «A General Safety Framework for Learning-Based Control in Uncertain Robotic Systems». En: *IEEE Transactions on Automatic Control* 64.7, págs. 2737-2752.

Goodwin, Graham C. y Kwai Sang Sin (2009). *Adaptive Filtering Prediction and Control*. Dover Publications, Inc.

Jian-Xin Xu, Sanjib K. Panda y Tong H. Lee (2009). *Real-time iterative learning control*. 1rst. Springer-Verlag.

Kober, Jens (2019). «Robot Learning». En: *Encyclopedia of Systems and Control*. London: Springer London, págs. 1-9.

Kröse, Ben y col. (ene. de 1993). «An introduction to neural networks». En: *J Comput Sci* 48.

Ledezma, Fernando Díaz y Sami Haddadin (2017). «First-order principles- based constructive network topologies: An application to robot inverse dynamics». En: *IEEE-RAS International Conference on Humanoid Robotics* N/A, págs. 438-445.

M. Q. Phan, R. W. Longman y K. L. Moore (2000). «Unified formulation of linear iterative learning control». En: *AAS/AIAA Space Flight Mech* AAA, págs. 00-106.

Michael Lutter, Christian Ritter y Jan Peters (2019). «Deep Lagrangian Networks: Using Physics as Model Prior for Deep Learning». En: *ICLR 2019* 1, N/A.

Mohammad Jafari Giovanny Marquez, John Selberg Manping Jia etc. (2021). «Feedback Control of Bioelectronic Devices Using Machine Learning». En: *IEEE Control Systems Letters* 5, págs. 1133-1138.

Moore, K. L. (1993). «Iterative learning control for deterministic systems». En: *Advances Industrial Control, New York: Springer-Verlag* N/A, N/A.

Muhammad Arif, Tadashi Ishihara e Hikaru Inooka (1999). «Iterative Learning Control Utilizing the Error Prediction Method». En: *Journal of Intelligent and Robotic Systems* 25, págs. 95-108.

N. Amann, H. Owens y E. Rogers (1996). «Iterative learning control for discrete-time systems with exponential rate of convergence». En: *Proc. Inst. Elect. Eng.-Control Theory Appl.* 143, págs. 217-224.

Norouzi, Armin (2019). «Robotic manipulator control using PD-type Fuzzy Iterative Learning Control». En: *IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)* N/A, N/A.

Oomen, Tom (2018). «Learning in machines: towards intelligent mechatronic systems through Iiterative control». En: *Mikroniek* 6, págs. 5-11.

Reyes, Fernando (2011). *Robótica. Control de robots manipuladores*. México: Alfaomega Grupo Editor S. A. de C. V., págs. 253-266.

Reyes-Cortes, Fernando y Basil M. Al-Hadithi (2020). «An asinh-type regulator for robot manipulators with global asymptotic stability». En: *Automatika* 61.4, págs. 574-586.

Saab, Samer y col. (2021). «Iterative Learning Control: Practical Implementation and Automation». En: *IEEE Transactions on Industrial Electronics* PP, págs. 1-1.

Suárez-Nuño, Rubén, Emmanuel Cruz-Zavala y Emmanuel Nuño (2019). «A Strict Lyapunov Function for the Finite-Time Regulation of Robot Manipulators with Bounded Inputs». En: *2019 American Control Conference (ACC)*, págs. 1761-1766.

Tahirovic, Adnan y Gianantonio Magnani (ene. de 2013). *Passivity-Based Model Predictive Control for Mobile Vehicle Motion Planning*.

van der Schaft, Arjan (2017). *L2-Gain and Passivity Techniques in Nonlinear Control*. 3rd. Springer
Interna-tional Publishing.

Wang, Danwei, Yongqiang Ye y Bin Zhang (ene. de 2014). «Practical Iterative Learning Control with Frequency Domain Design and Sampled Data Implementation». En: DOI: 10.1007/978-981-4585-60-6.

Zhang, Yueqing, Bing Chu y Zhan Shu (2019). «A Preliminary Study on the Relationship Between Iterative Learning Control and Reinforcement Learning∗∗This project is sponsored by the China Scholarship Council (CSC).» En: *IFAC-PapersOnLine* 52.29. 13th IFAC Workshop on Adaptive and Learning Control Systems ALCOS 2019, págs. 314 -319.