



UNIVERSIDAD DE GUANAJUATO

CAMPUS IRAPUATO-SALAMANCA
DIVISIÓN DE INGENIERÍAS

*Aplicación de Algoritmos de Optimización
Metaheurísticos en el Diseño de Controladores en
Sistemas No Lineales*

TESIS PROFESIONAL

QUE PARA OBTENER EL GRADO DE:

*MAESTRO EN INGENIERÍA ELÉCTRICA
INSTRUMENTACIÓN Y SISTEMAS DIGITALES*

PRESENTA:

Ing. José Manuel Vidó Ramírez

DIRECTORES DE TESIS:

Dr. Juan Gabriel Aviña Cervantes

Dr. Arturo García Pérez

Salamanca, Gto., a 1 de noviembre del 2021.

**M. en I. HERIBERTO GUTIÉRREZ MARTIN
COORDINADOR DE ASUNTOS ESCOLARES
P R E S E N T E.-**

Por medio de la presente, se otorga autorización para proceder a los trámites de impresión, empastado de tesis y titulación al alumno(a) José Manuel Vidó Ramírez del **Programa de Maestría en Ingeniería Eléctrica** y cuyo número de **NUA** es: 387748 del cual soy director. El título de la tesis es: Aplicación de Algoritmos de Optimización Metaheurísticos en el Diseño de Controladores en Sistemas No Lineales

Hago constar que he revisado dicho trabajo y he tenido comunicación con los sinodales asignados para la revisión de la tesis, por lo que no hay impedimento alguno para fijar la fecha de examen de titulación.



Dr. Juan Gabriel Aviña Cervantes

NOMBRE Y FIRMA
**DIRECTOR DE TESIS
SECRETARIO**

ATENTAMENTE



Dr. Arturo García Pérez

NOMBRE Y FIRMA
DIRECTOR DE TESIS



Dr. Carlos Hugo García Capulín

NOMBRE Y FIRMA
PRESIDENTE



Dr. Juan Carlos Gómez Carranza

NOMBRE Y FIRMA
VOCAL

Agradecimientos

- A mis padres Marcela y José, a mis hermanos Eduardo, Fernando y Mayra.
- A Zuleyka Oros y Carlos Valencia.
- Al Dr. Juan Gabriel Aviña Cervantes por permitirme trabajar como su tesista.
- A mis compañeros Angel, Viri y LuisMi

Agradecimientos institucionales

Hago una mención especial a las instituciones que permitieron obtener una formación integral. En particular, este proyecto de investigación fue apoyado por:

- El Departamento de Ingeniería Electrónica de la Universidad de Guanajuato, NUA 387748



- El Consejo Nacional de Ciencia y Tecnología CONACyT (México), Beca no. 752021/1007617.



En este proyecto de investigación se propone la aplicación del algoritmo Particle Swarm Optimization (PSO) como método para realizar la sintonización de un Controlador Proporcional-Integral-Derivativo (PID) en un convertidor de tensión de Corriente Directa (CD-CD). Este algoritmo de optimización es un esquema poblacional que permite obtener soluciones óptimas bajo una función objetivo (costo), mientras se establecen las condiciones de operación adecuadas. Los parámetros del algoritmo PSO se sintonizan adecuadamente junto a la función de costo y la función de transferencia del sistema para obtener el valor óptimo de las ganancias K_p , K_i y K_d del controlador que permitan obtener el voltaje de salida con el menor rizo posible y demás especificaciones de diseño. El convertidor de voltaje (elevador/reductor) CD-CD en su concepción es intrínsecamente un sistema no lineal, el cual se linealiza a través del Jacobiano de las ecuaciones de estado. Dicho sistema es afectado por la acción de un control PID sintonizado por las metaheurísticas. Dichas técnicas permiten desarrollar controladores eficientes para modelos no lineales o lineales altamente complejos, convirtiéndose en una alternativa a las técnicas clásicas de sintonización de controladores, tales como el método del lugar geométrico de las raíces o el método de respuesta en frecuencia.

Abstract

This research aims to design a Proportional-Integral-Derivative (PID) Controller tuned by a Particle Swarm Optimization (PSO) algorithm and applied on a CD-CD Boost Converter. Like most metaheuristic methods, this optimization algorithm is a population-based scheme that obtains optimal solutions following a fitness function under appropriate operating conditions. The PSO operative parameters are tuned jointly with the system's fitness and transfer function to obtain the optimal value of the controller gains (K_p , K_i , and K_d), generating a specific ripple output voltage, among other algorithmic constraints. The CD-CD Boost-Converter is intrinsically a nonlinear system, linearized using the Jacobian function of the state space equations. Such a system is affected by the PID controller action, tuned by metaheuristics. This method helped develop efficient controllers for highly complex non-linear or linear restrictive models, becoming an alternative to classical control tuning methods such as root-locus or frequency response.

Índice General

Agradecimientos	I
Agradecimientos institucionales	II
Resumen	IV
1. Introducción	12
1.1. Motivación	12
1.2. Objetivos	13
1.2.1. Objetivo General	13
1.2.2. Objetivos Específicos	13
1.3. Antecedentes	13
1.4. Justificación	14
1.5. Organización de la tesis	15

2. Fundamentos Teóricos	17
2.1. Algoritmos Metaheurísticos	17
2.1.1. Algoritmos de Swarm Intelligence	19
2.1.2. Particle Swarm Optimization	20
2.1.3. Parámetros de ajuste del Algoritmo PSO	22
2.1.4. Función Objetivo	23
2.1.5. Secuencia de Funcionamiento del PSO	23
2.2. Sistemas dinámicos	24
2.2.1. Sistemas Lineales	25
2.2.2. Sistemas No Lineales	25
2.2.3. Linealización de Sistemas No Lineales	26
2.3. Control Automático	27
2.3.1. Sistemas de Control Automático	28
2.3.2. Clasificación de Controladores	29
2.4. Convertidores de Tensión CD-CD	32
2.4.1. Convertidor Elevador - Boost Converter	33
2.4.2. Cálculo de los Parámetros de un Convertidor de Tensión CD-CD	34
3. Metodología	40
3.1. Modelo del Convertidor CD-CD	40
3.1.1. Parámetros del Convertidor de Tensión CD-CD	42
3.1.2. Punto de Equilibrio	43
3.1.3. Linealización Aproximada	44

3.1.4. Simulaciones	47
3.2. Algoritmo PSO	50
3.3. Controlador PID-PSO	51
4. Resultados numéricos	53
4.1. Pruebas con Sistemas Lineales	53
4.2. Controlador de un Convertidor de Tensión CD-CD	59
5. Conclusiones	63

Índice de Figuras

2.1. Ejemplo del comportamiento de una partícula del PSO [Mirjalili et al., 2020].	21
2.2. Diagrama de bloques de un sistema de control industrial.	28
2.3. Diagrama a bloques de un Controlador On/Off.	29
2.4. Diagrama a bloques de un Controlador Proporcional.	30
2.5. Diagrama a bloques de un Controlador Integral	30
2.6. Diagrama a bloques de un Controlador Derivativo.	31
2.7. Diagrama a bloques de un Controlador Proporcional Integral.	31
2.8. Diagrama a bloques de un Controlador Proporcional Derivativo.	32
2.9. Diagrama a bloques de un Controlador Proporcional Integral Derivativo. . .	32
2.10. Circuito electrónico de un convertidor elevador de tensión de CD-CD.	33
2.11. Funcionamiento del circuito con S_1 cerrado y S_2 abierto.	33
2.12. Funcionamiento del circuito con S_1 abierto y S_2 cerrado.	34

2.13. Gráficas de carga y descarga de energía para la bobina i_L , y la tensión de salida del capacitor V_o	35
2.14. Ciclo de trabajo al 50 %.	35
2.15. Voltaje de salida del circuito y corriente en el capacitor.	38
3.1. Modo 1 de funcionamiento de un circuito elevador.	41
3.2. Modo 2 de funcionamiento de un circuito elevador.	41
3.3. Circuito resultante del convertidor de tensión CD-CD.	43
3.4. Retrato de Fase del convertidor de tensión CD-CD.	45
3.5. Simulación a bloques del modelo matemático no lineal del convertidor de tensión CD-CD.	48
3.6. Simulación a bloques del modelo matemático lineal del convertidor de tensión CD-CD.	49
3.7. Respuesta en lazo abierto del sistema no lineal y lineal.	49
3.8. Diagrama a bloques del controlador PID-PSO	51
4.1. Respuesta del sistema a una entrada escalón unitario del sistema (4.1). . . .	54
4.2. Comparación gráfica de la respuesta al escalón unitario del sistema (4.1) controlado.	55
4.3. Respuesta del sistema a una entrada escalón unitario del sistema (4.5). . . .	56
4.4. Comparación gráfica de la respuesta al escalón unitario del sistema (4.5) controlado.	57
4.5. Respuesta del sistema a una entrada escalón unitario del sistema (4.7)	58
4.6. Comparación gráfica de la respuesta al escalón unitario del sistema (4.7) controlado.	58

4.7. Voltaje de salida del convertidor de tensión CD-CD.	61
4.8. Voltaje de salida del convertidor de tensión CD-CD controlado por retroalimentación de estados.	62

Índice de Tablas

3.1. Parámetros del algoritmo PSO	50
4.1. Comparación del desempeño de los controladores implementados para el sistema (4.1).	55
4.2. Comparación del desempeño de los controladores implementados para el sistema (4.5)	56
4.3. Comparación del desempeño de los controladores implementados para el sistema (4.7).	59
4.4. Parámetros del sistema convertidor de tensión CD-CD.	60
4.5. Parámetros del Controlador PID para 30 repeticiones.	60

1.1. Motivación

La teoría de control conocida hasta ahora está respaldada por años de investigación que la ponen como la más aceptada. Sin embargo, con el papel que desempeña actualmente el control automático en el avance de la ciencia y la ingeniería, es necesario proponer nuevas estrategias que prioricen la optimización de recursos y mejoren el rendimiento de los controladores, en sistemas dinámicos lineales y no lineales.

El desarrollo de este trabajo de tesis se da por el interés de ir más allá de los métodos convencionales para la sintonización de controladores, proponiendo nuevas técnicas estocásticas que permitan encontrar soluciones óptimas que demuestren un mejor rendimiento en la implementación de sistemas de control en sistemas lineales y no lineales. Es muy importante recalcar que las técnicas de control para sistemas lineales han sido ampliamente estudiadas y generalmente arrojan resultados satisfactorios. Sin embargo, muchos de los sistemas son no lineales, discontinuos, con retardos o saturables, cuyo modelado y control es exageradamente complejo y en muchos de esos casos, la linealización es la única alternativa. Es por ello que algunas de las técnicas de optimización metaheurísticas brindan una alternativa confiable para sistemas complejos.

1.2. Objetivos

1.2.1. Objetivo General

Analizar y evaluar técnicas de control utilizando el modelo metaheurístico Particle Swarm Optimization, aplicado a sistemas de control lineales y no lineales.

1.2.2. Objetivos Específicos

- Sintonizar controladores para sistemas lineales y no lineales mediante la técnica metaheurística de Particle Swarm Optimization.
- Evaluar el desempeño del modelo metaheurístico en la búsqueda de soluciones estables de control, basados en el sobrepico, el tiempo de asentamiento y el error en estado estable del sistema compensado.
- Aplicar controladores sintonizados con algoritmos metaheurísticos a un sistema eléctricos y electrónicos, lineales y no lineales cuya complejidad resulten en un reto significativo para ser resueltas mediante las técnicas clásicas de control.

1.3. Antecedentes

La idea de aplicar los principios Darwinianos para la resolución de problemas se remonta a la década de 1940, comenzando con dos áreas separadas: programación evolutiva y algoritmos genéticos. Actualmente, el término cómputo evolutivo describe la unión de estas dos áreas y comprende varias sub-áreas: programación evolutiva, algoritmos genéticos, programación genética, etc.

En la actualidad, estos algoritmos son comúnmente aplicados en diversas áreas de investigación, en especial, en el CA de telemática de la Universidad de Guanajuato, se aplican en el análisis de imágenes biomédicas [Aguirre-Ramos et al., 2019] y control de sistemas eléctricos de potencia. Por ejemplo, en [Valencia-Rivera et al., 2020] propusieron tres estrategias de control diferentes para una microred desbalanceada bajo cargas lineales y no lineales. La primera estrategia utiliza un algoritmo genético para el diseño de un regulador lineal cuadrático híbrido y un controlador proporcional integral. La segunda utiliza otro algoritmo genético para un controlador proporcional integral con el método del lugar geométrico de las raíces. La tercera estrategia fue el diseño de un controlador proporcional integral por el método de colocación de polos. Los resultados de las

simulaciones mostraron que la segunda estrategia fue la mejor debido a su robustez y efectividad.

En [Tran and Nguyen, 2018] utilizaron un algoritmo genético para encontrar los parámetros óptimos de un controlador PID para un sistema de control de altitud de un vehículo aéreo no tripulado (por sus siglas en inglés UAV-Unmanned Aerial Vehicle). Así, compararon los parámetros obtenidos del algoritmo genético, con los parámetros del método de Ziegler-Nichols, encontrando que los parámetros óptimos calculados con el algoritmo genético mostraron una precisión más alta en el seguimiento de trayectorias predeterminadas, disminuyendo el tiempo de vuelo del vehículo.

En [Sulttan et al., 2020] se simuló un controlador Proporcional-Integral, sintonizado con un algoritmo genético para controlar la congestión de datos en una red de internet y con ello mejorar su rendimiento. El algoritmo propuesto arrojó buenos resultados, ya que logró disminuir el tiempo de asentamiento y el sobrepaso. Además hubo una mejoría en el manejo de datos en comparación con el algoritmo sintonizado con el método de Ziegler-Nichols.

En [Solihin et al., 2011] propusieron la sintonización de un controlador PID utilizando el algoritmo de Optimización por Enjambre de Partículas (PSO) para un motor de corriente directa. Además, compararon sus resultados con un controlador sintonizado con el método de Ziegler-Nichols, notando que el controlador PID diseñado con PSO disminuyó el sobrepico en la respuesta del sistema.

En [Ou and Lin, 2006], los autores utilizaron dos técnicas evolutivas para la sintonización de controladores PID aplicados a sistemas típicos de la industria de segundo y tercer orden. Encontraron que, para este caso particular, la técnica PSO tiene mejor desempeño que los algoritmos genéticos, obteniendo soluciones de mejor calidad al minimizar la función objetivo. Además, el controlador sintonizado con PSO mejoró el tiempo de asentamiento en comparación con el controlador sintonizado con el algoritmo genético.

1.4. Justificación

En los últimos años, los algoritmos metaheurísticos han sido utilizados con el objetivo de optimizar sistemas no lineales o extremadamente complejos. Su principio de funcionamiento corresponde desde sus orígenes a la teoría de evolución de Darwin. En esta teoría se trabaja con dos conjuntos: un conjunto de soluciones candidatas, llamadas fenotipos o población, y un conjunto de cromosomas que representan

el conjunto de variables del modelo. Cada solución candidata de la población es codificada en una cadena de cromosomas o genotipos [Sandou, 2013].

Debido a que estos algoritmos se basan en la selección natural, los cromosomas evolucionan en cada iteración (generación) para producir nuevos individuos (solución). Cada generación se obtiene aplicando un conjunto fundamental de operadores genéticos (inicialización, cruza, mutación y selección) hasta obtener una solución óptima. Dicha solución permite que los sistemas trabajen de la mejor manera, sin importar su complejidad o las limitaciones que se deban tomar en cuenta.

La teoría clásica de control utiliza modelos simplificados de sistemas y reformulaciones de sus especificaciones. En este contexto, las leyes de control son calculadas usando algoritmos determinísticos, utilizando técnicas clásicas que fueron desarrolladas desde que comenzó el estudio del control automático y que hasta hace unos años eran imprescindibles en el estudio de la estabilidad de los sistemas [Eiben et al., 2003].

Sin embargo, este enfoque falla cuando el modelo es demasiado complejo y no se puede simplificar lo suficiente. Por ejemplo, en algunos casos, el diseñador debe tomar en cuenta todo tipo de limitaciones físicas que pueda tener el sistema y el grado de dificultad que representa el modelo de éste. Además, es de vital importancia encontrar la solución óptima para que el sistema trabaje en las mejores condiciones de diseño.

En este trabajo se plantea una nueva tendencia en el área de control automático utilizando el análisis y la evaluación de técnicas de control mediante modelos metaheurísticos de optimización, aplicados a sistemas de control lineales y no lineales. En particular, se explota el algoritmo de optimización de enjambre de partículas, PSO.

1.5. Organización de la tesis

En el Capítulo 1, se revisó el estado del arte de trabajos de investigación que han utilizado algoritmos metaheurísticos como método de sintonización de sistemas de control en distintas aplicaciones. Además, se plantearon los objetivos, el general y los específicos del presente trabajo de investigación.

En el Capítulo 2 se hablará de las principales características de los algoritmos metaheurísticos, su clasificación y algunas de las ventajas de su implementación. Además, se definirán conceptos básicos de sistemas lineales y no

lineales, linealización y sistemas de control no lineal. Finalmente, se describirán las principales características del convertidor de tensión CD-CD, comunmente denominado boost-converter.

En el Capítulo 3, se obtendrán los modelos matemáticos lineal y no lineal del convertidor de tensión CD-CD, y se describirá el funcionamiento del sistema de control sintonizado con el algoritmo metaheurístico.

En el Capítulo 4, se presentarán algunas comparaciones de sintonización de controladores con técnicas convencionales y técnicas metaheurísticas, y su aplicación en sistemas lineales y no lineales.

Finalmente, en el Capítulo 5 se sintetizan las conclusiones finales del trabajo de investigación, así como se describen los trabajos futuros que se pueden derivar de este estudio y que permitirían agregar relevancia a los métodos utilizados.

En este capítulo, se introducen los conceptos teóricos, técnicas, métodos y configuraciones necesarios para abordar la problemática de este proyecto de investigación.

2.1. Algoritmos Metaheurísticos

En las últimas dos décadas, las técnicas metaheurísticas de optimización se han vuelto muy populares. Algunas de ellas como los algoritmos genéticos (GE – Genetic Algorithms) [[Holland et al., 1992](#)], optimización por colonia de hormigas (ACO – Ant Colony Optimization) [[Dorigo et al., 2006](#)] y optimización por enjambre de partículas (PSO – Particle Swarm Optimization) [[Kennedy and Eberhart, 1995](#)] son bien conocidas, no solo por especialistas en computación, sino también por científicos de diferentes campos de investigación por el gran número de trabajos que aplican estas técnicas metaheurísticas en diferentes áreas de estudio [[Mirjalili et al., 2014](#)].

Existen cuatro razones principales por las que la utilización de estos algoritmos se ha vuelto común:

- **Simplicidad:** Estos algoritmos están basados en conceptos muy simples, inspirados principalmente en fenómenos físicos, en el comportamiento de animales o en conceptos evolutivos.

Esta simplicidad permite a los investigadores simular diferentes conceptos naturales, proponer nuevos algoritmos metaheurísticos, o mejorar los ya existentes.

- Flexibilidad: Estos algoritmos tienen la ventaja de poder ser aplicados a distintos problemas sin necesidad de realizar modificaciones importantes en la estructura del algoritmo.
- La mayoría de los algoritmos metaheurísticos son mecanismos que optimizan soluciones de manera estocástica. El proceso de optimización de los algoritmos metaheurísticos comienza con una solución aleatoria, sin necesidad de calcular derivadas para encontrar la solución óptima.

Esto hace que los algoritmos metaheurísticos sean altamente convenientes para resolver problemas reales de alta complejidad (no lineales, conmutados, saturables, o con retardos), los cuáles son extremadamente complejos de modelar o pueden presentar diversos puntos de estabilidad.

- En comparación con las técnicas convencionales de optimización, los algoritmos metaheurísticos tienen mayor facilidad para evadir mínimos locales. Este fenómeno es debido a su naturaleza estocástica que le permite evitar estancamiento en soluciones locales y permite realizar una búsqueda extensiva en todo el espacio de búsqueda.

En el mundo de la ingeniería, existe una gran variedad de problemas para los que se busca una solución. La gran diversidad de algoritmos metaheurísticos ha permitido buscar la mejor opción para cada problema.

Este principio está respaldado por el **Teorema No Free Lunch**: *Este teorema prueba que no existe algoritmo metaheurístico más adecuado para resolver todos los problemas de optimización. En otras palabras, un algoritmo metaheurístico en particular puede mostrar resultados satisfactorios en un conjunto de problemas. Sin embargo, el mismo algoritmo puede mostrar soluciones poco eficientes en otro conjunto de problemas, [Wolpert and Macready, 1997].* Obviamente, el teorema NFL hace que el campo de estudio sea muy activo, ya que cada vez se proponen nuevos algoritmos metaheurísticos.

Generalmente, los algoritmos metaheurísticos se dividen, según el tipo de solución, en dos clases:

- Basados en una sola solución, donde el proceso de búsqueda comienza con una solución candidata que va mejorando a cada iteración.

- Basados en población de soluciones, donde el proceso comienza con un conjunto de soluciones potenciales. En este caso, el proceso de búsqueda comienza con una población inicial aleatoria (múltiples soluciones), y esta población es mejorada a cada iteración, siguiendo un modelo predeterminado y cumpliendo con la función objetivo.

Los algoritmos metaheurísticos basados en una población de soluciones tienen algunas ventajas respecto a los basados en una sola solución, por ejemplo:

- Las soluciones candidatas comparten información acerca del espacio de búsqueda, lo que produce saltos repentinos hacia la parte del espacio que promete la mejor solución.
- Las soluciones candidatas se asisten entre sí para evitar soluciones óptimas locales.
- Los algoritmos metaheurísticos basados en población generalmente tienen una mayor exploración comparado con los algoritmos basados en una sola solución potencial.

Una de las categorías más interesantes de los algoritmos metaheurísticos basados en población de soluciones es Swarm Intelligence (SI) [[Beni and Wang, 1993](#)]. En su mayoría, estas técnicas están basadas en el comportamiento natural de colonias, enjambres, rebaños, etc.

Sin importar las diferencias entre los algoritmos metaheurísticos, una característica común es la división del proceso de búsqueda en dos etapas: exploración y explotación. La etapa de exploración se refiere al proceso de investigar el o las áreas que parezcan prometedoras dentro del espacio de búsqueda tan ampliamente como sea posible. Por otro lado, la explotación se refiere a la búsqueda local alrededor de las regiones obtenidas en el proceso de exploración.

Encontrar un balance apropiado entre estas dos fases es una tarea desafiante debido a la naturaleza estocástica de los algoritmos metaheurísticos, este inconveniente es un poco más severo en los métodos de optimización basados en gradientes.

2.1.1. Algoritmos de Swarm Intelligence

Los algoritmos metaheurísticos se clasifican, según el fenómeno que emulan, en tres clases principales: Algoritmos Evolutivos, Algoritmos basados en la física

y Algoritmos de Swarm Intelligence. Esta última clase de algoritmos imitan el comportamiento social de enjambres o manadas de criaturas de la naturaleza. El mecanismo de trabajo es similar al de los algoritmos basados en eventos físicos, sin embargo, la búsqueda se realiza simulando la inteligencia social y colectiva de las criaturas. Los algoritmos más populares de Swarm Intelligence son el Ant Colony Optimization, Bat-Inspired Algorithm y Particle Swarm Optimization (PSO). Este último será detallado a continuación, ya que es el algoritmo implementado para el desarrollo de este proyecto.

2.1.2. Particle Swarm Optimization

La Optimización por Enjambre de Partículas [Mirjalili et al., 2020], es uno de los algoritmos mejor vistos en la literatura de los metaheurísticos. A pesar de ser un modelo matemático muy simple, ha sido ampliamente usado en diversos campos de estudio para resolver problemas de optimización.

Este algoritmo está inspirado en el comportamiento de las aves en la naturaleza. Cada partícula es considerada para ser solución de un problema de optimización dado. Se compone de dos vectores: posición y velocidad. El vector de posición incluye los valores de cada una de las variables en el problema.

Por ejemplo, si el problema tiene dos parámetros, las partículas tendrán un vector de posición de dos dimensiones. Cada partícula será capaz de moverse en un espacio de búsqueda de n dimensiones, donde n es el número de variables.

Para actualizar la posición de cada partícula, se considera entonces el vector de *velocidad*. Este vector define la magnitud y la dirección del paso para cada dimensión y cada partícula de manera independiente.

La localización de las partículas se actualiza en cada paso de optimización usando la siguiente ecuación:

$$\vec{X}_i(t+1) = \vec{X}_i(t) + \vec{V}_i(t+1) \quad (2.1)$$

Donde $\vec{X}_i(t)$ representa la posición de la partícula i en la iteración t y $V_i(t+1)$ representa la velocidad de la partícula i en la iteración $t+1$. Esta ecuación muestra que la actualización de la posición es simple y que el componente principal es el vector de velocidad. Este vector se define como:

$$\vec{V}_i(t+1) = w\vec{V}_i(t) + c_1r_1(\vec{P}_i(t) - \vec{x}_i(t)) + c_2r_2(\vec{G}(t) - \vec{x}_i(t)) \quad (2.2)$$

Donde $\vec{X}_i(t)$ y $\vec{V}_i(t)$ representan la posición y velocidad, respectivamente, de la partícula i en la iteración t , w es el peso inercial, c_1 es el coeficiente individual, c_2 es el coeficiente social, r_1, r_2 son números aleatorios entre $[0, 1]$, $\vec{P}_i(t)$ es la mejor solución obtenida por la partícula i hasta la iteración t , y $\vec{G}(t)$ es la mejor solución obtenida por todas las partículas, es decir, el enjambre completo, hasta la iteración t .

La Ec. (2.2) muestra que el vector de velocidad está compuesto de tres partes. El primer componente, $w\vec{V}_i(t)$ mantiene la tendencia hacia la velocidad actual. Esta componente es multiplicada por un parámetro de inercia w . Mientras más grande sea este parámetro, mayor será la tendencia a mantener la velocidad previa.

El segundo componente, $c_1r_1(\vec{P}_i(t) - \vec{x}_i(t))$, simula la inteligencia individual de un ave memorizando y utilizando la mejor solución obtenida al momento por cada una de las partículas. El vector $\vec{P}_i(t)$ se actualiza en cada iteración en caso de que la partícula i encuentre una mejor solución. El impacto de este componente sobre el valor final de la velocidad puede incrementar o disminuir cambiando el valor de c_1 . Este parámetro es multiplicado por un valor aleatorio uniformemente distribuido entre 0 y 1 para dar un comportamiento aleatorio, ya que el algoritmo PSO es una técnica de optimización estocástica. En general, el segundo componente mantiene la tendencia hacia la mejor solución encontrada hasta ese instante, llamada "personal best".

El tercer componente, $c_2r_2(\vec{G}(t) - \vec{x}_i(t))$ imita la inteligencia social de una parvada de aves, en el cual la mejor solución obtenida por todas las partículas, llamada global best, es almacenada en $\vec{G}(t)$ y usada en este componente. Esto quiere decir que considerar la mejor solución encontrada hasta el momento por el enjambre gravita todas las partículas hacia un punto. El impacto de este componente puede afinarse cambiando el valor de c_2 .

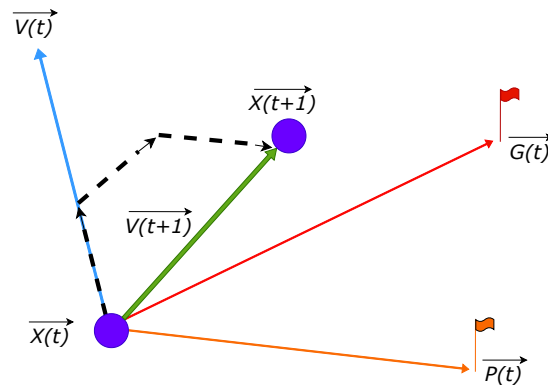


Figura 2.1: Ejemplo del comportamiento de una partícula del PSO [Mirjalili et al., 2020].

Con estos tres componentes, la siguiente posición de la partícula podrá ser definida. El ejemplo mostrado en la Figura 2.1 representa el comportamiento de las partículas, donde cada una de ellas considera la velocidad previa, el personal best y el global best para definir la velocidad actual y actualizar su posición. La siguiente posición depende en gran parte de los números aleatorios generados en r_1 y r_2 .

2.1.3. Parámetros de ajuste del Algoritmo PSO

La Ec. (2.2) muestra que el funcionamiento iterativo del PSO requiere la inicialización de algunos parámetros. El primero de ellos es el parámetro de inercia w , que mantiene la velocidad y dirección actual de la partícula. Enseguida, están los parámetros de aceleración c_1 y c_2 que, junto con los vectores aleatorios r_1 y r_2 , controlan la influencia estocástica de los componentes cognitivo y social en la velocidad de la partícula. Las constantes c_1 y c_2 también son parámetros de confianza, donde c_1 expresa cuánta confianza tiene una partícula en sí misma, mientras que c_2 expresa cuánta confianza tiene la partícula en sus partículas vecinas.

Existen diferentes combinaciones en los valores de c_1 y c_2 que dan como resultado un comportamiento diferente de las partículas.

- $c_1 = c_2 = 0$, las partículas se mantienen volando a su velocidad actual hasta topor con un vecindario en el espacio de búsqueda.
- $c_1 > 0$ y $c_2 = 0$, cada partícula realiza una búsqueda local y encuentra la mejor posición en su vecindario reemplazando la mejor posición actual por la nueva posición si ésta es mejor.
- $c_1 = 0$ y $c_2 > 0$, todo el enjambre de partículas es atraído a un solo punto.
- $c_1 = c_2$, es la configuración más utilizada, ya que las partículas son atraídas hacia el *personal best* de cada partícula.

Otro parámetro a tener en cuenta es el número de partículas del enjambre. Mientras más partículas haya en el enjambre, mayor espacio de búsqueda será cubierto en cada iteración, por lo tanto, el número de iteraciones disminuye. Sin embargo, esto produce un incremento en la complejidad computacional por iteración. Algunos estudios han probado que el algoritmo PSO tiene la habilidad de encontrar soluciones óptimas con enjambres pequeños de 10 hasta 30 partículas [Engelbrecht, 2007], por lo que el tamaño de este parámetro dependerá del problema en estudio.

El tamaño del vecindario es el parámetro que define el grado de interacción social en el enjambre. Entre más pequeño sea el tamaño del vecindario, menor es la interacción entre las partículas y la convergencia es más lenta. Para conocer el tamaño ideal del vecindario, es necesario experimentar con aumentos de tamaño para asegurar una alta diversidad inicial con una convergencia más rápida, a medida que las partículas se mueven en el espacio de búsqueda hacia un área prometedora.

El número de iteraciones es un parámetro dependiente del problema. Pocas iteraciones pueden detener la búsqueda prematuramente, mientras que un número grande de iteraciones tiene como consecuencia agregar complejidad computacional innecesaria.

2.1.4. Función Objetivo

La función objetivo, de costo o fitness, es la expresión matemática que compara los valores de referencia para un comportamiento deseado del sistema con los valores actuales buscando el mejor desempeño. En este punto, se utiliza alguna métrica donde normalmente se escoge la función de error cuadrático medio.

2.1.5. Secuencia de Funcionamiento del PSO

El proceso llevado a cabo en la implementación del algoritmo PSO comienza con:

- (1) La inicialización de las partículas con posición y velocidad aleatorias dentro del espacio de búsqueda. Posteriormente,
- (2) Se actualizan la posición y velocidad de acuerdo con las Ecs (2.1) y (2.2), y
- (3) Se evalúan estas posibles soluciones en la función fitness para determinar su rendimiento.
- (4) Si dicho rendimiento supera el de la mejor solución global o personal encontradas hasta ese momento, serán sustituidas por la solución actual.
- (5) Si el criterio de rendimiento es lo suficientemente bueno o se llega al número máximo de iteraciones, el algoritmo termina su proceso. Si no, regresa al paso (2), [Li et al., 2014]. Dicho proceso está descrito en el Algoritmo 1.

Algoritmo 1 Particle Swarm Optimization (PSO)

```
Inicialización  $N$ ,  $MaxIter$ ,  $c_1$ ,  $c_2$ ,  $w$ 
 $i = 0$ 
while  $i \geq MaxIter$  do
    Calcular nueva posición  $X(i)$  y velocidad  $V(i)$ 
    Evaluar solución en la función objetivo
    if  $f(Particula(i)) < f(Gbest)$  then
         $Gbest = Particula(i)$ 
    else if  $f(Particula(i)) < f(Pbest)$  then
         $Pbest = Particula(i)$ 
    end if
     $i = i + 1$ 
end while
```

2.2. Sistemas dinámicos

Un sistema es un conjunto de elementos que funcionan de manera individual para lograr un objetivo determinado. Este concepto no aplica solamente a sistemas físicos, sino también biológicos, fenómenos abstractos y dinámicos, como la economía.

Dentro del campo de la ingeniería de control, se estudian sistemas de diferente naturaleza; los más comunes son sistemas físicos cuya dinámica es modelada a través de ecuaciones que permitan el uso de herramientas matemáticas necesarias para su análisis.

Un sistema puede ser representado como,

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)), & x(t_0) &= x_0 \\ y(t) &= h(x(t), u(t)) \end{aligned} \tag{2.3}$$

Este modelo es conocido comúnmente como Modelo en Espacio de Estados. El análisis en espacio de estados centra la atención en tres tipos de variables: variables de entrada, variables de estado y variables de salida. Las variables de entrada son agrupadas en el vector $u \in \mathbb{R}^m$, las variables de estado son agrupadas en el vector $x \in \mathbb{R}^m$ y las variables de salida son agrupadas en el vector $y \in \mathbb{R}^m$.

2.2.1. Sistemas Lineales

Un sistema se denomina lineal si las funciones f y h de la Ec. (2.3) son lineales con respecto a x y a la entrada u . Un modelo en espacio de estados lineal puede ser representado de forma generalizada por las ecuaciones dinámicas variantes en tiempo:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}\tag{2.4}$$

donde A, B, C y D son matrices constantes. Este sistema es llamado Sistema Lineal Invariante en el tiempo. Cuando las matrices A, B, C y D son dependientes del tiempo, se habla de un Sistema Lineal Variante en el Tiempo.

Es bien sabido que los sistemas Dinámicos Lineales respetan el principio de superposición. Este principio establece que la respuesta producida por la aplicación simultánea de dos funciones de entradas diferentes es la suma de las dos respuestas individuales. Por tanto, para el sistema lineal, la respuesta a varias entradas se calcula tratando una entrada a la vez y sumando los resultados.

Los sistemas dinámicos lineales respetan el principio de superposición. Dicho principio establece que el resultado de la aplicación de dos funciones de entrada diferentes a un sistema es igual a la suma de las respuestas individuales. Con este principio, la respuesta de un sistema con varias entradas se realiza calculando las respuestas individuales y sumándolas.

2.2.2. Sistemas No Lineales

Se dice que un sistema es no lineal cuando no cumple con el principio de superposición. La mayoría de las relaciones físicas se representan por ecuaciones lineales. Aunque la mayoría de los casos solo lo son en algunos rangos de operación limitados. En ocasiones, las salidas pueden saturarse cuando son producidas por señales de entrada grandes, o puede haber un rango pequeño de variaciones de entrada a las cuales el componente es insensible [Ogata, 2003].

Un sistema no lineal se representa como:

$$\begin{aligned}\dot{x} &= f(x, u, t) \\ y &= h(x, u, t)\end{aligned}\tag{2.5}$$

donde x representa el vector de estados del sistema, $u(t)$ es el vector de entradas, $y(t)$ es el vector de salida, t es el tiempo, y f y h son un conjunto de funciones escalares.

La relación $\frac{d}{dt}x = \dot{x} = 0$ representa los puntos de operación del sistema (bien conocidos como puntos de equilibrio) donde se debe evaluar la estabilidad de este. Una de las herramientas gráficas útiles para conocer el comportamiento de un sistema no lineal es el retrato de fase, el cual está constituido por el conjunto de trayectorias del sistema para diferentes condiciones iniciales.

Dicho retrato de fase es útil para conocer la estabilidad del sistema en un punto de operación específico. Por ejemplo, un estado de equilibrio \bar{x} es asintóticamente estable para el sistema si las trayectorias en el espacio de fase convergen hacia el punto de equilibrio, y es inestable si las trayectorias que inician desde puntos cercanos al punto de equilibrio, divergen de este.

2.2.3. Linealización de Sistemas No Lineales

Todo sistema no lineal puede ser representado por su aproximación lineal. A este procedimiento se le denomina *linealización*.

La técnica de linealización aproximada es una representación que se obtiene de los términos de primer orden de la expansión en serie de Taylor de la no-linealidad que caracteriza al sistema como tal. Esta técnica es válida siempre y cuando las perturbaciones que afectan al sistema operando en el punto de equilibrio sean pequeñas.

Este método nos lleva a la representación de la Ec. (2.4), donde las matrices constantes, A, B y C, son llamadas Matrices Jacobianas, las cuales están representadas y evaluadas de la siguiente forma:

$$A = \left. \frac{\partial f}{\partial x} \right|_{(x,u)}; \quad B = \left. \frac{\partial f}{\partial u} \right|_{(x,u)}; \quad C = \left. \frac{\partial h}{\partial x} \right|_{(x,u)}; \quad (2.6)$$

Por esta razón, la linealización aproximada también es conocida como *Linealización Jacobiana* [Vaquero López and Milian del Valle, 2016].

2.3. Control Automático

En la actualidad, el control automático desempeña un papel muy importante en el avance de la ingeniería y la ciencia, tomando en cuenta que se ha convertido en una parte vital e integral en los sistemas de vehículos espaciales, sistemas robóticos, en procesos modernos de fabricación y en cualquier operación industrial que requiera el control de volumen, voltaje, etc.

Por lo cual, es muy importante estar familiarizado con la terminología y poder integrar adecuadamente las técnicas de ajuste de controladores mediante algoritmos metaheurísticos.

A continuación, se definen algunos conceptos de la teoría del control:

- Señal de error. Un controlador automático compara el valor a la salida de una planta con el valor de entrada del sistema. La diferencia entre ambos valores produce una señal de control proporcional que efectuará una acción de control en el sistema con el objetivo de disminuir esa diferencia hasta cero.
- Variable controlada y señal de control. La variable controlada es aquello que será controlado. La señal de control es la señal de salida del controlador que influye en el valor de la variable controlada. En otras palabras, controlar significa corregir la desviación del valor medido respecto al valor deseado.
- Planta. Una planta es todo aquel conjunto de elementos que funcionan juntos y cuyo objetivo es efectuar una operación en particular. También se define como cualquier objeto físico que se va a controlar, como un dispositivo mecánico, eléctrico, químico, etc.
- Proceso. Se define como proceso a toda operación o desarrollo natural continuo, marcado por una serie de cambios que se suceden unos a otros de forma relativamente fija y que llevan a un resultado o propósito determinado; una operación artificial o voluntaria que se hace de forma progresiva y consta de una serie de acciones controlados, sistemáticamente dirigidos hacia un propósito determinado.
- Perturbaciones. es una afectación negativa que influye en el valor de salida de un sistema. Cuando la perturbación se encuentra dentro del sistema, se habla de una perturbación interna. Cuando se encuentra fuera del sistema, se le llama externa, y es considerada una entrada.

- Sistema de control en lazo cerrado. Es un tipo de sistema que guarda una relación entre la señal de salida y la señal de entrada, comparándolas para utilizar la diferencia como medio de control. El controlador es alimentado con esta señal de error con el objetivo de corregir la salida del sistema hasta llegar a un valor de referencia deseado.
- Sistemas de control en lazo abierto. Son aquellos sistemas en donde no se guarda relación entre el valor de salida y la entrada del sistema.
- Diagramas a bloques. Un sistema de control está descrito por medio de una representación denominada diagrama de bloques. El diagrama de bloques de un sistema es una representación gráfica de cada uno de los componentes y las funciones que estos llevan a cabo. En estos diagramas se muestran el flujo de las señales involucradas en el funcionamiento del sistema. Algunas de las ventajas del uso de diagrama de bloques en la representación de sistemas residen en la facilidad para formar dicho diagrama, conectando los componentes y siguiendo el flujo de las señales.

2.3.1. Sistemas de Control Automático

La Figura 2.2 muestra un diagrama de bloques de un sistema de control automático que consiste en un controlador, un actuador, una planta y un sensor (elemento de medición).

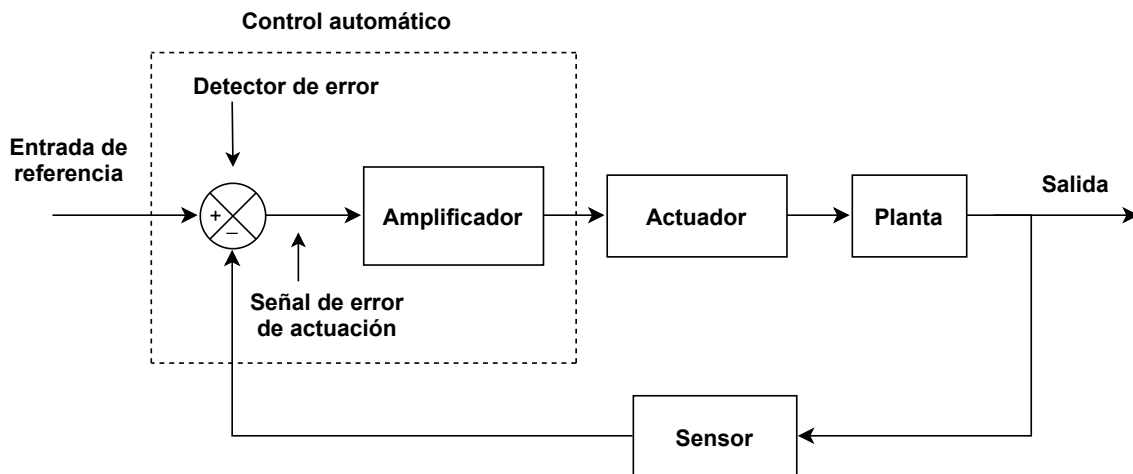


Figura 2.2: Diagrama de bloques de un sistema de control industrial.

El controlador toma la señal de error, generalmente de baja potencia, amplificándola para que sea entrada del actuador, que produce una señal de entrada

a la planta del sistema, a fin de que ésta produzca una señal de salida cercana a la señal de referencia.

El sensor es el dispositivo de conversión de señal que permite que la variable de salida sea transformada en otra variable que pueda usarse para comparar la salida con la señal de referencia. Este elemento se encuentra en el lazo de retroalimentación del sistema..

2.3.2. Clasificación de Controladores

En el área del control automático, la clasificación de los controladores se da de acuerdo a su acción de control, tales como:

1. De dos posiciones o controladores on-off
2. Controladores proporcionales
3. Controladores integrales
4. Controladores proporcionales-integrales
5. Controladores proporcionales-derivativos
6. Controladores proporcionales-integrales-derivativos

El tipo de controlador que se utiliza debe estar basado en la naturaleza de la planta y las condiciones de operación, que incluyen disponibilidad, costo, precisión, etc.

Control de dos posiciones o de encendido y apagado (on/off). En este sistema de control solo se tienen dos posiciones que, en la mayoría de los casos, es encendido y apagado. Este controlador es el más utilizado en sistemas de control industrial y domésticos debido a su simplicidad y bajo costo. En la Figura 2.3 se muestra el diagrama a bloques de este controlador.

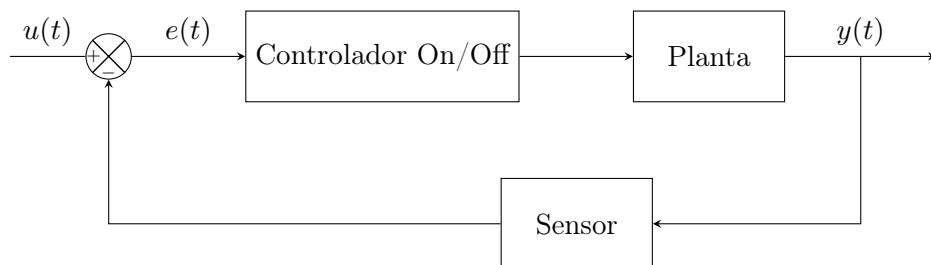


Figura 2.3: Diagrama a bloques de un Controlador On/Off.

Control proporcional. Para un controlador proporcional, la relación entre la salida $u(t)$ y la señal de error $e(t)$ es:

$$x(t) = K_p e(t) \quad (2.7)$$

donde K_p es la ganancia proporcional. En la Figura 2.4 se muestra el diagrama a bloques del Controlador P. En esencia, este controlador es un amplificador con ganancia ajustable.

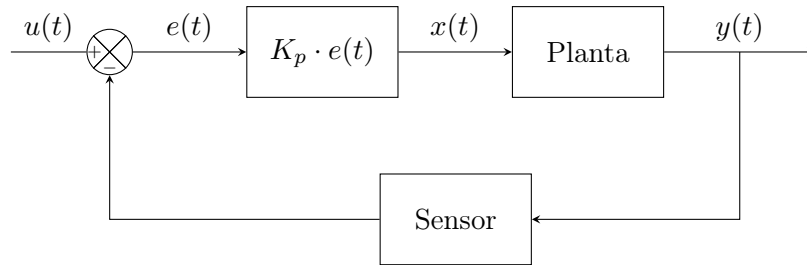


Figura 2.4: Diagrama a bloques de un Controlador Proporcional.

Control integral. En este tipo de controlador, el valor en su salida $x(t)$ se cambia a una razón proporcional a la señal de error $e(t)$. Es decir,

$$x(t) = K_i \frac{d}{dt} e(t), \quad (2.8)$$

donde K_i es una constante ajustable. Otra representación del controlador integral mostrado en la Figura 2.5 es la siguiente:

$$x(t) = K_i \int_{-\infty}^t e(t) dt, \quad (2.9)$$

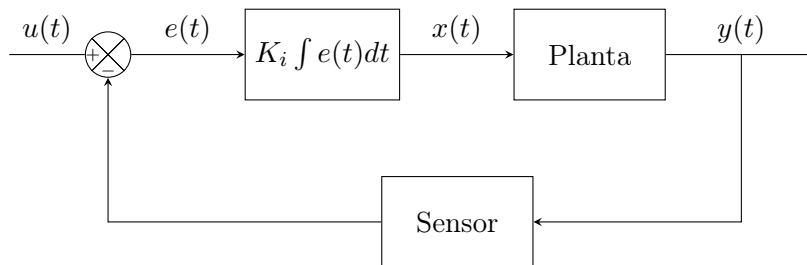


Figura 2.5: Diagrama a bloques de un Controlador Integral

Control derivativo. En este tipo de controlador, mostrado en la Figura 2.6

anticipa comportamiento futuro que va a tener la señal de error $e(t)$. Es decir,

$$x(t) = K_d \frac{de(t)}{dt} \quad (2.10)$$

donde K_d es el tiempo derivativo.

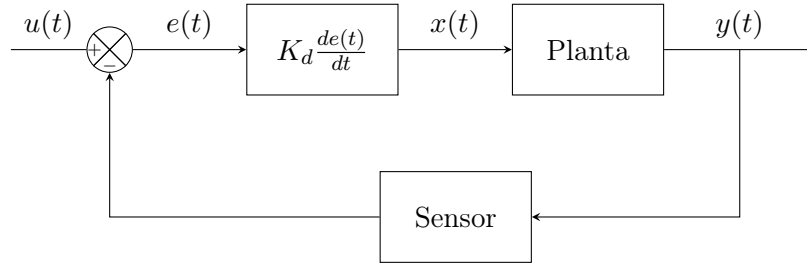


Figura 2.6: Diagrama a bloques de un Controlador Derivativo.

Control proporcional-integral (PI) En la Figura 2.7 se observa la acción de un controlador proporcional-integral, definida mediante:

$$x(t) = K_p e(t) + \frac{K_p}{T_i} \int_{-\infty}^t e(t) dt \quad (2.11)$$

o la función de transferencia del controlador es:

$$\frac{X(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} \right) \quad (2.12)$$

donde T_i se denomina el *tiempo integral*.

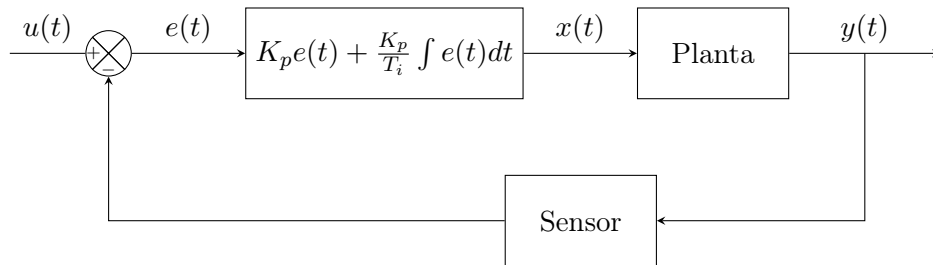


Figura 2.7: Diagrama a bloques de un Controlador Proporcional Integral.

Control proporcional-derivativo (PD) La acción de un controlador proporcional-derivativo se define mediante:

$$x(t) = K_p e(t) + K_p T_d \frac{d}{dt} e(t) \quad (2.13)$$

y su función de transferencia es:

$$H(s) = \frac{X(s)}{E(s)} = K_p(1 + T_d s) \quad (2.14)$$

donde T_d es el *tiempo derivativo*. En la Figura 2.8 se muestra el diagrama a bloques de este controlador.

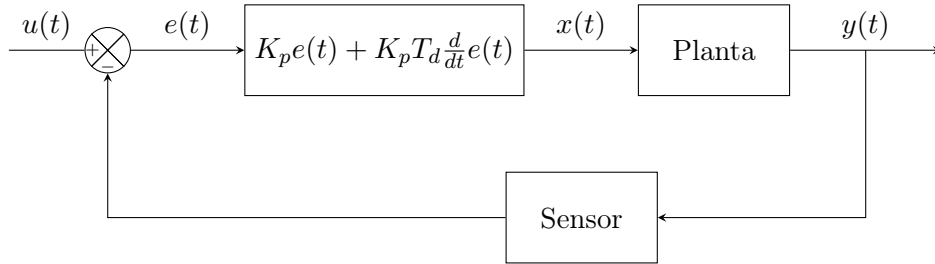


Figura 2.8: Diagrama a bloques de un Controlador Proporcional Derivativo.

Control proporcional-integral-derivativo La combinación del control proporcional, control integral y control derivativo, control PID se muestra en la Figura 2.9. Su ecuación está dada por:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_{-\infty}^t e(t) dt + K_p T_d \frac{d}{dt} e(t) \quad (2.15)$$

donde k_p es la ganancia proporcional, T_i es el tiempo integral y T_d es el tiempo derivativo.

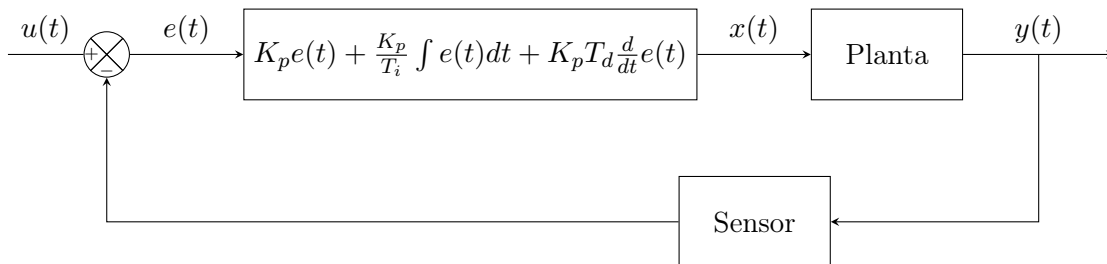


Figura 2.9: Diagrama a bloques de un Controlador Proporcional Integral Derivativo.

2.4. Convertidores de Tensión CD-CD

Los convertidores de tensión CD-CD convierten de forma directa un voltaje de corriente directa a otro voltaje de corriente directa mayor o menor.

2.4.1. Convertidor Elevador - Boost Converter

La Figura 2.10 muestra el circuito convertidor de tensión CD-CD utilizado en este estudio. El circuito-elevador consta de una fuente de voltaje E , un inductor L ,

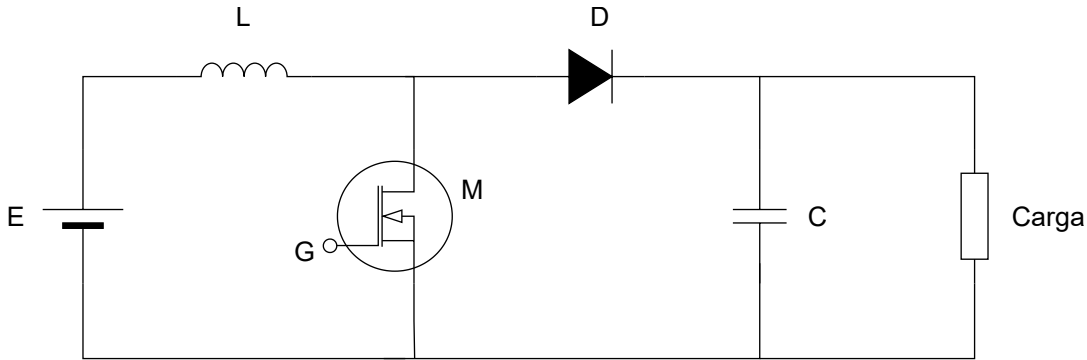


Figura 2.10: Circuito electrónico de un convertidor elevador de tensión de CD-CD.

que a su vez está conectado a un esquema de conmutación por medio de transistor MOSFET M de potencia, la frecuencia de conmutación es controlada por una señal de modulación por ancho de pulso (PWM) G .

La salida es recuperada por un circuito de salida, constituido por un capacitor C y una resistencia de carga R . La Figura 2.11 muestra el principio de funcionamiento del convertidor de tensión CD-CD. Se observa que inicialmente, se coloca en serie el

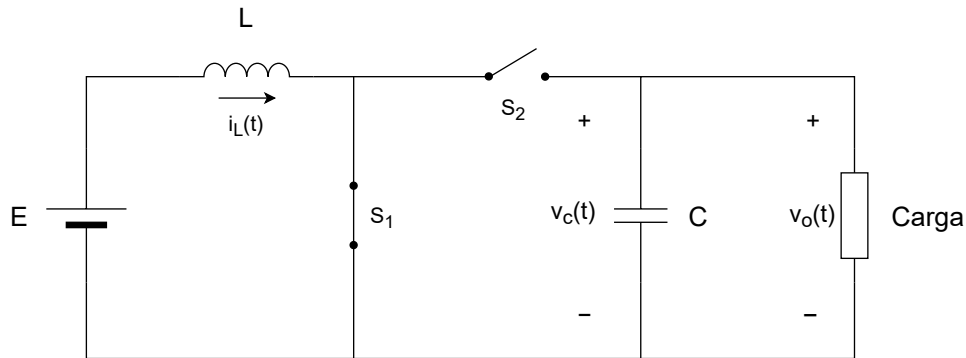


Figura 2.11: Funcionamiento del circuito con S_1 cerrado y S_2 abierto.

inductor L a la fuente de alimentación E , cerrando el circuito mediante el interruptor S_1 y manteniendo el interruptor S_2 abierto.

Cuando la corriente aumenta, el inductor comienza a almacenar energía, en un rango de tiempo corto, sin pasar al estado de saturación que produciría una demanda excesiva de corriente a la fuente de entrada. A su vez, el capacitor comienza su proceso de descarga de energía a través de la resistencia de carga R .

Posteriormente, en la Figura 2.12 los interruptores cambian de estado, se abre el interruptor S_1 y se cierra simultáneamente el interruptor S_2 . La energía almacenada

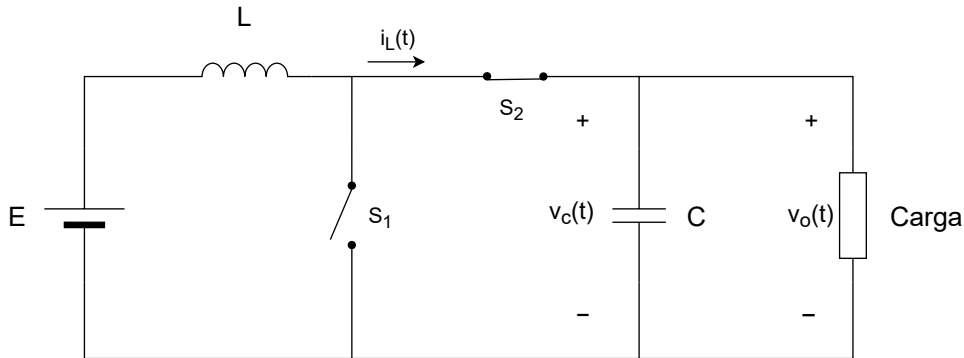


Figura 2.12: Funcionamiento del circuito con S_1 abierto y S_2 cerrado.

por el inductor es transferida junto a una fracción de la energía de la fuente de entrada en forma paulatina al circuito de salida.

El voltaje en la carga dependerá del valor inicial del capacitor, ya que la energía entregada por el inductor estará ahora almacenada en el capacitor. Cuando se repite el proceso de conmutación por varios ciclos, el voltaje de salida se incrementará de manera gradual hasta alcanzar un balance entre la corriente promedio del inductor y la corriente en la carga.

La Figura 2.13 muestra el comportamiento de carga y descarga de energía, tanto de la corriente de la bobina como del voltaje del capacitor a la salida.

2.4.2. Cálculo de los Parámetros de un Convertidor de Tensión CD-CD

El comportamiento esperado del convertidor de tensión CD-CD dependerá en su totalidad de los parámetros y valores de los componentes del circuito. Algunos de ellos deben ser especificados (fuente de alimentación) y otros calculados (ciclo de trabajo, inductancia y capacitancia) a partir de los parámetros especificados.

El ciclo de trabajo es una señal cuadrada que representa el tiempo que el interruptor S_1 estará encendido con respecto al periodo total de la señal, como se muestra en la Figura 2.14.

El voltaje a la salida del circuito está expresado por la siguiente ecuación,

$$V_C = \frac{E}{1 - D} \quad (2.16)$$

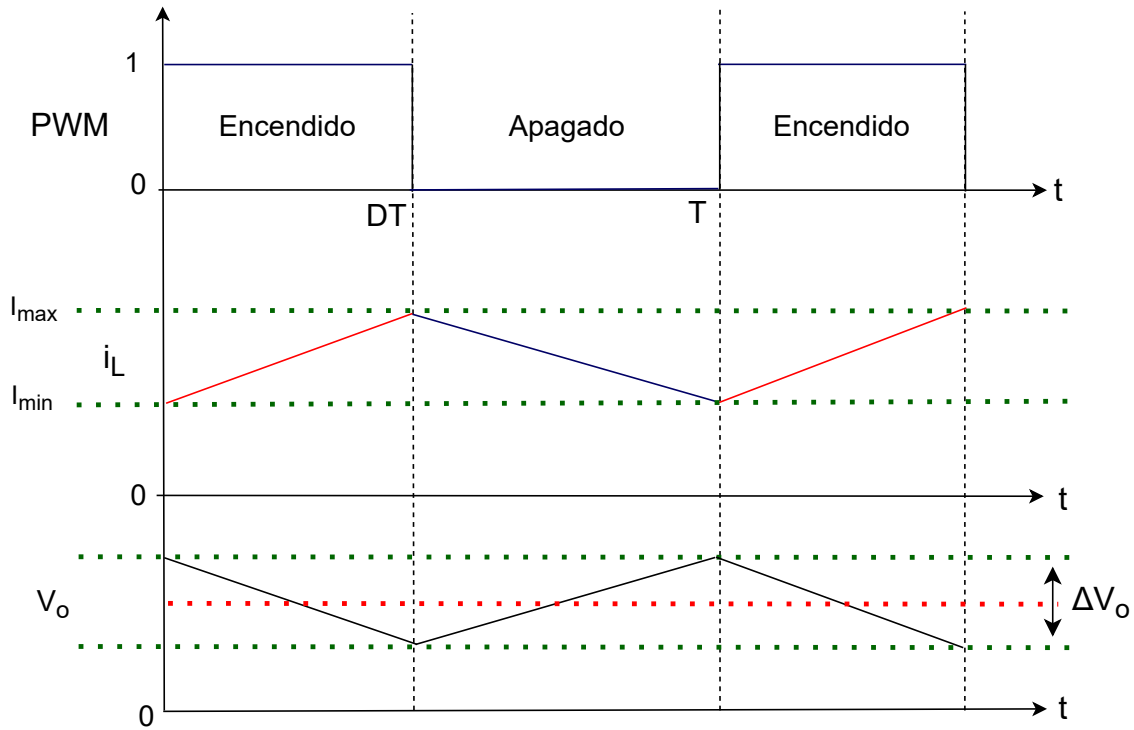


Figura 2.13: Gráficas de carga y descarga de energía para la bobina i_L , y la tensión de salida del capacitor V_o .

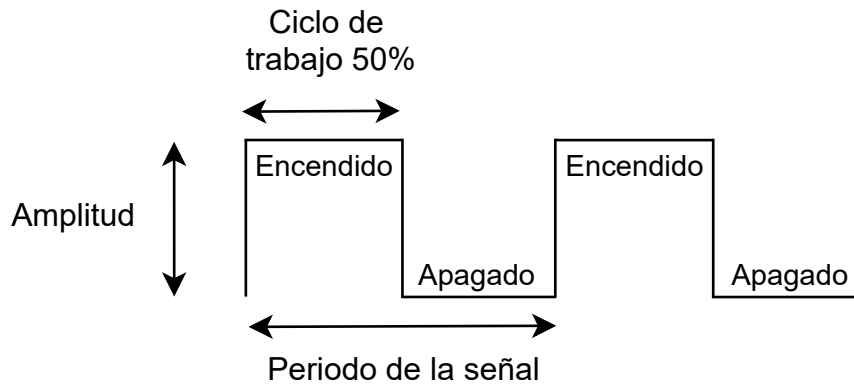


Figura 2.14: Ciclo de trabajo al 50 %.

donde V_C es el voltaje del capacitor y es el mismo que a la salida, E es el voltaje de la fuente de alimentación y D es el ciclo de trabajo.

Para conocer el valor del ciclo de trabajo, basta con especificar los valores de voltaje de la fuente de alimentación y de salida del circuito. Despejando el ciclo de trabajo de la Ec. (2.16), tenemos:

$$D = 1 - \frac{E}{V_C} \quad (2.17)$$

Conocido el valor D , se debe ahora encontrar el valor de inductancia del circuito. Para esto, se analiza la potencia en la entrada y salida del circuito de la Figura 2.10. La potencia de entrada P_i está dada por:

$$P_i = E * i_L \quad (2.18)$$

Mientras que la potencia de salida se calcula como:

$$P_o = \frac{V_c^2}{R} \quad (2.19)$$

Igualando las Ecs. (2.18) y (2.19), se tiene:

$$E * i_L = \frac{V_c^2}{R} \quad (2.20)$$

Sustituyendo el valor de V_C de la Ec. (2.16),

$$E * i_L = \frac{\left(\frac{E}{1-D}\right)^2}{R} \quad (2.21)$$

y despejando el valor de i_L , se tiene como resultado:

$$i_L = \frac{E}{(1-D)^2 R} \quad (2.22)$$

Ahora bien, analizando la malla del inductor, en el circuito de la Figura 2.11, aplicando la LVK, se tiene que la corriente que atraviesa el inductor está dada por:

$$\frac{di_L(t)}{dt} = \frac{E}{L} \quad (2.23)$$

En la Figura 2.13, se observa que la variación de corriente en el inductor es constante y positiva cuando el interruptor S_1 está cerrado, pudiendo entonces re-escribir la Ec. (2.23) como:

$$\frac{\Delta i_L}{\Delta t} = \frac{\Delta i_L}{DT} = \frac{E}{L} \quad (2.24)$$

Despejando la variación de corriente en el inductor, Δi_L , se tiene:

$$\Delta i_L = \frac{EDT}{L} \quad (2.25)$$

La corriente máxima y mínima en el inductor, I_{max} e I_{min} se calculan como:

$$I_{max} = i_L + \frac{\Delta i_L}{2} \quad ; \quad I_{min} = i_L - \frac{\Delta i_L}{2} \quad (2.26)$$

Sustituyendo las Ecs. (2.22) y (2.25) en la Ec. (2.26), se tiene:

$$I_{max} = \frac{E}{(1-D)^2 R} + \frac{EDT}{2L} \quad ; \quad I_{min} = \frac{E}{(1-D)^2 R} - \frac{EDT}{2L} \quad (2.27)$$

Para encontrar el valor de inductancia mínimo L_{min} para el funcionamiento del convertidor de tensión CD-CD, es necesario que I_{min} sea igual a 0, de modo que:

$$I_{min} = 0 = \frac{E}{(1-D)^2 R} - \frac{EDT}{2L} \quad (2.28)$$

Sabiendo que el periodo T de la señal es inversamente proporcional a su frecuencia f , el valor mínimo de inductancia L_{min} estará dado por

$$L_{min} = \frac{D(1-D)^2 R}{2f}. \quad (2.29)$$

Este valor de inductancia es el mínimo requerido para que el circuito opere en el Modo de Conducción Continua.

El último parámetro que se debe calcular es la capacitancia C , que está relacionada con el voltaje de salida del circuito convertidor de tensión CD-CD.

Idealmente, el voltaje a la salida del circuito debe ser constante. Sin embargo, no lo es. Este voltaje de salida presenta un rizo, llamado ΔV_o , como se muestra en la Figura 2.15

Partiendo de la definición de capacitancia en términos de carga Q del capacitor, se tiene:

$$C = \frac{Q}{V_o}; \quad Q = CV_o. \quad (2.30)$$

La variación de carga Q dependerá de la variación en el voltaje de salida V_o , de forma que:

$$\Delta Q = C\Delta V_o; \quad \Delta V_o = \frac{\Delta Q}{C} \quad (2.31)$$

Por otro lado, en la Figura 2.15 se observa que la variación de la carga Q en el capacitor está representada con el rectángulo situado bajo el eje temporal, cuya

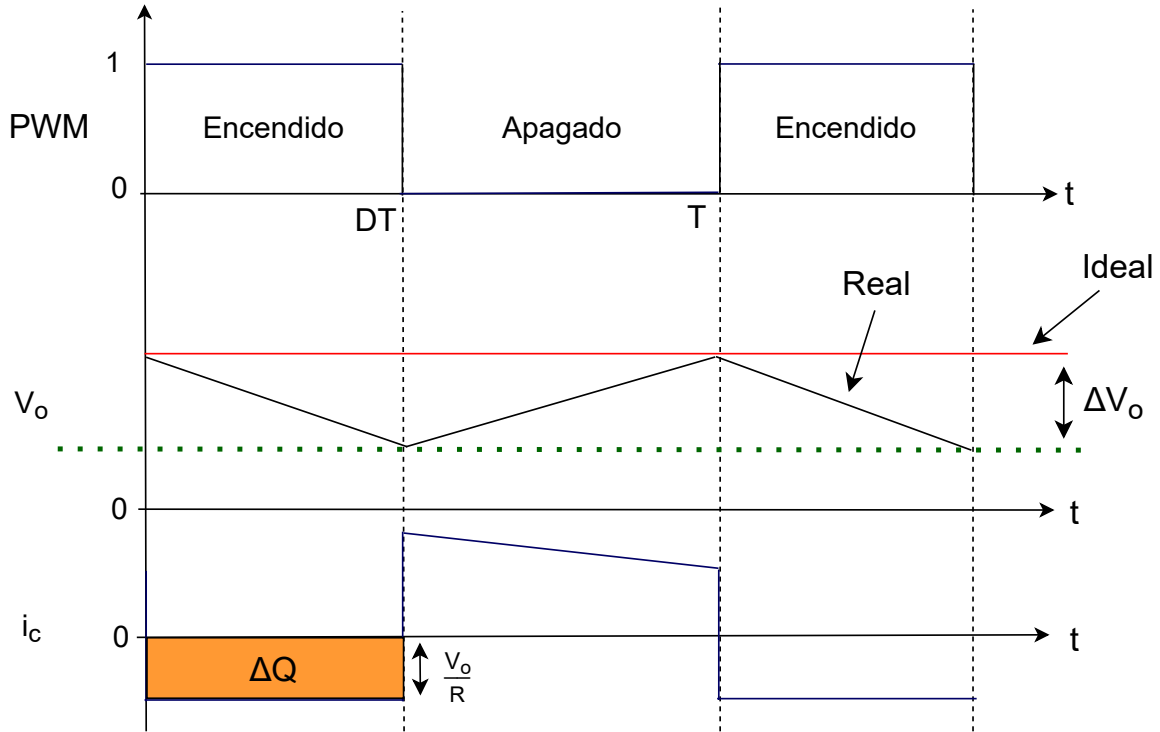


Figura 2.15: Voltaje de salida del circuito y corriente en el capacitor.

área es la corriente en el capacitor. Entonces:

$$\Delta Q = \left(\frac{V_o}{R} \right) DT \quad (2.32)$$

Sustituyendo la Ec. (2.32) en la Ec. (2.31), se tiene:

$$\Delta V_o = \frac{V_o DT}{RC} \quad (2.33)$$

Al determinar un valor de voltaje de rizo a la salida del circuito, se calcula el valor de la capacitancia C como sigue:

$$C = \frac{V_o DT}{R \Delta V_o} \quad (2.34)$$

De esta forma se tienen ahora las expresiones que permitirán calcular el valor de cada uno de los parámetros del convertidor de tensión CD-CD.

Todos los conceptos tratados en este capítulo serán de gran utilidad en la comprensión del resto del escrito. En el siguiente capítulo, se calcularán todos los parámetros del convertidor de tensión CD-CD y del algoritmo PSO. También se

2.4. CONVERTIDORES DE TENSIÓN CD-CD

hablará de la metodología planteada para llevar a cabo el control PID-PSO del convertidor de tensión CD-CD.

En este capítulo se presentará la metodología utilizada en cada una de las etapas de este proyecto de tesis. Comenzando con la obtención del modelo matemático del convertidor de tensión CD-CD y su implementación en Simulink. Posteriormente, se detallarán los parámetros utilizados para la implementación del algoritmo de optimización en Matlab. Finalmente, se describirá el proceso de sintonización del controlador PID-PSO.

3.1. Modelo del Convertidor CD-CD

El convertidor de tensión CD-CD mostrado en la Figura 2.10 tiene dos estados de funcionamiento. El primero de ellos se muestra en la Figura 3.1.

Se observa que el interruptor S_1 está cerrado, mientras que el interruptor S_2 está abierto. Esta configuración pone en funcionamiento la malla 1 del circuito. Aplicando la Ley de Voltajes de Kirchoff, se obtiene la ecuación de la malla por la que fluye la corriente i_L , provocando una caída de voltaje en el inductor V_L y otra caída en la resistencia interna de la bobina V_{RL} . Esta ecuación de malla está dada por:

$$E = L \frac{di_L}{dt} + i_L R_L \quad (3.1)$$

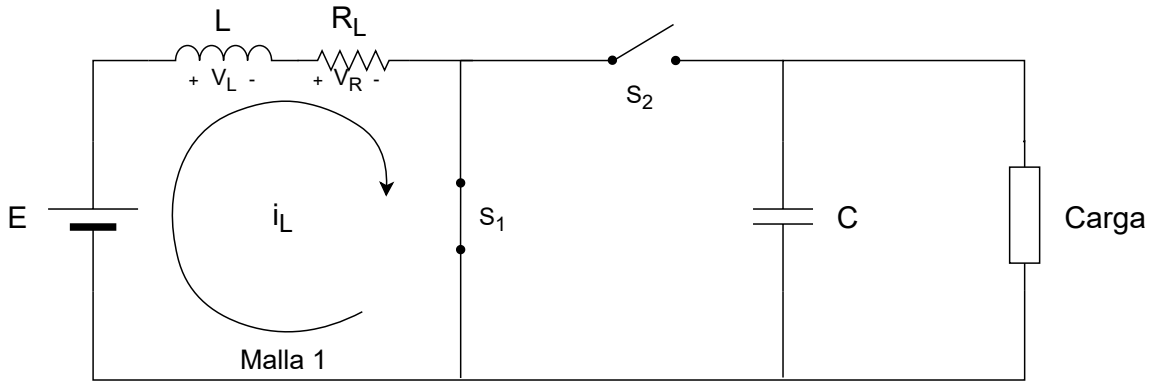


Figura 3.1: Modo 1 de funcionamiento de un circuito elevador.

Despejando la corriente del inductor, la ecuación queda expresada como:

$$\frac{di_L}{dt} = \frac{1}{L} (E - i_L R_L) \quad (3.2)$$

El segundo estado de funcionamiento del circuito se muestra en la Figura 3.2. Ahora, el interruptor S_1 se encuentra abierto y el interruptor S_2 cerrado. La corriente

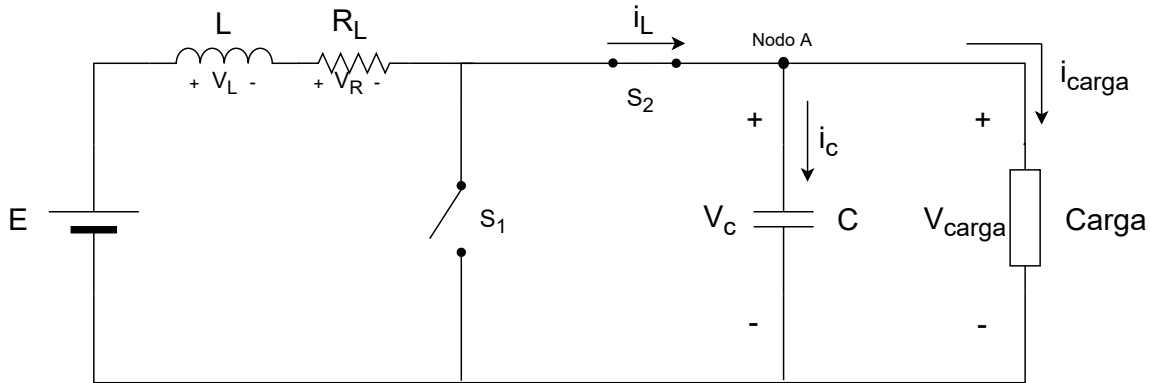


Figura 3.2: Modo 2 de funcionamiento de un circuito elevador.

entregada por la fuente de alimentación pasa a través del inductor y continua para entrar al Nodo A, donde se divide en i_C e i_{carga} , es decir:

$$i_L = i_C + i_{carga} \quad (3.3)$$

La corriente en el capacitor y en la carga están dadas por las Ecs. (3.4)

$$i_C = C \frac{dV_C}{dt}; \quad i_{carga} = \frac{V_C}{R_{carga}} \quad (3.4)$$

Sustituyendo la Ec. (3.4) en la Ec. (3.3), y despejando el voltaje de salida V_C ,

obtenemos:

$$\frac{dV_C}{dt} = \frac{1}{C} \left(i_L - \frac{V_C}{R} \right) \quad (3.5)$$

Para unificar las Ecs. (3.2) y (3.5), es necesario agregar el parámetro U que representa la posición del interruptor S_1 . Dicho parámetro tendrá un valor de $U = 1$ cuando S_1 esté cerrado; y $U = 0$ cuando S_1 esté abierto.

La unificación del sistema está dada por

$$\begin{pmatrix} \frac{d}{dt} i_L \\ \frac{d}{dt} V_C \end{pmatrix} = \begin{pmatrix} \frac{1}{L} (E - i_L R_L - U V_C) \\ \frac{1}{C} (U i_L - \frac{1}{R} V_C) \end{pmatrix} \quad (3.6)$$

La representación en espacio de estados de la Ec. (3.6) es una solución no lineal producida por el producto $U V_C$ y $U i_L$.

3.1.1. Parámetros del Convertidor de Tensión CD-CD

Los valores determinados para la fuente de alimentación (E) y el voltaje de salida (V_C) del convertidor son 200 V y 400 V, respectivamente. Por consiguiente, se deben calcular el resto de los parámetros del circuito.

A partir de la Ec. (2.17), podemos calcular el valor del ciclo de trabajo como:

$$D = 1 - \frac{E}{V_C} = 1 - \frac{200V}{400V} = 0.5 \quad (3.7)$$

El periodo (T) de la señal PWM cuyo ciclo de trabajo es 0.5 es de 0.0001s. La frecuencia es igual a:

$$f = \frac{1}{T} = \frac{1}{0.0001s} = 10,000Hz = 10 kHz. \quad (3.8)$$

El valor mínimo de inductancia L_{min} está dado por la Ec. (2.29):

$$L_{min} = \frac{D(1-D)^2 R}{2f} = \frac{0.5(1-0.5)^2(50 \Omega)}{2(10 kHz)} = 312.5 \mu H. \quad (3.9)$$

Sin embargo, este valor no es comercial, por lo que se utilizará el valor superior más cercano, en este caso 400 μH .

Continuando con el cálculo de los parámetros del convertidor de tensión CD-CD, el valor de la capacitancia C está dado por la Ec. (2.34), en donde es

necesario especificar un valor de voltaje de rizo ΔV_o . Es importante resaltar que, entre menor sea el valor elegido de voltaje de rizo, mayor deberá ser el valor de capacitancia. En este caso, se fijará un valor ΔV_o igual a 20 V. Por lo tanto,

$$C = \frac{(400 \text{ V})(0.5)(0.0001 \text{ s})}{(50 \Omega)(10 \text{ V})} = 50 \mu\text{F}. \quad (3.10)$$

De esta manera, el circuito resultante luego del cálculo de todos los parámetros, se muestra en la Figura 3.3.

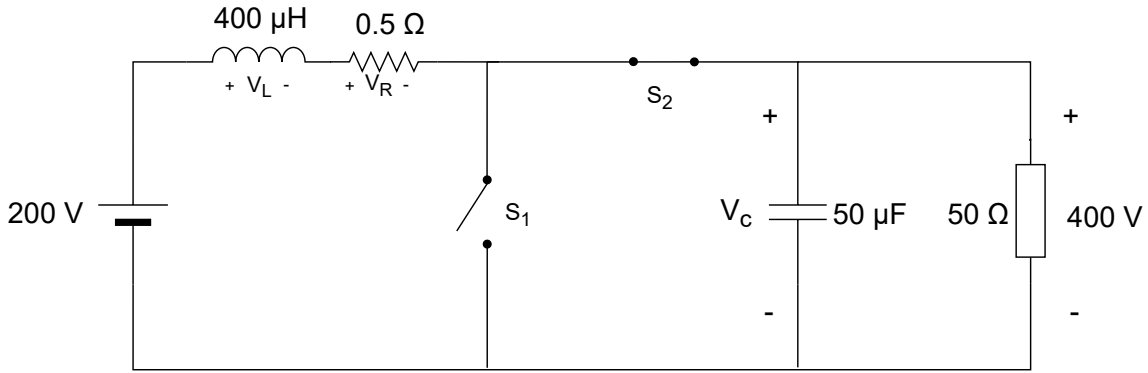


Figura 3.3: Circuito resultante del convertidor de tensión CD-CD.

3.1.2. Punto de Equilibrio

Como se mencionó en la sección anterior, el punto de equilibrio se calcula cuando la tasa de variación de las ecuaciones de estado es cero. Es decir,

$$\begin{pmatrix} \frac{d}{dt} i_L \\ \frac{d}{dt} V_o \end{pmatrix} = \begin{pmatrix} \frac{1}{L} (E - i_L R_L - U V_C) \\ \frac{1}{C} (U i_L - \frac{1}{R} V_C) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \quad (3.11)$$

Sustituyendo los parámetros del circuito, se tiene:

$$200 \text{ V} - i_L (0.1 \Omega) - (0.5) V_C = 0 \quad (3.12)$$

$$0.5 i_L - \frac{V_C}{50 \Omega} = 0 \quad (3.13)$$

Despejando i_L de la Ec. (3.12) se tiene:

$$i_L = \frac{200 V - 0.5 V_C}{0.1 \Omega} \quad (3.14)$$

Despejando i_L de la Ec. (3.13), se tiene:

$$i_L = \frac{V_C}{(0.5) 50 \Omega} \quad (3.15)$$

Igualando las Ecs. (3.14) y (3.15) para encontrar el valor de V_C , se tiene:

$$\frac{200 V - 0.5 V_C}{0.1 \Omega} = \frac{V_C}{(0.5) 50 \Omega},$$

$$(200 V - 0.5 V_C)((0.5) 50 \Omega) = V_C(0.1 \Omega),$$

$$5000 - 12.5 V_C = 0.1 V_C; \quad V_C = \frac{-5000}{12.6} = 396.8253 V,$$

Sustituyendo el valor de V_C en la Ec. (3.15) para obtener el valor de i_L :

$$i_L = \frac{396.8253 V}{(0.5) 50 \Omega}; \quad i_L = 15.873 A. \quad (3.16)$$

Entonces, el punto de operación es $(V_C, i_L) = (396.8253, 15.873)$. En la Figura 3.4, se muestra el retrato de fase del sistema, lo que permitirá conocer la estabilidad del sistema.

En el retrato de fase, se observa que las trayectorias convergen hacia el punto de operación, lo que demuestra la estabilidad del sistema en dicho punto.

3.1.3. Linealización Aproximada

El modelo matemático lineal del sistema se obtiene al aplicar la técnica de linealización aproximada al modelo de la Ec. (3.6). Como resultado de esta técnica, se obtienen tres matrices Jacobianas. La primera de ellas es la matriz A , representada

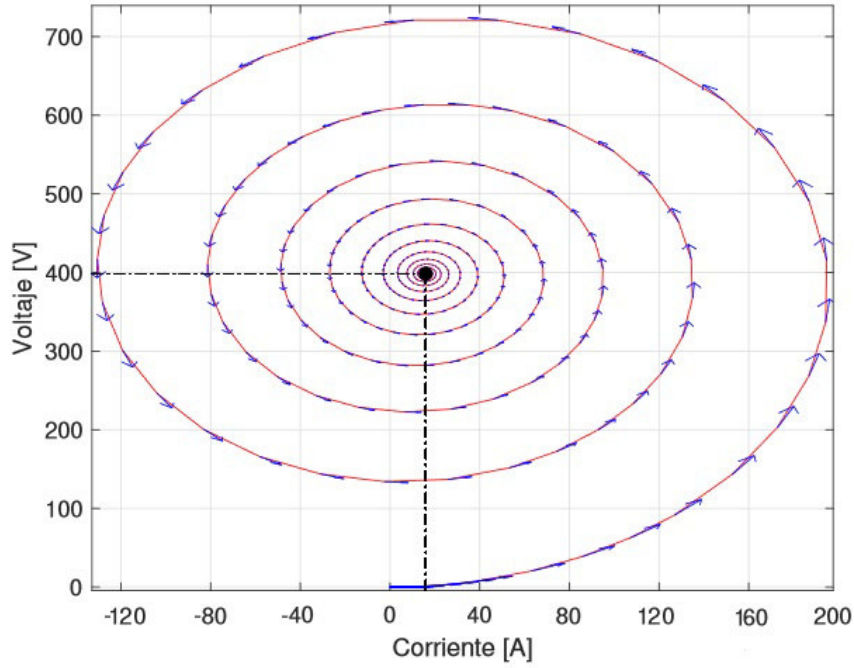


Figura 3.4: Retrato de Fase del convertidor de tensión CD-CD.

como

$$A = \frac{\partial f}{\partial x}|_{(x,u)} = \begin{pmatrix} \frac{\partial f_1}{\partial i_L} & \frac{\partial f_1}{\partial V_C} \\ \frac{\partial f_2}{\partial i_L} & \frac{\partial f_2}{\partial V_C} \end{pmatrix} \quad (3.17)$$

Donde f_1 representa $\frac{1}{L}(E - i_L R_L - U V_C)$ y f_2 a $\frac{1}{C}(U i_L - \frac{1}{R} V_C)$. Ahora, se calculan cada uno de los elementos de la matriz A:

$$A_{(1,1)} = \frac{\partial}{\partial i_L} \left[\frac{1}{L} (E - i_L R_L - U V_C) \right] = -\frac{R_L}{L} \quad (3.18)$$

$$A_{(1,2)} = \frac{\partial}{\partial V_C} \left[\frac{1}{L} (E - i_L R_L - U V_C) \right] = -\frac{U}{L} \quad (3.19)$$

$$A_{(2,1)} = \frac{\partial}{\partial i_L} \left[\frac{1}{C} \left(U i_L - \frac{1}{R} V_C \right) \right] = \frac{U}{C} \quad (3.20)$$

$$A_{(2,2)} = \frac{\partial}{\partial V_C} \left[\frac{1}{C} \left(U i_L - \frac{1}{R} V_C \right) \right] = -\frac{1}{RC} \quad (3.21)$$

$$(3.22)$$

Con estos elementos, se forma la matriz A de la siguiente manera:

$$A = \begin{pmatrix} -\frac{R_L}{L} & -\frac{U}{L} \\ \frac{U}{C} & -\frac{1}{RC} \end{pmatrix} \quad (3.23)$$

Sustituyendo los valores de los parámetros calculados, se tiene:

$$A = \begin{pmatrix} -\frac{0.1 \Omega}{400 [\mu H]} & -\frac{0.5}{400 [\mu H]} \\ \frac{0.5}{50 [\mu F]} & -\frac{1}{(50\Omega)(50 [\mu F])} \end{pmatrix} \quad (3.24)$$

Finalmente,

$$A = \begin{pmatrix} -250 & -1250 \\ 20000 & -400 \end{pmatrix} \quad (3.25)$$

La matriz B está representada como:

$$B = \frac{\partial f}{\partial u} \Big|_{(x,u)} = \begin{pmatrix} \frac{\partial f_1}{\partial U} \\ \frac{\partial f_2}{\partial U} \end{pmatrix} \quad (3.26)$$

Calculando los elementos de B , tenemos:

$$B_{(1,1)} = \frac{1}{L} \frac{\partial}{\partial U} \left(E - i_L R_L - U V_C \right) = -\frac{V_C}{L} \quad (3.27)$$

$$B_{(2,1)} = \frac{1}{C} \frac{\partial}{\partial U} \left(U i_L - \frac{V_C}{R} \right) = \frac{i_L}{C} \quad (3.28)$$

Este vector queda representado de la siguiente manera:

$$B = \begin{pmatrix} -\frac{V_C}{L} \\ \frac{i_L}{C} \end{pmatrix} \quad (3.29)$$

Sustituyendo los valores de los parámetros calculados, la matriz B queda como:

$$B = \begin{pmatrix} -\frac{V_C}{400 [\mu H]} \\ \frac{i_L}{50 [\mu F]} \end{pmatrix} \quad (3.30)$$

La matriz C está representa la salida del sistema y está relacionada con los estados del mismo (V_C e i_L). Particularmente, se desea estudiar el valor de V_C a la salida del sistema, por lo que la matriz C queda representada como:

$$C = \begin{pmatrix} 0 & 1 \end{pmatrix} \quad (3.31)$$

Unificando las Ecs. (3.17), (3.30), (3.31), el modelo matemático lineal aproximado queda como:

$$A = \begin{pmatrix} -250 & -1250 \\ 20000 & -400 \end{pmatrix}; \quad B = \begin{pmatrix} -\frac{V_C}{400 \mu H} \\ \frac{i_L}{50 \mu F} \end{pmatrix}; \quad C = \begin{pmatrix} 0 & 1 \end{pmatrix}. \quad (3.32)$$

3.1.4. Simulaciones

En este proyecto se realizan dos simulaciones de la respuesta lineal y no lineal del convertidor de tensión CD-CD. Es muy importante analizar las condiciones de estabilidad de un sistema no lineal posterior a su linealización. Generalmente las técnicas de análisis de estabilidad están muy bien fundamentadas para sistemas lineales, sin embargo, el rango de estabilidad es reducido alrededor del punto de operación. Es por ello que se requiere comparar cualitativamente la respuesta de un sistema lineal y no lineal en un punto de operación conocido y observar la respuesta en estas regiones para escoger el controlador que mejor se adapte a las condiciones de operación. Se puede realizar un análisis matemático directo de esta respuesta inicial, o se puede utilizar una herramienta a base de bloques en Simulink para estudiar dichos comportamientos de una forma más rápida utilizando los algoritmos numéricos asociados a estos bloques.

La primer simulación mostrada en la Figura 3.5 es el modelo matemático de la Ec. (3.6) mediante el uso de bloques de suma, multiplicación, integración. Se debe recordar que esta es la representación del modelo no lineal del circuito elevador de tensión DC-DC y que no cuenta con ninguna acción de control.

3.1. MODELO DEL CONVERTIDOR CD-CD

La variable U es una señal PWM que conmuta a alta frecuencia el funcionamiento del transistor. En la simulación se utiliza una señal cuadrada con un ancho de pulso de 50% y una amplitud de 1 V. Los bloques de integración $\frac{1}{s}$ se utilizan para realizar la integración de las variables i_L y V_C , ya que, según la Ec. (3.6), están expresadas como derivadas. Se utiliza el *solver ode45* para resolver las ecuaciones diferenciales que describen el sistema.

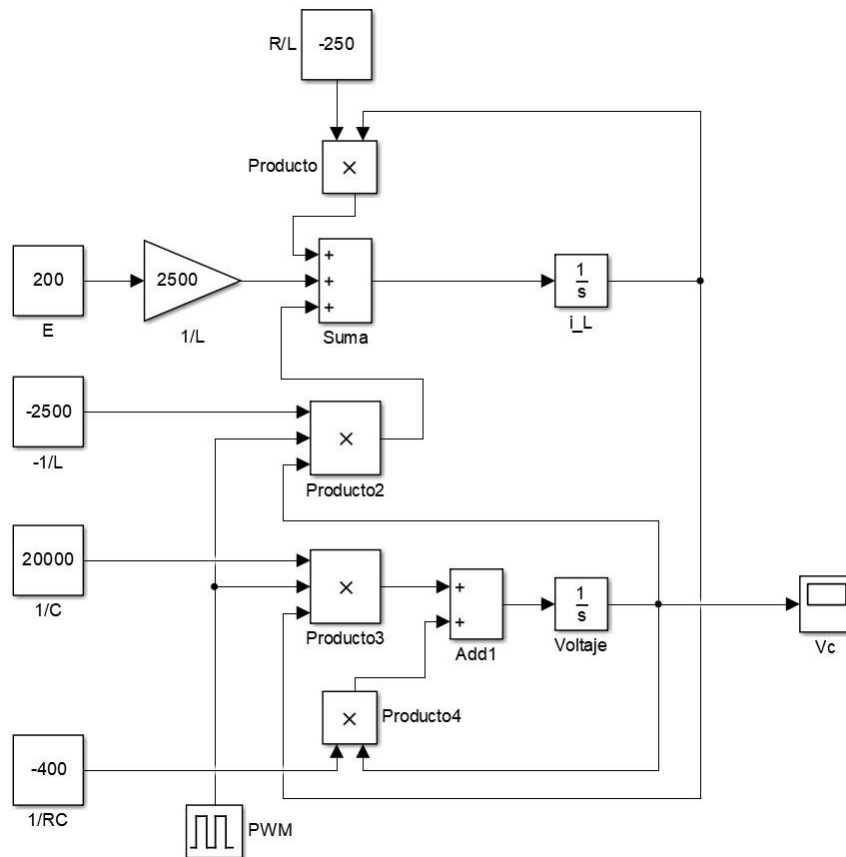


Figura 3.5: Simulación a bloques del modelo matemático no lineal del convertidor de tensión CD-CD.

Posteriormente, se simula el modelo lineal, mostrado en la Figura 3.6, en donde se utilizan los valores de las matrices obtenidas de la Ec. (3.32) luego del proceso de linealización aproximada.

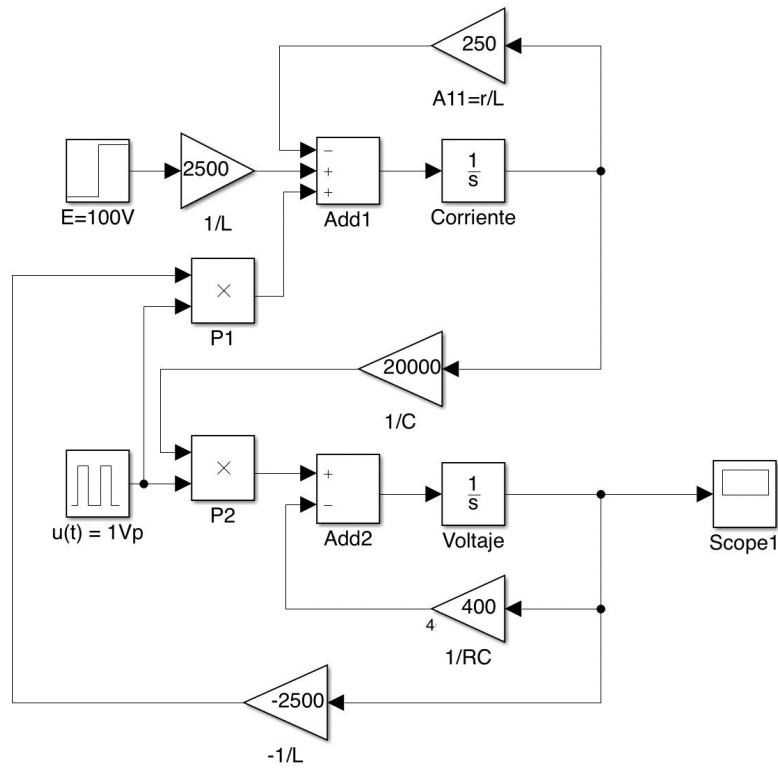


Figura 3.6: Simulación a bloques del modelo matemático lineal del convertidor de tensión CD-CD.

En la Figura. 3.7 se muestra la comparación de la respuesta de ambos sistemas para una entrada escalón. Se observa claramente que no existe alguna diferencia significativa en el comportamiento de los sistemas, a pesar de haber aplicado la linealización aproximada.

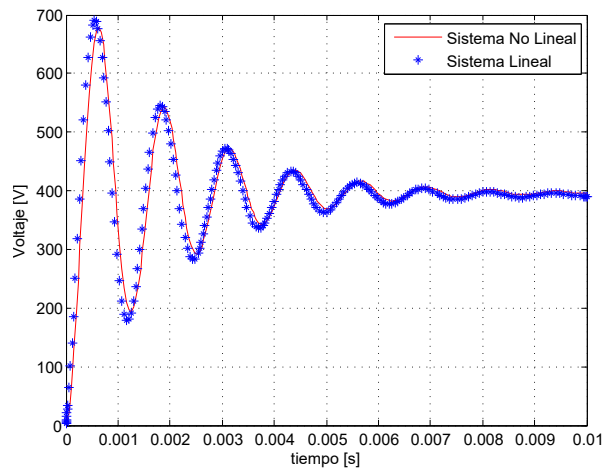


Figura 3.7: Respuesta en lazo abierto del sistema no lineal y lineal.

3.2. Algoritmo PSO

Como se mencionó en el Capítulo 2, el algoritmo PSO trabaja con un conjunto de partículas cuyo objetivo es explorar un espacio de búsqueda. Dichas partículas modifican su posición y velocidad en cada iteración para encontrar soluciones que son evaluadas en la función objetivo, afín de encontrar la solución óptima al problema. En este caso particular, se deben encontrar los valores de ganancias K_P , K_i y K_d del controlador PID que será implementado en el convertidor de tensión CD-CD para obtener una salida de voltaje controlada.

El algoritmo PSO fue codificado en el software Matlab y consta de cuatro partes principales que permiten su correcto funcionamiento. La primera de ellas es la inicialización, que consiste en declarar el valor de las variables N , $MaxIter$, $Voisinage$, w , c_1 , c_2 . Después de una exhaustiva prueba experimental, los valores seleccionados para estas variables se muestran en la Tabla 3.1.

Tabla 3.1: Parámetros del algoritmo PSO

Parámetro	Descripción	Valor
w	Peso inercial	0.5
c_1	Coefficiente individual	0.5
c_2	Coefficiente grupal	0.5
N	Número de individuos	25
$Voisinage$	Tamaño del vecindario	5
$MaxIter$	Número máximo de iteraciones	75

Posteriormente, con la posición y velocidad actuales, comienza el cálculo de la posición y velocidad nuevas para las partículas para buscar las soluciones que serán llamadas por la función *Calling parameters* que conforma la tercera parte del programa.

En la función *Calling parameters*, las ganancias K_P , K_i y K_d encontradas hasta el momento, son utilizadas para ejercer una acción de control sobre el convertidor de tensión CD-CD implementado en Simulink. Al obtener la salida del sistema controlado, se calcula el sobrepico y el tiempo de asentamiento del voltaje de salida del convertidor con la finalidad de ser evaluados en la *función objetivo*.

La función objetivo es la cuarta etapa, en donde se determina si el desempeño de las ganancias K_P , K_i y K_d es óptimo dentro de las condiciones establecidas o si se debe seguir repitiendo el proceso para la búsqueda de nuevas soluciones.

La función seleccionada para la minimización del error, está dada por la siguiente ecuación:

$$f(L_1, E_{r1}, E_{r2}) = \alpha \frac{|L_1 - L_o|}{L_o} + \beta \frac{|E_{r1} - E|}{E} + \gamma |E_{r2}|, \quad (3.33)$$

donde L_1 es el sobre pico calculado en el controlador actual, L_o es el sobre pico de referencia, E_{r1} es el error entre la entrada y la salida del controlador actual, E es la entrada de referencia del controlador, y E_{r2} es la diferencia entre el tiempo de asentamiento de referencia y el tiempo de asentamiento del controlador actual. Las constantes de ponderación α, β y γ son las que permiten distribuir la influencia de cada parámetro en la minimización del error en la función.

Después de varias pruebas experimentales, se determinaron estadísticamente los valores de α, β y γ que permitían cumplir juntamente con los parámetros de diseño, quedando como valores fijos $\alpha = 1.25, \beta = 0.5, \gamma = 0.5$.

3.3. Controlador PID-PSO

El algoritmo PSO es el encargado de la sintonización de las ganancias del controlador PID y con esto garantizar un control óptimo en condiciones nominales de funcionamiento al acoplarse al convertidor de tensión CD-CD. La selección apropiada

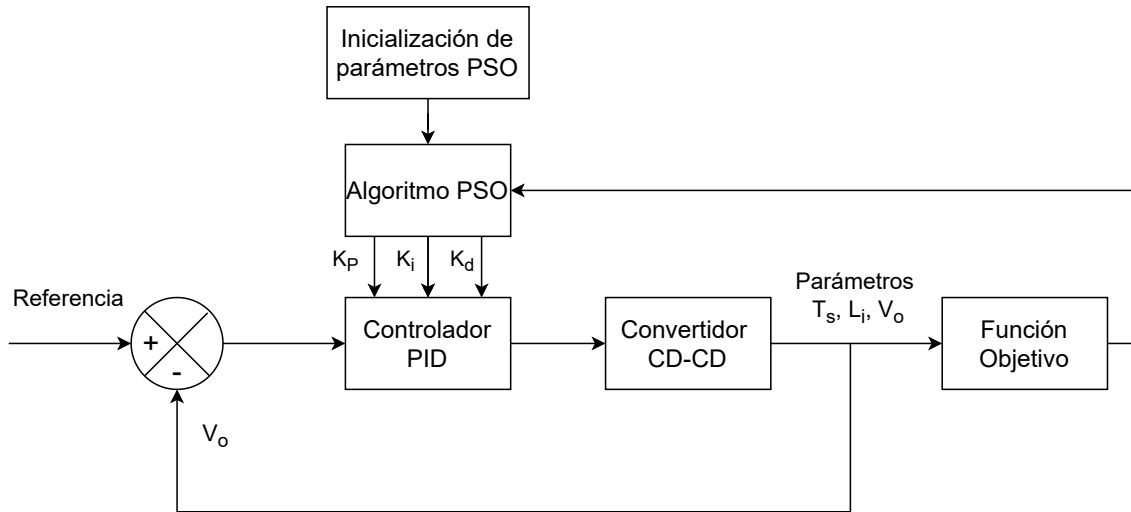


Figura 3.8: Diagrama a bloques del controlador PID-PSO

de los parámetros del controlador PID producirá una respuesta aceptable del sistema y dará lugar a la minimización de los índices formulados en el algoritmo del PSO. En la Figura 3.8 se observa el diagrama a bloques del controlador PID-PSO.

Después de haber analizado el boost-converter desde una perspectiva lineal y no-lineal utilizando un enfoque de análisis y de simulación, en el Capítulo 4 se presentarán los resultados numéricos de este trabajo de tesis.

En este capítulo se presentan los resultados de las diferentes pruebas realizadas para comprobar el funcionamiento y la convergencia del algoritmo PSO como método de sintonización de controladores implementados en sistemas lineales y no lineales.

4.1. Pruebas con Sistemas Lineales

Prueba 1: La función de transferencia de la Ec. (4.1) representa un sistema lineal de segundo orden,

$$G_1(s) = 10 \frac{1}{s(s+1)}. \quad (4.1)$$

La respuesta de este sistema a una entrada de escalón unitario se muestra en la Figura 4.1.

Dentro de las técnicas clásicas de control para sistemas lineales, el lugar geométrico de las raíces compite en practicidad y eficiencia con el método de respuesta en frecuencia, pero es mucho más rápido de diseñar y sus respuestas son más intuitivas al relacionar sus parámetros de respuesta en el dominio del tiempo. Aunque en esta tesis se utilizaron ambas técnicas para efectos de comparación o referencia. Se diseñaron dos controladores para el mismo sistema. El primero de ellos, al que se referirá como controlador calculado, fue sintonizado utilizando el método del lugar geométrico de las raíces [Ogata, 2003]. obteniendo la siguiente función de

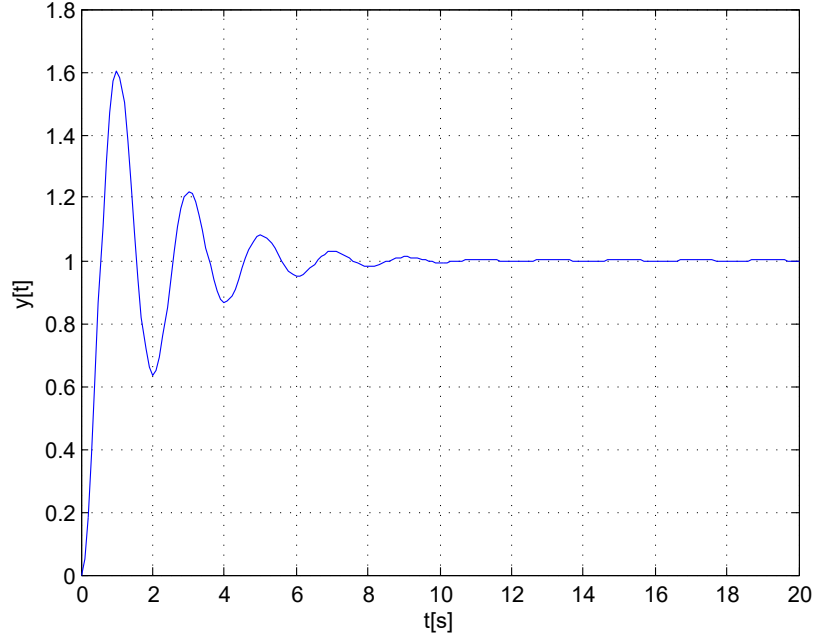


Figura 4.1: Respuesta del sistema a una entrada escalón unitario del sistema (4.1).

transferencia:

$$G_{cl}(s) = 0.9 \frac{s + 1}{s + 3} \quad (4.2)$$

El segundo controlador, referido como controlador optimizado, fue sintonizado con el algoritmo PSO configurado con los parámetros de la Tabla 3.1. La función objetivo a minimizar, está dada por la Ec. (4.3),

$$f(L_1, T_s) = \alpha \frac{|L_1 - L_o|}{L_o} + (1 - \alpha) \frac{|T_s - T_o|}{T_o} \quad (4.3)$$

donde $\alpha = 0.5$, $T_o = 2.0$ s y $L_o = 16.67\%$. El parámetro α es el encargado de ponderar la influencia de los parámetros L y T_s en la función. La función de transferencia calculada por el algoritmo fue:

$$G_{PSO1}(s) = 1.7 \frac{s + 0.839}{s + 3.681} \quad (4.4)$$

En la Figura 4.2 se observa la respuesta $y(t)$ del sistema en lazo cerrado para una entrada escalón unitario en ambos controladores. Dicha figura muestra que la respuesta del controlador sintonizado mediante el algoritmo PSO tuvo menor sobrepico y alcanzó el valor de referencia en menor tiempo comparado con la respuesta del controlador sintonizado por el método convencional.

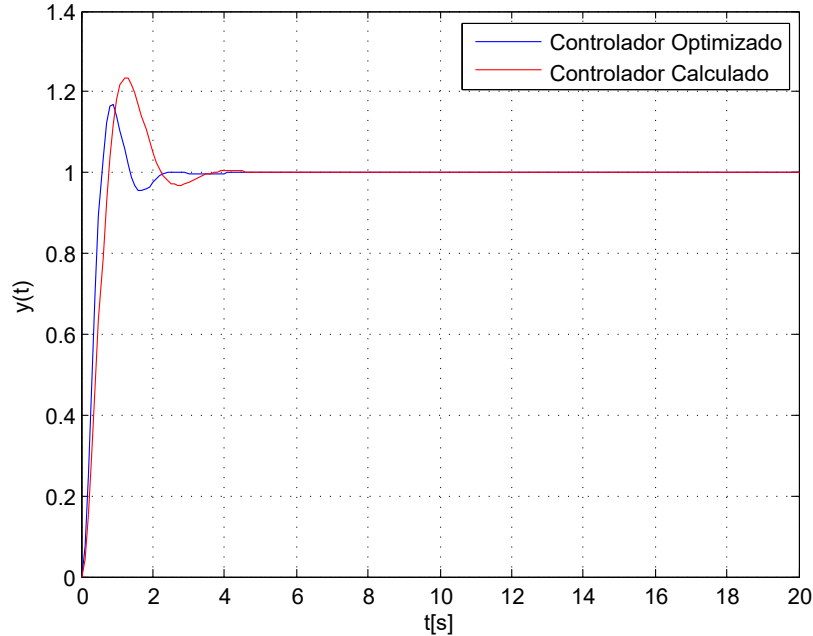


Figura 4.2: Comparación gráfica de la respuesta al escalón unitario del sistema (4.1) controlado.

La Tabla 4.1 muestra claramente que el controlador optimizado logró disminuir en un 6.53% el sobrepico del controlador calculado, así como la disminución del tiempo de asentamiento en un 44.59%.

Tabla 4.1: Comparación del desempeño de los controladores implementados para el sistema (4.1).

Controlador	Sobrepico L_o [%]	Tiempo de asentamiento T_s [s]
Método del lugar de las raíces	23.2	3.7
Algoritmo PSO	16.67	2.05

Prueba 2: En el siguiente ejemplo, se diseña un sistema de control para un vehículo espacial [Ogata, 2003], cuya función de transferencia está dada por

$$G_2(s) = \frac{4}{s(s + 0.5)}. \quad (4.5)$$

La respuesta de este sistema para una entrada escalón unitario, sin acción de control, se muestra en la Figura 4.3.

Se diseñaron dos controladores. El primero de ellos sintonizado con el método

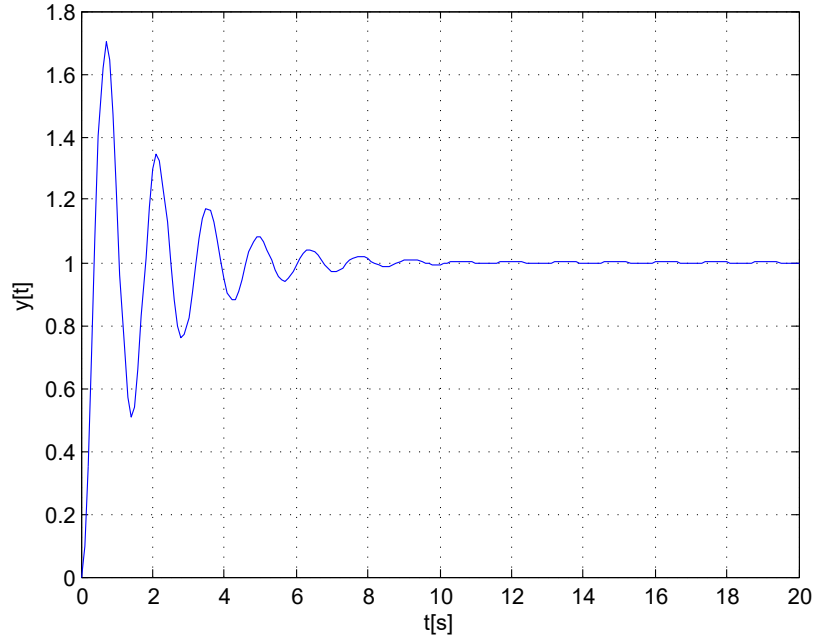


Figura 4.3: Respuesta del sistema a una entrada escalón unitario del sistema (4.5).

del lugar de las raíces [Ogata, 2003], y su función de transferencia está dada por:

$$G_{c2}(s) = 6.26 \left(\frac{s + 0.5}{s + 5.02} \right) \left(\frac{s + 0.2}{s + 0.01247} \right) \quad (4.6)$$

El segundo controlador fue sintonizado con el algoritmo PSO, utilizando la misma función objetivo de la Ec. (4.3). Las ganancias obtenidas del algoritmo PSO fueron $K_P = 0.1924$, $T_i = 0.2729$ s y $T_d = 2.0125$ s.

En la Figura 4.4, se observa la respuesta del sistema en lazo cerrado para una entrada escalón unitario para cada controlador implementado. En la Tabla 4.2 se muestra la comparación de parámetros de ambas respuestas del sistema. Es evidente que el controlador optimizado reduce significativamente el sobrepico en un 94.72% y el tiempo de asentamiento un 35.48%.

Tabla 4.2: Comparación del desempeño de los controladores implementados para el sistema (4.5)

Controlador	Sobrepico L_o [%]	Tiempo de asentamiento T_s [s]
Método del lugar de las raíces	20.84	3.10
Algoritmo PSO	1.10	2.0

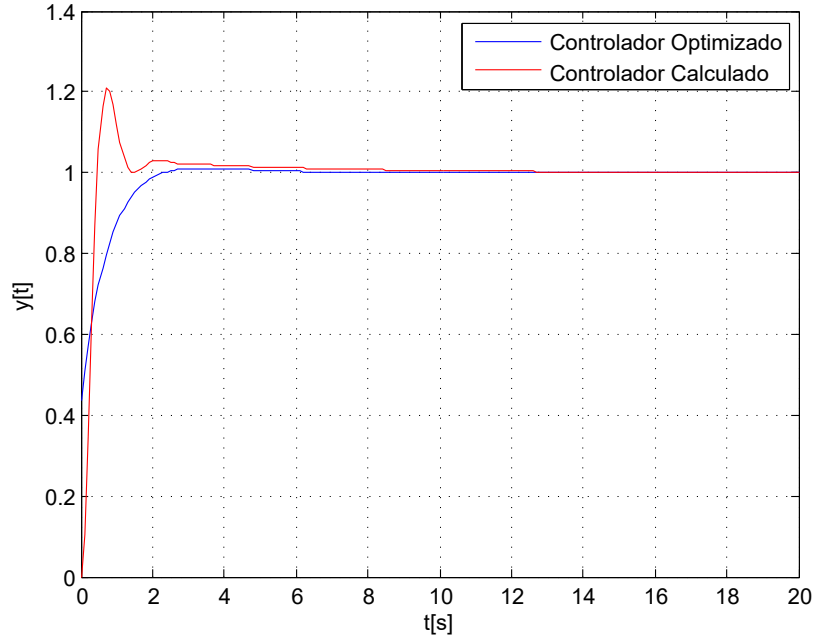


Figura 4.4: Comparación gráfica de la respuesta al escalón unitario del sistema (4.5) controlado.

Prueba 3: La Ec. (4.7) representa un sistema en lazo abierto de segundo orden,

$$G_3(s) = \frac{1}{s(s+1)(0.5s+1)}. \quad (4.7)$$

La respuesta de este sistema a una entrada de escalón unitario se muestra en la Figura 4.5.

Se diseñaron dos controladores: el primero de ellos fue sintonizado con el método de respuesta en frecuencia [Ogata, 2003]. La función de transferencia de este controlador está dada por

$$G_{c3} = 5 \left(\frac{10s+1}{100s+1} \right). \quad (4.8)$$

El segundo controlador se sintonizó con el algoritmo PSO. Se utilizó la misma función objetivo de la Ec. (4.3), obteniendo de dicho algoritmo las ganancias $K_P = 4.0688$, $T_i = 6.9658$ s y $T_d = 0.6275$ s. En la Figura 4.6 se muestra la comparación de ambos controladores, evidenciando un mejor rendimiento del sistema cuando es compensado con el controlador obtenido por el algoritmo PSO, disminuyendo en un 83% el sobrepico de respuesta del sistema al escalón unitario y en un 87.78% el tiempo de asentamiento.

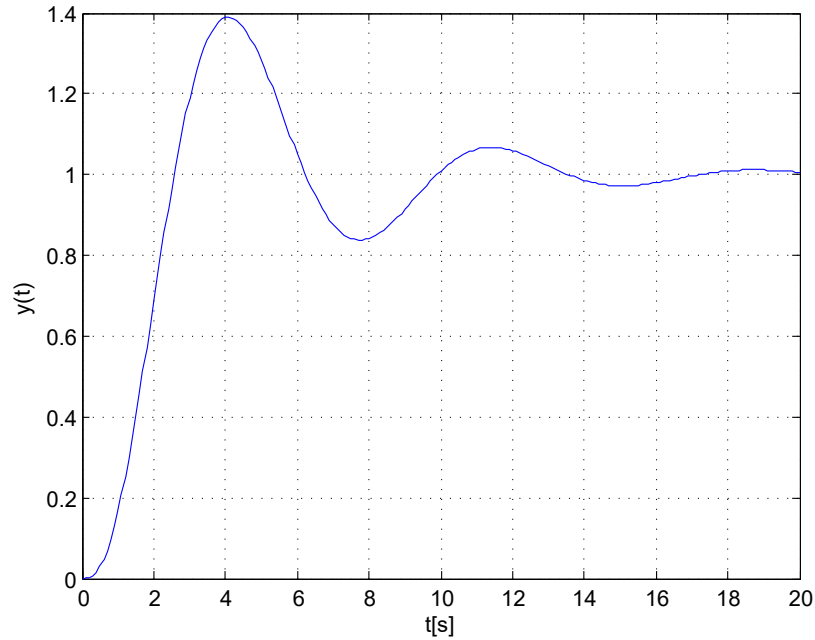


Figura 4.5: Respuesta del sistema a una entrada escalón unitario del sistema (4.7)

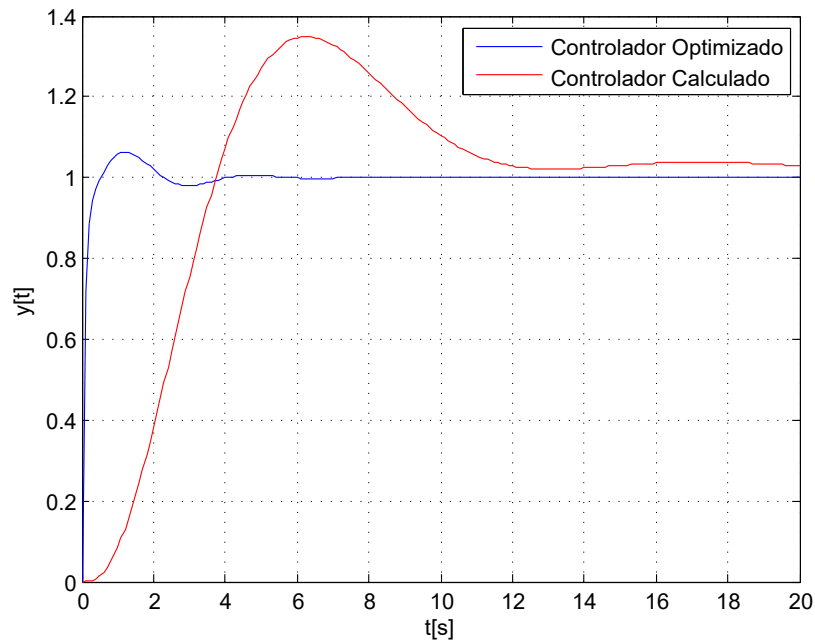


Figura 4.6: Comparación gráfica de la respuesta al escalón unitario del sistema (4.7) controlado.

La Tabla 4.3 muestra el sobrepico y el tiempo de asentamiento de las respuestas del sistema para cada controlador.

Tabla 4.3: Comparación del desempeño de los controladores implementados para el sistema (4.7).

Controlador	Sobrepico L_o [%]	Tiempo de asentamiento T_s [s]
Método de respuesta en frecuencia	38.09	16.37
Algoritmo PSO	6.3723	2.0

4.2. Controlador de un Convertidor de Tensión CD-CD

El algoritmo PSO fue implementado y simulado en Matlab[®], mientras que el convertidor CD-CD fue diseñado en Simulink[®]. Ambas tareas numéricas se trabajaron y acoplaron simultáneamente con el objetivo de obtener los parámetros óptimos del controlador.

Dicha configuración permite evaluar en el sistema o planta del convertidor CD-CD los valores sintonizados del control PID, generando una salida de tensión con los valores nominales de sobrepico, voltaje medio, tiempo de asentamiento y errores de salida que permiten recalculer mediante el optimizador basado en PSO las nuevas ganancias óptimas para el controlador. La función objetivo encargada de dicha evaluación está representada por la siguiente relación,

$$f(L_1, E_{r1}, E_{r2}) = \alpha \frac{|L_1 - L_o|}{L_o} + \beta \frac{|E_{r1} - E|}{E} + \gamma |E_{r2}|, \quad (4.9)$$

donde L_1 es el sobre pico calculado en el controlador actual, L_o es el sobre pico de referencia, E_{r1} es el error entre la entrada y la salida del controlador actual, E es la entrada de referencia del controlador, y E_{r2} es la diferencia entre el tiempo de asentamiento de referencia y el tiempo de asentamiento del controlador actual.

Después de varias pruebas experimentales, se determinaron estadísticamente los parámetros de regularización de la función *fitness* de α , β y γ que nos permitían cumplir juntamente con los parámetros de diseño, quedando como valores fijos $\alpha = 1.25$, $\beta = 0.5$, $\gamma = 0.5$.

En el caso particular donde se desea obtener un voltaje de salida de 400 V se utilizaron los valores de los componentes eléctricos del convertidor de tensión CD-CD mostrados en la Tabla 4.4.

Como punto de partida, en la Figura 3.7 se muestra la respuesta del modelo no lineal y el modelo linealizado, en lazo abierto y sin acción de control. Comparando

Tabla 4.4: *Parámetros del sistema convertidor de tensión CD-CD.*

Parámetro	Descripción	Valor
R	Resistencia de carga	50 [Ω]
C	Capacitancia	50 [μ F]
L	Inductancia	400 [μ H]
E	Fuente de alimentación	200 [V]
R_L	Resistencia en la inductancia	0.1 [Ω]

las respuestas, se determinó que el comportamiento de ambos sistemas es similar y controlable cerca de las condiciones de operación. Esta respuesta del modelo lineal implementado sin acción de control presenta un sobrepico de 689 V, es decir, 74 % más del valor de referencia, indicando que una acción de control es requerida.

En la Figura 3.4 se observa la trayectoria que siguen los estados, cuando el proceso es sometido a condiciones iniciales nulas, el punto de equilibrio se llega con cualquier condición inicial cercana a este punto. Este punto de equilibrio corresponde a un punto de espiral asintóticamente estable el cual lo podemos verificar calculando los eigenvalores de la matriz de estados del modelo planteado.

En la Figura 3.8 se muestra el diagrama de bloques del sistema integrando el controlador PID. Posteriormente, después de una exhaustiva evaluación del trabajo conjunto del convertidor de tensión CD-CD y el controlador sintonizado por el método de optimización PSO, se obtuvieron los resultados satisfactorios luego de 30 repeticiones.

Para un factor de amortiguamiento $\alpha = 0.5$, y las ganancias PID sintonizadas a través del algoritmo de optimización PSO, se muestran en la Tabla 4.5. Estos

Tabla 4.5: *Parámetros del Controlador PID para 30 repeticiones.*

Ganancia	Media (μ)	Desv. Stand. (σ)
K_P	2.97417×10^{-4}	$\pm 2.2018 \times 10^{-4}$
K_I	9.92×10^{-5}	$\pm 1.19 \times 10^{-4}$
K_D	7.73×10^{-5}	$\pm 1.15 \times 10^{-4}$

resultados se realizaron con 30 repeticiones para revisar su estabilidad y consistencia numérica, observándose una muy baja desviación estándar. Al aplicar el controlador PID sobre el sistema en estudio se observa una respuesta que cae dentro de los

parámetros de diseño, mostrada en la Figura 4.7.

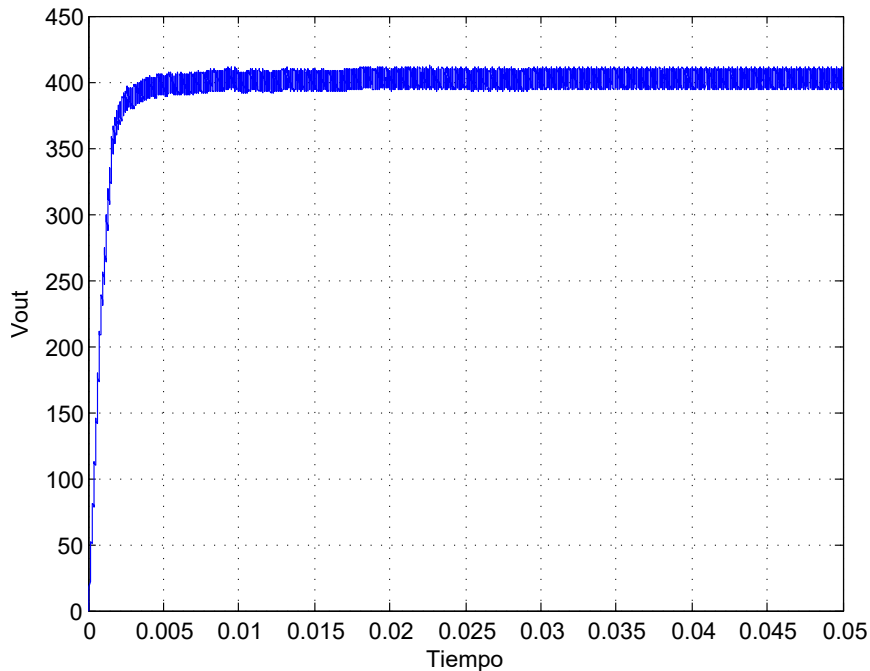


Figura 4.7: Voltaje de salida del convertidor de tensión CD-CD.

Claramente, se observa que el controlador actúa en el sistema suprimiendo el sobrepico y las oscilaciones presentes en la Figura 3.7 y manteniendo una salida constante, con un porcentaje de rizo dentro de un rango aceptable de acuerdo con lo previsto en el cálculo de los parámetros del convertidor, siguiendo la entrada de referencia y cumpliendo con un tiempo de asentamiento $T_s < 0.1s$ y un sobrepico $L_1 < 16\%$.

Además, en la Figura 4.8 se muestra la respuesta del sistema cuando se aplica un controlador sintonizado por el método de retroalimentación de estados, diseñado por [Valderrama et al., 2011], para un convertidor de tensión CD-CD.

Se observa que la acción de control ejercida por el controlador sintonizado con el algoritmo PSO logra que el sistema alcance el estado estable en un tiempo de 0.005 s, mientras que la respuesta del sistema con el controlador sintonizado por retroalimentación de estados tarda hasta 0.02 s.

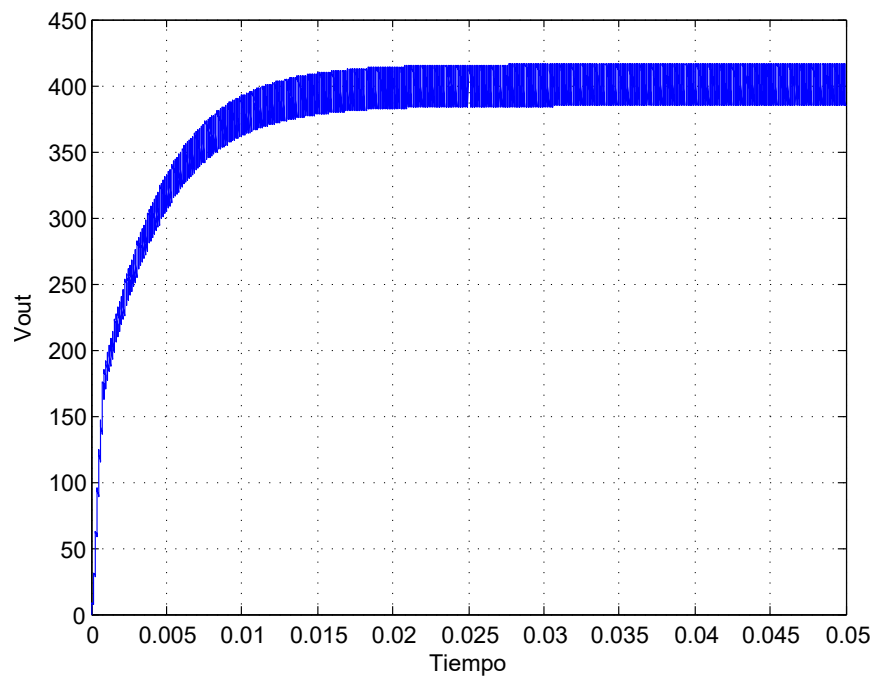


Figura 4.8: Voltaje de salida del convertidor de tensión CD-CD controlado por retroalimentación de estados.

Conclusiones

En este trabajo, se implementó el algoritmo metaheurístico PSO para sintonizar las ganancias K_P , K_i y K_d de algunos controladores aplicados a sistemas lineales. Se comparó el resultado de los controladores obtenidos mediante el algoritmo PSO con los ya obtenidos mediante el método del lugar geométrico de las raíces y el método de la respuesta en frecuencia.

El resultado de esta comparación demostró que los controladores sintonizados con el algoritmo PSO mostraron un mejor rendimiento, disminuyendo el tiempo de asentamiento y el sobrepico en la respuesta de los sistemas ante entradas de referencia escalón unitario.

De igual manera, se implementó el algoritmo para el ajuste de los parámetros de un controlador PID, encargado de controlar el convertidor de tensión de corriente directa DC-DC que en su forma estándar es un sistema no-lineal. Los resultados de las simulaciones confirmaron que el controlador sintonizado con el algoritmo PSO cumple con las condiciones de diseño establecidas las cuales incluyen, disminuir el sobre pico presente en la respuesta del sistema sin la acción de control y en la reducción significativa del tiempo de asentamiento original. Además, el controlador PID-PSO proporciona una respuesta adecuada, manteniendo el voltaje de salida en el valor de referencia deseado y con el rizo dentro de las especificaciones.

Bibliografía

- [Aguirre-Ramos et al., 2019] Aguirre-Ramos, H., Avina-Cervantes, J. G., Ilunga-Mbuyamba, E., Cruz-Duarte, J. M., Cruz-Aceves, I., and Gallegos-Arellano, E. (2019). Conic sections fitting in disperse data using differential evolution. *Applied Soft Computing*, 85:105769.
- [Beni and Wang, 1993] Beni, G. and Wang, J. (1993). Swarm intelligence in cellular robotic systems. In *Robots and biological systems: towards a new bionics?*, pages 703–712. Springer.
- [Dorigo et al., 2006] Dorigo, M., Birattari, M., and Stutzle, T. (2006). Ant colony optimization. *IEEE computational intelligence magazine*, 1(4):28–39.
- [Eiben et al., 2003] Eiben, A. E., Smith, J. E., et al. (2003). *Introduction to evolutionary computing*, volume 53. Springer.
- [Engelbrecht, 2007] Engelbrecht, A. P. (2007). *Computational intelligence: an introduction*. John Wiley & Sons.
- [Holland et al., 1992] Holland, J. H. et al. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.

- [Kennedy and Eberhart, 1995] Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization in: Neural networks. In *Proceedings IEEE International Conference on 1995*, pages 1942–1948.
- [Li et al., 2014] Li, M., Du, W., and Nian, F. (2014). An adaptive particle swarm optimization algorithm based on directed weighted complex network. *Mathematical problems in engineering*, 2014.
- [Mirjalili et al., 2020] Mirjalili, S., Dong, J. S., Lewis, A., and Sadiq, A. S. (2020). Particle swarm optimization: theory, literature review, and application in airfoil design. *Nature-inspired optimizers*, pages 167–184.
- [Mirjalili et al., 2014] Mirjalili, S., Mirjalili, S. M., and Lewis, A. (2014). Grey wolf optimizer. *Advances in engineering software*, 69:46–61.
- [Ogata, 2003] Ogata, K. (2003). *Ingeniería de control moderna*. Pearson Educación.
- [Ou and Lin, 2006] Ou, C. and Lin, W. (2006). Comparison between pso and ga for parameters optimization of pid controller. In *2006 International conference on mechatronics and automation*, pages 2471–2475. IEEE.
- [Sandou, 2013] Sandou, G. (2013). *Metaheuristic optimization for the design of automatic control laws*. John Wiley & Sons.
- [Solihin et al., 2011] Solihin, M. I., Tack, L. F., and Kean, M. L. (2011). Tuning of pid controller using particle swarm optimization (pso). In *Proceeding of the International Conference on Advanced Science, Engineering and Information Technology*, volume 1, pages 458–461.
- [Sulttan et al., 2020] Sulttan, M. Q., Jaber, M. H., and Shneen, S. W. (2020). Proportional-integral genetic algorithmcontroller for stability of tcp network. *International Journal of Electrical and Computer Engineering (IJECE)*, 10(6):6225–6232.
- [Tran and Nguyen, 2018] Tran, H. K. and Nguyen, T. N. (2018). Flight motion controller design using genetic algorithm for a quadcopter. *Measurement and Control*, 51(3-4):59–64.

- [Valderrama et al., 2011] Valderrama, F. F., Moreno, H., and Vega, H. M. (2011). Análisis, simulación y control de un convertidor de potencia dc-dc tipo boost. *Ingenium Revista de la facultad de ingeniería*, 12(24):44–55.
- [Valencia-Rivera et al., 2020] Valencia-Rivera, G. H., Merchan-Villalba, L. R., Tapia-Tinoco, G., Lozano-Garcia, J. M., Avina-Cervantes, M. A. I.-M., and Gabriel, J. (2020). Hybrid lqr-pi control for microgrids under unbalanced linear and nonlinear loads. *Mathematics*, 8(7):1096.
- [Vaquero López and Milian del Valle, 2016] Vaquero López, J. and Milian del Valle, C. L. (2016). Control no lineal mediante linealización aproximada aplicada en el sistema no lineal posición de un anillo sobre un aro rotatorio. *Revista Cubana de Ciencias Informáticas*, 10(2):89–99.
- [Wolpert and Macready, 1997] Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82.