

**DISEÑO Y OPTIMIZACIÓN DE UN CONTROLADOR BASADO EN LÓGICA
DIFUSA PARA UN CONVERTIDOR DE POTENCIA TIPO REDUCTOR DC/DC.**

**ANDRÉS FELIPE SARMIENTO BARBOSA
ANGIE DANIELA SIERRA MOYANO**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICOMECHANICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
BUCARAMANGA**

2019

**DISEÑO Y OPTIMIZACIÓN DE UN CONTROLADOR BASADO EN LÓGICA
DIFUSA PARA UN CONVERTIDOR DE POTENCIA TIPO REDUCTOR DC/DC.**

**ANDRÉS FELIPE SARMIENTO BARBOSA
ANGIE DANIELA SIERRA MOYANO**

Trabajo de grado para optar por el título de Ingeniero Electrónico

**Director:
JAVIER ENRIQUE SOLANO MARTÍNEZ
PhD en Ingeniería Eléctrica**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICOMECHANICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA**

2019

DEDICATORIA

A mis padres, quienes siempre creyeron en mí y me apoyaron a lo largo del camino; sin ellos nada de esto hubiera sido posible. Su apoyo, amor incondicional y ejemplo serán por siempre mi guía.

A Daniela, quien desde el primer día ha sido siempre mi luz en el camino. Es imposible agradecer lo suficiente por haber encontrado una compañera, no solo de tesis, si no de vida, tan increíble como lo es ella. Que este logro sea solo el principio del maravilloso futuro que sé que te espera.

A mis abuelos, mis tíos y mis primos quienes me ayudaron de múltiples formas y quienes siempre me apoyaron.

A mis mejores amigos, mis compañeros de universidad, quienes siempre me brindaron apoyo y amistad incondicional y la sana competencia para esforzarme a ser mejor cada día.

Y por último pero no menos importante, a mis profesores. Tuve la oportunidad de contar con docentes de una calidad invaluable, quienes me enseñaron la belleza de aprender.

A todos ustedes, mis más sinceros agradecimientos, este logro es de todos nosotros.

Andrés Felipe Sarmiento Barbosa

DEDICATORIA

Primeramente agradezco a Dios quien fue mi guía, mi sustento y mi fortaleza, quien dio la gracia a mis padres y los recursos para que fuese posible lograr mi sueño de convertirme en ingeniera electrónica de una gran institución como lo es la Universidad Industrial de Santander.

A mis padres que con esfuerzo y dedicación me motivaron todos los días hasta que llegase a la meta.

A mi hermano menor que me retó para no claudicar y que fuese siempre su ejemplo.

A mi compañero de tesis quien fue mi apoyo durante todo este camino y con quien trabajamos en equipo desde el principio hasta el último momento.

A mis compañeros de la carrera que me motivaban a seguir adelante, a mi grupo de estudio que me ayudó a perseverar y no rendirme en el trayecto.

A los profesores especialmente a nuestro director Javier Solano quien nos ayudó en el desarrollo de este proyecto, al profesor Ricardo Alzate que trabajó con nosotros durante todo el proceso y muchos otros que con sus enseñanzas marcaron mi vida.

A todos lo que formaron parte de este proceso y gracias a ellos estoy acá, ¡MIL GRACIAS!

Angie Daniela Sierra Moyano

CONTENIDO

	Pág.
INTRODUCCIÓN	14
1. OBJETIVOS	16
1.1 OBJETIVO GENERAL	16
1.2 OBJETIVOS ESPECÍFICOS	16
2. MOTIVACIÓN	17
3. ANTECEDENTES	18
4. CONVERTIDOR	19
5. CONTROLADORES	22
5.1 CONTROLADOR PID	22
5.2 LA LÓGICA DIFUSA Y EL CONTROLADOR FUZZY	23
6. MÉTODO DE OPTIMIZACIÓN	26
7. CASO DE ESTUDIO	30
8. IMPLEMENTACIÓN	37
8.1. CONTROLADOR FUZZY	37
9. RESULTADOS	44
9.1. CONTROLADOR PID:	51
9.2 COMPARACIÓN DE LOS RESULTADOS:	53
10. CONCLUSIONES	56
BIBLIOGRAFÍA	58

LISTA DE FIGURAS

	Pág.
Figura 1. Esquema del convertidor Buck.	20
Figura 2. Esquema de simulación para controlador Fuzzy.	25
Figura 3. Diagrama de flujo correspondiente al algoritmo general propuesto.	29
Figura 4. Esquema de control utilizado.	31
Figura 5. Controlador Fuzzy utilizado.	31
Figura 6. Convertidor Buck utilizado.	32
Figura 7. Derivador numérico.	32
Figura 8. Integrador numérico.	32
Figura 9. Funciones de pertenencia iniciales para el error (E).	34
Figura 10. Funciones de pertenencia iniciales para la derivada del error (EE).	35
Figura 11. Funciones de pertenencia iniciales la salida (OUT).	35
Figura 12. Gráfica para tensión de salida ante cambio en la tensión de referencia.	36
Figura 13. Funciones de pertenencia optimizadas para el error.	44
Figura 14. Funciones de pertenencia optimizadas para la derivada del error.	45
Figura 15. Funciones de pertenencia optimizadas para la salida del controlador.	45
Figura 16. Resultados de simulación para funciones de pertenencia optimizadas.	46
Figura 17. Evolución del factor de centrado.	47
Figura 18. Evolución del factor de simetría.	48
Figura 19. Funciones de pertenencia optimizadas para el error (variación de resistencia de carga).	49
Figura 20. Funciones de pertenencia optimizadas para la derivada del error (variación de resistencia de carga).	50

Figura 21. Funciones de pertenencia optimizadas para la salida (variación de resistencia de carga).....	50
Figura 22. Diagrama de bloques del sistema usando controlador PID.	52
Figura 23. Salida usando controlador PID.	52
Figura 24. Variación de la resistencia de carga usando controlador PID.....	53
Figura 25. Cálculo de la integral del error absoluto.....	54

LISTA DE TABLAS

	Pág.
Tabla 1. Valores del convertidor	30
Tabla 2. Reglas Lógicas para el controlador.....	33
Tabla 3. Valores de las ganancias.	35
Tabla 4. Valores de las ganancias optimizados para variación de la tensión de referencia.	49
Tabla 5. Parámetros del controlador PID.	51
Tabla 6. Comparación de resultados del controlador Fuzzy optimizado y sin optimizar.	54
Tabla 7. Resultados ante una variación en la tensión de entrada.....	54
Tabla 8. Resultados ante una variación en la resistencia de carga.	55

RESUMEN

TÍTULO: DISEÑO Y OPTIMIZACIÓN DE UN CONTROLADOR BASADO EN LÓGICA DIFUSA PARA UN CONVERTIDOR DE POTENCIA TIPO REDUCTOR DC/DC*.

AUTORES: ANDRÉS FELIPE SARMIENTO BARBOSA
ANGIE DANIELA SIERRA MOYANO **

PALABRAS CLAVE: unciones de pertenencia, lógica difusa, algortimos genéticos, controlador PID, convertidor de potencia, ganancias.

DESCRIPCIÓN

A continuación se propone un método de optimización para controladores de Lógica Difusa utilizando Algoritmos Genéticos. Durante el desarrollo de este trabajo de grado se discuten parámetros cuya optimización resulta relevante para el adecuado funcionamiento del convertidor de potencia; además, se brinda una breve introducción de aspectos como: la planta a controlar, el método de control y el método de optimización utilizado para facilitar el aprendizaje del funcionamiento de los controladores Fuzzy. Se propone un método que optimiza la posición de las funciones de pertenencia junto con las ganancias correspondientes al error, la derivada del error, y la ganancia de salida. El método de diseño se aplica a un convertidor de potencia tipo reductor DC/DC con el fin de regular su tensión de salida ante cambios en el voltaje de referencia y la resistencia de carga. Se presentan nuevos conceptos como factor de simetría y factor de centrado para hacer frente a los problemas de reducción de parámetros a optimizar y eficiencia en la selección aleatoria. Los resultados obtenidos mediante el algoritmo general propuesto se comparan con un controlador de tipo proporcional, integral y derivativo (PID) diseñado mediante sintonización, y se utilizan como criterios de comparación la integral del error absoluto y el tiempo de establecimiento.

* Proyecto de grado

** Facultad de Ingenierías Fisicomecánicas Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones Director: Javier Enrique Solano Martínez PhD en Ingeniería Eléctrica

ABSTRACT

TITLE: DISEÑO Y OPTIMIZACIÓN DE UN CONTROLADOR BASADO EN LÓGICA DIFUSA PARA UN CONVERTIDOR DE POTENCIA TIPO REDUCTOR DC/DC*.

AUTORES: ANDRÉS FELIPE SARMIENTO BARBOSA
ANGIE DANIELA SIERRA MOYANO **

KEYWORDS: membership functions, fuzzy logic, genetic algorithms, PID controller, power converter, gains..

DESCRIPCIÓN

A genetic algorithm-based method for Fuzzy Logic Controller parameter optimization is proposed in this paper. The relevance of the parameters to be optimized and their impact on the correct functioning of the power converter is discussed along with brief insights on the used concepts, such as the control plant, which is a DC/DC Buck converter, the principles of Fuzzy Logic and Fuzzy Logic Controllers and the fundamentals of Genetic Algorithms. A method for optimizing the membership functions position along with the gains corresponding to the error, the derivative of the error and the output of the controller is proposed. The proposed design method is tested on a DC/DC Buck type power converter in order to regulate its output voltage (voltage of the capacitor) in presence of reference voltage variations and load resistance variations. New concepts are introduced such as the symmetry factor (SF) and the centroid factor (CF) in order to deal with the problem of a high number of optimization parameters and the efficiency of the random selection. The results obtained using the proposed method are compared with the results of a proportional, derivative and integral (PID) controller designed using PID tuning, and the criteria for the comparison of the controllers are integral of the absolute error and settling time.

* Proyecto de grado

** Facultad de Ingenierías Fisicomecánicas Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones Director: Javier Enrique Solano Martínez PhD en Ingeniería Eléctrica

INTRODUCCIÓN

Los convertidores de potencia son circuitos de eléctricos utilizados en múltiples aplicaciones, el control de estos se realiza por técnicas variadas como el PID ^{1 2 3} o Lógica Difusa (FLC) ^{4 5 6 7}.

Los trabajos presentados en la literatura muestran que el control de lógica difusa optimizado tiene resultados favorables, sin embargo, estos se enfocan en la optimización del FLC pero pocos se plantean una optimización simultánea de los parámetros del FLC y el controlador global, lo cual se plantea en este artículo.

Como primera estrategia de control se utiliza el controlador FLC (como se presenta en ⁸) y además se propone un algoritmo para el diseño automatizado y optimizado del mismo, usando el software de Matlab, Simulink el cual se contrasta con un PID

¹ AMEUR A., HARMAS M.N and ABDERREZEK H.. "Stochastic Optimization Methods Based Robust Control for DC/DC Buck Converter" in 5th International Conference on Control, Decision and Information Technologies (CoDIT'18).IEEE. 2018

² WANG C.Y., OU Y. C., WU C. F. and SHIUE M. T.. "A Voltage-Mode DC-DC Buck Converter with Digital PID Controller" in 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD). 2015

³ PARK H. S., and KIM H. J.. "Simultaneous Control of DC-DC Converters by DSP Controller", IEEE PEDS. Indonesia. 2001.pp 89-93

⁴ DÍAZ SAÉNZ H. D., PÉREZ MUNEVAR J.A.."Modelado, simulación e implementación de un sistema de control de tensión para un convertidor Buck, basado en lógica difusa."Colombia. 2007

⁵ CHALMERS S. J., "A design methodology for fuzzy logic controllers", Synthesis and Optimization of Logic Systems, IEEE Colloquium on. 1994. Cap 10,pp 1-4

⁶ BENDAOU K., KRIT S., KABRANE M., OUADANI H., ELASKRI M., KARIMI K., ELBOUSTY H. and ELMAIMOUNI L.."Fuzzy Logic Controller (FLC):Application to control DC-DC buck converter." Laboratory of Engineering Sciences and Energies, Polydisciplinary Faculty of Ouarzazate. Ibn Zohr University- Agadir. Morocco. 2017

⁷ HOLGADO J.. "Design improvement in fuzzy controllers.", Department of Electronic and Control Engineering, University of Cadiz. Spain.1996. pp. 2038-2044

⁸ SWATHY K, SHRUTIKA JANTRE MS., YOGITA JADHAV Ms., SUSHIL Mr LABDE. M., PRATIK KADAM Mr.. "Design and Hardware Implementation of Closed Loop Buck Converter Using Fuzzy Logic Controller.", Proceedings of the 2nd International conference on Electronics, Communication and Aerospace Technology.ICECA. 2018.pp. 175-180

que actúe sobre la misma planta para así verificar la eficiencia del algoritmo diseñado.

Este trabajo de grado se distribuye de la siguiente forma: En la primera sección se habla del convertidor *Buck* y su modelo matemático⁹, en la sección 2 del controlador (FLC y PID), en la 3 se da una breve introducción a la lógica difusa, en la sección 4 se analiza caso de estudio, en la 5 se menciona implementación de los controladores en el software de Matlab (Simulink) y finalmente, en la sección 6 se muestran los resultados obtenidos.

⁹ HART D. W.. \textit{Power Electronics.}. 1ra. ed. McGraw Hill. 2010. Capítulo 6.

1. OBJETIVOS

1.1 OBJETIVO GENERAL

Diseñar un controlador basado en lógica difusa para el control de la tensión de salida de un convertidor de potencia DC/DC implementando un algoritmo genético para optimizar los parámetros del controlador.

1.2 OBJETIVOS ESPECÍFICOS

- Constituir un escenario de simulación para un convertidor DC/DC reductor sometido a perturbaciones de suministro y/o de carga.
- Implementar una estrategia de control basada en lógica difusa para regular la tensión de salida del circuito.
- Emplear una rutina basada en algoritmos genéticos para optimizar los parámetros del controlador diseñado.

2. MOTIVACIÓN

El diseño de los controladores de Lógica Difusa ha sido estudiado con distintos enfoques ¹⁰, ¹¹ siendo comúnmente utilizado en sistemas complejos en los cuales el control convencional no representa una opción adecuada.

Con este proyecto se busca desarrollar e implementar un algoritmo para el diseño de sistemas de control óptimos (rápidos y con el menor error posible) aplicados a convertidores de potencia de tipo DC/DC. Para garantizar la mejor respuesta del sistema se propone el uso de controladores basados en Lógica Difusa cuyos parámetros (funciones de pertenencia y ganancias) son optimizados mediante Algoritmos Genéticos, lo que permitirá verificar la funcionalidad de este tipo de control y su eficiencia con respecto a los métodos convencionales (Control Proporcional, Integral y Derivativo).

¹⁰ CHEN C.H.. "Programmable Fuzzy Logic Device for Sequential Fuzzy Logic Synthesis." in 2001 IEEE International Fuzzy Systems Conference. 2001. pp. 107-110

¹¹ HU Y. and RAMAMOON P. "Design of Stable Recurrent Fuzzy-logic-controllers." Department of Electrical & Computer Engineering, University of Cincinnati, Ohio. 1994. pp. 918-923.

3. ANTECEDENTES

Se toma como referencia el proyecto desarrollado en la Universidad Industrial de Santander: “MODELADO, SIMULACIÓN E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL DE TENSIÓN PARA UN CONVERTIDOR BUCK BASADO EN LÓGICA DIFUSA” ¹², para entender el funcionamiento de este tipo de control no convencional y encontrar alternativas para hacerlo más eficiente de tal forma que sea posible obtener una respuesta precisa, disminuyendo el error según el criterio especificado.

Es importante mencionar que los controladores de lógica difusa son ampliamente utilizados en áreas en las cuales implementar un control convencional es difícil, ineficiente y/o costoso; por tanto, en este trabajo se presenta una alternativa que permite optimizar los parámetros del sistema de tal forma que se garantice un óptimo funcionamiento del controlador.

Se han propuesto múltiples formas de optimizar los parámetros de un controlador Fuzzy. Se pueden optimizar sus funciones de pertenencia a la entrada y salida, las ganancias y las reglas lógicas así, como optimizaciones múltiples. En este caso nos interesa optimizar las ganancias y funciones de pertenencia a la entrada y salida, dejando siempre constantes las reglas lógicas para garantizar que el sistema se comporte de forma estable y con el menor error posible.

¹² DÍAZ SAÉNZ H. D., PÉREZ MUNEVAR J.A.. Op. Cit.

4. CONVERTIDOR

En esta sección se habla del convertidor de potencia reductor DC/DC, su modelo matemático y sus principales características.

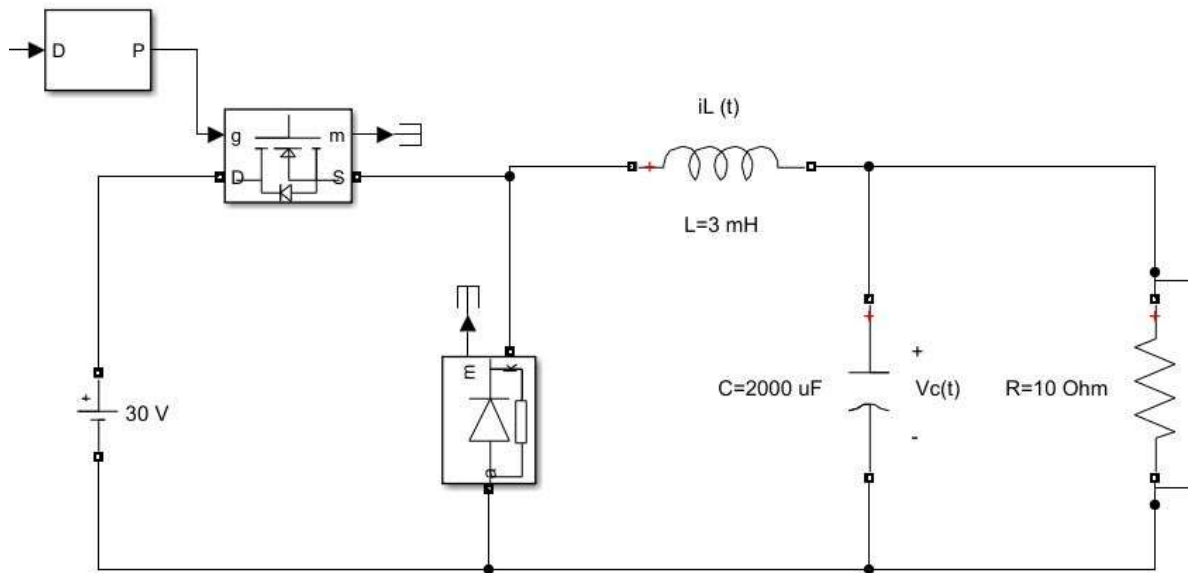
Los convertidores de potencia tienen como función principal transformar energía por medio de diferentes procesos. Teniendo en cuenta sus múltiples aplicaciones, existen diferentes tipos de configuraciones que se obtienen como combinaciones de los circuitos básicos (Reductor o Buck y elevador o Boost).

El convertidor utilizado en este análisis es el Buck, empleado en entornos en los que se necesita trabajar con un sistema que funcione con la menor cantidad de fuentes de voltaje y entregue un valor fijo a la salida menor o igual al de la entrada ($V_{out} < V_{in}$).

Este circuito cumple con esta esa función utilizando las características de almacenamiento de energía de dos elementos no lineales: Capacitor e inductor. Su funcionamiento se basa en dos estados: en el primero cuando la señal PWM está en alto (1) y la energía de la fuente se transfiere al capacitor y la carga; y el segundo, cuando el ciclo PWM está en bajo (0) y la energía se transfiere desde el inductor a la resistencia de salida para así obtener un menor voltaje de salida que de entrada (E).

En la figura 1 se muestra el esquema básico del convertidor a analizar.

Figura 1. Esquema del convertidor Buck.



Para que el convertidor funcione de manera óptima, es necesario que se cumplan algunas condiciones en el diseño del mismo. Se puede garantizar linealidad en su modelado mientras la corriente que pasa por el inductor sea continua. De igual manera, el valor del capacitor influirá directamente en el rizado de la tensión de salida, dependiendo así este valor del rizado que permita la aplicación del convertidor. Un análisis detallado se puede encontrar en ¹³. Las ecuaciones 1 y 2 relacionan la continuidad de la corriente y el rizado del voltaje de salida respectivamente con los valores de los elementos del circuito.

$$L_{min} = (1-DC)R2f \quad (1)$$

$$C = \frac{1-DC}{8LV_oV_{of}2} \quad (2)$$

DC = Ciclo útil.

¹³ HART D. W.. Op. Cit.

L_{min} = Representa el mínimo valor de la inductancia necesario para garantizar la continuidad de la corriente.

ΔV_o = Representa el rizado de la tensión de salida.

Es posible representar el sistema en el espacio de estados, como se expresa en ¹⁴, mediante el siguiente razonamiento. Definiendo el vector de estado como en la ecuación 3, se toman casos individuales que dependen del estado del interruptor. Estos casos pueden distinguirse incluyendo una nueva variable D. Cuando D vale 1, implica que el interruptor está cerrado y cuando D vale 0, significa que está abierto. Se pueden obtener así las ecuaciones de estado que se evidencian en 4 y 5.

$$x(t) = [v_C(t) \ i_L(t)] \quad (3)$$

$$\begin{bmatrix} \dot{v}_C(t) \\ \dot{i}_L(t) \end{bmatrix} = \begin{bmatrix} -\frac{1}{RC} & \frac{1}{C} & -\frac{1}{L} & 0 \end{bmatrix} \begin{bmatrix} v_C(t) \\ i_L(t) \end{bmatrix} + \begin{bmatrix} 0 & \frac{D}{L} \end{bmatrix} E \quad (4)$$

$$y(t) = [1 \ 0] \begin{bmatrix} v_C(t) \\ i_L(t) \end{bmatrix} \quad (5)$$

Esto se puede representar como función de transferencia tal como se muestra en la ecuación 6.

$$\frac{Y(s)}{U(s)} = \frac{DC}{LC\left(s^2 + \frac{s}{RC} + \frac{1}{LC}\right)} \quad (6)$$

¹⁴ OGATA K., \textit{Ingeniería de Control Moderno}. 5ta. ed. Pearson. 2010. Capítulo 2

5. CONTROLADORES

En esta sección se describen los dos controladores a implementar, tanto el PID como el *Fuzzy*.

5.1 CONTROLADOR PID

El controlador de tipo PID utiliza 3 elementos esenciales para garantizar que la señal se ajuste a la referencia de manera rápida con el menor error posible y en el menor tiempo de establecimiento estos son: Una constante proporcional, un tiempo de integración y otro de derivación, como se plantea en la ecuación 7.

$$CV = P + \frac{1}{Tis} + Td \frac{N}{1+N\frac{1}{s}} \quad (7)$$

Donde N es el coeficiente de filtrado, P la constante proporcional, I la constante integral que es inversamente proporcional al Tiempo de integración (Ti) y finalmente D que representa el Tiempo de derivación (Td).

Para el diseño de este controlador se utiliza la herramienta PID Tuner de Matlab donde a partir de la función de transferencia de la planta se calculan las constantes que garanticen un comportamiento estable en el sistema sin importar las variaciones que se presenten en cualquiera de los elementos del circuito.

Una vez calculadas las constantes del controlador descrito en la ecuación 7 se procede a simular el sistema en lazo cerrado para verificar su comportamiento ante una entrada de tipo escalón y validar la eficiencia del método.

5.2 LA LÓGICA DIFUSA Y EL CONTROLADOR FUZZY

La lógica difusa es un tipo de inteligencia artificial que permite manejar el concepto de verdades parciales, en contraste con la lógica booleana, donde los bits pueden tomar únicamente valores entre 0 y 1. Esto le brinda al sistema la posibilidad de lidiar con datos con alta incertidumbre sin contar con información numérica precisa. Lo anterior le permite a los controladores de lógica difusa resolver problemas en casos en los que no se cuenta con modelos matemáticos específicos del sistema, plantas que presentan comportamientos no lineales o incluso, problemas semánticos.

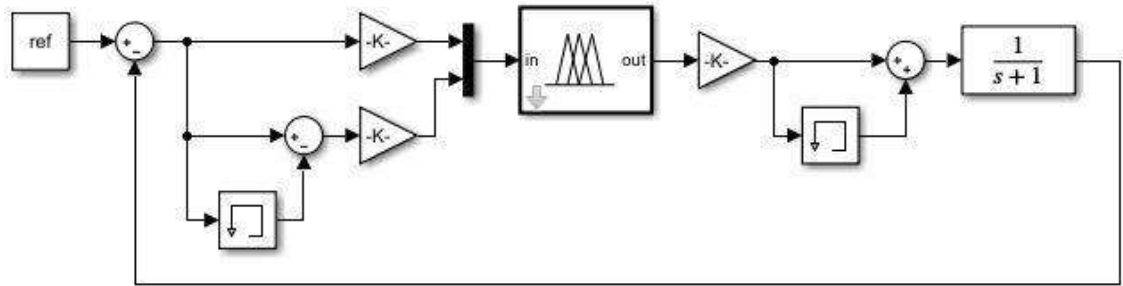
El controlador de lógica difusa o *FLC*, surge como una alternativa al control PID y consiste en utilizar la lógica difusa como método para realizar las decisiones requeridas del sistema. Este consta de cuatro partes fundamentales:

- **Funciones de pertenencia a la entrada:** consiste en una cantidad de funciones matemáticas distribuidas en el rango de -1 a 1. Cuando alguna de las entradas recibe un valor, clasifica este con base en las funciones de pertenencia. Esta clasificación se hace evaluando la función en el punto que ingresa. Generalmente a estas funciones se le asignan nombres que representan el efecto cualitativo de los valores de entrada. Ejemplos de esto son: "Flujo alto", "Presión muy baja", "Voltaje medio", etc. Se puede tener la cantidad de funciones de pertenencia a la entrada que se desee, y estas pueden tener la forma que se requiera. El paquete Fuzzy de MATLAB ofrece funciones de los siguientes tipos: triangular, trapezoidal, campana de Gauss, sigmoide, diferencia de sigmoides, producto de sigmoides, con forma de "pi", con forma de "s", y con forma de "z". Un valor de entrada puede ser clasificado en múltiples funciones de pertenencia. Las funciones de pertenencia de una entrada aplican únicamente a esta entrada.

- **Sistema de reglas lógicas:** Cuando las entradas han sido clasificadas con sus funciones de pertenencia, el sistema de reglas lógicas se encarga del procesamiento de estas. Consiste en un conjunto de condiciones para las entradas que definen las salidas. El número de reglas lógicas se calcula mediante la permutación de todas las posibles entradas. Por lo tanto, para un sistema de 5 funciones de pertenencia por entrada y dos entradas, obtenemos 25 reglas lógicas. Un ejemplo de regla lógica puede ser: "Si la presión es muy alta y la temperatura es muy alta, abrir válvula completamente". Las reglas lógicas pueden ser de tipo and o de tipo or, permitiendo múltiples combinaciones de estas.
- **Funciones de pertenencia a la salida:** Existen dos tipos de controladores Fuzzy, los cuales se definen por la forma de sus funciones de pertenencia a la salida. El primero, el fuzzy de tipo "mamdani", consiste en utilizar funciones de pertenencia a la salida de la misma forma que las funciones de pertenencia a la entrada. Por lo tanto, la salida consiste en la ponderación de las múltiples funciones de pertenencia a la salida que hayan sido seleccionadas por las reglas lógicas. El segundo tipo de Fuzzy, el "Sugeno", tiene constantes por funciones de pertenencia a la salida.
- **Elementos para la adecuación de las señales:** Teniendo en cuenta que las entradas y salidas de los sistemas de Fuzzy sólo aplican para valores en el rango de -1 a 1, es necesario que las entradas se encuentren normalizadas para el rango de interés. De igual manera, la salida tiene que normalizarse de forma que pueda causar cambios efectivos en la planta acorde a lo que estas salidas representan. Para algunas aplicaciones es necesario, además, implementar elementos adicionales a la salida, como lo pueden ser integradores, saturación, entre otros.

En Simulink, un esquema típico de simulación para un controlador Fuzzy que tome el error y la derivada del error como entradas y realice una integración a la salida, se puede apreciar en la figura 2.

Figura 2. Esquema de simulación para controlador Fuzzy.



6. MÉTODO DE OPTIMIZACIÓN

En esta sección se habla del método de optimización empleado, el cual será, como se dijo antes: Algoritmos Genéticos.

Este método consiste en aplicar el concepto de la evolución biológica en la optimización de uno o múltiples parámetros. Para realizar este método es necesario identificar plenamente el parámetro o los parámetros a optimizar, contar con un modelo que permita evaluar los impactos que tiene la variación de los valores de estos parámetros en el sistema y finalmente, una función de costo, la cual permite cuantificar el rendimiento de los mismos.

Una rutina de algoritmos genéticos típica empieza definiendo una población inicial, la cual se denomina Alpha. Esta población se selecciona generalmente de manera aleatoria, sin embargo, es necesario definir un rango para los valores que pueden tomar los diversos parámetros, y en algunos casos, es posible utilizar distribuciones de probabilidad no necesariamente uniformes, como lo puede ser una distribución de probabilidad Gaussiana, cuando se cree que los mejores resultados se encuentran en una media de datos. El tamaño de la población inicial depende de múltiples factores, como lo son: la cantidad de parámetros a optimizar, la cantidad de valores que estos pueden tomar, el tiempo de simulación tolerable, entre otros. En los casos en los que los parámetros a optimizar toman valores continuos, una buena forma de decidir el tamaño de la población inicial consiste en definir el mínimo cambio de impacto. Por ejemplo, cuando se trata de optimizar el espacio de la habitación de una casa, a pesar de que la distancia puede tomar valores continuos y, por tanto, infinitos. Cualquier valor inferior a unos pocos centímetros no tendrá un impacto considerable.

Una vez definido el tamaño de la población inicial, se evalúa el sistema con los valores que vayan surgiendo de forma aleatoria y se organiza la información de acuerdo a su criterio de relevancia, es decir, se clasifican los valores por su desempeño. Posteriormente, se selecciona una cantidad relevante de datos cuyo desempeño haya sido mayor y se le permite a esta población de datos pasar a la siguiente "generación". Llamaremos a los datos que, por su alto desempeño logren pasar a la siguiente generación, la población Beta.

Los datos que hayan resultado del filtrado anterior pueden ser combinados para producir lo que denominamos como mutaciones. Las mutaciones pueden ser de diversos tipos como cruces directos, pequeños cambios a parámetros que en su generación presentaron resultados relevantes, o realizar cambios significativos a un solo parámetro perteneciente a un conjunto de parámetros eficientes. En general, puede haber tipos de mutaciones casi ilimitados que dependen tanto de la creatividad del programador como de la utilidad que aporten los mismos al caso específico. La población que se produce debido a estas mutaciones recibirá el nombre de Gamma.

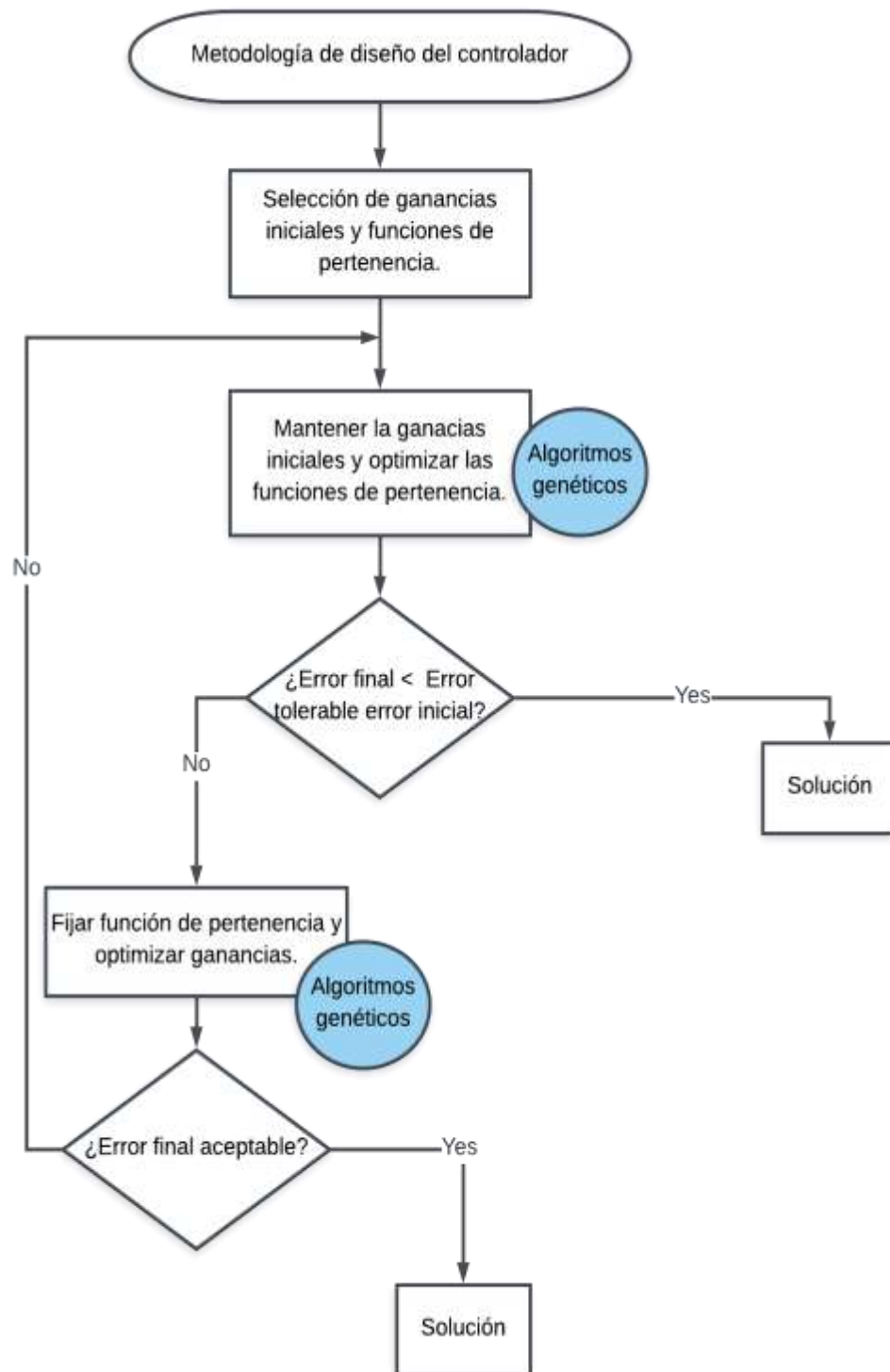
Una vez finalizadas las mutaciones, se procede a pasar a la siguiente generación, la cual requiere generar una nueva población aleatoria Alpha, mientras se mantienen los resultados de las poblaciones Beta y Gamma. Los resultados de Alpha se organizan de acuerdo al criterio o función de costo y se comparan con la población Beta para conformar una nueva población Beta. El proceso se repite la cantidad de veces que se considere necesario hasta hallar una solución con un criterio deseado. A la cantidad de veces que se repita el proceso se le denomina Generaciones.

Es importante notar que la búsqueda aleatoria que se realiza para hallar la población Alpha puede ser a su vez optimizada con el paso de las generaciones. Esto es, si se notan tendencias en los resultados que obtienen ciertos valores, puede acotarse

el rango de los valores generados de tal manera que produzca solo datos con mejores resultados. Por ejemplo, si notamos en un algoritmo genético al realizar una cantidad determinada de generaciones que los mejores resultados siempre se obtienen con valores del parámetro a optimizar superiores a alguna cantidad específica, ajustaremos la distribución de probabilidad de tal manera que produzca muchos menos valores aleatorios que se encuentren por debajo de esa cantidad. Este proceso puede hacerse automáticamente utilizando distribuciones de tipo Campana de Gauss cuya media y desviación estándar sean dinámicos y se ajusten de acuerdo a los parámetros con mejores resultados.

El algoritmo general a implementar consiste en la optimización individual de las funciones de pertenencia, seguida por la optimización de las ganancias. Ambas optimizaciones se hacen mediante los algoritmos genéticos propuestos. Este proceso se repite hasta encontrar una solución tolerable (Ver diagrama de flujo, figura 3).

Figura 3. Diagrama de flujo correspondiente al algoritmo general propuesto.



7. CASO DE ESTUDIO

Esta sección contiene los siguientes ítems: Selección de parámetros del convertidor, definición de funciones de pertenencia y ganancias iniciales del controlador *Fuzzy* y simulación del mismo bajo las condiciones dadas (incluyendo los diagramas empleados).

El método descrito en la sección anterior se usa para controlar la tensión de salida de un convertidor Buck como el que se usa en ¹⁵. Este convertidor cumple las reglas de diseño expresadas en ¹⁶, los valores de los elementos del circuito se encuentran en la tabla 1. El convertidor presentará variaciones en la tensión de referencia y en la resistencia de carga. Como punto de referencia inicial para el sistema de lógica difusa, se utilizarán las funciones de pertenencia expresadas en ¹⁷[13] y el esquema de control que se muestra en la figura 4.

Tabla 1. Valores del convertidor

Elemento	Valor	Unidad
R	10	Ω
L	3	mH
C	2000	μF
E	30	V
f	15	kHz

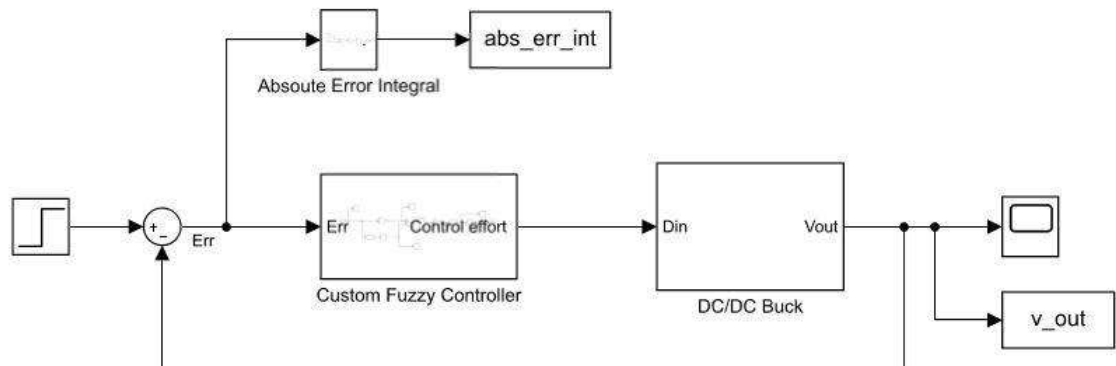
Fuente: SWATHY K, SHRUTIKA JANTRE MS., YOGITA JADHAV Ms., SUSHIL Mr LABDE. M., PRATIK KADAM Mr.. "Design and Hardware Implementation of Closed Loop Buck Converter Using Fuzzy Logic Controller.", Proceedings of the 2nd International conference on Electronics, Communication and Aerospace Technology.ICECA. 2018.pp. 175-180

¹⁵ SWATHY K, SHRUTIKA JANTRE MS., YOGITA JADHAV Ms., SUSHIL Mr LABDE. M., PRATIK KADAM Mr..Op. Cit

¹⁶ HART D. W.. Op. Cit

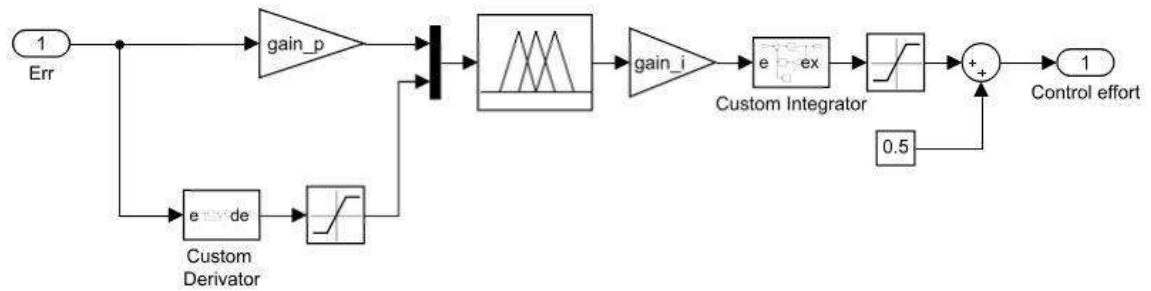
¹⁷ SWATHY K, SHRUTIKA JANTRE MS., YOGITA JADHAV Ms., SUSHIL Mr LABDE. M., PRATIK KADAM Mr..Op. Cit

Figura 4. Esquema de control utilizado.



La implementación del controlador Fuzzy y del convertidor utilizado se pueden ver en las figuras 5 y 6 respectivamente. Los bloques *custom derivator* y *custom integrator* se aprecian en las figuras 7 y 8 respectivamente ¹⁸.

Figura 5. Controlador Fuzzy utilizado.



¹⁸ DÍAZ SAÉNZ H. D., PÉREZ MUNEVAR J.A.. Op. Cit.

Figura 6. Convertidor Buck utilizado.

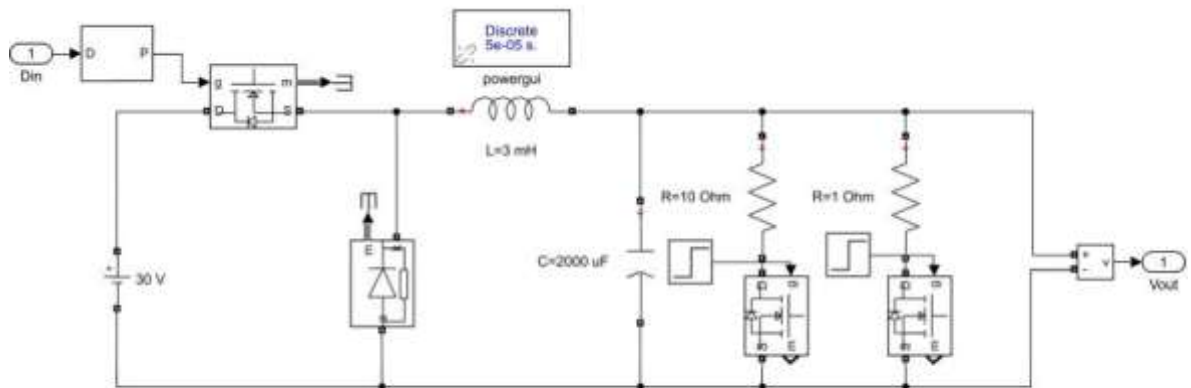


Figura 7. Derivador numérico.

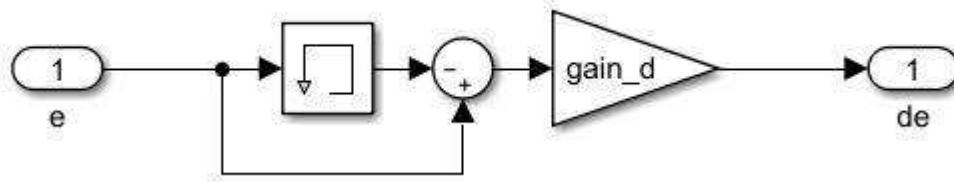
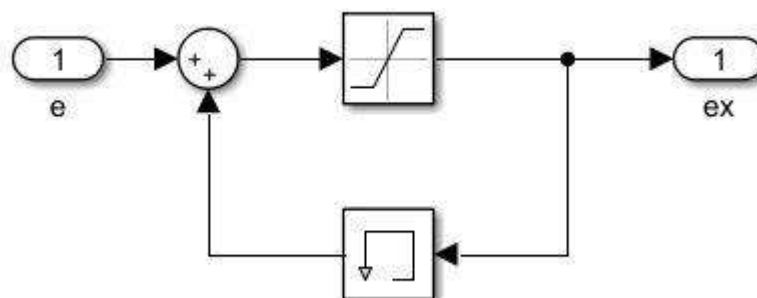


Figura 8. Integrador numérico.



Para la definición de las funciones de pertenencia es necesario tener en cuenta que estas representan diferentes escalas de los valores de las entradas y las salidas del sistema; la convención a utilizar es:

- NB significa *Negative Big*, es decir, valor fuertemente negativo o alto negativo.
- NM significa *Negative Medium*, es decir, valor negativo medio.
- NS significa *Negative Small*, es decir, valor débilmente negativo o bajo negativo.
- ZE significa *Zero*, es decir, cero o valor neutral.
- PS significa *Positive Small*, es decir, valor débilmente positivo o bajo positivo.
- PM significa *Positive Medium*, es decir, valor positivo medio.
- PB significa *Positive Big*, es decir, valor fuertemente positivo o alto positivo.

La salida entonces depende de los valores mencionados anteriormente para el error y su derivada. Por ejemplo, si el error es NB, significa que la salida del sistema es mucho mayor que la referencia, por lo que nos interesaría hacer que esta baje fuertemente a un nivel cercano a la referencia. Sin embargo, si a la vez el cambio en el error es PB, significa que este cambio en el error ya se está corrigiendo y por lo tanto no es necesario que la salida sea un cambio fuerte, es decir, se espera que la salida sea ZE. Esta regla lógica puede expresarse como: Si E es NB y EE es PB, entonces OUT es Z. Las reglas lógicas se presentan en la tabla 2. En esta, la primera fila representa la entrada E, mientras que la primera columna representa la entrada EE. Las casillas en color blanco corresponden a los valores de la salida. Las reglas lógicas presentan una simetría diagonal y, puesto que son ellas quienes mantienen el sentido de los valores de la entrada ¹⁹, no serán modificadas en el presente trabajo. Sin embargo, la optimización de las reglas lógicas mediante algoritmos genéticos podría ser un interesante objeto de estudio para una continuación de este trabajo.

Tabla 2. Reglas Lógicas para el controlador

ee \ e	NB	NM	NS	Z	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZE
NM	NB	NB	NB	NM	NS	ZE	PS

¹⁹ SWATHY K, SHRUTIKA JANTRE MS., YOGITA JADHAV Ms., SUSHIL Mr LABDE. M., PRATIK KADAM Mr.. Op. Cit.

ee e	NB	NM	NS	Z	PS	PM	PB
NS	NB	NB	NM	NS	ZE	PS	PM
Z	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PM	PB	PB
PM	NS	ZE	PS	PM	PB	PB	PB
PB	ZE	PS	PM	PB	PB	PB	PB

Fuente: SWATHY K, SHRUTIKA JANTRE MS., YOGITA JADHAV Ms., SUSHIL Mr LABDE. M., PRATIK KADAM Mr.. "Design and Hardware Implementation of Closed Loop Buck Converter Using Fuzzy Logic Controller.", Proceedings of the 2nd International conference on Electronics, Communication and Aerospace Technology.ICECA. 2018.pp. 175-180

De acuerdo a lo anterior, se utilizan las funciones de pertenencia que se muestran en las figuras 9, 10 y 11, y las ganancias la tabla 3. Ante estas condiciones se obtiene una integral del error absoluto de 0.0288 para un cambio en la tensión de referencia de 15[V] a 22[V], junto con la gráfica que se muestra en la figura 12.

Figura 9. Funciones de pertenencia iniciales para el error (E).

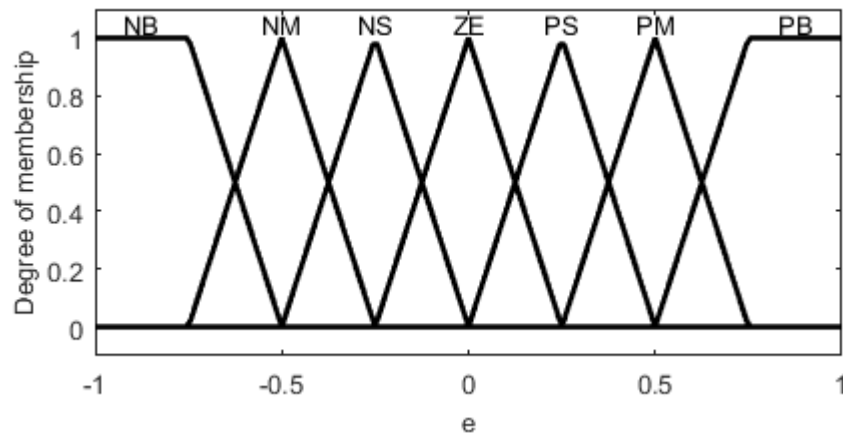


Figura 10. Funciones de pertenencia iniciales para la derivada del error (EE).

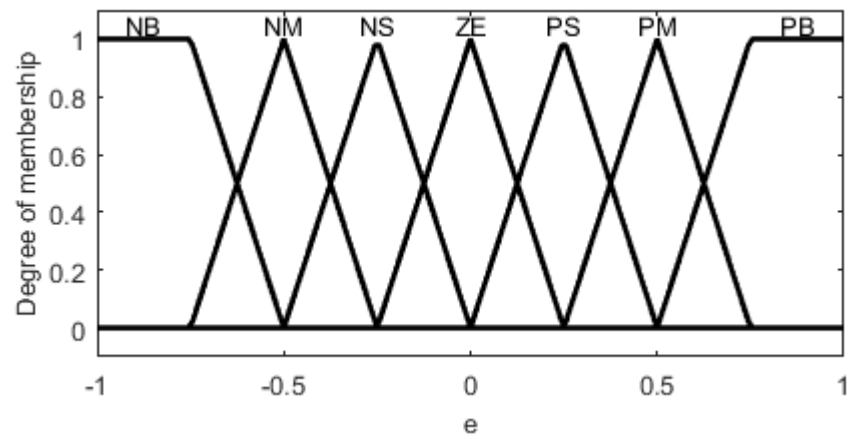


Figura 11. Funciones de pertenencia iniciales la salida (OUT).

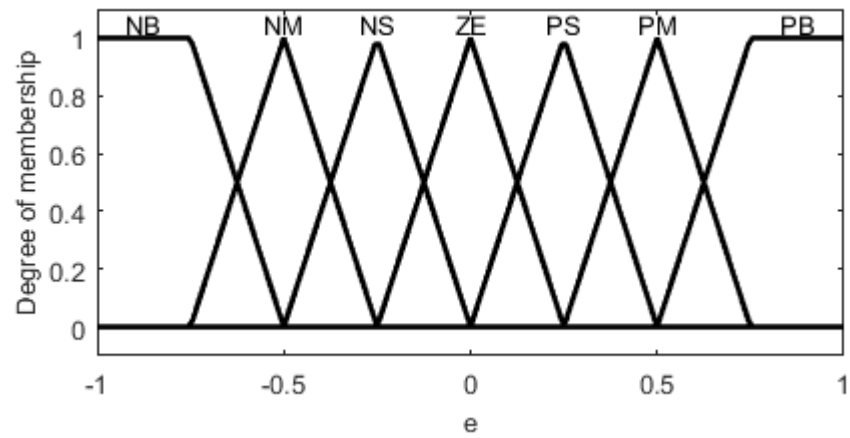
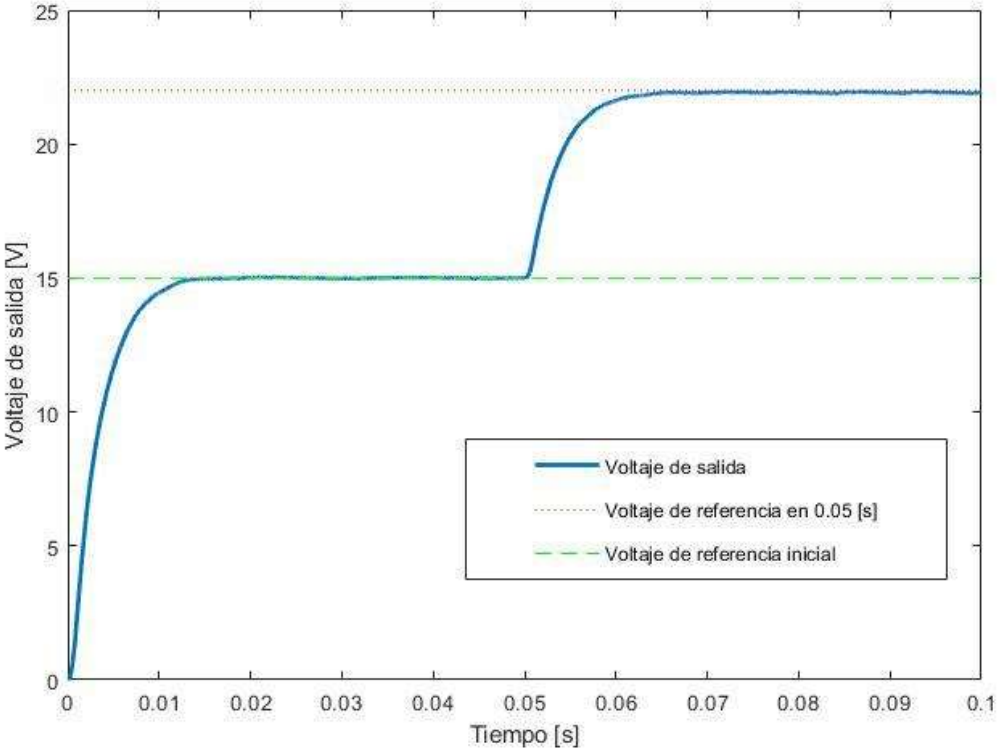


Tabla 3. Valores de las ganancias.

Ganancia	Valor
Error	0.05
Derivada del error	3.333
Salida (integración)	100

Figura 12. Gráfica para tensión de salida ante cambio en la tensión de referencia.



8. IMPLEMENTACIÓN

En esta sección se describe el algoritmo utilizado para la optimización del controlador *Fuzzy*.

El algoritmo principal que se presenta en la figura 3 (Diagrama de flujo), se implementó en un script de MATLAB utilizando un ciclo *while* que ejecuta el programa hasta encontrar dicha solución.

8.1. CONTROLADOR FUZZY

La variación de las funciones de pertenencia implica variar la posición de las mismas, la cual se representa como los argumentos de cada función individual; por lo tanto, esto implica la variación de 57 parámetros (el error, la derivada del error y la salida, constan de 7 funciones de pertenencia cada uno. De esas 7 funciones de pertenencia, 5 son triángulos, que se definen mediante 3 parámetros, y 2 son trapecios, que se definen utilizando 2 parámetros). Teniendo en cuenta que estas funciones tienen significados, como, por ejemplo, alto positivo, las funciones de pertenencia deben mantenerse relativamente cerca de sus valores iniciales, con el fin de no perder estos significados. Es decir, no es aceptable que los valores de medio negativo sean más positivos que los valores de medio positivo. Lo anterior garantiza que las reglas lógicas asociadas a cada una de estas funciones seguirán manteniendo el sentido original.

Para solucionar este problema se utiliza una distribución de probabilidad de tipo Campana de Gauss que genera valores aleatorios para cada función de pertenencia individual con base en el valor central de la función original. Esto significa que,

puesto que en las funciones de pertenencia iniciales el centro de ZE es cero (tanto en ambas entradas como en la salida), los valores aleatorios para ZE tendrán una alta probabilidad de salir en las vecindades de cero. Este parámetro se representa en la media de la distribución Gaussiana; el otro parámetro es la desviación estándar, la cual nos indica qué tan lejos de la media se encuentran los valores que aún tienen una alta probabilidad de aparecer. Este parámetro nos representa justamente la capacidad de mutar de las funciones, por lo que, si le permitimos ser un parámetro dinámico, podemos hacer que las funciones muten más en caso de que los resultados se encuentren estancados. Esto se discutirá de manera más formal más adelante, junto con los dos conceptos que se presentan a continuación.\

Debido a la gran cantidad de parámetros a optimizar, se buscó una forma de acortar la búsqueda aleatoria en la mayor cantidad posible. Una de estas ideas fue la noción de que un conjunto de funciones de pertenencia simétricas con respecto al cero tendría un mejor desempeño. De ser esto cierto, la cantidad de parámetros se vería reducida a la mitad, puesto que solo sería necesario generar una mitad de las funciones de pertenencia ya que la otra sería exactamente igual. Sin embargo, no se encontró ninguna evidencia teórica concluyente sobre porqué esto habría de ser cierto. No obstante, no regular la simetría de las funciones en la generación aleatoria de las mismas, causaría que el mayor porcentaje de estas fuese fuertemente asimétrico (la probabilidad de que un conjunto de 7 funciones de pertenencia generado aleatoriamente mediante probabilidad gaussiana con una desviación estándar de 0.25 sea simétrico, es apenas del $1.28e-11\%$ para uno de los casos más ideales). Otra propuesta para reducir el número de parámetros a optimizar consiste en centrar las funciones de pertenencia, lo que significa que los cortes con el eje x de las funciones de pertenencia triangulares siempre caen en el centro de su función vecina. Esto implica que sólo se genera un parámetro que corresponde al punto central de cada triángulo, y los demás parámetros quedan definidos inmediatamente, lo cual reduce en un tercio la cantidad de posibles combinaciones. Nuevamente, no se encontraron evidencias teóricas de que esto fuese

necesariamente cierto, y de igual manera, permitir una generación puramente aleatoria de parámetros para las funciones de pertenencia implicaría que un gran porcentaje de las mismas no sería centrada.

Con el fin de corregir estos problemas, se proponen los factores de simetría y de centrado. Estos factores son pensados de tal manera que representen numéricamente (del 0 al 1) qué tan simétrico o qué tan centrado es un grupo de funciones de pertenencia, en donde el cero representa asimetría total (en el caso del factor de centrado) o falta de centrado total (para el caso del factor de centrado), y, por lo tanto, 1 representaría simetría absoluta o centrado absoluto, según corresponda.

Estos factores se implementan corrigiendo un conjunto de funciones de pertenencia previamente generados de manera aleatoria. Esto implica la suposición de que este conjunto será asimétrico y muy poco centrado o nulamente centrado, suposición que, es estadísticamente válida para fines prácticos.

Para el caso del factor de simetría, su implementación consiste en tomar los pares simétricos (ej: PM y NM) y promediar sus componentes (el valor absoluto de los componentes negativos sumado a los componentes positivos y multiplicado por 0.5), obteniendo así una función promedio. Las nuevas funciones PM y NM serán entonces ponderaciones entre su valor anterior y el promedio, cuyos pesos son uno menos el factor de simetría y el factor de simetría respectivamente. Matemáticamente podemos expresar esto como en la ecuación 8.

$$MF_{avg} = (MF(+) + |MF(-)|) \cdot 0,5$$

$$MF(+) = MF(+)_old \cdot (1 - SF) + MF_{avg} \cdot SF$$

$$(8) MF(-) = MF(+)_old \cdot (1 - SF) - MF_{avg} \cdot SF$$

Sin embargo, en el caso de las funciones de pertenencia ZE, su par simétrico con respecto al valor central de la función de pertenencia que lo describe es cero. Por lo tanto, el promedio de este valor siempre será cero.

Esto implica que un factor de simetría de cero no alterará las funciones, mientras que un factor de simetría de uno hará que cada par simétrico sea exactamente igual al promedio entre ambos, haciéndolos iguales y por lo tanto simétricos. Sin embargo, es importante resaltar que el factor de simetría (SF) puede tomar cualquier valor entre 0 y 1, lo cual permite incorporar el concepto de simetría parcial o relativa.

La implementación del factor de centrado por su lado consiste en hacer que los puntos laterales de las funciones de pertenencia triangulares se asemejen a los puntos centrales de sus funciones vecinas, y la cantidad que define qué tanto se asemejan relativamente es el factor de centrado (CF). Matemáticamente, podemos expresar esto como se muestra en la ecuación 9.

$$MF_i(1) = MF_i(1)_{old} \cdot (1 - CF) + MF_{i-1}(2) \cdot CF \quad MF_i(3) = MF_i(3)_{old} \cdot (1 - CF) + MF_{i+1}(2) \cdot CF \quad (9)$$

Teniendo en cuenta que representamos una función de pertenencia triangular por 3 puntos, en los cuales MF(1) representa el corte del triángulo con cero por la izquierda, MF(2) el punto central y MF(3) el corte del triángulo con el eje cero por la derecha. Esto implica que cuanto mayor sea el factor de centrado, más cerca estarán los puntos laterales de los triángulos, de los puntos centrales de sus vecinos. Nuevamente, este factor puede tomar cualquier valor entre cero y uno.

Permitir la variación dinámica de estos parámetros (en conjunto con la desviación estándar que nos proporciona la velocidad de mutación), nos permite estudiar las hipótesis de qué tanto afectan la simetría y el centrado de las funciones de pertenencia al desempeño de las mismas durante la ejecución de los algoritmos

genéticos. Podemos valernos de los mismos algoritmos genéticos para guiar los valores de estos factores hacia aquellos que nos brinden un mejor criterio, manteniendo, sin embargo, una capacidad de mutar en los mismos. Lo anterior podemos realizarlo si permitimos que los factores sean dados por distribuciones de Gauss, al igual que en la variación de los parámetros de las funciones de pertenencia, cuya media es nuestro parámetro dinámico optimizable y la desviación estándar es, nuevamente, la velocidad de mutación.

Permitiendo una velocidad de mutación elevada, como lo es una desviación estándar de 0.5, y poniendo la media de nuestros factores de centrado y simetría en 0.5, cubriremos con una densidad de probabilidad casi uniforme el rango de 0 a 1 para la primera generación. Al finalizar la primera simulación, podemos calcular la media y la desviación estándar para los factores de los conjuntos de funciones de pertenencia que obtuvieron un mejor desempeño. Estos parámetros pueden reemplazar la media y la desviación estándar seleccionada inicialmente para estos factores, restringiendo los posibles valores hacia aquellos que proveen mejores resultados. Nótese que en caso de que nuestras hipótesis sean inválidas, la media se alejará de 1 para el factor para el que lo anterior sea cierto, y la desviación estándar será grande. Si la hipótesis es válida, la mayoría de los conjuntos de funciones de pertenencia tenderán hacia ser simétricos o centrados según sea el caso y, por lo tanto, reduciremos fuertemente la cantidad de parámetros a optimizar (se reducirán en un 94.12% para el caso en el que ambas hipótesis sean ciertas).

De igual manera, permitiendo la variación dinámica de la velocidad de mutación de los parámetros de las funciones de pertenencia, y optimizándola a cada generación, podemos comprobar qué tanto se ve afectado el desempeño de estas funciones cuando las funciones originales se alejan de su media que, como se explicó antes, implica también su significado ante las reglas lógicas. Nótese que en caso de que los mejores resultados se obtengan al alejar a las funciones más bien poco con

respecto a su media, el conjunto de posibilidades para la optimización se verá, de igual manera, fuertemente reducido.

Se seleccionó una población inicial de 50 simulaciones aleatorias bajo los parámetros que fueron previamente especificados, sobre un total de 50 generaciones para la optimización de las funciones de pertenencia. Esto da como resultado un total de 2500 simulaciones. Por cada generación, se selecciona a la mejor mitad como la población Beta.

Para la población Gamma, se utilizan dos tipos de mutaciones. La primera mutación recibe tres grupos de funciones de pertenencia y combina las funciones de pertenencia de error del primero, con las funciones de pertenencia de la derivada del error del segundo y las funciones de pertenencia a la salida del tercero. La segunda mutación recibe dos conjuntos de funciones de pertenencia y crea un promedio entre estas, es decir, promedia los parámetros de las funciones de pertenencia del error para crear una nueva, hace lo mismo con la derivada del error y la salida. Por cada generación se producen 10 mutaciones, y los grupos de funciones de pertenencia a mutar se seleccionan aleatoriamente (distribución de probabilidad uniforme) entre la mejor mitad.

Para modificar los parámetros de las funciones, se accede directamente al objeto FIS que crea MATLAB. La ruta para modificar los parámetros específicos depende de si es una entrada (y qué entrada es) o salida, y la función de pertenencia a la cual pertenece ese parámetro específico. Por ejemplo, si quisiéramos modificar el valor central de la función de pertenencia correspondiente a PS para la derivada del error, la ruta a seguir sería, (nombre del objeto FIS). Inputs(2). Membership Functions(4). Parameters(2). Para las funciones triangulares, los parámetros deben ir en orden ascendente; en caso de no hacer esto, el programa genera un error. Para corregir esto en la generación aleatoria basta con usar la función sort de MATLAB para garantizar que siempre queden en orden ascendente.

En caso de que la optimización de las funciones de pertenencia no brinde un resultado satisfactorio, se procede a utilizar los algoritmos genéticos para optimizar las ganancias en las entradas y las salidas. Empíricamente, se sabe que la variación de estos parámetros puede resultar con un impacto igual o mayor en el desempeño del sistema que la variación de las funciones de pertenencia; sin embargo, el rango de estas no posee una cota explícita, por lo que dificulta el rango de valores aleatorios a generar en la primera generación.

Para abordar este problema, se implementó un barrido en escalas de diez sobre los valores iniciales, los cuales brindan una buena aproximación original. Este barrido se realiza como se muestra en la ecuación 10, en donde G representa la ganancia a generar aleatoriamente, G_{avg} representa la media de las ganancias con mejores resultados, y x representa un valor aleatorio (distribución de probabilidad uniforme).

$$G = G_{avg} \cdot 10^{-1+2x} \quad (10)$$

Empíricamente, se observó que el valor de la constante de integración, es decir la ganancia de salida, tiene cierta dependencia con el valor de la ganancia correspondiente a la derivada del error, por lo que se procede a mantener fija una (en este caso será la ganancia de salida), reduciendo adicionalmente las variables a optimizar.

9. RESULTADOS

En esta sección se presentan los resultados obtenidos, teniendo en cuenta las variaciones tanto en la tensión de entrada como en la resistencia de carga, junto con una comparación de la respuesta del sistema a cada controlador.

Al ejecutar el código planteado en la sección anterior, para una variación de la tensión de referencia de 15[V] a 22 [V] que ocurre a los 0.05[s] de simulación, se obtuvieron las funciones de pertenencia que se evidencian en las figuras 13,14 y 15.

Figura 13. Funciones de pertenencia optimizadas para el error.

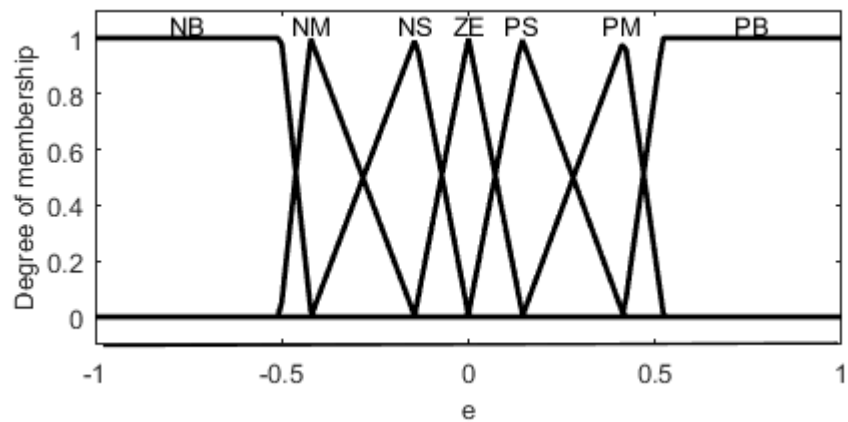


Figura 14. Funciones de pertenencia optimizadas para la derivada del error.

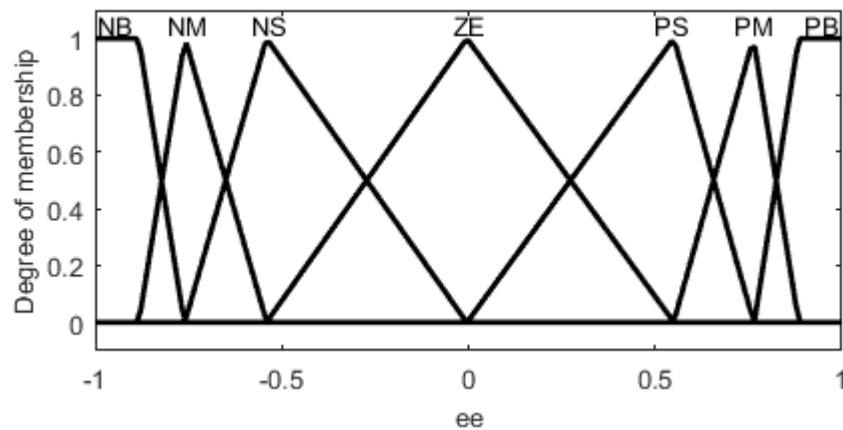
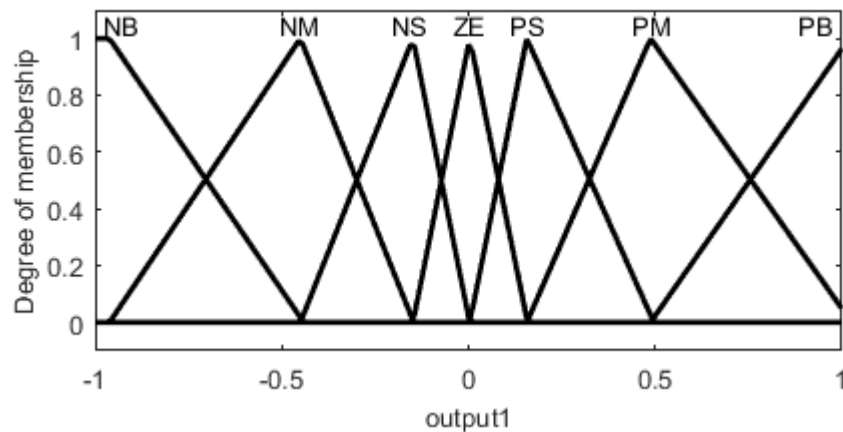
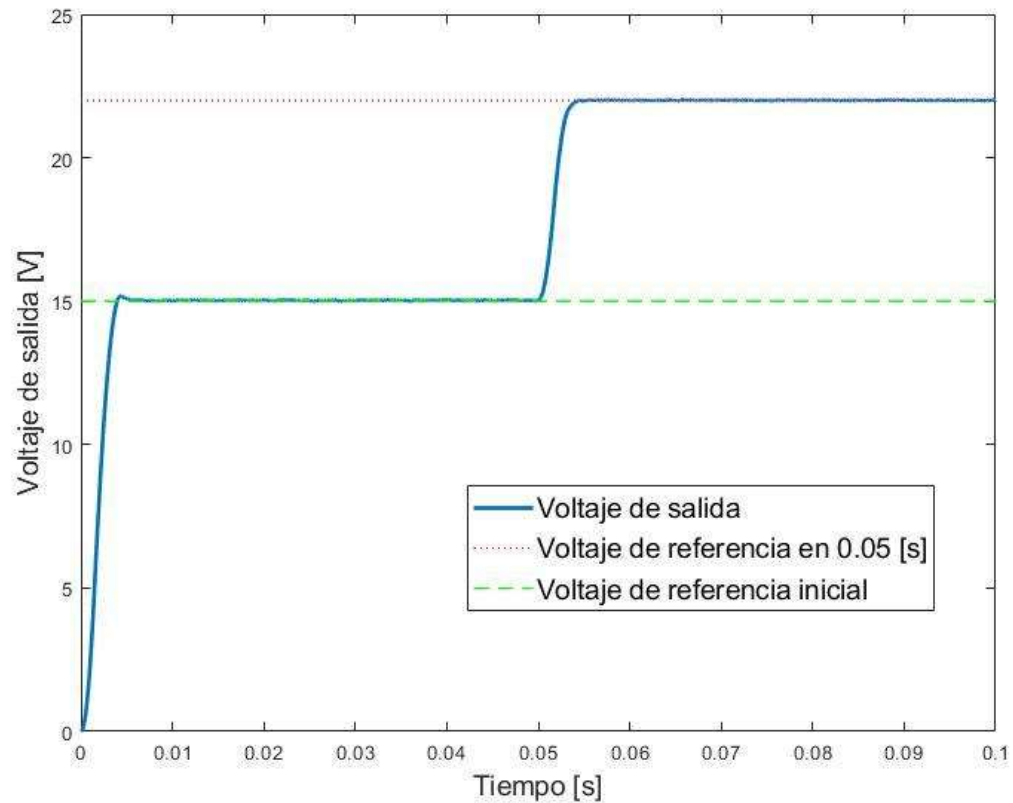


Figura 15. Funciones de pertenencia optimizadas para la salida del controlador.



Con estas funciones de pertenencia, la integral del error absoluto del sistema de control fue de 0.0132, lo cual representa una mejora del 54,1667% con respecto a las funciones de pertenencia iniciales. Estas funciones de pertenencia tienen un factor de simetría del 87% lo cual representa una simetría relativamente alta; sin embargo, no total como se puede evidenciar a la salida, en la cual los trapecios para alto positivo y alto negativo tienen valores considerablemente diferentes. El factor de centrado obtenido fue del 97.56%, lo cual se puede considerar totalmente centrado en la práctica. Los resultados de simulación se presentan en la figura 16.

Figura 16. Resultados de simulación para funciones de pertenencia optimizadas.



En las figuras 17 y 18 se puede apreciar la variación de los factores de centrado y simetría respectivamente, recordando que la media de estos (así como su variación), se acercan a aquellos valores que brinden los mejores resultados. Claramente puede verse que, con el paso de las simulaciones, la media de estos valores tiende rápidamente hacia valores elevados (alrededor de 0.85), lo cual valida la hipótesis sobre la efectividad de las funciones de pertenencia centralizadas y simétricas planteada anteriormente.

Figura 17. Evolución del factor de centrado.

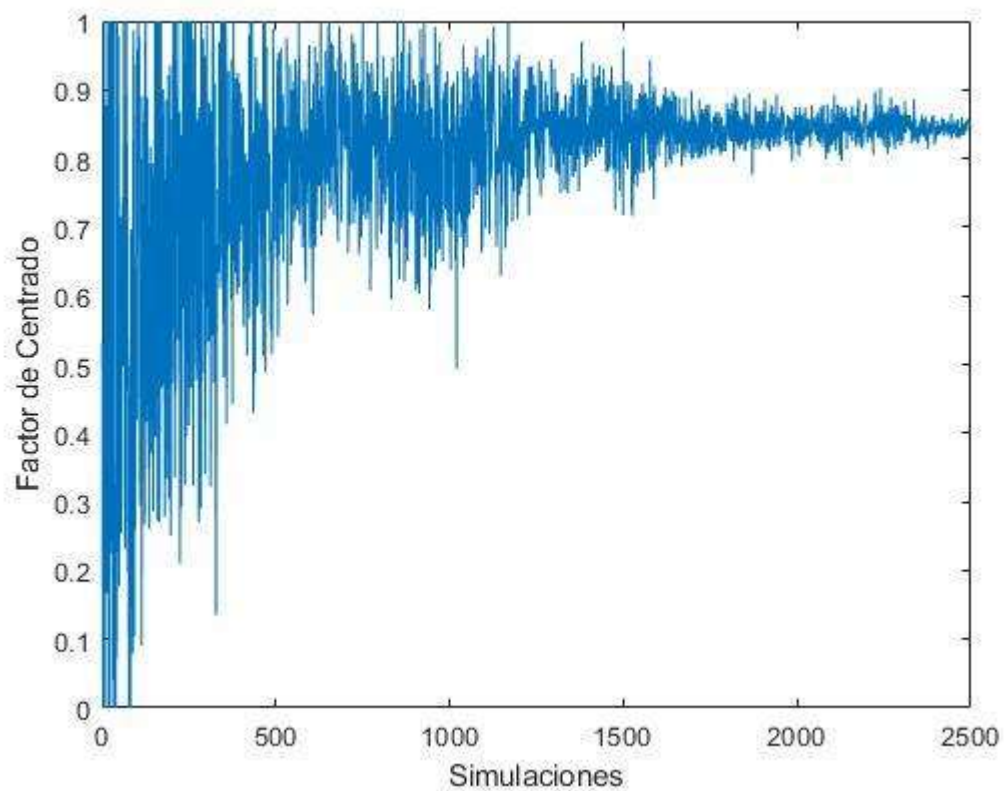
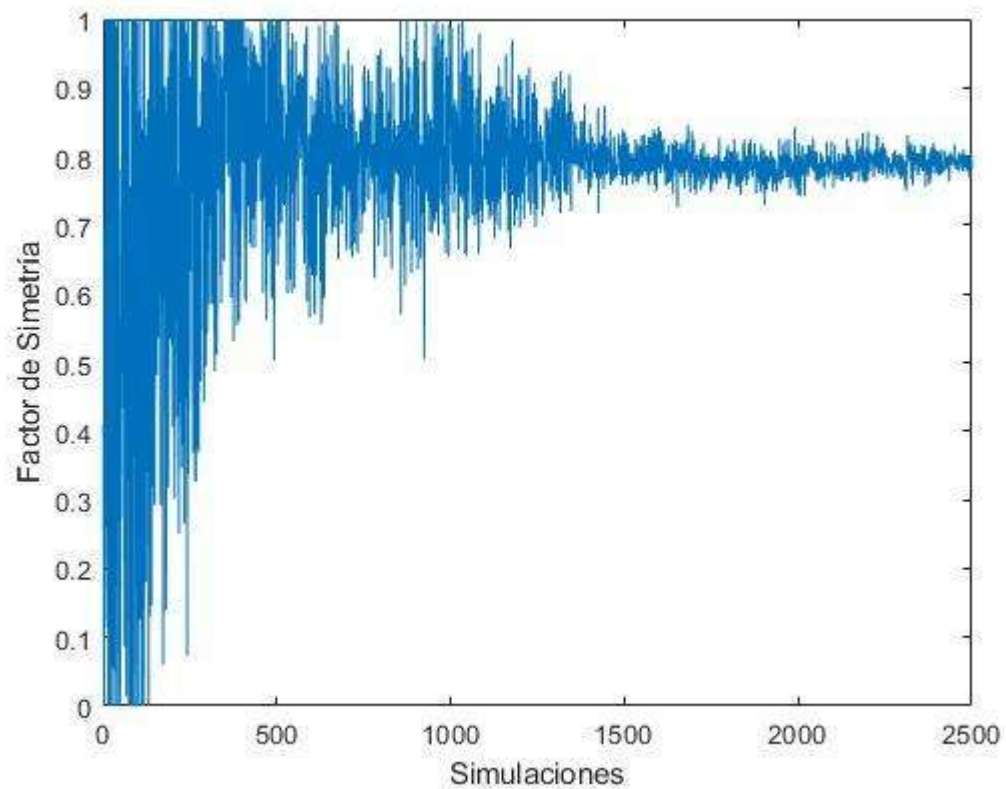


Figura 18. Evolución del factor de simetría.



Las funciones de pertenencia anteriores se obtuvieron durante el primer ciclo de optimización, por lo tanto, fue necesario volver a correr el código (ahora con la nueva referencia) para ver si, o bien se encontraban nuevas funciones de pertenencia que mejoraran los criterios especificados, o bien se encontraban ganancias que permitieran mejorar los resultados obtenidos. Al cabo de 2 ciclos de optimización de ganancias, se encontró que las ganancias que se muestran en la tabla 4 le proporcionan al sistema una integral del error absoluto de 0.0126, el cual representa una mejora del 56,25% con respecto al error original, y del 4,54% con respecto a las ganancias originales para funciones de pertenencia optimizadas.

Tabla 4. Valores de las ganancias optimizados para variación de la tensión de referencia.

Ganancia	Valor
Error	0.04
Derivada del error	2.5
Salida (integración)	100

Posteriormente, se ejecutó el código para una variación en la resistencia de carga. El valor inicial para la resistencia es de $10\ \Omega$ y el valor final de la misma es de Ω , y la variación ocurre a los 0.05 [s]. Con estas condiciones, se encontró que el conjunto de funciones de pertenencia que se evidencia en las figuras 19, 20 y 21, obtuvo una integral del valor absoluto de 0.009. Este resultado se obtuvo a los 6015 ciclos de simulación, y nuevamente, los factores de simetría y de centrado evolucionaron con tendencia hacia 1, siendo de 1 y 0.9751 respectivamente para el resultado obtenido. En cuanto a la optimización de ganancias, tras varios ciclos de optimización, el resultado no mejoró ante la optimización de las ganancias.

Figura 19. Funciones de pertenencia optimizadas para el error (variación de resistencia de carga).

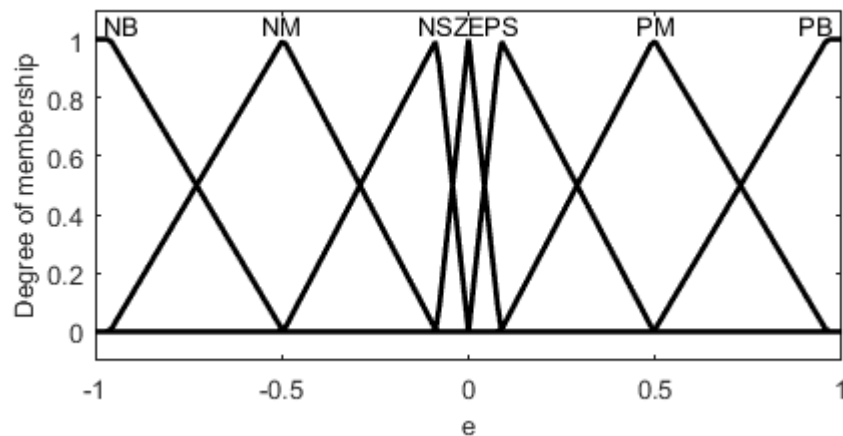


Figura 20. Funciones de pertenencia optimizadas para la derivada del error (variación de resistencia de carga).

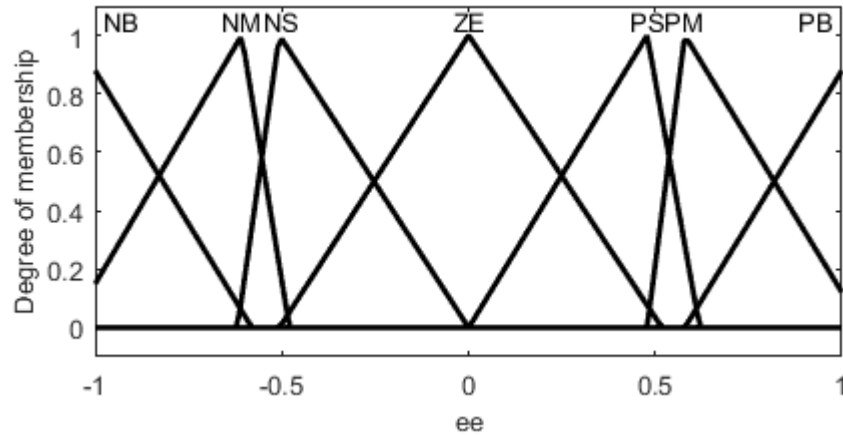
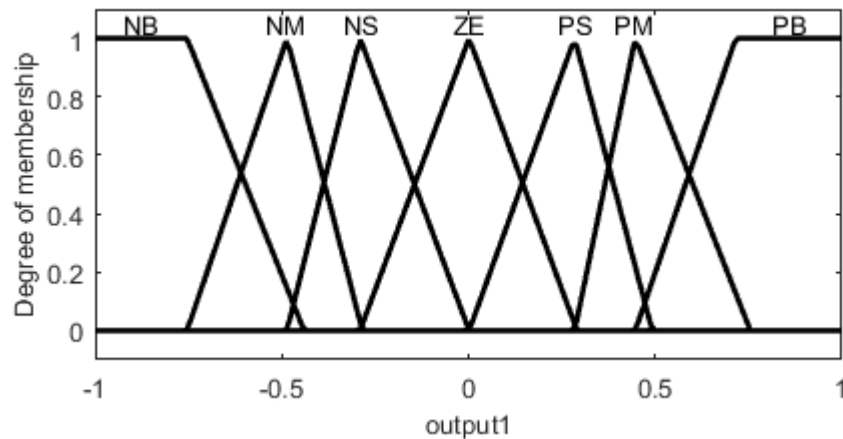


Figura 21. Funciones de pertenencia optimizadas para la salida (variación de resistencia de carga).



Finalmente, se toman los parámetros resultantes del proceso de optimización ante una variación en la tensión de entrada y en la resistencia de carga.

En primer lugar, se toman los parámetros obtenidos variando la resistencia de carga y se ejecutan para la variación en la tensión de referencia de lo que se obtiene un error absoluto de 0,0134. De igual forma, se hace el experimento contrario

obteniendo una integral del error absoluto de 0,0111. Comparando los resultados pertinentes a cada escenario, es posible concluir que los parámetros obtenidos para variación de la resistencia de carga proveen un mejor desempeño global.

9.1. CONTROLADOR PID:

Para el diseño del controlador PID descrito en la sección 2 se utilizó la herramienta PID TUNER de MATLAB, la cual se encarga de tomar la función de transferencia de la planta (convertidor Buck) y sintonizarla con las constantes adecuadas para garantizar que el sistema funcione de forma óptima.

Para el caso del convertidor Buck se obtuvieron las siguientes constantes (Tabla 5) que serán reemplazadas en la ecuación 7.

Tabla 5. Parámetros del controlador PID.

Constante	Valor
P	2.578
I	826.402
D	1.195e-3
N	16416.2263

Una vez obtenidas las constantes del controlador PID, se hace una simulación en lazo cerrado del sistema planteado en la figura 22, para verificar el comportamiento del sistema ante una variación en la referencia. Este resultado se puede observar en la figura 23.

Figura 22. Diagrama de bloques del sistema usando controlador PID.

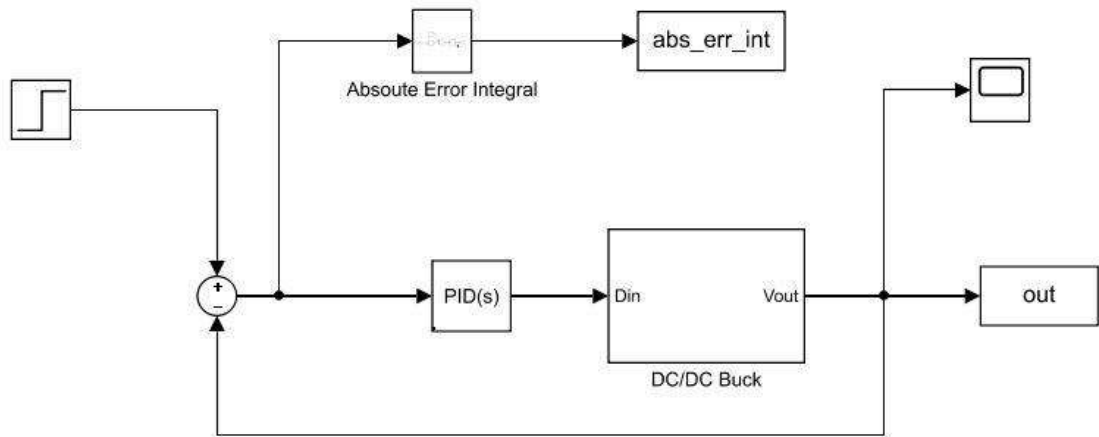
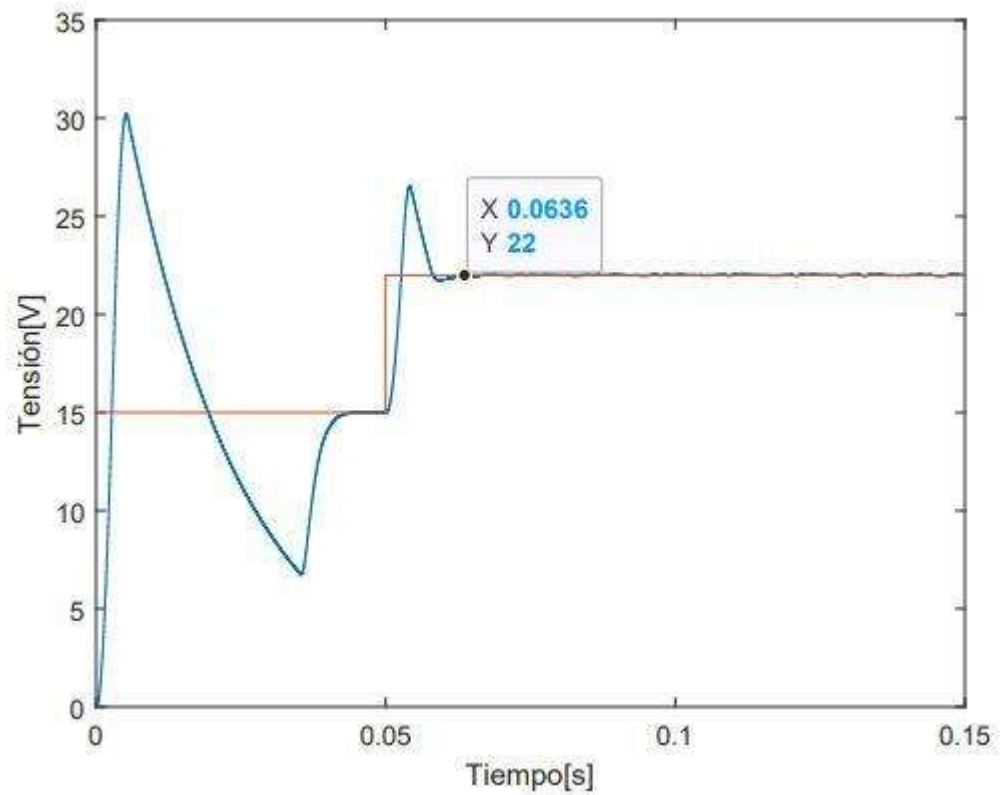
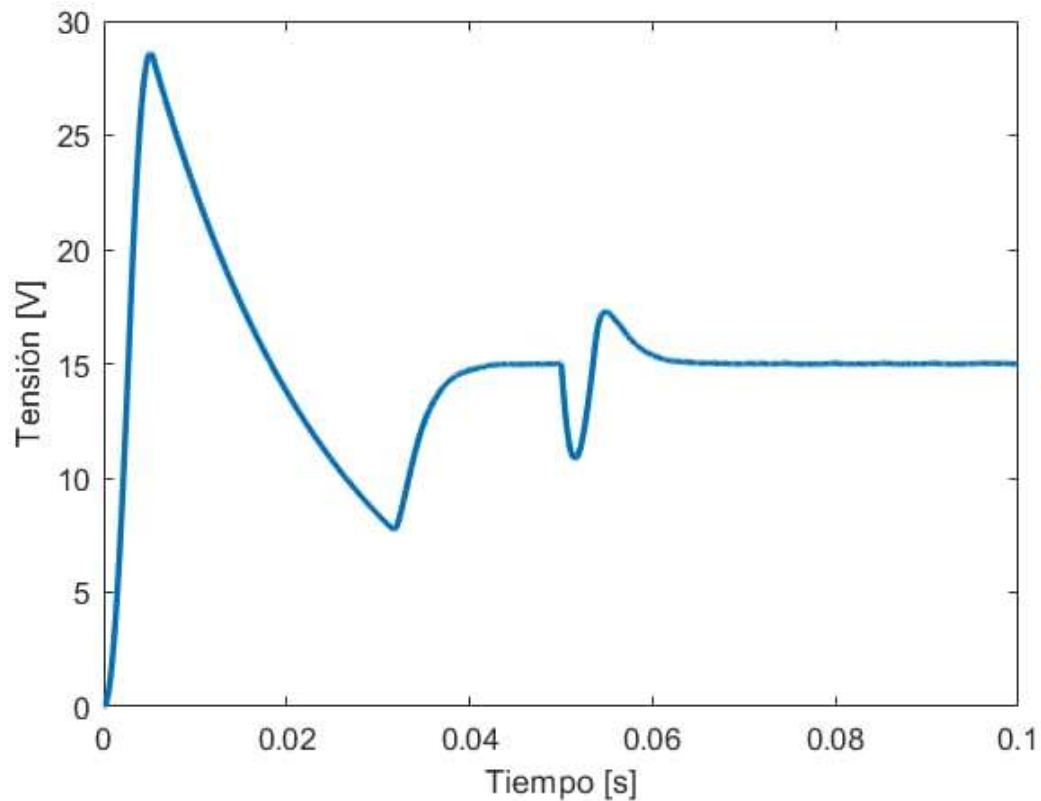


Figura 23. Salida usando controlador PID.



Finalmente, se propone una variación en la resistencia de carga para verificar el comportamiento del sistema y la eficiencia del controlador PID ante este cambio. Los resultados se muestran en la figura 24.

Figura 24. Variación de la resistencia de carga usando controlador PID.



9.2 COMPARACIÓN DE LOS RESULTADOS:

Para verificar la eficiencia de los controladores diseñados se tomarán como referencia dos parámetros que son: la integral del error absoluto (Cuyo diagrama de bloques se muestra en la figura 25) y el tiempo de establecimiento. En las tablas 6,7 y 8 se presenta la comparación de los resultados obtenidos.

Según estos resultados es posible concluir que para este sistema (Convertidor Buck) el controlador Fuzzy optimizado con algoritmos genéticos presenta mejores resultados que el controlador de lógica difusa estándar (sin optimizar) y un control PID, especialmente ante variaciones en la resistencia de carga, lo que lo convierte en una alternativa más eficiente estabilizar este circuito.

Figura 25. Cálculo de la integral del error absoluto.

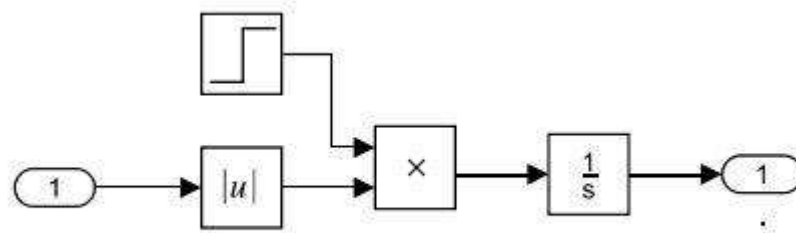


Tabla 6. Comparación de resultados del controlador Fuzzy optimizado y sin optimizar.

Tipo de controlador	Integral del error absoluto	Tiempo de establecimiento
Fuzzy optimizado	0.0126	0.05442[s]
Fuzzy estándar	0.02	0.0635[s]

Tabla 7. Resultados ante una variación en la tensión de entrada.

Tipo de controlador	Integral del error absoluto	Tiempo de establecimiento
Fuzzy optimizado	0.0126	0.05442[s]
PID	0.277	0.0636[s]

Tabla 8. Resultados ante una variación en la resistencia de carga.

Tipo de controlador	Integral del error absoluto	Tiempo de establecimiento
Fuzzy optimizado	0.009	0.054[s]
PID	0.0634	0.0704[s]

10. CONCLUSIONES

En este trabajo de grado se presentó un algoritmo para la optimización de múltiples parámetros de un controlador de Lógica Difusa basado en algoritmos genéticos. Este algoritmo encontró conjuntos de funciones de pertenencia y ganancias que le permitieron superar en eficiencia (siendo el criterio de esta la integral del error absoluto) a las funciones de pertenencia propuestas en ²⁰ y a un controlador PID, además, se plantearon hipótesis con respecto a la eficiencia de las funciones de pertenencia simétricas y centradas, las cuales fueron validadas con los datos de simulación.

Las conclusiones obtenidas con el desarrollo de este trabajo de grado según lo mencionado anteriormente son:

* Se toma como punto de partida un sistema bastante común y simple, para verificar la eficiencia de este método, lo cual permitirá en futuras ocasiones llevarlo a la práctica en sistemas más complejos con mayor facilidad ya que este cálculo no depende de los parámetros del sistema sino puramente de las funciones de pertenencia y ganancias del Fuzzy.

* Según los resultados obtenidos anteriormente el controlador de lógica difusa optimizado demuestra ser más eficiente que su estructura convencional y el controlador PID diseñado.

* El sistema se comporta adecuadamente ante variaciones en la resistencia de carga y la tensión de entrada lo que permite reafirmar que este controlador

²⁰ Ibíd.

funciona adecuadamente para este convertidor de potencia y podría ser llevado a la práctica.

BIBLIOGRAFÍA

AMEUR A., HARMAS M.N and ABDERREZEK H.. "Stochastic Optimization Methods Based Robust Control for DC/DC Buck Converter" in 5th International Conference on Control, Decision and Information Technologies (CoDIT'18).IEEE. 2018.

BENDAOU K., KRIT S., KABRANE M., OUADANI H., ELASKRI M., KARIMI K., ELBOUSTY H. and ELMAIMOUNI L.."Fuzzy Logic Controller (FLC):Application to control DC-DC buck converter." Laboratory of Engineering Sciences and Energies, Polydisciplinary Faculty of Ouarzazate. Ibn Zohr University- Agadir. Morocco. 2017.

CHALMERS S. J.. "A design methodology for fuzzy logic controllers", Synthesis and Optimization of Logic Systems, IEEE Colloquium on. 1994. Cap 10,pp 1-4.

CHEN C.H.. "Programmable Fuzzy Logic Device for Sequential Fuzzy Logic Synthesis." in 2001 IEEE International Fuzzy Systems Conference. 2001. pp. 107-110.

DÍAZ SAÉNZ H. D., PÉREZ MUNEVAR J.A.."Modelado, simulación e implementación de un sistema de control de tensión para un convertidor Buck, basado en lógica difusa.".Colombia. 2007.

HART D. W.. \textit{Power Electronics.}. 1ra. ed. McGraw Hill. 2010. Capítulo 6.

HOLGADO J.. "Design improvement in fuzzy controllers.", Department of Electronic and Control Engineering, University of Cadiz. Spain.1996. pp. 2038-2044.

HU Y. and RAMAMOON P. "Design of Stable Recurrent Fuzzy-logic-controllers." Department of Electrical & Computer Engineering, University of Cincinnati, Ohio. 1994. pp. 918-923.

KHAN E. and VENKATAPURAM P.. "Neufuz: Neural Network Based Fuzzy Logic Design Algorithms", Embedded Systems Division, National Semiconductor, Santa Clara. IEEE. pp 647-654.

OGATA K., *Ingeniería de Control Moderno*. 5ta. ed. Pearson. 2010. Capítulo 2.

PARK H. S., and KIM H. J.. "Simultaneous Control of DC-DC Converters by DSP Controller", IEEE PEDS. Indonesia. 2001. pp 89-93.

SWATHY K, SHRUTIKA JANTRE MS., YOGITA JADHAV Ms., SUSHIL Mr LABDE. M., PRATIK KADAM Mr.. "Design and Hardware Implementation of Closed Loop Buck Converter Using Fuzzy Logic Controller.", Proceedings of the 2nd International conference on Electronics, Communication and Aerospace Technology. ICECA. 2018. pp. 175-180.

WANG C.Y., OU Y. C., WU C. F. and SHIUE M. T.. "A Voltage-Mode DC-DC Buck Converter with Digital PID Controller" in 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD). 2015.