

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
Facultad de Ciencias Químicas e Ingeniería
Maestría y Doctorado en Ciencias e Ingeniería



**OPTIMIZACIÓN MULTIOBJETIVO EVOLUTIVA DE
CONTROLADORES DIFUSOS TIPO-2 PARA ROBOTS
AUTÓNOMOS MÓVILES**

TRABAJO DE TESIS

Presentado por

RICARDO MARTÍNEZ SOTO

Para obtener el Grado de

DOCTOR EN CIENCIAS

Director de Tesis

DR. OSCAR CASTILLO LOPEZ

Co-Director de Tesis

DR. LUIS TUPAK AGUILAR BUSTOS

DR. ANTONIO RODRIGUEZ DIAZ

TIJUANA, BC., DICIEMBRE 2011

Universidad Autónoma de Baja California

FACULTAD DE CIENCIAS QUÍMICAS E INGENIERÍA

COORDINACIÓN DE POSGRADO E INVESTIGACIÓN

FOLIO No. 067

Tijuana, B. C., a 11 de noviembre de 2011

C. RICARDO MARTINEZ SOTO
Pasante de: Doctor en Ciencias
Presente

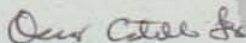
El tema de trabajo y/o tesis para su examen profesional, en la
Opción TESIS

Es propuesto, por los C. Dres. Oscar Castillo López, Luis Tupak Aguilar Bustos y Antonio Rodríguez Díaz

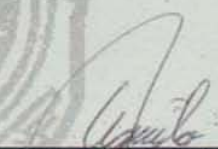
quienes serán los responsables de la calidad de trabajo que usted presente, referido al tema Optimización multiobjetivo evolutiva de controladores difusos tipo-2 para robots autónomos móviles.

el cual deberá usted desarrollar, de acuerdo con el siguiente orden:

- I.- INTRODUCCIÓN
- II.- NOTACION Y PRELIMINARES
- III.- DESARROLLO
- IV.- RESULTADOS
- V.- CONCLUSIONES
- VI.- REFERENCIAS



Dr. Oscar Castillo López
Director de Tesis

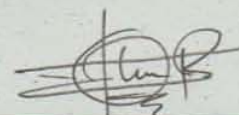


Dr. Luis Tupak Aguilar Bustos
Co-Director de Tesis

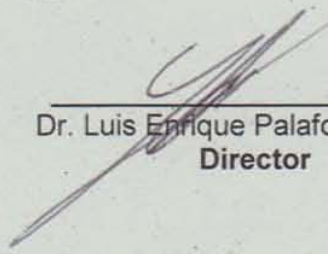


Q. Noemí Hernández Hernández
Sub-Director Secretario

FACULTAD DE CIENCIAS
QUÍMICAS E INGENIERÍA



Dr. Antonio Rodríguez Díaz
Co-Director Externo



Dr. Luis Enrique Palafox Maestre
Director

Resumen

En este trabajo de tesis se realizó el estudio de métodos de optimización para controladores difusos tipo-1 y tipo-2, aplicados a robots autónomos móviles. Se diseñó un método de optimización híbrido (Híbrido PSO-GA) a partir de la combinación de otros dos métodos de optimización, como son algoritmos genéticos (GA) y optimización por medio de nube de partículas (PSO) con la finalidad de generar controladores difusos óptimos para el seguimiento de trayectorias de los robots autónomos móviles. Se hicieron simulaciones con diferentes parámetros para los métodos de optimización y se analizaron con el fin de determinar si el método híbrido propuesto mejora los resultados obtenidos por los otros dos métodos de optimización usados de manera individual.

Abstract

We describe a proposed hybrid optimization method (PSO-GA) using the combination of two existing optimization methods, which are genetic algorithms (GA) and particle swarm optimization (PSO) in order to generate optimal type-1 and type-2 fuzzy logic controllers of trajectory tracking control for autonomous mobile robots. Simulations were made with different parameters for the optimization methods and were analyzed to determine if the proposed hybrid optimization method improves the results obtained by the two existing optimization methods used individually.

Agradecimientos

Primeramente, quiero dar gracias a Dios por permitirme llegar a este momento y lograr una meta más.

A mis padres Ricardo y María de los Ángeles por alentarme a continuar mis estudios y creer en mí; por darme su apoyo constante e incondicional, amor y comprensión, por estar siempre a mi lado y enseñarme que todo esfuerzo al final es recompensa.

A mi hermano Héctor y mis hermanas Julia, Cristina y Patricia por apoyar mis decisiones y brindarme su amistad.

A Denisse mi hoy esposa y su familia por el apoyo y la confianza para seguir adelante.

A mis asesores de Tesis; al Dr. Oscar Castillo López, al Dr. Luis Tupak Aguilar, a la Dra. Patricia Melin, y al Dr. Antonio Rodríguez por su asesoría científica, además al Dr. Guillermo Licea, al Dr. Juan Ramón Castro, a la Dra. Olivia Mendoza y al Dr. Fevrier Valdez por compartir sus conocimientos, sus consejos, amistad, y estímulo para seguir adelante en mi vida profesional.

A la Universidad Autónoma de Baja California y al Consejo Nacional de Ciencia y Tecnología (CONACYT) por la beca otorgada No. 175887 para concretar mis estudios de Doctorado en Ciencias.

ÍNDICE

RESUMEN	2
ABSTRACT	3
AGRADECIMIENTOS	4
1. INTRODUCCIÓN	9
1.1 ORGANIZACIÓN DEL TRABAJO	11
1.2. RESUMEN DE CONTRIBUCIONES	12
1.2.1. Journal Indexado	12
1.2.2. Capítulos en libro	12
1.2.3. Artículos en memorias de congreso	13
2. NOTACIÓN Y PRELIMINARES	15
2.1 LÓGICA DIFUSA	15
2.1.1 Introducción a Lógica Difusa	15
2.1.2 Percepción difusa	16
2.1.3 Conjuntos difusos	16
Definiciones básicas y terminología	16
Operaciones de conjuntos difusos	18
2.1.4 Funciones de membresía	19
2.1.5 Variables lingüísticas	23
2.1.6 Reglas Si-Entonces difusas	24
2.1.7 Sistemas de inferencia difusos	25
Base de reglas	25
Fusificador	26
Motor de inferencia	26
Defusificador	26
2.2 CONTROL DIFUSO.....	27
2.2.1 Aspectos fundamentales en la ingeniería de control	27
2.2.2 Proceso del diseño de control.....	28
Selección de la metodología del diseño	28
Determinación de los objetivos del diseño técnico	28
Desarrollo del modelo de la planta.....	28
2.2.3 Controlador lógico difuso	28
Arquitectura de Takagi-Sugeno para el control difuso.....	29
Aspectos de diseño del control difuso.....	29
2.3. LÓGICA DIFUSA TIPO-2	29
2.3.1 Definición de conjunto difuso Tipo-2	29
2.3.2 Operaciones en conjuntos difusos Tipo-2	32
Unión de conjuntos difusos Tipo-2	33
Intersección de conjuntos difusos Tipo-2.....	33
Complemento de un conjunto difuso Tipo-2.....	33
2.4. ALGORITMOS GENÉTICOS.....	34
2.4.1 Definición de algoritmo genético	34
2.4.2 Representación del cromosoma.....	34
2.4.3 Función objetivo y función de aptitud.....	35
Escalamiento lineal	35
2.4.4 Métodos de selección	36
Método de la ruleta	36
2.4.5 Operaciones genéticas	36
Cruce.....	36
Cruce de valores reales	37
A continuación se muestra los diferentes tipos de cruces utilizando valores reales.	37

Cruce discreto	37
Cruce intermedio.....	37
Cruce lineal	37
Cruce lineal extendido	37
Cruce de valores binarios.....	38
A continuación se muestra los diferentes tipos de cruces utilizando valores binarios.....	38
Cruce de un solo punto.....	38
Cruce multipunto	38
Cruce uniforme	38
Mutación.....	38
2.5. OPTIMIZACIÓN DE PARTÍCULAS	39
3. DESARROLLO	42
3.1 CASOS DE APLICACIÓN	43
3.1.1 Sistema lineal estable (Planta 1).....	43
3.1.2 Sistema lineal no estable (Planta 2).....	43
3.1.3 Péndulo con rueda de inercia	44
Experimento.....	46
3.1.4 Robot autónomo móvil.....	46
Objetivo de control	48
Control de seguimiento	49
3.2 Algoritmos Genéticos.....	52
Planta1	52
Diseño del controlador difuso tipo-1.....	52
Diseño del controlador difuso tipo-2.....	53
Configuración del algoritmo genético.....	55
Límites inferiores y superiores de los parámetros de las funciones de membresía.....	56
Reglas difusas del controlador difuso	57
Planta2	58
Diseño del controlador difuso tipo-1.....	58
Diseño del controlador difuso tipo-2.....	59
Configuración del algoritmo genético.....	61
Límites inferiores y superiores de los parámetros de las funciones de membresía.....	62
Reglas difusas del controlador	63
Péndulo de rueda de inercia	64
Configuración del algoritmo genético.....	64
Robot autónomo móvil.....	65
Diseño del controlador difuso	65
Simulación del robot autónomo móvil	67
Configuración de algoritmos genéticos para el control de seguimiento.....	67
Representación del cromosoma.....	68
Diseño de un controlador difuso Tipo-2.....	70
3.3 OPTIMIZACIÓN POR MEDIO DE NUBE DE PARTÍCULAS (PARTICLE SWARM OPTIMIZATION).....	70
Planta 1	71
Diseño del controlador difuso tipo-1.....	71
Diseño del controlador difuso tipo-2.....	71
Configuración del PSO	72
Límites inferiores y superiores de los parámetros de las funciones de membresía.....	73
Reglas difusas del controlador difuso	74
Planta 2	75
Diseño del controlador difuso	75
Diseño del controlador difuso tipo-2.....	77
Configuración del PSO	77
Límites inferiores y superiores de los parámetros de las funciones de membresía.....	77
Reglas difusas del controlador	79
PSO: Péndulo de rueda de inercia	79

Configuración del método de optimización por medio de nube de partículas (PSO).....	80
Robot autónomo móvil.....	80
Diseño del controlador difuso.....	80
Simulación del robot autónomo móvil.....	82
Configuración de PSO para el control de seguimiento.....	82
Diseño de un controlador difuso Tipo-2.....	84
3.4 MÉTODO DE OPTIMIZACIÓN HÍBRIDO (PSO-GA)	84
Planta 1	86
Diseño del controlador difuso	86
Diseño del controlador difuso tipo-2.....	87
Configuración del método híbrido PSO-GA.....	88
Límites inferiores y superiores de los parámetros de las funciones de membresía.....	88
Reglas difusas del controlador difuso	89
Planta 2	90
Diseño del controlador difuso.....	90
Diseño del controlador difuso tipo-2.....	91
Configuración del método híbrido PSO-GA	92
Límites inferiores y superiores de los parámetros de las funciones de membresía.....	92
Reglas difusas del controlador	93
Robot autónomo móvil.....	94
Diseño del controlador difuso	94
Simulación del robot autónomo móvil.....	96
Configuración del método híbrido PSO-GA	96
Diseño de un controlador difuso Tipo-2.....	98
4. RESULTADOS.....	99
4.1 RESULTADOS DEL USO DE LOS ALGORITMOS GENÉTICOS	99
4.1.1 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 1.....	99
Experimento (Mejor Simulación).....	101
4.1.2 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 1.....	101
Experimento (Mejor Simulación).....	103
4.1.3 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 2.....	104
Experimento (Mejor Simulación).....	106
4.1.4 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 2.....	108
Experimento (Mejor Simulación).....	109
4.1.5 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía GGG para la Planta 1	111
Experimento (Mejor Simulación).....	112
4.1.6 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-2 con funciones de membresía GGG para la Planta 1	114
Experimento (Mejor Simulación).....	115
4.1.7 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía GGG para la Planta 2	117
Experimento (Mejor Simulación).....	118
4.1.8 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-2 con funciones de membresía GGG para la Planta 2	120
Experimento (Mejor Simulación).....	121
4.1.9 Resultados del uso de algoritmos genéticos para encontrar las ganancias k_1 , k_2 y k_3 del controlador PID del péndulo de rueda de inercia.....	123
Experimento (Mejor Simulación).....	124
4.1.10 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 para un robot autónomo móvil.....	126

Experimento (Mejor Simulación).....	128
4.2 RESULTADOS DEL USO DE OPTIMIZACIÓN POR MEDIO DE PARTÍCULAS (PSO)	131
4.2.1 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados manualmente	131
Experimento (Mejor Simulación).....	132
4.2.2 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados manualmente	134
Experimento (Mejor Simulación).....	135
4.2.3 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GGG para la Planta 1 con parámetros de PSO asignados manualmente	137
Experimento (Mejor Simulación).....	138
4.2.4 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía GGG para la Planta 2 con parámetros de PSO asignados manualmente	140
Experimento (Mejor Simulación).....	141
4.2.5 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados de forma aleatoria.....	143
Experimento (Mejor Simulación).....	144
4.2.6 Resultados del uso de PSO para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados de forma aleatoria.....	146
Experimento (Mejor Simulación).....	147
4.2.7 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados de forma aleatoria.....	149
Experimento (Mejor Simulación).....	150
4.2.8 Resultados del uso de PSO para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados de forma aleatoria.....	152
Experimento (Mejor Simulación).....	153
4.2.9 Resultados del uso de PSO para encontrar las ganancias k_1 , k_2 y k_3 del controlador PID del péndulo de rueda de inercia	153
Experimento (Mejor Simulación).....	156
4.2.10 Resultados del uso de PSO para el controlador difuso tipo-1 para un robot autónomo móvil	157
Experimento (Mejor Simulación).....	159
4.3 RESULTADOS DEL USO DEL MÉTODO DE OPTIMIZACIÓN HÍBRIDO PSO-GA	162
4.3.1 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados manualmente	162
Experimento (Mejor Simulación).....	162
4.3.2 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados manualmente	164
Experimento (Mejor Simulación).....	166
4.3.3 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados manualmente	168
Experimento (Mejor Simulación).....	169
4.3.4 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados manualmente	171
Experimento (Mejor Simulación).....	172
4.3.5 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-1 para un robot autónomo móvil	174
Experimento (Mejor Simulación).....	175
4.3.6 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-2 para un robot autónomo móvil	178
Experimento (Mejor Simulación).....	179
5. CONCLUSIONES	182
6. REFERENCIAS	186

1. Introducción

En la actualidad, los robots móviles son utilizados para gran variedad de tareas, debido a la facilidad que tienen para desplazarse en terrenos o ambientes inciertos.

Los robots móviles son sistemas no holonómicos debido a las constantes impuestas en su cinemática. Las ecuaciones que describen estas constantes, no pueden ser integradas simbólicamente para obtener relaciones explícitas entre la posición del robot en coordenadas locales y globales. Estas restricciones se originan debido al contacto de las ruedas con el suelo, reduciendo las velocidades alcanzables por el robot móvil. De ahí que los problemas de control han atraído la atención de la comunidad de control en los últimos años [43].

Se han aplicado diferentes métodos para resolver los problemas de control de movimiento. Kanayama [32] propuso un método de control de seguimiento estable para un vehículo no holonómico usando una función de Lyapunov, Lee [46] resolvió el control de seguimiento usando desplazamiento hacia atrás (backstepping) en [47] con constantes de saturación. Además, la mayoría de los diseños reportados confían en control inteligente tal como control con lógica difusa [7], [8], [30], [45], [68], [70], [72], [77], [78] y redes neuronales [24], [64], [71], [75].

Sin embargo, la mayoría de las publicaciones mencionadas, se han concentrado en los modelos cinemáticos de los robots móviles, los cuáles son controlados por la entrada de velocidad, mientras han dejado de prestar atención a los problemas de control de la dinámica de sistemas no holonómicos, donde la fuerza y los pares son las verdaderas entradas.

La computación inteligente, es un método innovador para construir sistemas inteligentes. Los problemas complejos del mundo real, requieren sistemas inteligentes que combinen conocimiento, técnicas y metodologías de diferentes fuentes. Esta computación inteligente consiste de varios paradigmas, incluyendo redes neuronales, lógica difusa, razonamiento aproximado y métodos de optimización como los algoritmos genéticos, particle swarm optimization, entre otros.

El cerebro humano interpreta información imprecisa e incompleta por medio de órganos perceptivos. La teoría de la lógica difusa provee un cálculo sistemático para lidiar con esa información de forma lingüística, y realiza computación numérica usando etiquetas lingüísticas estipuladas por funciones de membresía. Además, una selección de reglas difusas “si-entonces” forman el componente principal de un sistema de inferencia difusa (FIS) que puede modelar de forma efectiva la experiencia humana en una aplicación específica.

Aunque un sistema de inferencia difusa tiene una representación estructurada del conocimiento en la forma de reglas difusas “si-entonces”, carece de la adaptabilidad para lidiar con el cambio del ambiente externo [11].

La inteligencia natural es el producto de millones de años de evolución biológica. Simular complejos procesos biológicos evolutivos nos puede conducir a descubrir cómo la evolución propulsa los sistemas vivientes hacia niveles más altos de inteligencia. Por esto, se ha puesto mayor atención a las técnicas de computación evolutiva tales como algoritmos genéticos (GAS) [3], [4], [16], [17], [18], los cuales están basados en el principio de evolución de selección natural; y sistemas bio-inspirados como es Particle Swarm Optimization (PSO) [5], [6], [15] el cual está basado en el comportamiento de las aves en el medio ambiente. Estos métodos de optimización son solo algunos de los más populares, pero existen muchos otros tipos más. En si estos dos tipos de métodos de optimización [25], son muy buenos y resuelven diferentes tipos de problemas, hasta el combinarlos con sistemas difusos hacen que se vuelvan un complemento mas

para la optimización de los mismos y obtener mejores resultados; un vez trabajados, con estos métodos decidimos crear un sistema de optimización híbrido [33],[42],[50],[65],[69],[73] que ayude a obtener controladores difusos óptimos más eficientes y para problemas con mayor complejidad; y es en este punto donde hacemos la propuesta de crear un método de optimización híbrido llamado PSO-GA haciendo uso de los métodos de optimización GA y PSO. Se han realizado anteriormente trabajos con algoritmos genéticos y optimización por medio de nube de partículas para encontrar controladores difusos óptimos para robots autónomos móviles, pero en éste trabajo de tesis en particular se buscan controladores difusos tipo-1 y tipo-2 para robots autónomos móviles con ruedas utilizando el método híbrido PSO-GA que al compararlo con los métodos de optimización mencionados anteriormente se pretende demostrar su robustez.

1.1 Organización del trabajo

El presente trabajo de tesis se encuentra estructurado por capítulos, los cuales se describen brevemente a continuación:

- *Capítulo 2:* En este capítulo se muestran los conceptos básicos acerca de los métodos utilizados en este trabajo de tesis de tal manera que se pueda tener una amplia visión de los temas tratados; como lo son, la Lógica Difusa Tipo-2, Algoritmos Genéticos, Optimización por medio de nube de Partículas, Métodos de Optimización Híbridos, entre otros.
- *Capítulo 3:* Se describe el desarrollo de los métodos de optimización y de los casos de aplicación utilizados. Primeramente se presentan los dos métodos de forma individual como son Algoritmos Genéticos y Optimización por medio de nube de Partículas, las cuales fueron utilizadas en los casos de aplicación llamados sistema lineal estable (planta 1), sistema lineal no estable (planta 2), péndulo con rueda de inercia y al final con un robot autónomo móvil. Al final del capítulo se presenta el método de optimización propuesto, que es una hibridación de algoritmos genéticos con optimización por medio de nube de partículas al cual fué llamado PSO-GA Híbrido, el cual también se utilizaron los mismos casos de aplicación para ser probado. También en éste capítulo se muestran el tipo de configuración utilizada para los

controladores difusos y la sintonización de los parámetros del caso de aplicación del péndulo con rueda de inercia.

- *Capítulo 4:* En este capítulo se presenta los resultados para cada caso de aplicación, como son primeramente la tabla de resultados que muestra los parámetros utilizados por el método de optimización, los tiempos de ejecución del método y al final el error promedio alcanzado; además de la tabla, se presentan las gráficas que muestran el comportamiento del método de optimización, las funciones de membresía óptimas y la respuesta del controlador óptimo obtenido para cada uno de los métodos de optimización en cada caso de aplicación.
- *Capítulo 5:* En este capítulo quedan plasmadas las conclusiones a las que llegamos después de haber realizado el presente proyecto de Tesis.
- *Capítulo 6:* Finalmente, en este capítulo se encuentran las referencias, que presenta la bibliografía estudiada para la realización de éste proyecto de Tesis.

1.2. Resumen de Contribuciones

A continuación se muestra la contribución de artículos desarrollados en este proyecto de tesis, los cuales se presentaron en journal indexado, capítulo en libro y memoria de congreso.

1.2.1. Journal Indexado

Ricardo Martinez-Soto, Oscar Castillo, Luis T. Aguilar, Antonio Rodriguez, "Gain Optimization for Inertia Wheel Pendulum Stabilization using Particle Swarm Optimization and Genetic Algorithms", "Aceptado" en *International Journal of Innovative Computing, Information & Control*, (IJICIC), 2012.

1.2.2. Capítulos en libro

Ricardo Martinez, Oscar Castillo, Luis T. Aguilar, "Optimization of Interval Type-2 Fuzzy Logic Controllers for a Perturbed Autonomous Wheeled Mobile Robot using Genetic Algorithms", *Soft Computing for Hybrid Intelligence systems*, pp. 3-18, 2008.

Ricardo Martinez-Soto, Oscar Castillo, Luis T. Aguilar, Antonio Rodriguez, "Evolutionary Optimization of Type-2 Fuzzy Logic Systems Applied to Linear Plants", *Evolutionary Design of Intelligent Systems in Modeling, Simulation and Control*, Springer, pp. 17- 31, ISBN: 978-3-642-04513-4, 2009.

Ricardo Martinez-Soto, Oscar Castillo, Luis T. Aguilar, Antonio Rodríguez, "Optimization of Type-2 Fuzzy Logic Controllers using PSO Applied to Linear Plants", Oscar Castillo, Witold Pedrycz, Janusz Kacprzyk in *Soft Computing for Intelligent Control and Mobile Robotics*, Springer, Alemania, pp. 181-193, 2010

1.2.3. Artículos en memorias de congreso

Ricardo Martinez, Oscar Castillo, Luis T. Aguilar, Antonio Rodríguez, "Optimization of Type-2 Fuzzy Logic Controllers for Mobile Robots Using Evolutionary Methods", *Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2009)* San Antonio, TX, USA , pp. 4909 -4914, 2009.

Ricardo Martinez, Oscar Castillo, Luis T. Aguilar, Antonio Rodríguez, "Genetic Optimization of Interval Type-2 Fuzzy Logic Controllers Applied to Linear Plants," *V Encuentro Regional Académico ERA`09* los días 11 y 12 de Noviembre del 2009, Tijuana, Baja California, México, ISBN : 978-607-414-113-9, pp. 84-89, 2009.

Ricardo Martinez, Oscar Castillo, Luis T. Aguilar, Antonio Rodríguez, "Bio-inspired Optimization methods of Fuzzy Logic Controllers Applied to Linear Plants", *WSC14, 2009 Online World Conference on Soft Computing in Industrial Applications WWW, November 17-29, 2009. Publish: Soft Computing in Industrial Applications, Advances in Soft Computing*, Volume 75, pp. 245-252, 2010.

Ricardo Martinez, Oscar Castillo, Luis T. Aguilar, Antonio Rodríguez, "Type-2 Fuzzy Logic Controllers Optimization using Genetic Algorithms and Particle Swarm Optimization", *2010 IEEE Conference on Granular Computing (GrC 2010)*, Aug 14-16 , Silicon Valley USA, pp.724-727, 2010.

Ricardo Martinez, Oscar Castillo, Luis T. Aguilar, Patricia Melin, "Fuzzy Logic Controller Optimization using Genetic Algorithms and Particle Swarm Optimization", *9th Mexican International Conference on Artificial Intelligence (MICAI 2010)*, November 8-13 Pachuca, Mexico, pp. 475-486, 2010.

Ricardo Martinez, Antonio Rodriguez, Luis T. Aguilar, "Gain Optimization for Inertia Wheel Pendulum Stabilization using a Bio-Inspired and Evolutionary Methods", *Proceeding of the World Conference on Soft Computing (WCSC 2011)*, San Francisco, CA; pp. 246-251, May 23-26, 2011.

2. Notación y Preliminares

Este capítulo introduce las definiciones y conceptos básicos de la lógica difusa Tipo-1 y Tipo-2, así como control difuso, algoritmos genéticos y optimización por medio de nube de partículas, los cuales sirvieron de base para la realización de este proyecto de tesis.

2.1 Lógica difusa

2.1.1 Introducción a Lógica Difusa

Zadeh introdujo el término lógica difusa en su trabajo seminal “Conjuntos Difusos”, el cual describió las matemáticas de la teoría de conjuntos difusos (1965). Platón colocó la base para lo que se conocería como lógica difusa, indicando que había una tercera región más allá de falso y verdadero. Fué Lukasiewicz quien primero propuso una alternativa sistemática a la bivaluada lógica de Aristóteles. El tercer valor que Lukasiewicz propuso puede ser mejor traducido como “posible”, y lo asignó a un valor numérico entre verdadero y falso [42].

La lógica difusa provee la oportunidad para modelar condiciones que son definidas imprecisamente. Las técnicas difusas en la forma de razonamiento aproximado proveen soporte de decisiones y sistemas expertos con capacidades de razonamiento poderosos.

La tolerancia de lo difuso en el proceso del pensamiento humano sugiere que mucha de la lógica detrás del procesamiento del pensamiento no es la lógica tradicional bi-valorada, o incluso la lógica multi-valorada, si no la lógica con verdades difusas, con conectividades difusas y reglas difusas o inferencia [1].

2.1.2 Percepción difusa

Nuestra interacción con el entorno da paso a una percepción del mismo, la cual puede considerarse difusa debido a la forma de expresarnos o asignar valores lingüísticos en situaciones cotidianas. El lenguaje natural maneja conceptos vagos e imprecisos, por ejemplo, para verificar la temperatura del agua de una regadera al abrir y cerrar las llaves del agua fría y caliente, no lo hacemos con la precisión de un instrumento de medición, más bien nos referimos a ella como fría, tibia, caliente, muy caliente, etc., y estas características dan paso al uso de reglas también difusas que son del estilo “si esto, entonces aquello” en donde el antecedente y consecuente de la regla puede ser una variable lingüística y la percepción puede ser diferente para cada individuo.

2.1.3 Conjuntos difusos

Los conjuntos difusos son una extensión de la teoría clásica de conjuntos y como su nombre lo implica, es un conjunto sin límites bien establecidos. Esto es, la transición de “pertener a un conjunto” a “no pertenecer a un conjunto” es gradual, y ésta suave transición se caracteriza por grados de membresía que dan a los conjuntos difusos flexibilidad en el modelado de expresiones lingüísticas comúnmente usadas [11], tales como “El clima es frío” o “Gustavo es alto”.

Definiciones básicas y terminología

Sea X un espacio de objetos y x un elemento de X . Un conjunto clásico A , $A \subseteq X$ está definido como una colección de elementos u objetos $x \in X$, tales que cada x puede pertenecer o no pertenecer al conjunto A . Por definición, una función característica para cada elemento x en X ,

podemos representar un conjunto clásico A por un conjunto de pares ordenados $(x, 0)$ o $(x, 1)$, los cuales indican $x \notin A$ o $x \in A$, respectivamente [11].

A diferencia de un conjunto convencional, un conjunto difuso expresa el grado al cual un elemento pertenece a un conjunto. De ahí que la función característica de un conjunto difuso permite tener valores entre 0 y 1, lo cual denota el grado de membresía de un elemento en un conjunto dado [11]. Las siguientes definiciones serán utilizadas a lo largo del documento y fueron recopiladas de [11]. Dichas definiciones son necesarias para conocer la metodología de los sistemas difusos y así poder comprender como funciona un controlador difuso en este trabajo de tesis.

Definición 2.1 (*Conjuntos difusos y funciones de membresía*): Si X es una colección de objetos denotados genéricamente por x , entonces un conjunto difuso A en X está definido como un conjunto de pares ordenados:

$$A = \{(x, \mu_A(x)) \mid x \in X\} \quad (2.1)$$

donde $\mu_A(x)$ es llamada la función de membresía (o FM para abreviar) para el conjunto difuso A . La función de membresía mapea cada elemento de X a un grado de membresía (o valor de membresía) entre 0 y 1.

Definición 2.2 (*Variables lingüísticas y valores lingüísticos*): Supongamos que $X = \text{“edad”}$. Entonces, podemos definir conjuntos difusos “joven”, “mediana edad” y “viejo” que son caracterizados por funciones de membresía $\mu_{joven}(x)$, $\mu_{medianaedad}(x)$, y $\mu_{viejo}(x)$, respectivamente. Justo como una variable puede asumir varios valores, una variable lingüística “edad” puede asumir diferentes valores lingüísticos, tales como “joven”, “mediana edad” y “viejo” en este caso. Si “edad” asume el valor de “joven”, entonces tenemos la expresión “edad es joven”, y así para los otros valores. Las funciones de membresía típicas para estos valores lingüísticos se muestran en la Figura 2.1, donde el universo de X está totalmente cubierto por las funciones de membresía y la transición de una función de membresía hacia otra es suave y gradual.

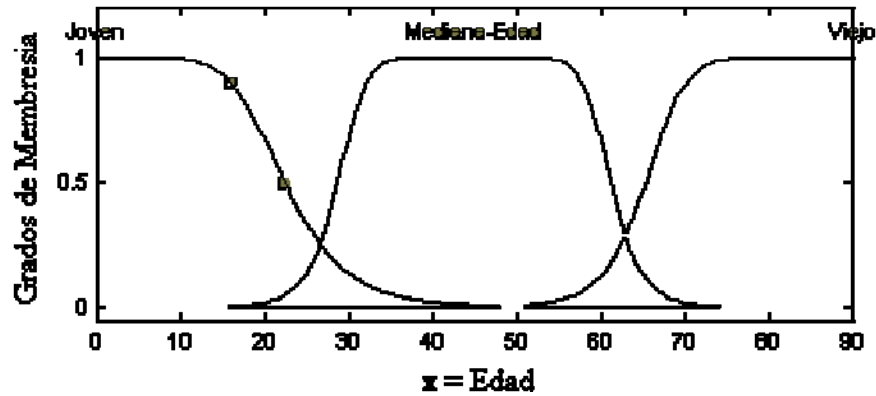


Figura 2.1: Funciones de membresía típicas de los valores lingüísticos “joven”, “mediana edad”, y “viejo”.

Operaciones de conjuntos difusos

La unión, la intersección y el complemento son las operaciones más básicas en los conjuntos clásicos [11].

Los conjuntos difusos también tienen operaciones similares a las de los conjuntos clásicos, y serán definidas a continuación.

Definición 2.3 (Subconjunto difuso): Un conjunto difuso A está contenido en un conjunto difuso B (o, equivalentemente, A es un subconjunto de B) si y solo si $\mu_A(x) \leq \mu_B(x)$ para toda x .

$$A \subseteq B \leftrightarrow \mu_A(x) \leq \mu_B(x) \tag{2.2}$$

Definición 2.4 (Unión): La unión de dos conjuntos difusos A y B , es un conjunto difuso C , escrito como $C = A \cup B$ o $C = A \vee B$, cuyas funciones de membresía están relacionadas a aquellas de A y B por:

$$\mu_c(x) = \max(\mu_A(x), \mu_B(x)) = \mu_A(x) \vee \mu_B(x). \tag{2.3}$$

Definición 2.5 (Intersección): La intersección de dos conjuntos difusos A y B , es un conjunto difuso C , escrito como $C = A \cap B$ o $C = A \wedge B$, cuya función de membresía está relacionada con las de A y B por:

$$\mu_c(x) = \min(\mu_A(x), \mu_B(x)) = \mu_A(x) \wedge \mu_B(x). \quad (2.4)$$

Definición 2.6 (*Complemento negación*): El complemento de un conjunto difuso A , denotado por \bar{A} , está definida como:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x). \quad (2.5)$$

Definición 2.7 (*Producto cartesiano y Co-Producto*): Sean A y B conjuntos difusos en X y Y respectivamente. El producto cartesiano de A y B , denotado por $A \times B$, es un conjunto difuso en el producto espacio $X \times Y$ con la función de membresía:

$$\mu_{A \times B}(x, y) = \min(\mu_A(x), \mu_B(y)). \quad (2.6)$$

Simultáneamente el co-producto $A + B$ es un conjunto difuso con la función de membresía:

$$\mu_{A+B}(x, y) = \max(\mu_A(x), \mu_B(y)). \quad (2.7)$$

2.1.4 Funciones de membresía

Un conjunto difuso está caracterizado por su función de membresía. Una forma conveniente de describirla es expresándola como una fórmula matemática [11].

Las funciones de membresía pueden tomar muchas formas. A continuación se presentan los tipos de membresía más comunes.

Definición 2.8 (*Función de membresía triangular*): Una función de membresía triangular está especificada por tres parámetros $\{a, b, c\}$, de la siguiente manera:

$$\text{triangle}(x; a, b, c) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0 & c \leq x. \end{cases} \quad (2.8)$$

Los parámetros $\{a, b, c\}$ (con $a < b < c$) determinan las coordenadas en x de las tres esquinas de la función de membresía fundamental.

La Figura 2.2-A ilustra una función de membresía triangular definida por: $\text{triangle}(x, 20, 60, 80)$.

Definición 2.9 (*Función de membresía trapezoidal*): Una función de membresía trapezoidal está especificada por cuatro parámetros (a, b, c, d) de la siguiente manera:

$$\text{trapezoid}(x; a, b, c, d) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ 1 & b \leq x \leq c \\ \frac{d-x}{d-c} & c \leq x \leq d \\ 0 & d \leq x. \end{cases} \quad (2.9)$$

Los parámetros $\{a, b, c, d\}$ (con $a < b < c < d$) determinan las coordenadas en x de las cuatro esquinas de la función de membresía trapezoidal. La Figura 2.2-B muestra una función de membresía trapezoidal definida por: $\text{trapezoid}(x; 10, 30, 70, 95)$. Debido a sus fórmulas simples y su eficiencia computacional, las funciones de membresía triangular y trapezoidal han sido muy usadas principalmente en implementaciones de tiempo real.

Definición 2.10 (*Función de membresía gaussiana*): Una función de membresía gaussiana se especifica por dos parámetros $\{c, \sigma\}$:

$$\text{gaussian}(x; c, \sigma) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma} \right)^2}. \quad (2.10)$$

Una función de membresía gaussiana está determinada completamente por c y σ ; c representa el centro de la función de membresía y σ determina su amplitud. La Figura 2.2-C muestra una función de membresía gaussiana definida por $\text{gaussian}(x, 50, 20)$.

Definición 2.11 (*Función de membresía campana generalizada*): Una función de membresía de campana generalizada se especifica por tres parámetros $\{a, b, c\}$:

$$\text{bell}(x; a, b, c) = \frac{1}{1 + \left| \frac{x - c}{a} \right|^{2b}}, \quad (2.11)$$

donde el parámetro b es generalmente positivo. (Si b es negativo, la forma de ésta función de membresía se convierte en una campana invertida hacia abajo). La Figura 2.2-D ilustra una función de membresía de campana generalizada definida por: $\text{bell}(x; 20, 4, 50)$.

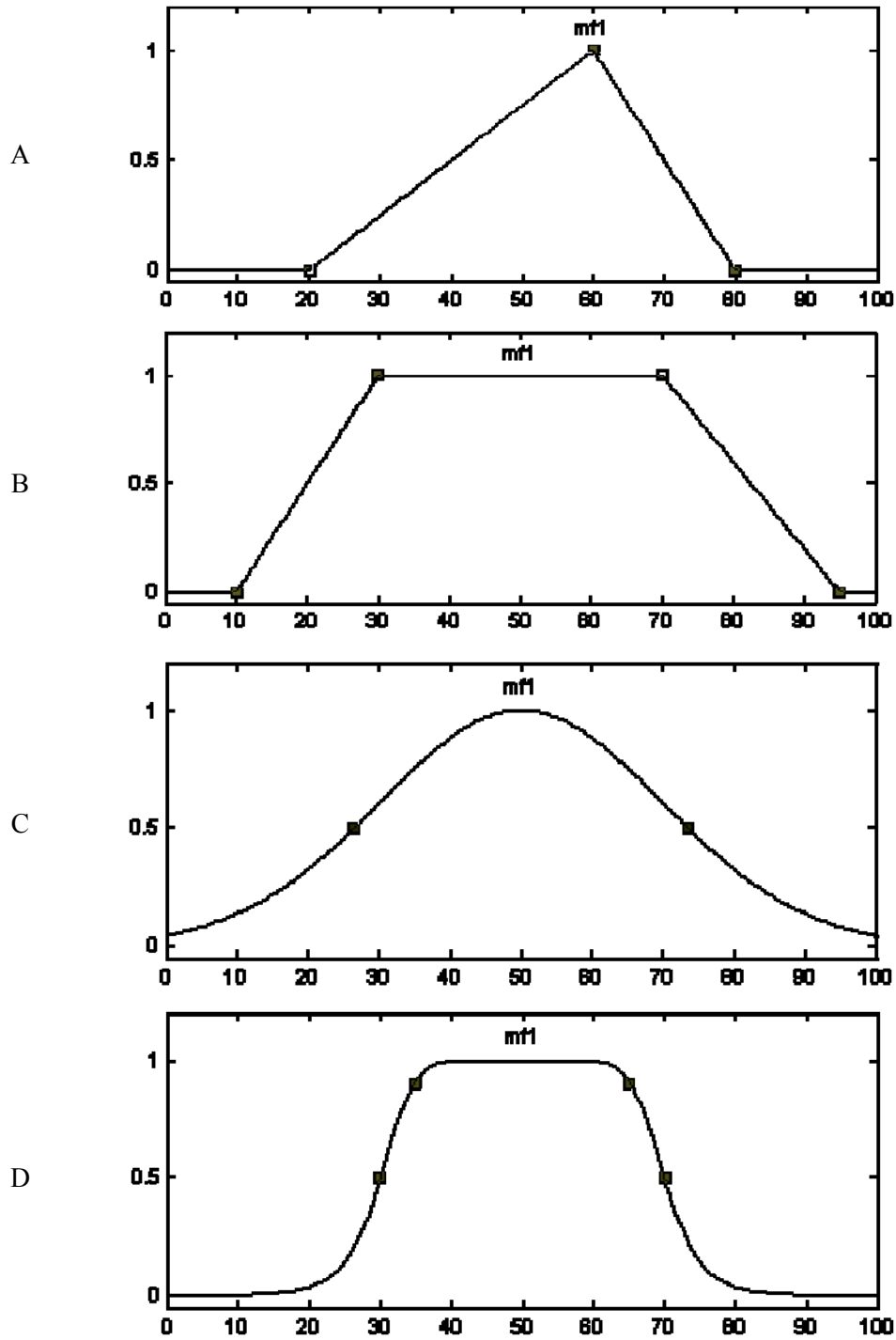


Figura 2.2: Función de membresía A) triangular: $triangle(x, 20, 60, 80)$, B) trapezoidal: $trapezoid(x, 10, 30, 70, 95)$, C) gaussiana: $gaussian(x, 50, 20)$ y D) campana generalizada: $bell(x, 20, 4, 50)$.

Por su suavidad y su notación concisa, las funciones de membresía gaussiana y campana generalizada se están volviendo populares para especificar conjuntos difusos. Las funciones gaussianas son bien conocidas en probabilidad y estadística, y poseen propiedades útiles tales como la invariancia en la multiplicación (el producto de dos gaussianas es una gaussiana con un factor escalar) y en las transformadas de Fourier (la transformada de Fourier de una gaussiana es una gaussiana). La función de membresía de campana tiene un parámetro más que la gaussiana, así que tiene un grado más de libertad para ajustar el escalón en los puntos de cruce. Aunque las funciones de membresía gaussianas y campanas logran suavidad, no pueden especificar funciones de membresía asimétricas, las cuales son importantes para ciertas aplicaciones [11].

A continuación se definen las funciones de membresía sigmoideas, las cuales pueden ser abiertas a la izquierda o a la derecha.

Definición 2.12 (*Función de membresía sigmoidea*): Una función de membresía sigmoidea está definida por:

$$\text{sig}(x; a, c) = \frac{1}{1 + \exp[-a(x - c)]}, \quad (2.12)$$

donde a controla la cuesta en el punto de cruce $x = c$.

Dependiendo del signo del parámetro a , una función de membresía sigmoidea es inherentemente abierta por la derecha o la izquierda y así es apropiada para representar conceptos como “muy grande” o “muy negativo”. Las funciones sigmoideas de éste tipo son empleadas ampliamente como función de activación de las redes neuronales artificiales.

2.1.5 Variables lingüísticas

Una variable lingüística está representada por una quintupla $(x, T(x), X, G, M)$ en donde x es el nombre de la variable; $T(x)$ es el conjunto de términos de x , esto es, el conjunto de sus valores lingüísticos o términos lingüísticos; X es el universo en discurso; G es una regla sintáctica la cual genera los términos en $T(x)$; y M es una regla semántica la cual asocia a cada valor lingüístico A con su significado $M(A)$, donde $M(A)$ denota un conjunto difuso en X . [11]

2.1.6 Reglas Si-Entonces difusas

Una regla difusa si-entonces (también conocida como regla difusa, implicación difusa o declaración condicional difusa) asume la forma:

Si x es A entonces y es B ,

donde A y B son valores lingüísticos definidos por conjuntos difusos en universos X y Y , respectivamente. A menudo “ x es A ” es llamado el antecedente o la premisa, mientras que “ y es B ” es llamado el consecuente o conclusión [11].

La interpretación de una regla difusa, es un proceso de tres partes: (a) Asignar a los antecedentes de todas las sentencias difusas un grado de membresía entre 0 y 1; (b) si hay múltiples partes en el antecedente, aplicar operadores lógicos difusos y resolver el antecedente a un solo número entre 0 y 1, es el resultado de soporte para toda la regla para formar el conjunto difuso de salida; y (c) aplicar el método de implicación. Si la regla tiene más de un antecedente, el operador difuso se aplica para obtener un número que represente el resultado de aplicar dicha regla [44].

Cada regla difusa produce un solo número que representa la fuerza de disparo para esa regla. La fuerza de disparo es usada para formar el conjunto difuso de salida que representa la parte del consecuente de la regla. El método de implicación está definido como la forma del consecuente (el conjunto difuso de salida), basado en el antecedente. La entrada para el proceso de implicación es un solo número dado por el antecedente, y la salida es un conjunto difuso [44].

Después de que se obtiene la fuerza de disparo de las reglas, se necesita combinar las salidas de los conjuntos difusos correspondientes en un conjunto difuso compuesto. El proceso de combinar conjuntos de salida difusos en un solo conjunto es llamado *agregación*, un proceso que unifica las salidas de todas las reglas. Esencialmente, la agregación toma todos los conjuntos difusos que representan la salida para cada regla y las combina en un conjunto difuso que es usado como entrada para el proceso de defusificación. La agregación solo ocurre una vez para cada variable de salida. Las entradas para el método de agregación son conjuntos difusos modificados obtenidos como la salida del proceso de implicación. La salida del proceso de agregación es un solo conjunto difuso que representa la variable de salida. Ya que el método de agregación es conmutativo, el orden en el cuál las reglas son ejecutadas no es importante [44].

Algunos ejemplos de reglas difusas son los siguientes:

- Si el nivel de agua es alto, la humedad de la tierra es alta.
- Si hay muchas nubes, entonces la probabilidad de lluvia es alta.
- Si la temperatura del horno es normal, entonces la velocidad de la banda transportadora es normal.

2.1.7 Sistemas de inferencia difusos

Un sistema de inferencia difuso (FIS), define un mapeo no lineal del vector de datos de entrada a salidas escalares, usando reglas difusas. Un sistema de inferencia difuso emplea reglas, sin embargo, a diferencia de las reglas de los sistemas expertos convencionales, una regla difusa localiza una región del espacio a través de la función de la superficie en vez de aislar un punto en la superficie. Para un punto dado, más de una regla puede ser disparada. También en un FIS, múltiples regiones son combinadas en el espacio de salida para producir una región compuesta [1], [13], [44].

Los componentes de un sistema de inferencia difuso son: El fusificador, el motor de inferencia, la base de reglas y el defusificador, los cuales serán descritos a continuación. En la Figura 2.3 se muestra el modelo general de un sistema de inferencia difuso.

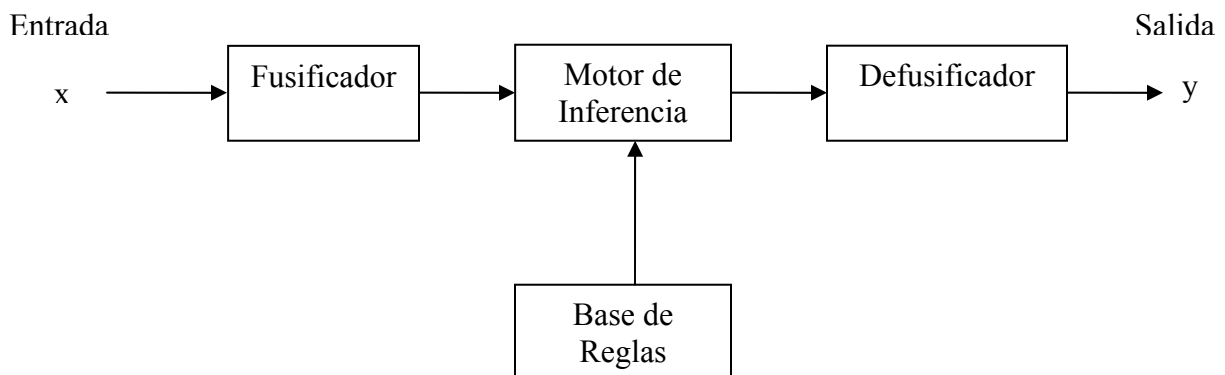


Figura 2.3: Diagrama a bloques de un sistema de inferencia Difuso.

Base de reglas

La base de reglas, contiene reglas lingüísticas que son provistas por expertos. También es posible extraer las reglas de datos numéricos. Una vez que las reglas han sido establecidas, el

sistema de inferencia difuso puede ser visto como un sistema que mapea un vector de entrada a un vector de salida [44].

Fusificador

El fusificador mapea números de entrada en sus correspondientes membresías difusas. Esto es requerido para activar reglas que están en términos de variables lingüísticas. El fusificador toma los valores de entrada y determina el grado al cual pertenecen para cada conjunto difuso por medio de funciones de membresía [44].

Motor de inferencia

El motor de inferencia determina el grado al cual el antecedente se satisface para cada regla. Si el antecedente de una regla dada tiene más de una cláusula, se aplican los operadores difusos para obtener un número que representa el resultado del antecedente para esa regla. Es posible que más de una regla se dispare al mismo tiempo. Las salidas para todas las reglas son agregadas. Durante la agregación, los conjuntos difusos que representan la salida para cada regla se combinan en un solo conjunto difuso. Las reglas difusas son disparadas en paralelo, lo cual es una de los aspectos importantes de los sistemas de inferencia difusos ya que el orden en el cual las reglas son disparadas no afecta a la salida [1], [44].

Defusificador

El defusificador mapea los conjuntos difusos de salida en un número bien establecido. Dado un conjunto difuso que abarque un rango de valores de salida, el defusificador regresa un número. El método más popular de defusificación es el de centroide [70], el cual consiste en encontrar el centro de gravedad del conjunto difuso resultante después del proceso de inferencia, o más exactamente su coordenada x . En términos matemáticos, un centroide es el punto de una figura geométrica cuyas coordenadas se igualan al promedio de todos los otros puntos que comprenden la figura. Este punto es el centro de gravedad de la figura; en términos sencillos, el centro de gravedad para una salida difusa es el valor del dato de la salida. El método del centro de gravedad es el más utilizado porque proporciona un resultado más representativo, basado en los valores ponderados de la salida de varias funciones de membresía [70].

El valor de salida final C para un sistema difuso con N reglas, se expresa mediante la fórmula:

$$y = \frac{\sum_{i=1}^N b_i \int \mu(i)}{\sum_{i=1}^N \int \mu(i)}, \quad (2.13)$$

donde b_i representa el centro de la función de membresía del consecuente de la regla i , y $\int \mu(i)$ es el área bajo la función de membresía $\mu(i)$ del consecuente en la regla i [70].

2.2 Control difuso

2.2.1 Aspectos fundamentales en la ingeniería de control

El término ingeniería de control se refiere a la disciplina cuyo principal objetivo es regular y generalmente controlar el comportamiento de un sistema físico. El término sistema físico se refiere a una colección de objetos físicos o entidades que juntas sirven para un propósito específico o una función predecible de acuerdo a las leyes físicas [63].

La terminología usada tal como planta, controlador, referencia, error, entrada, salida y perturbación son significados estándar para referirse a los aspectos significativos de un sistema de control, véase la Tabla 2.1 [63].

Tabla 2.1 Terminología usada en la ingeniería de control.

TERMINO	NOTACIÓN ESTÁNDAR	SIGNIFICADO
Planta	P	Es el sistema físico que se desea controlar
Controlador	C	Es el sistema computacional y/o el algoritmo usado para controlar el sistema físico
Entrada	u	Son los atributos controlables de la planta.
Salida	y	Son los atributos observables de la planta
Referencia	y_d o r	Es el valor deseado para las salidas de la planta
Error	e	Es la diferencia entre la referencia y la salida.
Perturbación	d	Son las entradas incontrolables de la planta

El componente principal del sistema de control es el controlador, cuyo propósito es el de cumplir los objetivos de comportamiento que se formulan dado el problema de control [63].

2.2.2 Proceso del diseño de control

El diseño del proceso de control en la ingeniería, envuelve la selección de las técnicas de control o la metodología, la determinación de objetivos de diseño técnicos, el desarrollo del modelo de la planta y la selección de la estructura del controlador y los parámetros.

Selección de la metodología del diseño

Las metodologías de control en la ingeniería se categorizan como basadas en tiempo o basadas en la frecuencia del dominio [63].

Determinación de los objetivos del diseño técnico

Esta tarea requiere interpretar, refinar, y cuantificar los objetivos de desempeño externos en un conjunto de objetivos de diseño técnicos compatibles con la metodología de diseño dada [63].

Desarrollo del modelo de la planta

Con el objetivo de diseñar adecuadamente una estrategia de control, se debe tener un modelo predictivo de la planta. En general, tal modelo es una descripción matemática del comportamiento del sistema físico dado y está derivado de acuerdo a leyes físicas aplicables.

2.2.3 Controlador lógico difuso

El control lógico difuso intenta acercarse a términos de naturaleza informal del diseño de proceso de control. En la arquitectura de Sugeno, se puede ver el control con lógica difusa como si se tradujeran directamente las especificaciones de desempeño y las observaciones del comportamiento externo de la planta en una estrategia de control basada en reglas lingüísticas y funciones matemáticas a su salida [63].

Arquitectura de Takagi-Sugeno para el control difuso

La extensión natural de la noción del control difuso es llamado arquitectura Takagi-Sugeno (alternativamente Takagi-Sugeno-Kang, TSK), por el que el conjunto de reglas define regiones de acción de las leyes de control convencional algebraico diferencial. El paradigma básico para el control lógico difuso esta dado por r reglas de la siguiente forma:

$$\begin{aligned} & \text{Si } x_1(t) \text{ es } M_{i1}, x_2(t) \text{ es } M_{i2}, \dots, \text{ y } x_n(t) \text{ es } M_{in}, \\ & \text{then } y = a_{i0} + a_{i1}x_1(t) + a_{i2}x_2(t). \end{aligned}$$

donde X_1, X_2, \dots, X_n son las variables antecedentes y la variables consecuentes es y . Por otra parte, $M_{i1}, M_{i2}, \dots, M_{in}$ son los conjuntos difusos definidos sobre los dominios respectivos de definiciones de X_1, X_2, \dots, X_n mientras $a_{i0}, a_{i1}, \dots, a_{in}$ son coeficientes constantes que caracterizan la relación lineal definida por las i reglas en el conjunto de reglas, $i = 1, 2, \dots, r$.

Aspectos de diseño del control difuso

Mientras que el enfoque basado en reglas ofrece medios flexibles para representar un algoritmo de control, no es claro como se debe idear tal conjunto de reglas. Con este fin, Sugeno y Takagi [49] sugirieron los siguientes medios como bases para la derivación de la base de reglas:

- Interrogación de un operador humano.
- Observación de un operador humano en acción.
- Modelado difuso del proceso.

Los primeros dos métodos son aplicables en situaciones en las cuales se tiene conocimiento previo de la operación del sistema físico dado. Sin embargo, se puede confiar en un modelo intuitivo del comportamiento del sistema físico y en particular entender los patrones de respuesta cuando hay cambios en sus entradas [63].

2.3. Lógica difusa Tipo-2

2.3.1 Definición de conjunto difuso Tipo-2

Considérese la transición de un conjunto ordinario a un conjunto difuso. Cuando no podemos determinar la membresía de un elemento en un conjunto como 0 o 1, usamos conjuntos difusos de Tipo-1. Similarmente, cuando las circunstancias son tan difusas que tenemos problemas para determinar el grado de membresía inclusive como un número bien definido entre $[0, 1]$, usamos conjuntos difuso de Tipo-2 [14], [34], [35], [36]; como un concepto que fue introducido por Zadeh en 1975.

Imaginemos quitar la definición de la función de membresía Tipo-1 de la Figura 2.4 (a) moviendo los puntos del triángulo ya sea, para la izquierda o la derecha y no necesariamente en las mismas cantidades, como en la Figura 2.4 (b). Después, en un valor específico de x , digamos x' , no es más un solo valor para la función de membresía; en lugar de eso, la función de membresía toma valores donde sea que la línea vertical intercepta la indefinición. Esos valores no necesitan ser pesados de la misma manera; por lo tanto, podemos asignar una distribución de amplitud para todos esos puntos. Hacer esto para toda $x \in X$, podemos crear una función de membresía tridimensional, -una función de membresía Tipo-2- que caracteriza a un conjunto difuso de Tipo-2 [27]. Las siguientes definiciones serán utilizadas a lo largo del documento y fueron recopiladas de [27].

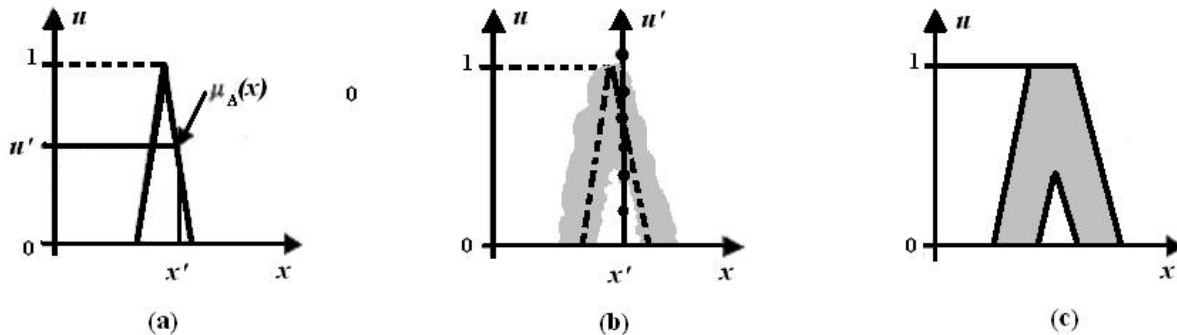


Figura 2.4: (a) Función de membresía Tipo-1, (b) Función de membresía Tipo-1 difusa, y (c) Huella de Incertidumbre.

Las siguientes definiciones describen el funcionamiento de los sistemas difusos tipo-2.

Definición 2.13: En un conjunto difuso Tipo-2 denotado como \tilde{A} , se caracteriza por una función de membresía Tipo-2 $\mu_{\tilde{A}}(x, u)$, donde $x \in X$ y $u \in J_x \subseteq [0, 1]$, por ejemplo:

$$\tilde{A} = \{ (x, u), \mu_{\tilde{A}}(x, u) \mid \forall x \in X, \forall u \in J_x \subseteq [0,1] \}, \quad (2.14)$$

donde $0 \leq \mu_{\tilde{A}}(x, u) \leq 1$. \tilde{A} también puede ser expresado como:

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x, u) / (x, u) \quad J_x \subseteq [0,1], \quad (2.15)$$

donde $\int \int$ denota la unión sobre toda x y u admisibles.

Definición 2.14: En cada valor de x , digamos $x = x'$, el plano 2D cuyos ejes son u y $\mu_{\tilde{A}}(x', u)$ es llamado un segmento vertical de $\mu_{\tilde{A}}(x', u)$. Una función de membresía secundaria es un segmento vertical de $\mu_{\tilde{A}}(x', u)$. Esto es, $\mu_{\tilde{A}}(x = x', u)$ para $x' \in X$ y $\forall u \in J_{x'} \subseteq [0,1]$, por ejemplo:

$$\mu_{\tilde{A}}(x = x', u) \equiv \mu_{\tilde{A}}(x') = \int_{u \in J_{x'}} f_{x'}(u) / u \quad J_{x'} \subseteq [0,1], \quad (2.16)$$

dónde $0 \leq f_{x'}(u) \leq 1$. Porque $\forall x' \in X$, se deja la notación primaria en $\mu_{A'}(x')$ y se refiere a $\mu_{A'}(x)$ como una función de membresía secundaria; esto es, un conjunto difuso de Tipo-1, en el cual también nos referimos como un conjunto secundario.

Basado en el concepto de conjuntos secundarios, podemos interpretar un conjunto difuso Tipo-2 como la unión de todos los conjuntos secundarios. Por ejemplo, usando la (2.16), podemos re-exresar \tilde{A} en una forma de segmento vertical, como:

$$\tilde{A} = \{ (x, \mu_{\tilde{A}}(x)) \mid \forall x \in X \}, \quad (2.17)$$

o como:

$$\tilde{A} = \int_{x \in X} \mu_{\tilde{A}}(x) / x = \int_{x \in X} \left[\int_{u \in J_x} f_x(u) / u \right] / x \quad J_x \subseteq [0,1]. \quad (2.18)$$

Definición 2.15: El dominio de una función de membresía secundaria es llamada la función de membresía primaria de x . En (2.18), J_x es la función de membresía primaria de x , donde $J_x \subseteq [0,1]$ para $\forall x \in X$.

Definición 2.16: La amplitud de una función de membresía secundaria es llamada un grado secundario. En (2.18), $f_x(u)$ es un grado secundario; En (2.14), $\mu_{\tilde{A}}(x', u')$ ($x' \in X, u' \in J_{x'}$) es un grado secundario.

Para funciones de membresía secundarias por intervalo, definimos la función de membresía principal como ocurre en la unión de todos los puntos medios de las funciones de membresía. Nótese que cuando toda la incertidumbre de la función de membresía desaparece, una función de membresía de tipo 2 se reduce a su principal función de membresía.

Definición 2.17: La incertidumbre en las funciones de membresía de conjuntos difusos tipo 2, \tilde{A} consiste en una región definida a la que llamamos la huella de incertidumbre (FOU), ésta es la unión de todas las funciones de membresía primarias.

$$FOU(\tilde{A}) = \bigcup_{x \in X} J_x. \quad (2.19)$$

El término FOU es muy útil, porque no solo enfoca nuestra atención en las incertidumbres inherentes en una función de membresía específica de Tipo-2, cuya Figura es una consecuencia directa de la naturaleza de estas incertidumbres, sino que también provee una muy conveniente descripción verbal del dominio entero de un soporte para todos los grados secundarios de una función de membresía difusa de Tipo-2.

2.3.2 Operaciones en conjuntos difusos Tipo-2

Considérese dos conjuntos de Tipo-2 \tilde{A} y \tilde{B} :

$$\tilde{A} = \int_X \mu_{\tilde{A}}(x) / x = \int_X \left[\int_{J_x} f_x(u) / u \right] / x \quad J_x^u \subseteq [0,1], \quad (2.20)$$

$$\tilde{B} = \int_X \mu_B(x)/x = \int_X \left[\int_{J_u^w} g_x(w)/w \right] / x \quad Jw \subseteq [0,1]. \quad (2.21)$$

Unión de conjuntos difusos Tipo-2

La unión de \tilde{A} y \tilde{B} es otro conjunto difuso de Tipo-2, como sucede en la unión de conjuntos difusos A y B de Tipo-1. De ahí que:

$$\tilde{A} \cup \tilde{B} \Leftrightarrow \mu_{\tilde{A} \cup \tilde{B}}(x, v) = \int_{x \in X} \mu_{\tilde{A} \cup \tilde{B}}(x)/x = \int_{x \in X} \left[\int_{v \in J_x^v \subseteq [0,1]} h_x(v)/v \right] / x, \quad (2.22)$$

donde:

$$\int_{v \in J_x^v \subseteq [0,1]} h_x(v)/v = \varphi \left(\int_{u \in J_x^u} f_x(u)/u, \int_{w \in J_x^w} g_x(w)/w \right) = \varphi(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)). \quad (2.23)$$

y φ es una función t-norma de las funciones de membresía secundarias, $\mu_{\tilde{A}}(x)$ y $\mu_{\tilde{B}}(x)$, los cuales son conjuntos difusos Tipo-1. φ es una función t-norma porque la unión de dos conjuntos difusos Tipo-1 es equivalente al t-norma de sus funciones de membresía.

Intersección de conjuntos difusos Tipo-2

La intersección de \tilde{A} y \tilde{B} es otro conjunto difuso Tipo-2, de ahí que:

$$\tilde{A} \cap \tilde{B} \Leftrightarrow \mu_{\tilde{A} \cap \tilde{B}}(x, v) = \int_{x \in X} \mu_{\tilde{A} \cap \tilde{B}}(x)/x. \quad (2.24)$$

El desempeño de $\mu_{\tilde{A} \cap \tilde{B}}(x)$ es el mismo que el desempeño de $\mu_{\tilde{A} \cup \tilde{B}}(x)$, excepto que en el caso presente φ es el mínimo o función producto \wedge ; de ahí:

$$\mu_{\tilde{A} \cap \tilde{B}}(x) = \int_{u \in J_x^u} \int_{w \in J_x^w} f_x(u) * g_x(w) / (u \wedge w) \quad x \in X. \quad (2.25)$$

Complemento de un conjunto difuso Tipo-2

El complemento de \tilde{A} es otro conjunto difuso Tipo-2, de la siguiente forma:

$$\bar{A} \Leftrightarrow \mu_{\bar{A}}(x, v) = \int_{x \in X} \mu_{\bar{A}}(x) / x. \quad (2.26)$$

En (2.26), $\mu_{\bar{A}}(x)$ es una función de membresía secundaria. Usando el principio de extensión, encontramos que:

$$\mu_{\bar{A}}(x) = \int_{u \in J_x^u} f_x(u) / (1-u) \equiv \neg \mu_{\bar{A}}(x) \quad x \in X, \quad (2.27)$$

donde \neg denota la llamada operación de negación. El uso de la notación $\neg \mu_{\bar{A}}(x)$ para indicar la negación de la segunda función de membresía $\mu_{\bar{A}}(x)$ es otra notación.

A través de (2.27) se puede realizar la negación de una función de membresía secundaria $\mu_{\bar{A}}(x)$, $1-u$ debe ser computado en $\forall u \in J_x^u$, y el grado secundario de $\mu_{\bar{A}}(x)$ en $1-u$ es el grado secundario correspondiente de $\mu_{\bar{A}}(x)$, $f_x(u)$. De acuerdo a (2.26), esto debe ser hecho $\forall x \in X$, para obtener $\mu_{\bar{A}}(x, v)$ [69].

2.4. Algoritmos genéticos

2.4.1 Definición de algoritmo genético

Un algoritmo genético (GA), es una técnica de programación que imita a la evolución biológica como estrategia para resolver problemas [76].

Los algoritmos genéticos suponen que la solución de cualquier problema es un individuo y puede ser representado por un conjunto de parámetros. Estos parámetros son considerados como genes de un cromosoma y pueden ser estructurados por una cadena de valores en forma binaria. Un valor positivo, generalmente conocido como valor de aptitud es usado para reflejar el grado de “bondad” del cromosoma para el problema, el cual será altamente relacionado con su valor objetivo [29].

2.4.2 Representación del cromosoma

Antes de que un algoritmo genético pueda ponerse a trabajar en un problema, se necesita un método para codificar las soluciones potenciales del problema de forma que una computadora

pueda procesarlas [76]. La codificación de la representación del cromosoma puede variar de acuerdo a la naturaleza del problema [29].

Un enfoque común es codificar las soluciones como cadenas binarias: secuencias de 1s y 0s, donde el dígito de cada posición representa el valor de algún aspecto de la solución. Otro método similar consiste en codificar las soluciones como cadenas de enteros o números decimales, donde cada posición, de nuevo, representa algún aspecto particular de la solución. Este método permite una mayor precisión y complejidad que el método comparativamente restringido de utilizar sólo números binarios, y a menudo “está intuitivamente más cerca del espacio de problemas” [76].

2.4.3 Función objetivo y función de aptitud

Una función objetivo, es un mecanismo de medición que es usada para evaluar el estado de un cromosoma. Como cada cromosoma pasa por el mismo modo de evaluación, el rango de este valor varía de un cromosoma a otro. Para mantener la uniformidad, los valores del objeto son mapeados a una función de aptitud con un mapa Ψ donde el dominio de F generalmente es mayor que cero.

$$\Psi: O \rightarrow F. \quad (2.28)$$

Escalamiento lineal

El valor de aptitud f_i del cromosoma i tiene una relación lineal con el valor objetivo o_i como:

$$f_i = a o_i + b \quad (2.29)$$

donde a y b son seleccionados para reforzar la calidad del valor objetivo promedio y el valor de aptitud promedio, y causar que la aptitud escalada mayor sea un múltiplo específico de la aptitud promedio.

Este método puede reducir el efecto de desvío genético produciendo un buen cromosoma. Sin embargo, puede introducir un valor de función de aptitud negativo, el cual debe ser evitado en las operaciones de GA. De ahí que la selección de a y b son dependientes del conocimiento del rango de los valores objetivo [29].

Existen tres métodos para evitar que el valor de aptitud sea negativo y son: Truncamiento de sigma (Sigma Truncation), escalamiento elevado a la potencia (Power Law Scaling) y el esquema de Categorización (Ranking).

2.4.4 Métodos de selección

Para generar un buen hijo, se necesita un buen mecanismo de selección de padres. Este es un proceso usado para determinar el número de pruebas para un individuo en particular usado en la reproducción. La oportunidad de seleccionar un cromosoma como padre deberá ser directamente proporcional al número de hijos producidos [29].

Método de la ruleta.

El método básico de selección por ruleta, es un muestreo estocástico con reemplazo (SSR). El tamaño del segmento y la probabilidad de la selección permanecen igual en toda la fase de selección y los individuos son seleccionados de acuerdo a los siguientes procedimientos [29]:

Bias, define la diferencia absoluta entre individuos en la probabilidad actual y la esperada para la selección. Un bias óptimo de cero se alcanza cuando la probabilidad de un individuo iguala el número de intentos esperados [29].

Extensión (Spread), es un rango en el posible número de intentos que un individuo puede alcanzar. Si $g(i)$ es el número actual de intentos debido a cada individuo i , entonces la extensión mínima es la extensión más pequeña que técnicamente permite un bias cero. La extensión de un método mide su consistencia.

Eficiencia, está relacionada con el tiempo total de complejidad del algoritmo. El algoritmo de selección debe así alcanzar un bias cero, al mismo tiempo que mantenga la extensión mínima y no contribuya a un incremento de tiempo en la complejidad del GA [29].

2.4.5 Operaciones genéticas

Cruce

El cruce produce nuevos individuos combinando la información contenida en sus padres. Dependiendo de la representación de las variables de los individuos pueden ser aplicados los siguientes algoritmos [31].

Cruce de valores reales

A continuación se muestra los diferentes tipos de cruces utilizando valores reales.

Cruce discreto.

El cruce discreto realiza un intercambio de los valores de las variables entre los individuos. El cruce discreto puede ser usado con cualquier tipo de variables (binarias, reales o símbolos) [31].

Cruce intermedio

El cruce intermedio es una metodología que sólo se aplica a variables reales (no a variables binarias). Aquí, los valores de la variable de los hijos se escogen en alguna parte alrededor y entre los valores de las variables de los padres [31].

Los hijos son producidos de acuerdo a la siguiente regla:

$$\text{Hijo} = \text{padre1} + \text{Alpha} (\text{Padre2} - \text{Padre1}), \quad (2.30)$$

donde Alpha es un factor escalar seleccionado uniformemente en forma aleatoria sobre un intervalo $[-d, 1, +d]$. En el cruce intermedio $d=0$, para el cruce intermedio extendido, $d_i > 0$.

Una buena selección es $d=0.25$. Cada variable en el hijo es el resultado de combinar las variables de acuerdo a la expresión anterior con un nuevo valor de Alpha seleccionado para cada variable [31].

Cruce lineal

El cruce lineal es similar al cruce intermedio, excepto que se usa solo un valor de Alpha para todas las variables y puede generar cualquier punto en la línea definida por los padres [31].

Cruce lineal extendido

El cruce lineal extendido genera hijos en una dirección definida por los padres. Prueba más seguido fuera del área definida por los padres y en la dirección del primer padre. El punto para el hijo está definido por características del operador de mutación del algoritmo genético del criador. Es solo aplicable a valores reales (no binarios o enteros) [31].

Cruce de valores binarios

A continuación se muestra los diferentes tipos de cruces utilizando valores binarios.

Cruce de un solo punto

En el cruce de un solo punto, una posición de cruce es seleccionada uniformemente de manera aleatoria y las variables se intercambian entre los individuos en este punto, después, dos nuevos hijos son creados [31].

Cruce multipunto

Para el cruce multipunto, el número de variables de un individuo son seleccionadas de manera aleatoria sin duplicados y clasificadas en orden ascendente. Después, las variables entre los puntos de cruce sucesivos son intercambiadas entre los dos padres para producir dos nuevos hijos. La sección entre la primera variable y el primer punto de cruce no es intercambiado entre los individuos. La idea detrás del cruce multipunto y muchas de las variaciones del operador de cruce, es que las partes de la representación del cromosoma que contribuyen en mayoría al desempeño de un individuo en particular, no necesariamente están contenidos en sub cadenas adyacentes. La naturaleza del cruce multipunto parece alentar la exploración del espacio de trabajo, más bien que favoreciendo la convergencia de individuos altamente aptos en los primeros resultados de la búsqueda, haciendo así, la búsqueda más robusta [31].

Cruce uniforme

Los cruces de un sólo punto y multipunto definen puntos de cruce como sitios entre lugares geométricos donde un individuo se puede partir. El cruce uniforme generaliza este esquema para hacer cada lugar geométrico un punto de cruce potencial. Se crea de forma aleatoria, una máscara de cruce, del mismo largo de la estructura del individuo, y la paridad de los bits en la máscara indican cual padre suministrará al hijo con cuales bits [31].

Mutación

Originalmente, la mutación fue diseñada solo para cromosomas representados en forma binaria. Para adoptar el concepto de introducir variaciones en el cromosoma, se ha diseñado una mutación para cromosomas representados con números reales:

$$g = g + \Psi(\mu, \sigma), \quad (2.31)$$

donde g es el gene de valor real; ψ es una función aleatoria la cual puede ser gaussiana o normalmente distribuida; μ , σ son la media y la varianza relacionadas con la función aleatoria, respectivamente [76].

2.5. Optimización de Partículas

La formación de equipos se ha observado en muchas especies de animales [22]. Para algunas especies de animales, los equipos o grupos son controlados por un líder (generalmente se refiere al alpha macho o hembra), por ejemplo una manada de leones, una tropa de caballos salvajes, una tropa de babuinos, por nombrar algunos. Uno de los primeros estudios sobre sociedad animal es el trabajo de Eugène N. Marais en sus estudios del chacmababoon salvaje (*Papioursinusursinus*) en los años 1900's [22]. En estas sociedades el comportamiento de los individuos es fuertemente dictado por una jerarquía social. Lo más interesante es el comportamiento auto-organizado de las especies que viven en grupos donde ningún líder es identificado, por ejemplo una parvada de pájaros, un banco de peces, o una manada de ovejas. Dentro de este grupos social de individuos no se tiene un conocimiento global del grupo, pero se tiene un conocimiento global del ambiente. A pesar de esto, tienen la capacidad recolectar y moverse juntos, basándose entre interacciones locales e individuos.

Un gran número de estudios sobre comportamiento colectivo y social de animales han sido hechos, por ejemplo (por mencionar algunos) [22]:

- Comportamiento de la congregación de aves.
- Comportamiento de los peces.
- El comportamiento de cacería de las ballenas jorobadas.
- El comportamiento de los monos salvajes, etc.

La optimización con enjambre de partículas, más conocida en la literatura científica como *Particle Swarm Optimization* (PSO) [12], [37], [38], [66], [74] nace, al igual que otras técnicas

estocásticas de cálculo evolutivo, en un intento por imitar y mimetizar el comportamiento de procesos naturales. El PSO junto con la optimización con una colonia de hormigas o *Ant Colony Optimization* (ACO) son los dos métodos más populares y utilizados en el área de la inteligencia computacional. Ambos tratan de imitar los comportamientos sociales de un colectivo a partir de la interacción de los individuos entre sí y con el entorno. Los orígenes del PSO como método estocástico de optimización global se remontan a los estudios iniciados por Kennedy y Eberhart en [20], quienes se fijan como objetivo inicial simular gráficamente el movimiento sincronizado e impredecible de grupos tales como los bancos de peces o las bandadas de aves, intrigados por la capacidad de estos grupos para separarse, reagruparse o encontrar alimento. En línea con trabajos previos en el ámbito de la biología y de la sociología, que concluyen que el comportamiento, inteligencia y movimiento de estas agrupaciones, entre las cuales se podría incluir con un cierto grado de abstracción a los seres humanos, está relacionado directamente con la capacidad de los individuos para compartir información y aprovecharse de la experiencia acumulada por sus congéneres, en [20] se modela dicho comportamiento de forma matemática utilizando expresiones simples que revelan su potencial como método de optimización.

En la terminología utilizada en PSO, Kennedy y Eberhart [20], [21] introducen el término general *partícula* o *agente* para representar a los peces, pájaros, abejas, hormigas o cualquier otro tipo de individuos que exhiban un comportamiento social como grupo, en forma de una colección de agentes que interactúan entre sí. De acuerdo con los fundamentos teóricos del método, el movimiento de cada una de estas partículas hacia un objetivo común en dos dimensiones está condicionado por dos factores básicos, la memoria autobiográfica de la partícula o nostalgia y la influencia social de todo el enjambre. A nivel computacional, como método de optimización, esta filosofía puede extenderse a un espacio N -dimensional de acuerdo con el problema bajo análisis. La posición instantánea de cada una de las partículas de la población en el espacio N -dimensional representa una solución potencial, siendo N el número de incógnitas del problema original. Básicamente, el proceso evolutivo se reduce a mover cada partícula dentro del espacio de soluciones con una velocidad que variará de acuerdo a su velocidad actual, a la memoria de la partícula y a la información global que comparte el resto del enjambre, utilizando una función de fitness para cuantificar la calidad de cada partícula en función de la posición que ésta ocupe.

Más allá de la propia naturaleza del método, los esquemas existentes para su implementación son muy diversos. Dependiendo de cómo se actualicen las posiciones de las partículas surgen las versiones síncrona y asíncrona del algoritmo [28], [40]. Adicionalmente, dependiendo de cómo se haga influir la experiencia acumulada por el enjambre sobre el movimiento de cada una de las partículas que lo integran, se puede distinguir entre PSO local y global [39].

3. Desarrollo

En este capítulo se presentan los métodos de optimización aplicados a dos plantas lineales, a un péndulo de rueda de inercia y a un robot autónomo móvil.

Los métodos de optimización utilizados en esta investigación fueron algoritmos genéticos (GA, Genetic Algorithms), optimización por medio de nube de partículas (PSO, Particle Swarm Optimization), y un método híbrido (Hybrid PSO-GA); primeramente se realizaron pruebas de forma individual con cada método y posteriormente se hizo una hibridación con los dos métodos, para formar un método de optimización combinado, en el cual se obtuvo mejores resultados en los casos con mayor complejidad que con los métodos aplicados individualmente.

A continuación se describen primeramente cada uno de los casos de estudio (Planta 1, Planta 2, Péndulo de rueda de inercia y Robot autónomo móvil) utilizados en esta investigación para continuar después con la descripción de los métodos de optimización.

3.1 Casos de Aplicación

3.1.1 Sistema lineal estable (Planta 1)

Para el estudio de los métodos de optimización se utilizaron diferentes casos de aplicación, el primero de ellos se le denomina planta 1 a una función de transferencia la cual el objetivo fue controlar encontrar un controlador difuso que ayudara a llegar a una referencia dada de manera estable [51], [53], [54], [55], [57], [58].

La planta 1 se representa con la siguiente función de transferencia (3.1):

$$g(s) = \frac{w_n^2}{s^2 + 2\varepsilon w_n s + w_n^2} \quad \varepsilon = 0.5, \quad w_n = 2 \quad (3.1)$$

donde w_n es la frecuencia natural y ε (épsilon) es el coeficiente de amortiguamiento.

En la Figura 3.1 se presenta el modelo de la planta 1 en diagrama de bloques programado en Simulink, para probar el controlador difuso optimizado de manera simulada.

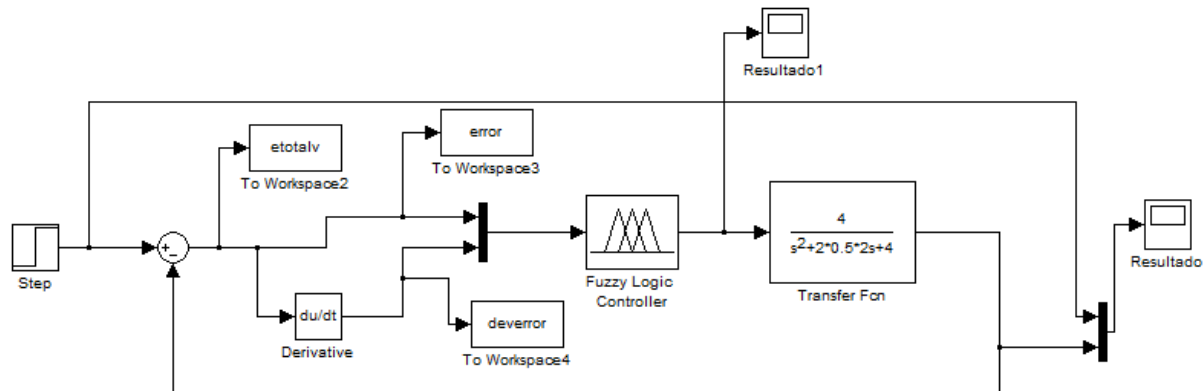


Figura 3.1: Diagrama de bloques en Simulink de la planta 1.

3.1.2 Sistema lineal no estable (Planta 2)

La planta 2 es otro de los casos de aplicación utilizada para encontrar su controlador difuso óptimo. Comparando la planta 1 con la planta 2, esta segunda es más complicada de estabilizar y controlar por lo que se cataloga como caso con más grado de complejidad [51], [53], [54], [55], [57], [58].

La planta 2 se representa con la siguiente función de transferencia (3.2):

$$g(s) = \frac{1}{s^2+4} \quad (3.2)$$

A continuación se presenta el modelo de la planta 2 en diagrama de bloques programado en Simulink, para probar el controlador difuso optimizado de manera simulada.

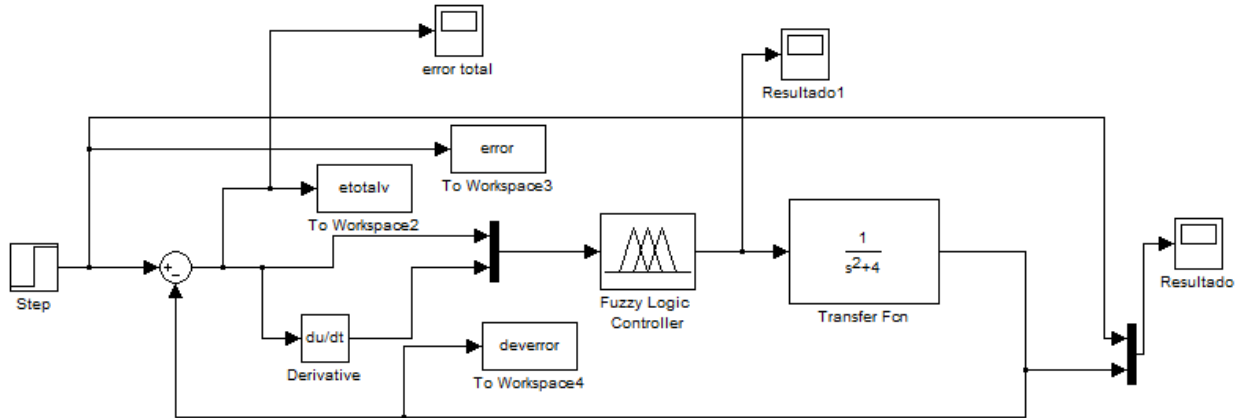


Figura 3.2: Diagrama de bloques en Simulink de la planta 2.

3.1.3 Péndulo con rueda de inercia

El péndulo de rueda de inercia fue otra de las aplicaciones que utilizamos para probar los métodos de optimización en esta investigación. En este caso no se generaron sistemas difusos, solo se realizó la búsqueda de las ganancias óptimas que necesita el controlador PID para estabilizar el péndulo. Se utilizó el modelo matemático del péndulo para probar los métodos de optimización de manera simulada y una vez obtenidos los valores óptimos, éstos fueron probados de manera experimental en el equipo Mechatronic Kit de Quanser [2], [23], [61], [62].

La dinámica del péndulo de rueda de inercia se describe como sigue:

$$\begin{bmatrix} J_1 & J_2 \\ J_2 & J_2 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} h \sin q_1 \\ f_s \dot{q}_2 \end{bmatrix} = \begin{bmatrix} 0 \\ k \end{bmatrix} u \quad (3.3)$$

donde $q_1(t) \in \mathbf{R}$ es el ángulo absoluto del péndulo, tomado en cuenta de acuerdo a las agujas del reloj desde la posición vertical descendente; $q_2(t) \in \mathbf{R}$ es el ángulo absoluto del disco; $f_s(\dot{q}_2)$ representa la fuerza de fricción viscosa que afecta el actuador, donde $f_s > 0$ es el coeficiente de fricción viscosa; J_1, J_2 y h son parámetros físicos positivos que dependen de la dimensión geométrica y de la distribución de la masa-inercia; $\tau = ku$ es el límite escalar aplicado al torque (ver Figura 3.3) . La fricción del péndulo es bastante pequeña por esta razón no se tomará en cuenta. Se debe tomar en cuenta que el sistema es no lineal y subactuado. Iniciamos definiendo la siguiente ley de control [2]

$$u = -k_1 q_1 - k_2 \dot{q}_1 - k_3 \dot{q}_2 \quad (3.4)$$

donde k_1, k_2, k_3 son las ganancias.

El péndulo es influenciado por la aceleración de la rueda; debido a la física del motor, este se saturará si la velocidad de la rueda es muy alta. Por tanto, es conveniente tratar de lograr el doble objetivo de estabilizar el péndulo y para mantener la velocidad pequeña. Para lograr esto se propuso el uso de los métodos de optimización (PSO y GA) para encontrar los valores óptimos de las ganancias $k_i, i = 1, 2, 3$, para obtener una rápida estabilización en el menor tiempo posible y con un mínimo error en estado estable de $x(q_1)$.

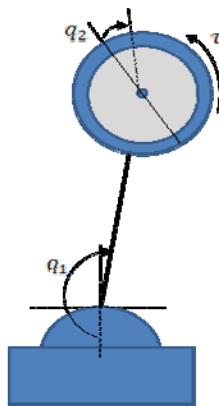


Figura 3.3: Péndulo de rueda de inercia.

Experimento

Los resultados de experimento se basaron en el péndulo de rueda de inercia de laboratorio de Mechatronics Control Kit manufacturado por Quanser Inc., donde $J_1 = 4.572 \times 10^{-3}$, $J_2 = 2.495 \times 10^{-5}$, $h = 0.3544$, $k = 0.00494$, y $\beta = 10$ [2]. Consiste en un péndulo físico (q_1) con un motor/rueda flotante montada en el extremo libre del péndulo (q_2). La rueda es accionada por un motor de corriente directa de imán permanente de 24 Volts, y el par de acoplamiento entre la rueda y el péndulo se puede utilizar para controlar el movimiento del sistema. El ángulo del péndulo es medido por un encoder. La configuración del experimento incluye una PC equipada con una interface C6713 DSK Quanser y una tarjeta amplificadora PWM. El algoritmo fue implementado utilizando lenguaje C. La frecuencia de muestra para el algoritmo de implementación fue de 400 Hz. El coeficiente de viscosidad $f_s = 8.80 \times 10^{-5}$ fue identificado por la aplicación de un procedimiento de [23].

3.1.4 Robot autónomo móvil

El robot móvil utilizado como caso de aplicación, es un vehículo autónomo debe tener la capacidad de seguir trayectorias en ambientes predecibles o inciertos y se muestra en la Figura 3.4 [52], [56], [59], [60].

El cuerpo del robot es simétrico alrededor de un eje perpendicular y el centro de masa está en el centro geométrico del cuerpo [19]. Tiene dos ruedas fijas al eje que atraviesan el punto de referencia C , representado por $\{C, X_m, Y_m\}$, y una rueda orientable pasiva colocada frente al eje.

El plano de movimiento se representa por el eje cartesiano $\{0, X, Y\}$. La posición del robot autónomo móvil en el plano está dada por el vector $q = (x, y, \theta)^T$, donde (x, y) son las coordenadas del punto de referencia C con respecto al eje cartesiano de referencia, y θ es la orientación del robot móvil, con respecto al eje de referencia x .

Las dos ruedas fijas, son controladas por motores independientes y la rueda pasiva es para prevenir que el robot se vuelque conforme se mueva en un plano. Se asume que el movimiento de la rueda pasiva puede ser ignorado en la dinámica del robot móvil.

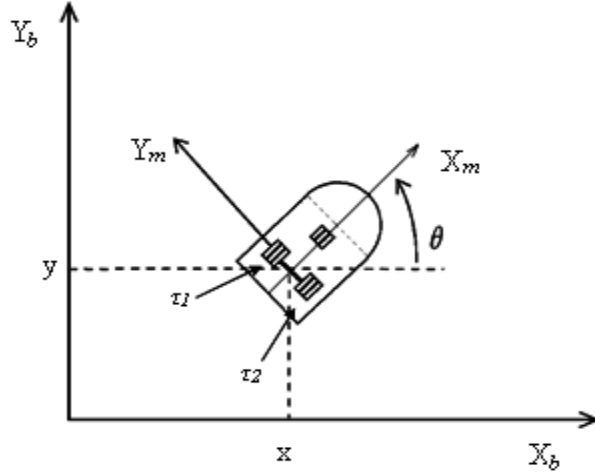


Figura 3.4: Robot autónomo móvil.

La ecuación de movimiento que rige al robot móvil está dada por [46].

$$\dot{q} = \underbrace{\begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix}}_{J(q)} \underbrace{\begin{bmatrix} v \\ w \end{bmatrix}}_{\vartheta}, \quad (3.5)$$

$$M(q)\dot{\vartheta} + C(q, \dot{q})\vartheta + D\vartheta = \tau + P(t) \quad (3.6)$$

donde $q = (x, y, \theta)^T$ es el vector de coordenadas generalizadas, $\vartheta = (v, w)^T$ es el vector de velocidades; $\tau = (\tau_1, \tau_2)$ es el vector de pares aplicados a las ruedas del robot, $M(q) \in \mathbb{R}^{2 \times 2}$ es una matriz de inercia de 2×2 positiva definida; $C(q, \dot{q})\vartheta$ es el vector de fuerzas centrípeta y de Coriolis; $D \in \mathbb{R}^{2 \times 2}$ es una matriz diagonal positiva definida y $P \in \mathbb{R}^2$ es una perturbación uniformemente limitada.

La ecuación (3.5) representa la cinemática del sistema, donde (x, y) es la posición en el mundo de coordenadas $X - Y$; θ es el ángulo entre la dirección del móvil y el eje x ; v y w son las velocidades lineales y angulares, τ_1 y τ_2 son los pares de las ruedas derecha e izquierda, respectivamente [26].

El sistema tiene una restricción no holonómica definida por siguiente ecuación [46]:

$$\dot{x} \cos \theta - \dot{y} \sin \theta = 0 \quad (3.7)$$

Esta limitante indica que las ruedas actuadas solo se mueven en dirección de su eje de rodamiento, y no se deslizan al contacto con el piso, esto limita las velocidades que alcanza el robot móvil en cualquier configuración q [9].

Objetivo de control

La propuesta de control para el robot móvil es la siguiente: Dadas una trayectoria $q_d(t)$ y orientación deseadas, se debe diseñar un controlador difuso que aplique un par adecuado τ , tal que las posiciones medidas $q(t)$ alcancen la trayectoria de referencia $q_d(t)$. Esto se representa por:

$$\lim_{t \rightarrow \infty} \|q_d(t) - q(t)\| = 0. \quad (3.8)$$

Para alcanzar el objetivo de control, se deriva $\tau(t)$ de una $\vartheta_r(t)$ específica que controla el sistema de manejo usando un controlador lógico difuso.

El propósito de este controlador difuso, es alcanzar la estabilidad de un robot móvil en una trayectoria deseada, donde el enfoque del desplazamiento hacia atrás (backstepping) juega un rol muy importante.

El sistema que conforman las ecuaciones (3.5) y (3.6) está en interconexión en cascada; esto es, la cinemática del subsistema (3.5), es controlada solo indirectamente a través del vector de velocidad v . Leyes de estabilización de control para sistemas en forma jerárquica pueden ser diseñadas usando el método de backstepping, el cual consiste en los siguientes dos pasos:

1. Se debe encontrar un vector de velocidad ideal $\vartheta_r = \vartheta$ tal que la cinemática del sistema (3.5) sea estable.
2. Se debe encontrar una entrada τ tal que:

$$\|\vartheta_r(t) - \vartheta(t)\| = 0, \quad \forall t \geq t_s, \quad (3.9)$$

donde t_s es el tiempo de alcance.

En (3.9), consideramos que los robots móviles reales tienen ruedas actuadas, así que la entrada de control es τ ; esto es, la entrada de control real τ es designada para estabilizar la dinámica del sistema (3.6) sin desestabilizar la cinemática del sistema (3.5) forzando $\vartheta \in \mathbb{R}^2$ para alcanzar el vector de velocidad virtual $\vartheta_r \in \mathbb{R}^2$ en un tiempo finito. Entonces, si (3.9) se satisface en asintóticamente ($i, e; t_s = \infty$), entonces ϑ será diferente de v durante $t < \infty$, con la resultante de que el robot móvil no estará ni posicionado ni orientado el punto deseado.

El diagrama a bloques de la Figura 3.5, corresponde al sistema de control del robot autónomo móvil.

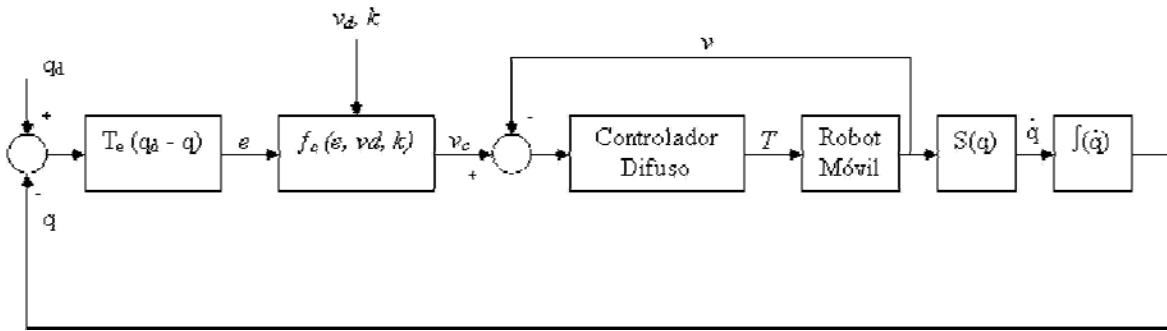


Figura 3.5: Diagrama a bloques del sistema de control.

Control de seguimiento

Para el control de seguimiento se considera la propuesta de Kanayama [32] y Nelson [67]. Suponiendo que la trayectoria de referencia q_d satisface:

$$\dot{q}_d = \begin{pmatrix} \cos \theta_d & 0 \\ \sin \theta_d & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_d \\ w_d \end{pmatrix}, \quad (3.10)$$

donde $\theta_d(t)$ es la orientación deseada, y $v_d(t)$ y $w_d(t)$ son las velocidades lineal y angular deseadas.

Usando el marco local del robot, las coordenadas del error están definidas por la siguiente ecuación:

$$\begin{pmatrix} \dot{\tilde{q}}_1 \\ \dot{\tilde{q}}_2 \\ \dot{\tilde{q}}_3 \end{pmatrix} = \underbrace{\begin{pmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{T_e(\theta)} \begin{pmatrix} x_d - x \\ y_d - y \\ \theta_d - \theta \end{pmatrix}, \quad (3.11)$$

donde $(x_d(t), y_d(t))$ es la posición deseada en las coordenadas del mundo X-Y, \tilde{q}_1 y \tilde{q}_2 son las coordenadas del vector de error de posición, y \tilde{q}_3 es el error de orientación.

El error de seguimiento está dado por la siguiente ecuación:

$$\begin{pmatrix} \tilde{q}_1 \\ \tilde{q}_2 \\ \tilde{q}_3 \end{pmatrix} = \begin{pmatrix} w\tilde{q}_2 - v + v_d \cos \tilde{q}_3 \\ -w\tilde{q}_1 + v_d \sin \tilde{q}_3 \\ w_d - w \end{pmatrix}, \quad (3.12)$$

donde los términos corresponden a la velocidad real y deseada, obtenida diferenciando (3.11) con respecto al tiempo .

Para presentar el resultado de esta subdivisión, debemos establecer los siguientes teoremas [10].

Teorema 1: [41] (*estabilidad uniforme*). Sea el origen $x = 0$ un punto de equilibrio de $\dot{x} = f(x, t)$ y $D \subset \mathbb{R}^n$ contenido en el dominio del origen $x = 0$. Sea $V: [0, \infty] \times D \rightarrow \mathbb{R}$ una función continuamente diferenciable tal que:

$$W_1(x) \leq V(x, t) \leq W_2(x), \quad (3.13)$$

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(x, t) \leq 0, \quad (3.14)$$

para toda $t \geq 0$ y para toda $x \in D$, donde $W_1(x)$ y $W_2(x)$ son funciones continuas positivas definidas en D . Entonces, el origen $x = 0$ es uniformemente estable.

Teorema 2: [41] (*estabilidad asintóticamente uniforme*). Asumiendo que las suposiciones del Teorema 1 se satisfacen, sujeto a (3.14) entonces:

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(x, t) \leq -W_3(x) \quad (3.15)$$

para toda $t \geq 0$ y para toda $x \in D$, donde $W_3(x)$ es una función continua positiva definida en D . Entonces el origen $x = 0$ es asintóticamente estable uniforme.

Teorema 3: Sean las ecuaciones del error de seguimiento (3.12), gobernadas por la ley de control (velocidades virtuales)

$$\begin{aligned} v_r &= v_d \cos \tilde{q}_3 + \gamma_1 \tilde{q}_1 \\ w_r &= w_d + \gamma_2 v_d \tilde{q}_2 + \gamma_3 \sin \tilde{q}_3 \end{aligned} \quad (3.16)$$

donde γ_1, γ_2 y γ_3 son constantes positivas. Si $v = v_r$ y para toda $w = w_r$ en (3.6), entonces, el origen del sistema de lazo-cerrado es asintóticamente estable en forma uniforme.

Prueba: bajo el control (3.16), el sistema de lazo-cerrado toma la forma de:

$$\begin{pmatrix} \dot{\tilde{q}}_1 \\ \dot{\tilde{q}}_2 \\ \dot{\tilde{q}}_3 \end{pmatrix} = \begin{pmatrix} w_d \tilde{q}_2 + \gamma_2 v_d \tilde{q}_2^2 + \gamma_3 \tilde{q}_2 \sin \tilde{q}_3 - \gamma_1 \tilde{q}_1 \\ -w_d \tilde{q}_1 - \gamma_2 v_d \tilde{q}_1 \tilde{q}_2 - \gamma_3 \tilde{q}_1 \sin \tilde{q}_3 + v_d \sin \tilde{q}_3 \\ -\gamma_2 v_d \tilde{q}_2 - \gamma_3 \sin \tilde{q}_3 \end{pmatrix} \quad (3.17)$$

tal que, el origen $(\tilde{q}_1, \tilde{q}_2, \tilde{q}_3)^T = 0$ es un punto de equilibrio para el sistema de lazo-cerrado, pero no único, porque \tilde{q}_3 adopta diferentes posiciones ($i, e; \tilde{q}_3 = 0, \pi, \dots, n\pi$). El teorema de estabilidad asintótica invoca una pauta para obtener los límites de los valores de γ_i , el cual garanticen la convergencia del error $\tilde{q} \in \mathbb{R}^3$ a cero. Para este propósito introducimos la función candidata de Lyapunov:

$$V(\tilde{q}) = \frac{1}{2} \tilde{q}_1^2 + \frac{1}{2} \tilde{q}_2^2 + \frac{1}{\gamma_2} (1 - \cos \tilde{q}_3) \quad (3.18)$$

la cual es positiva definida. Tomando en cuenta la derivada de $V(\tilde{q})$ con respecto al tiempo, a lo largo de la solución del sistema de lazo-cerrado (3.17), obtenemos:

$$\dot{V}(\tilde{q}) = \tilde{q}_1 \dot{\tilde{q}}_1 + \tilde{q}_2 \dot{\tilde{q}}_2 + \frac{1}{\gamma_2} \dot{\tilde{q}}_3 \sin \tilde{q}_3 = -\gamma_1 \tilde{q}_1^2 - \frac{\gamma_3}{\gamma_2} (\sin \tilde{q}_3)^2 \leq 0 \quad (3.19)$$

Concluyendo entonces, que para cualquier constante positiva γ_i , el sistema de lazo-cerrado es uniformemente estable. Para completar la prueba nótese que $\tilde{q}_1, \tilde{q}_2, \tilde{q}_3 \in L_\infty^n$; $\dot{\tilde{q}}_1, \dot{\tilde{q}}_2 \in L_\infty^n$; y $\dot{\tilde{q}}_3 \in L_\infty^n$ donde

$$\begin{aligned}
L_2^n &= \{x(t): \mathbb{R}_+ \mapsto \mathbb{R}^n \mid \|x(t)\|_2^2 = \int_0^\infty \|x(t)\|_2^2 dt < \infty\} \\
L_\infty^n &= \{x(t): \mathbb{R}_+ \mapsto \mathbb{R}^n \mid \|x(t)\|_\infty^2 = \sup \|x(t)\|_2^2 < \infty\}
\end{aligned}
\tag{3.20}$$

por lo tanto concluimos, que aplicando el lema de Barbalat, \tilde{q}_1 y \tilde{q}_3 convergen en el origen. Finalmente invocando el Teorema de Matrosov, la convergencia se concluye el \tilde{q}_2 al origen.

Una vez descritos los casos de aplicación continuamos con la descripción de los métodos de optimización con la configuración utilizada para cada caso de aplicación.

A continuación se describen los métodos de optimización utilizados para encontrar controladores difusos y búsqueda de parámetros óptimos de controladores tradicionales PID.

3.2 Algoritmos Genéticos

El primer método de optimización utilizado en este trabajo de tesis fueron los algoritmos genéticos ya que se tiene más tiempo trabajando con ellos en el área de controladores difusos. Se escogieron varias plantas de diferentes grados de complejidad las cuales con ayuda de los algoritmos genéticos obtuvimos controladores difusos óptimos de tipo-1 y tipo-2.

Planta1

Diseño del controlador difuso tipo-1

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de la planta 1, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error (e) y la derivada del error (ev), fueron tomadas como variables de entrada, y una variable de Salida (S) [51], [55].

De acuerdo a trabajos previos con controladores difusos se decidió tomar el mismo tipo de funciones de membresía por la semejanza entre los mismos. Las funciones de membresía son para las variables de entrada son de tipo Gaussiana para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (Z). De igual forma se decidió probar con las funciones de membresía de tipo gaussiana para todos los términos lingüísticos Negativo (N), Cero (Z) y Positivo (P) de las variables de entrada.

El intervalo usado para los dos tipos de controlador difuso fué de $[-2 \ 2]$.

Las Figura 3.6 A) muestran las variables de entrada del controlado difuso tipo-1 catalogado como GTG (Gaussiana-Triangular-Gaussiana) y B) muestra las funciones de membresía del controlador difuso tipo-1 catalogado como GGG (Gaussiana-Gaussiana-Gaussiana).

Diseño del controlador difuso tipo-2

Para el controlador difuso tipo-2 se utilizó el mismo diseño que el caso del controlador tipo-1 ya que se tomó el mejor controlador tipo 1, la cual sirvió como base para buscar solo la incertidumbre y así obtener el controlador difuso tipo-2 óptimo, de igual forma se utilizaron los tipos de controladores difusos mencionados anteriormente [51], [55].

Las Figura 3.6 muestran las variables de entrada del controlado difuso tipo-2 catalogado como C) GTG (Gaussiana-Triangular-Gaussiana) y D) muestra las funciones de membresía del controlador difuso tipo-2 catalogado como GGG (Gaussiana-Gaussiana-Gaussiana).

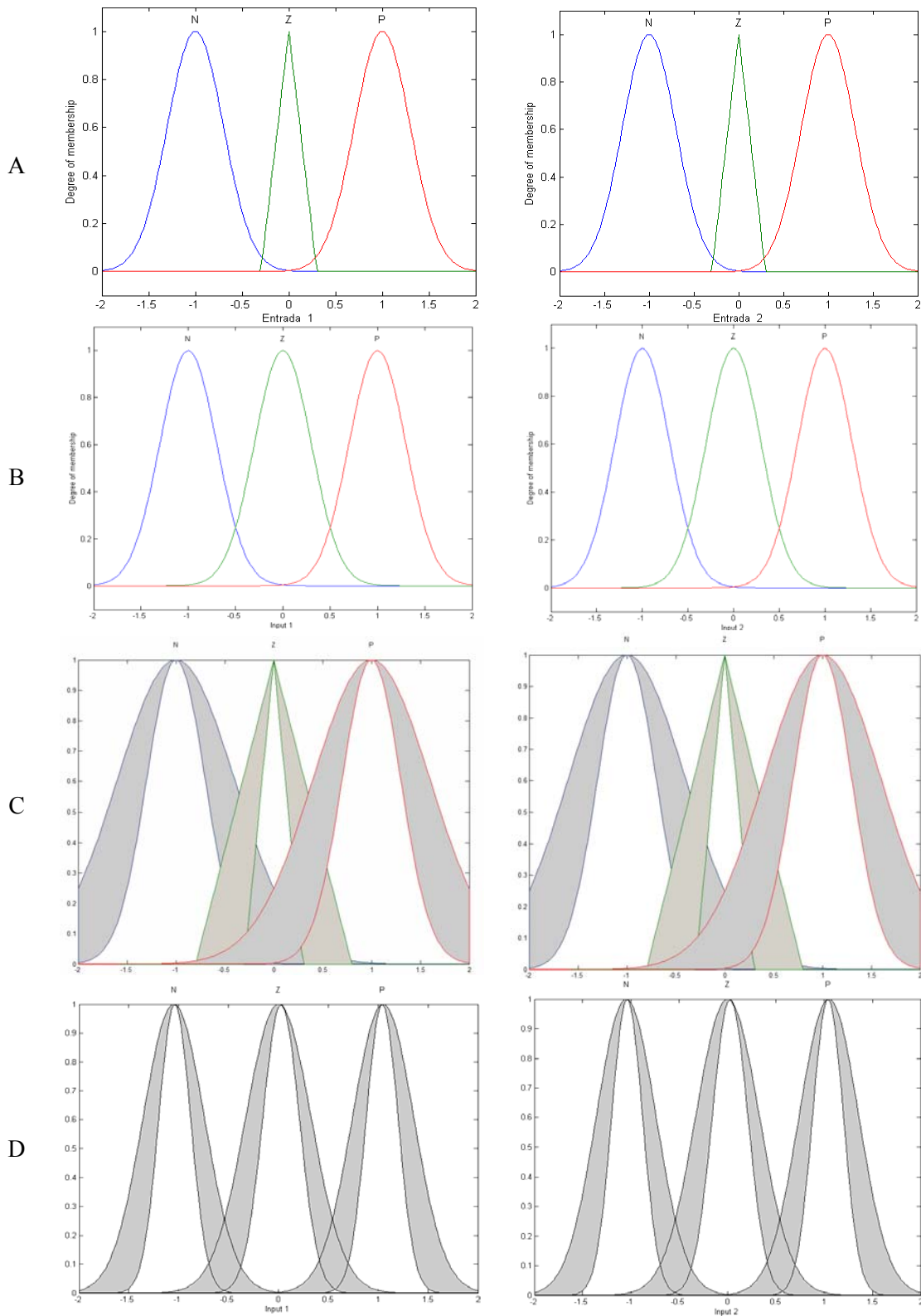


Figura 3.6: Entrada 1 (error "e") y entrada 2 (derivada error "ev") del controlador difuso tipo-1 A) GTG y B) GGG y entrada 1 (error "e") y entrada 2 (derivada error "ev") del controlador difuso tipo-2 C) GTG y D) GGG.

Configuración del algoritmo genético

Las variables de las funciones de membresía para el controlador difuso tipo GTG se representó con un cromosoma de 17 posiciones de valores reales, las primeras 14 posiciones representan los valores de los parámetros de las funciones de membresía gaussiana y triangular de las variables de entrada y las 3 posiciones restantes son para la variable de salida, se variaron los valores de los parámetros de cruce y mutación y el tipo de cruce fué de un solo punto. La representación gráfica del cromosoma se muestra en la Figura 3.7.

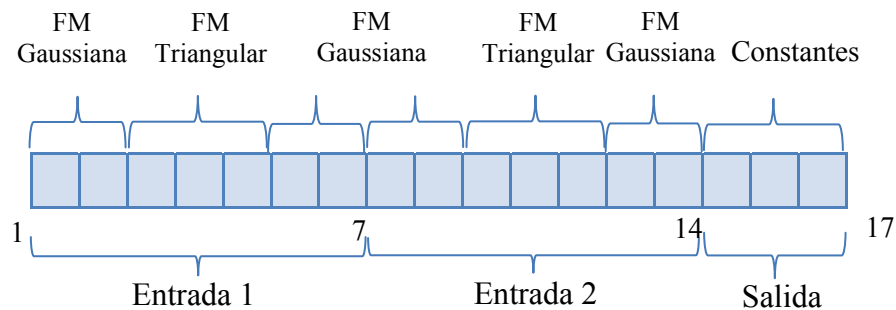


Figura 3.7: Representación del cromosoma para los parámetros de las funciones de membresía.

Para las variables de las funciones de membresía para el controlador difuso tipo GGG se representó con un cromosoma de 15 posiciones de valores reales, las primeras 12 posiciones representan los valores de los parámetros de las funciones de membresía gaussiana y triangular de las variables de entrada y las 3 posiciones restantes son para la variable de salida, se variaron los valores de los parámetros de cruce y mutación y el tipo de cruce fué de un solo punto. La representación gráfica del cromosoma se muestra en la Figura 3.8.

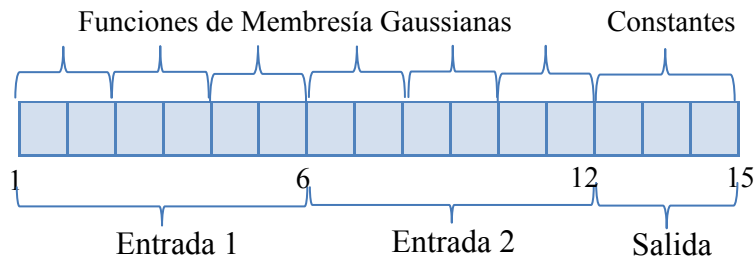


Figura 3.8: Representación del cromosoma para los parámetros de las funciones de membresía.

Limites inferiores y superiores de los parámetros de las funciones de membresía

Tabla 3.1 se presenta los límites inferiores y superiores utilizados para la búsqueda de los parámetros de las funciones de membresía de los controladores difusos tipo GTG y GGG para la planta 1.

La Figura 3.9 muestra los tipos de funciones de membresía utilizados para el controlador difuso de tipo A) GTG y B) GGG.

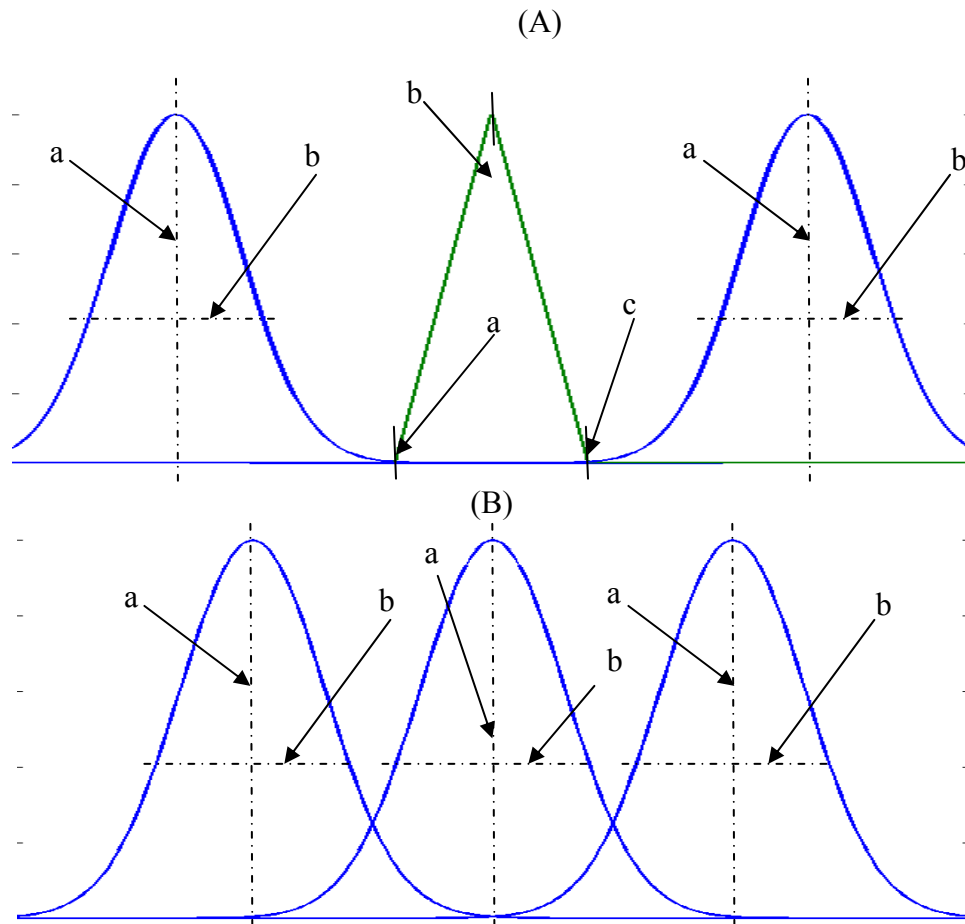


Figura 3.9: Tipos de funciones de membresía utilizados A) GTG y B) GGG.

En la Tabla 3.1 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía (GTG).

Tabla 3.1: Límites inferiores y superiores de las funciones de membresía (GTG).

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	0.3	0.6
	b	-1	-1
Triangular	a	-0.3	-0.8
	b	0	0
	c	0.3	0.8
Gaussiana	a	0.3	0.6
	b	1	1

En la Tabla 3.2 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía (GGG).

Tabla 3.2: Límites inferiores y superiores de las funciones de membresía (GGG).

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	0.3	0.6
	b	-1.25	-0.75
Gaussiana	a	0.1	0.4
	b	0	0
Gaussiana	a	0.3	0.6
	b	0.75	1.25

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con el algoritmo genético y la planta desarrollada en Simulink®. El objetivo es minimizar el error obtenido por la diferencia entre la referencia y el resultado del controlador.

Reglas difusas del controlador difuso

Se utilizaron 9 reglas difusas para este controlador debido a las 3 funciones de membresía por cada entrar que tenemos y éstas fueron usadas en base a la experiencia en otros controladores similares estudiados anteriormente.

Las 9 reglas difusas IF- THEN utilizadas son las siguientes:

- R1: Si error es N y derivadaerror es N, Entonces S es N
- R2: Si error es N y derivadaerror es Z, Entonces S es N

- R3: Si error es N y derivadaerror es P, Entonces S es Z
- R4: Si error es Z y derivadaerror es N, Entonces S es N
- R5: Si error es Z y derivadaerror es Z, Entonces S es Z
- R6: Si error es Z y derivadaerror es P, Entonces S es P
- R7: Si error es P y derivadaerror es N, Entonces S es Z
- R8: Si error es P y derivadaerror es Z, Entonces S es P
- R9: Si error es P y derivadaerror es P, Entonces S es P

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.3, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.3: Reglas difusas.

e / e_v	N	Z	P
N	N/N	N/Z	N/P
Z	Z/N	Z/Z	Z/P
P	P/N	P/Z	P/P

Planta2

Diseño del controlador difuso tipo-1

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de la planta 2, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error (e) y la derivada del error (e_v), fueron tomadas como variables de entrada, y una variable de Salida (S) [51], [55].

Las funciones de membresía de las variables de entrada son de tipo Gaussiana para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (Z). De igual forma se decidió probar con las funciones de membresía de tipo gaussiana para todos los términos lingüísticos Negativo (N), Cero (Z) y Positivo (P) de las variables de entrada. El intervalo usado para los dos tipos de controlador difuso fué de [-10,10].

Las Figuras 3.10 A) muestran las variables de entrada del controlador difuso tipo-1 catalogado como GTG (Gaussiana-Triangular-Gaussiana) y B) muestran las variables de entrada del controlado difuso tipo-1 catalogado como GGG (Gaussiana-Gaussiana-Gaussiana).

Diseño del controlador difuso tipo-2

Para el controlador difuso tipo-2 se utilizo el mismo diseño que el caso del controlador tipo-1 ya que se tomo el mejor controlador tipo 1, la cual sirvió como base para buscar sólo la incertidumbre y así obtener el controlador difuso tipo-2 óptimo, de igual forma se utilizaron los tipos de controladores difusos mencionados anteriormente [51], [55].

Las Figuras 3.10 C) muestran las variables de entrada del controlado difuso tipo-2 catalogado como GTG (Gaussiana-Triangular-Gaussiana) y D) muestran las variables de entrada del controlado difuso tipo-2 catalogado como GGG (Gaussiana-Gaussiana-Gaussiana).

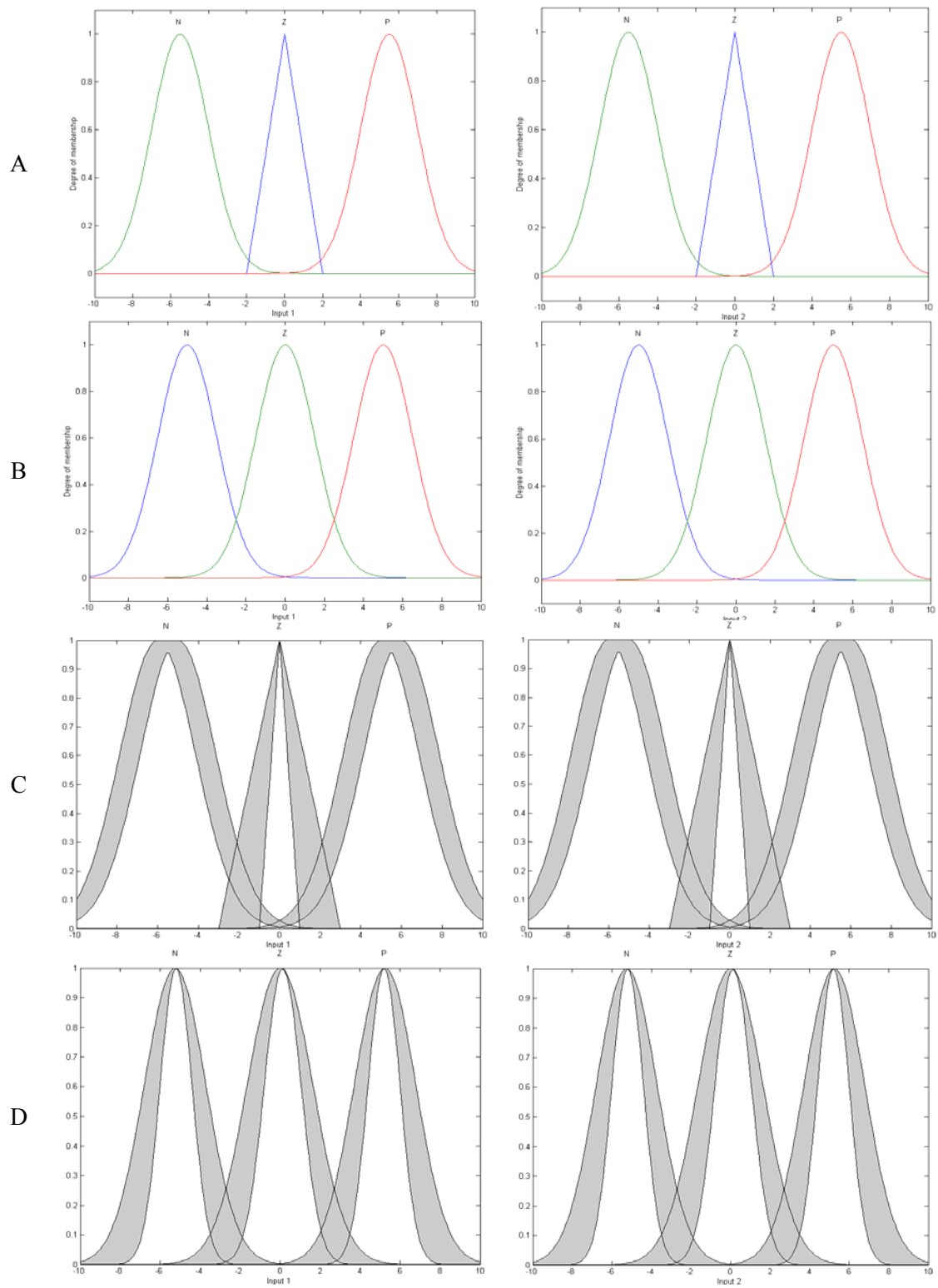


Figura 3.10: Entrada 1 (error "e") y entrada 2 (derivada de error "ev") del controlador difuso tipo-1 A) GTG y B) GGG y entrada 1 (error "e") y entrada 2 (derivada de error "ev") del controlador difuso tipo-2 C) GTG y D) GGG.

Configuración del algoritmo genético

Las variables de las funciones de membresía se representó con un cromosoma de 17 posiciones de valores reales, las primeras 14 posiciones representan los valores de los parámetros de las funciones de membresía gaussiana y triangular de las variables de entrada y las 3 posiciones restantes son para la variable de salida, se variaron los valores de los parámetros de cruce y mutación y el tipo de cruce fué de un sólo punto. La representación gráfica del cromosoma se muestra en la Figura 3.11.

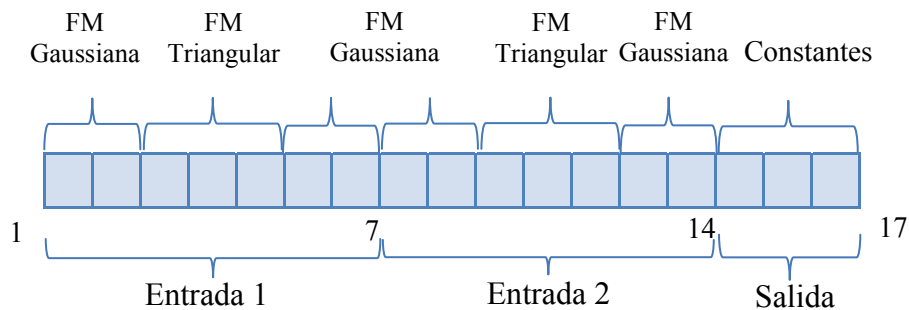


Figura 3.11: Representación del cromosoma para los parámetros de las funciones de membresía.

Para las variables de las funciones de membresía para el controlador difuso tipo GGG se representó con un cromosoma de 15 posiciones de valores reales, las primeras 12 posiciones representan los valores de los parámetros de las funciones de membresía gaussiana y triangular de las variables de entrada y las 3 posiciones restantes son para la variable de salida, se variaron los valores de los parámetros de cruce y mutación y el tipo de cruce fué de un sólo punto. La representación gráfica del cromosoma se muestra en la Figura 3.12.

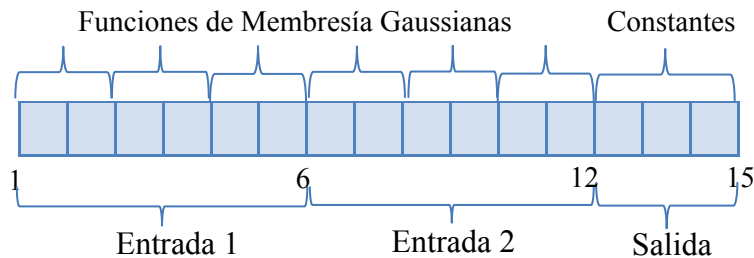


Figura 3.12: Representación del cromosoma para los parámetros de las funciones de membresía.

Limites inferiores y superiores de los parámetros de las funciones de membresía

Tabla 3.4 se presenta los límites inferiores y superiores utilizados para la búsqueda de los parámetros de las funciones de membresía de los controladores difusos para la planta 1. La Figura 3.13 muestra los tipos de funciones de membresía utilizados para el controlador difuso A) GTG y B) GGG.

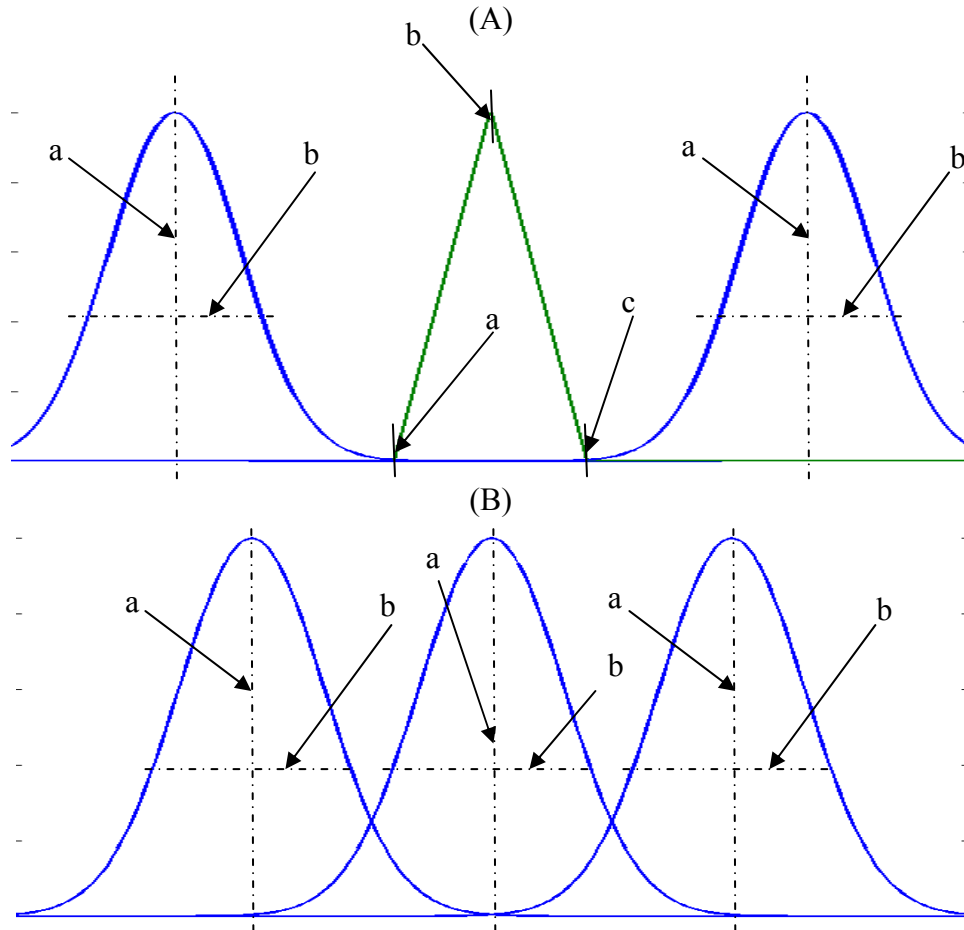


Figura 3.13: Tipos de funciones de membresía utilizados A) GTG y B) GGG.

En la Tabla 3.4 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía.

Tabla 3.4: Límites inferiores y superiores de las funciones de membresía.

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	1.8	2.8
	b	-5	-5
Triangular	a	-0.5	-4
	b	0	0
	c	0.5	4
Gaussiana	a	1.8	2.8
	b	5	5

En la Tabla 3.5 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía (GGG).

Tabla 3.5: Límites inferiores y superiores de las funciones de membresía (GGG).

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	1.8	2.8
	b	-4	-6
Gaussiana	a	0.5	1.5
	b	0	0
Gaussiana	a	1.8	2.8
	b	4	6

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con el algoritmo genético y la planta desarrollada en Simulink®. El objetivo es minimizar el error obtenido por la diferencia entre la referencia y el resultado del controlador.

Reglas difusas del controlador

Se utilizaron 9 reglas difusas para este controlador debido a las 3 funciones de membresía por cada entrar que tenemos y éstas fueron usadas en base a la experiencia en otros controladores similares estudiados anteriormente.

Las 9 reglas difusas IF- THEN utilizadas son las siguientes:

- R1: Si error es N y derivadaerror es N, Entonces S es N
- R2: Si error es N y derivadaerror es Z, Entonces S es N
- R3: Si error es N y derivadaerror es P, Entonces S es Z
- R4: Si error es Z y derivadaerror es N, Entonces S es N
- R5: Si error es Z y derivadaerror es Z, Entonces S es Z
- R6: Si error es Z y derivadaerror es P, Entonces S es P
- R7: Si error es P y derivadaerror es N, Entonces S es Z
- R8: Si error es P y derivadaerror es Z, Entonces S es P
- R9: Si error es P y derivadaerror es P, Entonces S es P

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.6, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.6: Reglas difusas.

e / e_v	N	Z	P
N	N/N	N/Z	N/P
Z	Z/N	Z/Z	Z/P
P	P/N	P/Z	P/P

Péndulo de rueda de inercia

Configuración del algoritmo genético

En el caso del péndulo los algoritmos genéticos no se utilizaron para crear un controlador difuso, sino que se utilizó para buscar los valores óptimos de las ganancias del controlador pid del péndulo de rueda de inercia. Para esto se utilizó un algoritmo genético de 3 posiciones de valores reales para las variables llamadas gain 1, gain 2 y gain 3 del controlador PID para estabilizar el péndulo [61], [62].

En la configuración del algoritmo genético se fueron cambiando el número de individuos, generaciones, se variaron los valores de mutación y cruce de tipo simple, usando como método de selección de tipo ruleta, y un número máximo de generaciones utilizado como sistema de paro del algoritmo.

La representación del cromosoma es la siguiente:

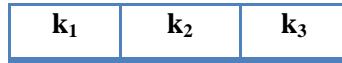


Figura 3.14: Representación del cromosoma del algoritmo genético.

Se utilizaron un rango de valores para la búsqueda de cada ganancia (k_1 , k_2 y k_3) y la Tabla 3.7 muestra los valores mínimos y máximos basándonos en las condiciones físicas del sistema real.

Tabla 3.7: Límites inferiores y superiores del rango de búsqueda de las ganancias del péndulo.

	Valores mínimos	Valores máximos
K_1	50	250
K_2	5	50
K_3	0.001	0.1

Robot autónomo móvil

Diseño del controlador difuso

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de seguimiento del robot autónomo móvil, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error de las velocidades lineal (v_d) y angular (w_d), que no son más que la diferencia entre la velocidad deseada y la velocidad real del móvil, fueron tomadas como variables de entrada, y como variables de salida se tomó el par derecho (τ_1) e izquierdo (τ_2) [52], [56], [59], [60].

Las funciones de membresía de las variables de entrada son de tipo trapezoidal para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (C). El intervalo usado para el controlador difuso fué de [-50 50].

Las Figuras 3.15 muestran las variables de entrada y variables de salida.

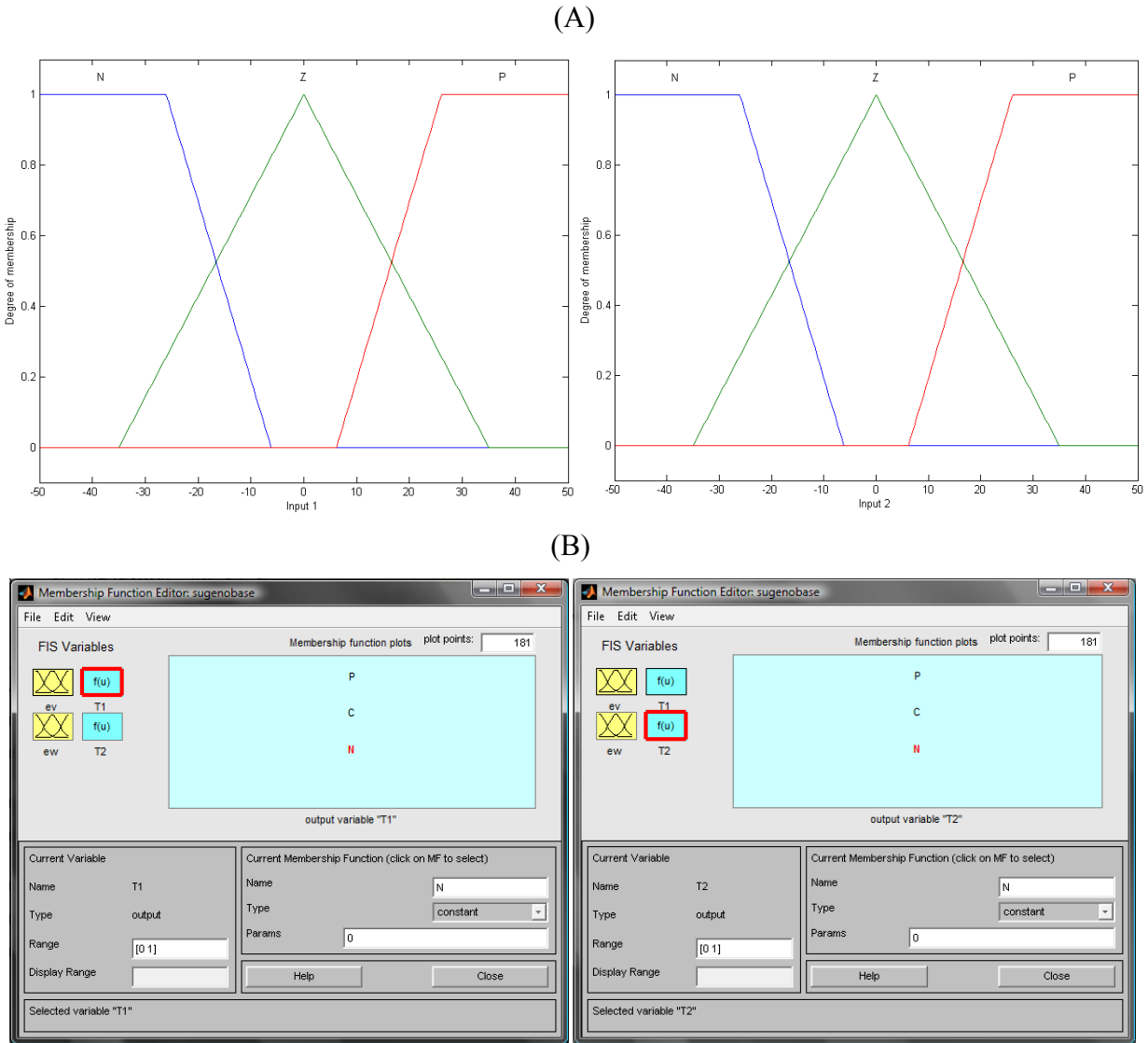


Figura 3.15: A) Error en la velocidad lineal y error de velocidad angular y B) Torque derecho y torque izquierdo.

Las 9 reglas difusas utilizadas son las siguientes:

- R1: Si V_{Lineal} es Z y $V_{angular}$ es Z , Entonces $T1$ es C y $T2$ es C
- R2: Si V_{Lineal} es Z y $V_{angular}$ es P , Entonces $T1$ es C y $T2$ es P
- R3: Si V_{Lineal} es Z y $V_{angular}$ es N , Entonces $T1$ es C y $T2$ es N
- R4: Si V_{Lineal} es P y $V_{angular}$ es Z , Entonces $T1$ es P y $T2$ es C
- R5: Si V_{Lineal} es P y $V_{angular}$ es P , Entonces $T1$ es P y $T2$ es P
- R6: Si V_{Lineal} es P y $V_{angular}$ es N , Entonces $T1$ es P y $T2$ es N
- R7: Si V_{Lineal} es N y $V_{angular}$ es Z , Entonces $T1$ es N y $T2$ es C

- R8: Si VLineal es N y Vangular es P, Entonces T1 es N y T2 es P
- R9: Si VLineal es N y Vangular es N, Entonces T1 es N y T2 es N

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.8, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.8: Reglas difusas.

e_v / e_w	N	C	P
N	N/N	N/C	N/P
C	C/N	C/C	C/P
P	P/N	P/C	P/P

Simulación del robot autónomo móvil

Una vez que se diseñó el controlador, se hizo la simulación tomando en cuenta las Ecuaciones (3.5),(3.6), (3.10), (3.11) y (3.12). La Figura 3.16 muestra el sistema de control hecho en Simulink®.

Las condiciones para la simulación son las siguientes:

$$M(q) = \begin{bmatrix} 0.3749 & -0.0202 \\ -0.0202 & 0.3739 \end{bmatrix}, \quad D = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, \quad C(q, \dot{q}) = \begin{bmatrix} 0 & 0.1350\dot{\theta} \\ -0.1350\dot{\theta} & 0 \end{bmatrix} \quad (3.21)$$

La trayectoria deseada es la siguiente:

$$\vartheta_d(t) = \begin{cases} v_d(t) = 0.2(1 - \exp(-t)) \\ w_d(t) = 0.4 \sin(0.5t) \end{cases}. \quad (3.22)$$

Configuración de algoritmos genéticos para el control de seguimiento

Se usaron algoritmos genéticos para optimizar los parámetros de las funciones de membresía de entrada y salida del controlador difuso. El objetivo es que en la simulación, los errores en la velocidad lineal y angular sean cero.

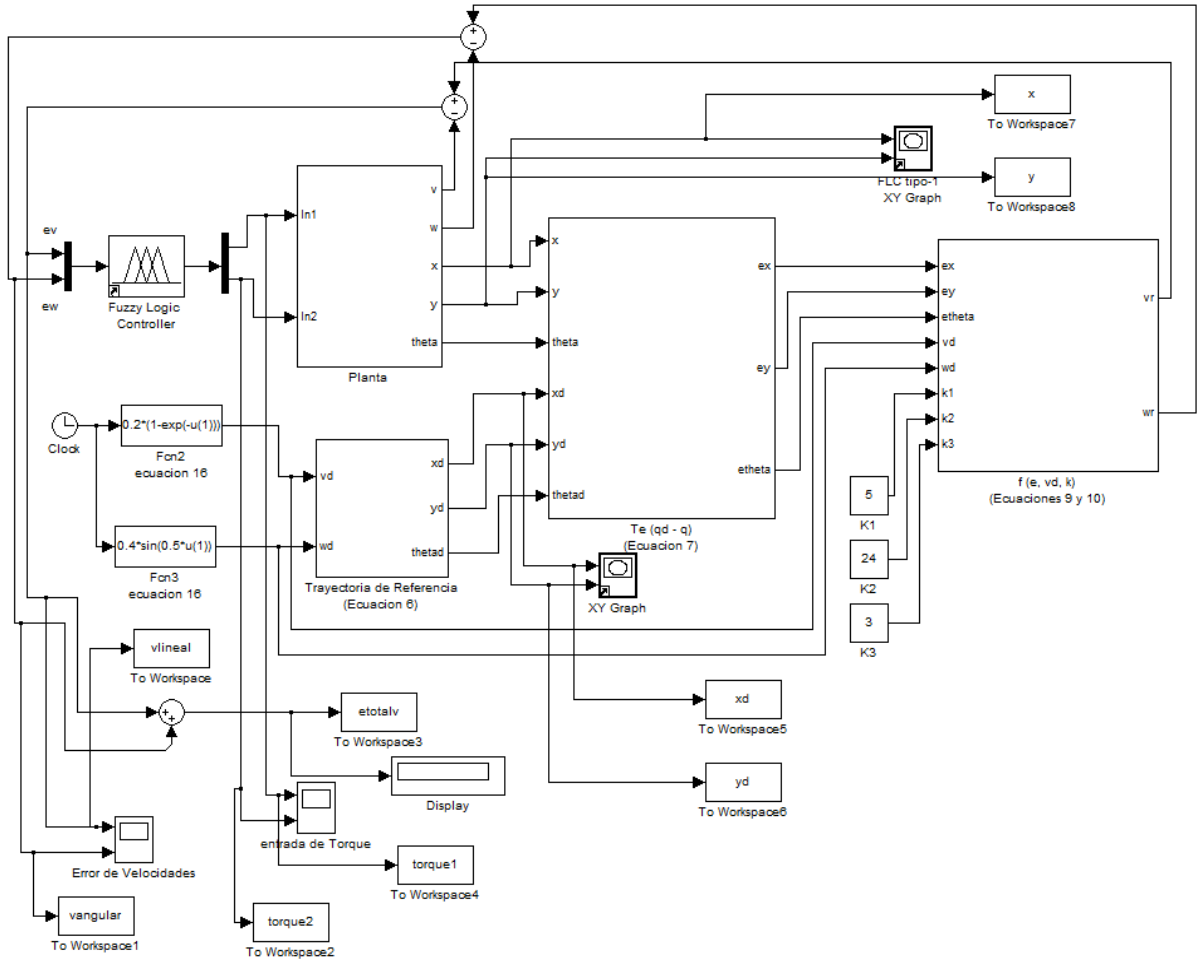


Figura 3.16: Sistema de control en Simulink®.

Representación del cromosoma

El cromosoma contiene 28 posiciones, las primeras 22 posiciones representan los valores de los parámetros de las funciones de membresía trapezoidal y triangular de las variables de entrada y las 6 posiciones restantes son para las variables de salida. La representación gráfica del cromosoma se muestra en la Figura 3.17.

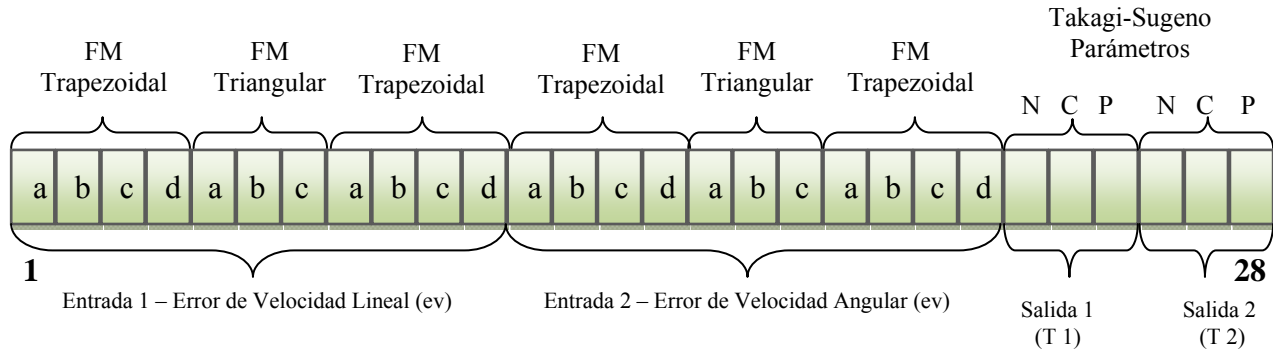


Figura 3.17: Representación del cromosoma para los parámetros de las funciones de membresía.

La Figura 3.18 muestra los tipos de funciones de membresía utilizados para el controlador difuso.

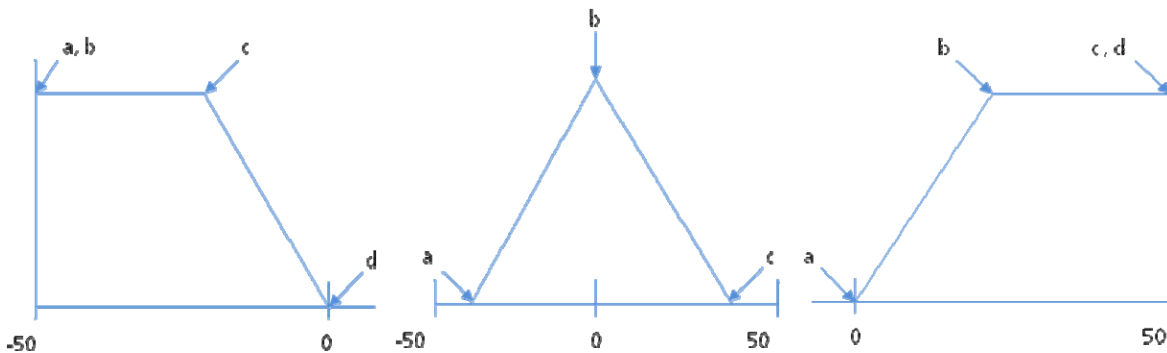


Figura 3.18: Tipos de funciones de membresía utilizados.

En la Tabla 3.9 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía.

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con el algoritmo genético y el sistema en Simulink®. El objetivo es minimizar los errores de las velocidades lineal y angular del sistema. Los valores usados para k_1 , k_2 y k_3 son 5, 24 y 3, respectivamente.

Tabla 3.9: Límites inferiores y superiores de las funciones de membresía.

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Trapezoidal	a	-50	-50
	b	-50	-50
	c	-15	-5.05
	d	-1.5	-0.5
Triangular	a	-5	-1.75
	b	0	0
	c	1.75	5
Trapezoidal	a	0.5	1.5
	b	5.05	15
	c	50	50
	d	50	50

Diseño de un controlador difuso Tipo-2

Tomando como referencia el controlador difuso de Tipo-1 que fue considerado el mejor, se diseñó un controlador difuso Tipo-2, en el cual los valores de las funciones de membresía primarias fueron las mismas que las de Tipo-1. La idea de extender el controlador difuso de Tipo-1 a Tipo-2, es compararlos entre sí agregando perturbación a la planta.

Los parámetros para la amplitud de la huella de incertidumbre fueron optimizados con el uso de algoritmos genéticos. Los resultados obtenidos para la optimización de los parámetros de las funciones de membresía secundarias.

3.3 Optimización por medio de nube de partículas (Particle swarm optimization)

Optimización por medio de nube de partículas es otro de los métodos de optimización que se estudiaron en esta tesis, aplicados a controladores difusos y búsqueda de parámetros de controladores clásicos como los PID. De igual manera se hicieron uso de los casos de aplicación utilizados con los algoritmos genéticos que se muestran a continuación.

Planta 1

Diseño del controlador difuso tipo-1

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de la planta 1, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error (e) y la derivada del error (ev), fueron tomadas como variables de entrada, y una variable de Salida (S) [53], [54], [57].

De acuerdo a trabajos previos con controladores difusos se decidió tomar el mismo tipo de funciones de membresía por la semejanza entre los mismos. Las funciones de membresía son para las variables de entrada son de tipo Gaussiana para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (Z). De igual forma se decidió probar con las funciones de membresía de tipo gaussiana para todos los términos lingüísticos Negativo (N), Cero (Z) y Positivo (P) de las variables de entrada. El intervalo usado para los dos tipos de controlador difuso fue de $[-22]$.

La Figura 3.19 A) muestran las variables de entrada del controlado difuso tipo-1 catalogado como *GTG* (Gaussiana-Triangular-Gaussiana), B) muestran las variables de entrada del controlado difuso tipo-1 catalogado como *GGG* (Gaussiana-Gaussiana-Gaussiana) y C) muestran las variables de entrada del controlado difuso tipo-2 catalogado como *GTG*.

Diseño del controlador difuso tipo-2

Para el controlador difuso tipo-2 se utilizó el mismo diseño que el caso del controlador tipo-1 ya que se tomó el mejor controlador tipo 1, la cual sirvió como base para buscar sólo la incertidumbre y así obtener el controlador difuso tipo-2 óptimo, de igual forma se utilizaron los tipos de controladores difusos mencionados anteriormente.

Las Figuras 3.19 C) muestran las variables de entrada del controlado difuso tipo-2 catalogado como *GTG*.

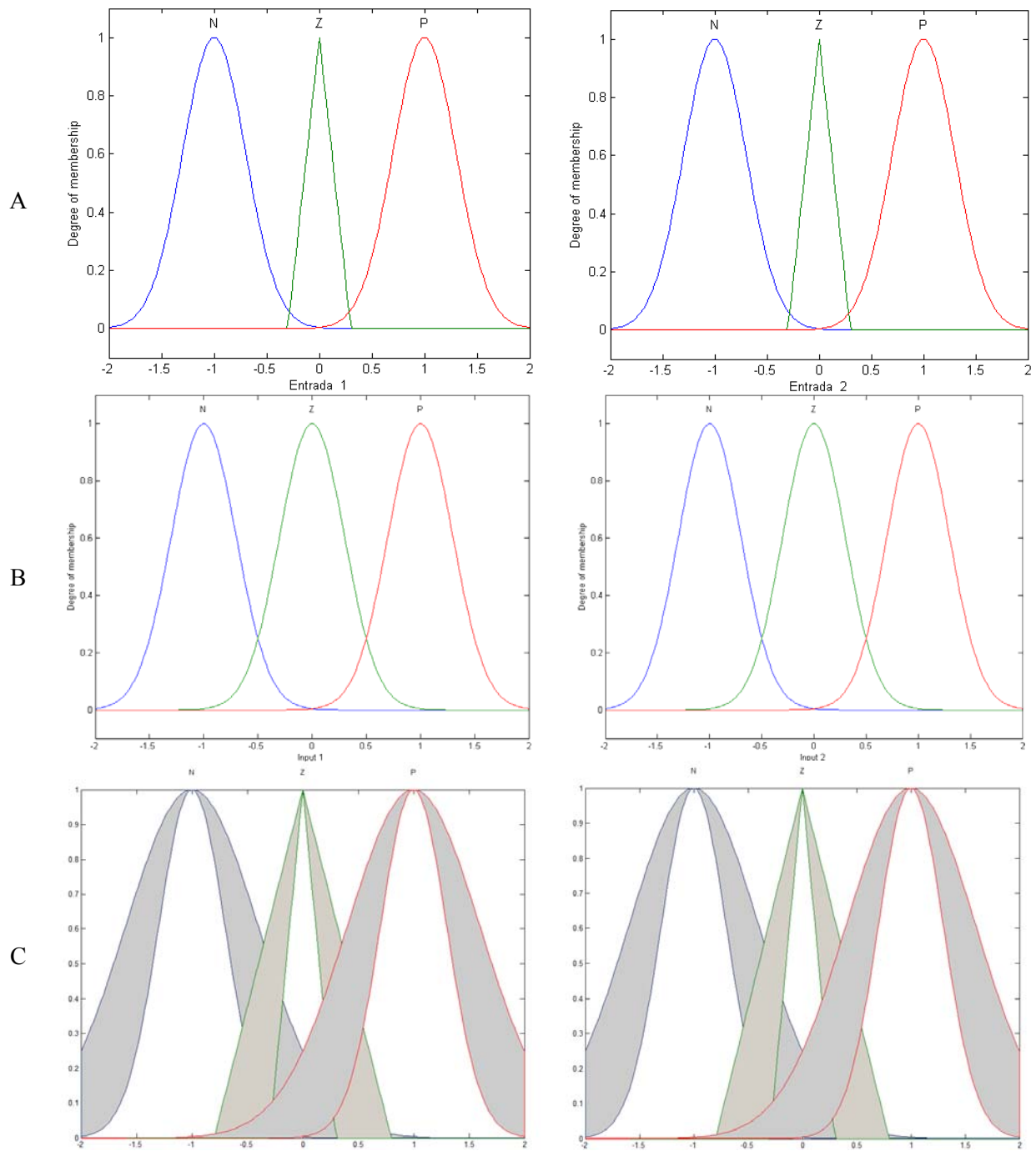


Figura 3.19: Entrada 1 (error "e") y entrada 2 (derivada error "ev") del controlador difuso tipo-1 A) GTG y B) GGG y entrada 1 (error "e") y entrada 2 (derivada error "ev") del controlador difuso tipo-2 C) GTG.

Configuración del PSO

Se utilizó el método de optimización PSO para encontrar el mejor controlador difuso tipo -1 y tipo-2 para la planta 1. Se asignaron valores a las variables cognitivas, social y de inercia de

PSO. Para las variable cognitiva y social se asignaron de forma manual con rangos de 0.001 a 1y el rango para la variable de inercia es 0.005 a 1; que más adelante se mostraran los resultados obtenidos, después se le asigno de manera aleatoria valores a la variables de PSO utilizando los rangos mencionados anteriormente.

Límites inferiores y superiores de los parámetros de las funciones de membresía

Tabla 3.10 presenta los límites inferiores y superiores utilizados para la búsqueda de los parámetros de las funciones de membresía de los controladores difusos tipo *GTG* y *GGG* para la planta 1.

La Figura 3.20 muestra los tipos de funciones de membresía utilizados para el controlador difuso de tipo GTG.

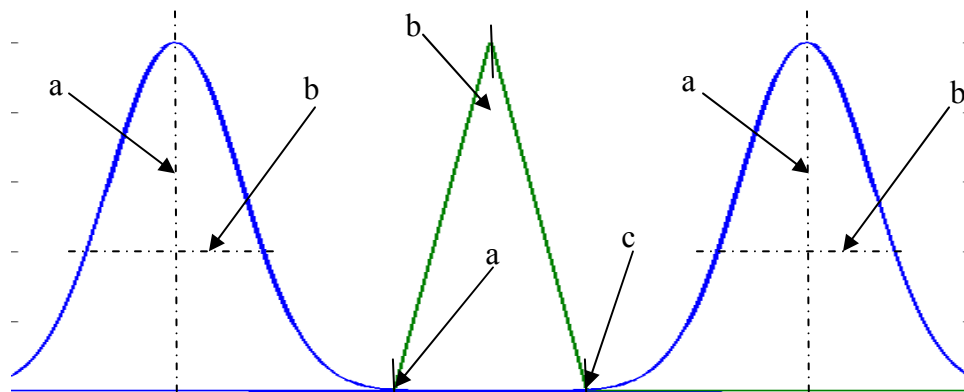


Figura 3.20 Tipos de funciones de membresía utilizados GTG.

En la Tabla 3.10 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía del controlador difuso tipo-1 (GTG).

Tabla 3.10: Límites inferiores y superiores de las funciones de membresía (GTG).

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	0.3	0.6
	b	-1.2	-0.8
Triangular	a	-0.8	-0.3
	b	0	0
	c	0.3	0.8
Gaussiana	a	0.3	0.6
	b	0.8	1.2

En la Tabla 3.11 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía del controlador difuso tipo-2 (GTG).

Tabla 3.11: Límites inferiores y superiores de las funciones de membresía (GTG).

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	0.13	0.14
	b	-1.2	-1
	a'	0.22	0.43
	b'	-1.2	-1
Triangular	a	-1	-0.75
	b	0	0
	c	0.15	0.25
	a'	-0.75	-0.25
	b'	0	0
	c'	0.5	0.75
Gaussiana	a	0.13	0.14
	b	1	1.2
	a'	0.22	0.43
	b'	1	1.2

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con el algoritmo genético y la planta desarrollada en Simulink®. El objetivo es minimizar el error obtenido por la diferencia entre la referencia y el resultado del controlador.

Reglas difusas del controlador difuso

Se utilizaron 9 reglas difusas para este controlador debido a las 3 funciones de membresía por cada entrada que tenemos y éstas fueron usadas en base a la experiencia en otros controladores similares estudiados anteriormente.

Las 9 reglas difusas IF- THEN utilizadas son las siguientes:

- R1: Si error es N y derivadaerror es N, Entonces S es N
- R2: Si error es N y derivadaerror es Z, Entonces S es N
- R3: Si error es N y derivadaerror es P, Entonces S es Z

- R4: Si error es Z y derivadaerror es N, Entonces S es N
- R5: Si error es Z y derivadaerror es Z, Entonces S es Z
- R6: Si error es Z y derivadaerror es P, Entonces S es P
- R7: Si error es P y derivadaerror es N, Entonces S es Z
- R8: Si error es P y derivadaerror es Z, Entonces S es P
- R9: Si error es P y derivadaerror es P, Entonces S es P

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.12, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.12: Reglas difusas.

e / e_v	N	Z	P
N	N/N	N/Z	N/P
Z	Z/N	Z/Z	Z/P
P	P/N	P/Z	P/P

Planta 2

Diseño del controlador difuso

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de la planta 2, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error (e) y la derivada del error (e_v), fueron tomadas como variables de entrada, y una variable de Salida (S) [53], [54], [57]..

Las funciones de membresía de las variables de entrada son de tipo Gaussiana para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (Z). De igual forma se decidió probar con las funciones de membresía de tipo gaussiana para todos los términos lingüísticos Negativo (N), Cero (Z) y Positivo (P) de las variables de entrada. El intervalo usado para los dos tipos de controlador difuso fué de [-1010].

La Figura 3.21 A) muestran las variables de entrada del controlador difuso tipo-1 catalogado como *GTG* (Gaussiana-Triangular-Gaussiana), B) muestran las variables de entrada

del controlado difuso tipo-1 catalogado como *GGG* (Gaussiana-Gaussiana-Gaussiana) y C) muestran las variables de entrada del controlado difuso tipo-2 catalogado como *GTG*.

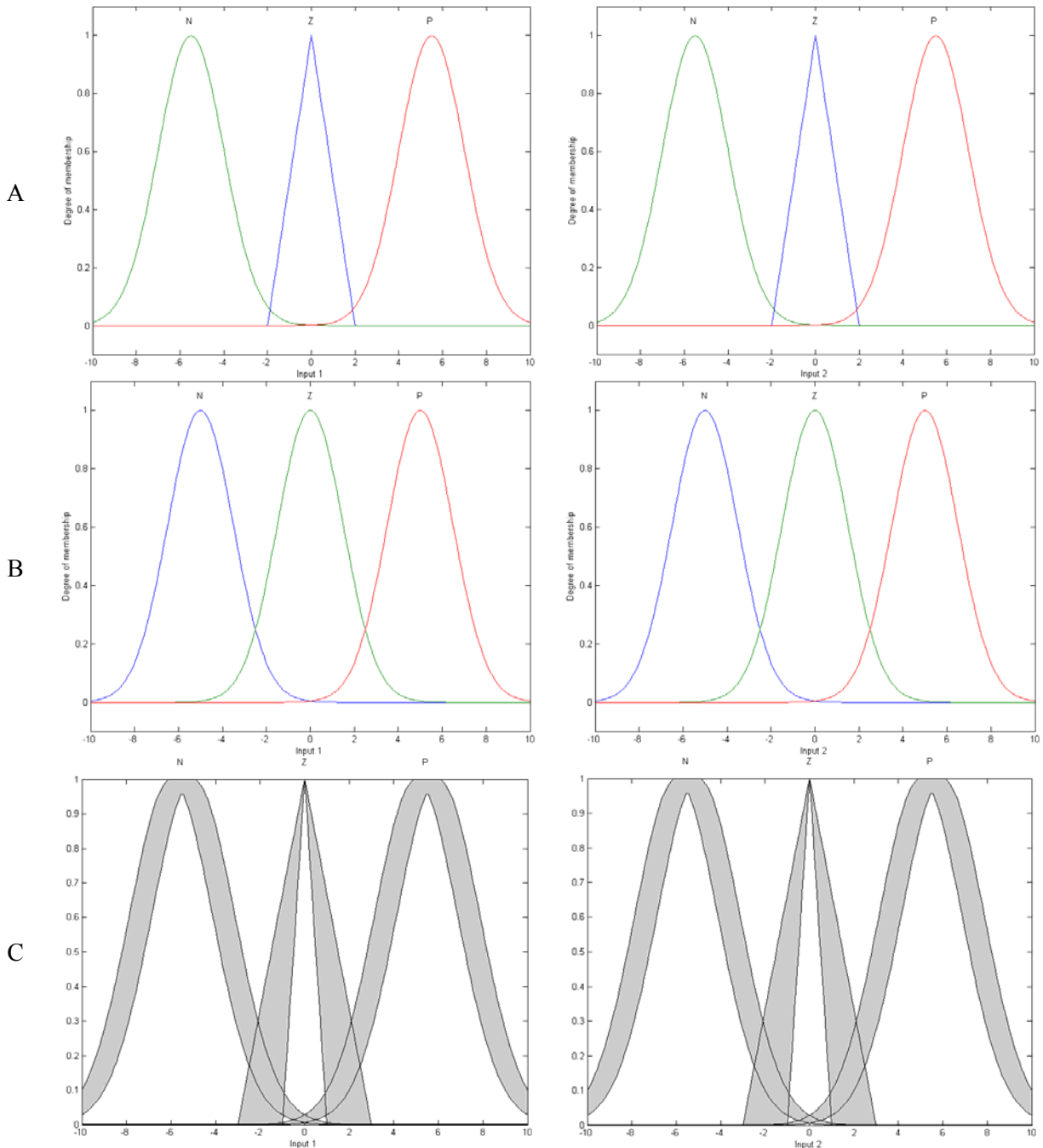


Figura 3.21: Entrada 1 (error "e") y entrada 2 (derivada de error "ev") del controlador difuso tipo-1 A) *GTG* y B) *GGG* y entrada 1 (error "e") y entrada 2 (derivada de error "ev") del controlador difuso tipo-2 C) *GTG*.

Diseño del controlador difuso tipo-2

Para el controlador difuso tipo-2 se utilizó el mismo diseño que el caso del controlador tipo-1 ya que se tomó el mejor controlador tipo 1, la cual sirvió como base para buscar sólo la incertidumbre y así obtener el controlador difuso tipo-2 óptimo, de igual forma se utilizaron los tipos de controladores difusos mencionados anteriormente. Las Figuras 3.21 C) muestran las variables de entrada del controlado difuso tipo-2 catalogado como *GTG*.

Configuración del PSO

Se utilizó el método de optimización PSO para encontrar el mejor controlador difuso tipo -1 y tipo-2 para la planta 1. Se asignaron valores a las variables cognitivas, social y de inercia de PSO. Para las variable cognitiva y social se asignaron de forma manual con rangos de 0.001 a 1y el rango para la variable de inercia es 0.005 a 1; que más adelante se mostraran los resultados obtenidos, después se le asigno de manera aleatoria valores a la variables de PSO utilizando los rangos mencionados anteriormente.

Límites inferiores y superiores de los parámetros de las funciones de membresía

Tabla 3.13 se presenta los límites inferiores y superiores utilizados para la búsqueda de los parámetros de las funciones de membresía de los controladores difusos para la planta 2. La Figura 3.22 muestra los tipos de funciones de membresía utilizados para el controlador difuso *GTG*.

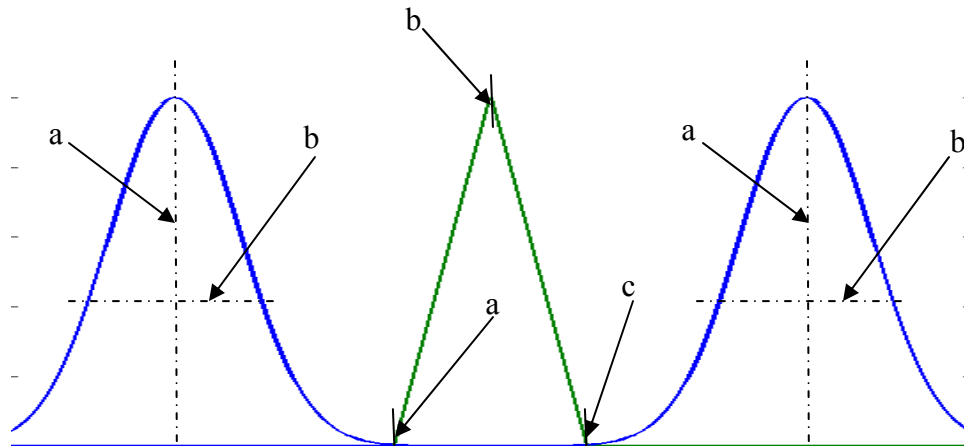


Figura 3.22: Tipos de funciones de membresía utilizados.

En la Tabla 3.13 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía del controlador difuso tipo-1.

Tabla 3.13: Límites inferiores y superiores de las funciones de membresía.

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	1.8	2.8
	b	-5	-5
Triangular	a	-0.5	-4
	b	0	0
	c	0.5	4
Gaussiana	a	1.8	2.8
	b	5	5

En la Tabla 3.14 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía del controlador difuso tipo-2.

Tabla 3.14: Límites inferiores y superiores de las funciones de membresía.

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	1.8	2.8
	b	-7	-3
	a'	1.5	1.6
	b'	-7	-3
Triangular	a	-3	-2.5
	b	0	0
	c	0.5	2
	a'	-2	-1
	b'	0	0
	c'	1	3
Gaussiana	a	1.8	2.8
	b	3	7
	a'	1.5	1.6
	b'	3	7

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con el algoritmo genético y la planta desarrollada en Simulink®. El objetivo es minimizar el error obtenido por la diferencia entre la referencia y el resultado del controlador.

Reglas difusas del controlador

Se utilizaron 9 reglas difusas para este controlador debido a las 3 funciones de membresía por cada entrar que tenemos y estas fueron usadas en base a la experiencia en otros controladores similares estudiados anteriormente.

Las 9 reglas difusas IF- THEN utilizadas son las siguientes:

- R1: Si error es N y derivadaerror es N, Entonces S es N
- R2: Si error es N y derivadaerror es Z, Entonces S es N
- R3: Si error es N y derivadaerror es P, Entonces S es Z
- R4: Si error es Z y derivadaerror es N, Entonces S es N
- R5: Si error es Z y derivadaerror es Z, Entonces S es Z
- R6: Si error es Z y derivadaerror es P, Entonces S es P
- R7: Si error es P y derivadaerror es N, Entonces S es Z
- R8: Si error es P y derivadaerror es Z, Entonces S es P
- R9: Si error es P y derivadaerror es P, Entonces S es P

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.15, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.15: Reglas difusas.

e / e_v	N	Z	P
N	N/N	N/Z	N/P
Z	Z/N	Z/Z	Z/P
P	P/N	P/Z	P/P

PSO: Péndulo de rueda de inercia

Como se mencionó en la sección 3.1.3 utilizamos el método de optimización por medio de nube de partículas (PSO) para encontrar los valores óptimos de las ganancias del controlador PID del péndulo de rueda de inercia [61], [62].

Configuración del método de optimización por medio de nube de partículas (PSO)

Como se menciona en la sección 3.1.3.2 el método optimizador no se utilizó para crear un controlador difuso, sino para buscar los valores óptimos de las ganancias del controlador PID del péndulo de rueda de inercia. El objetivo del método de optimización es encontrar los valores óptimos para las ganancias del controlador PID llamadas k_1 , k_2 y k_3 .

Los parámetros del método PSO que se fueron cambiando son la variable cognitiva y social con un rango de 0.001 a 1 en cada variable y el rango para la variable inercia es de 0.005 a 1, primeramente de forma manual y después se implementó que de manera aleatoria asignara valores del mismo rango para la búsqueda de las ganancias del controlador PID. El sistema de paro del PSO fue hasta llegar a un número máximo de iteraciones.

Se utilizaron un rango de valores para la búsqueda de cada ganancia (k_1 , k_2 y k_3) y la Tabla 3.16 muestra los valores mínimos y máximos basándonos en las condiciones físicas del sistema real.

Tabla 3.16: Límites inferiores y superiores del rango de búsqueda de las ganancias del péndulo.

	Valores mínimos	Valores máximos
K_1	50	250
K_2	5	50
K_3	0.001	0.1

Robot autónomo móvil

Diseño del controlador difuso

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de seguimiento del robot autónomo móvil, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error de las velocidades lineal (v_d) y angular (w_d), que no son más que la diferencia entre la velocidad deseada y la velocidad real del móvil, fueron tomadas como variables de entrada, y como variables de salida se tomó el par derecho (τ_1) e izquierdo (τ_2).

Las funciones de membresía de las variables de entrada son de tipo trapezoidal para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (C). El intervalo usado para el controlador difuso fue de $[-50 \ 50]$.

Las Figuras 3.24 A) muestran las variables de entrada y B) muestran las variables de salida.

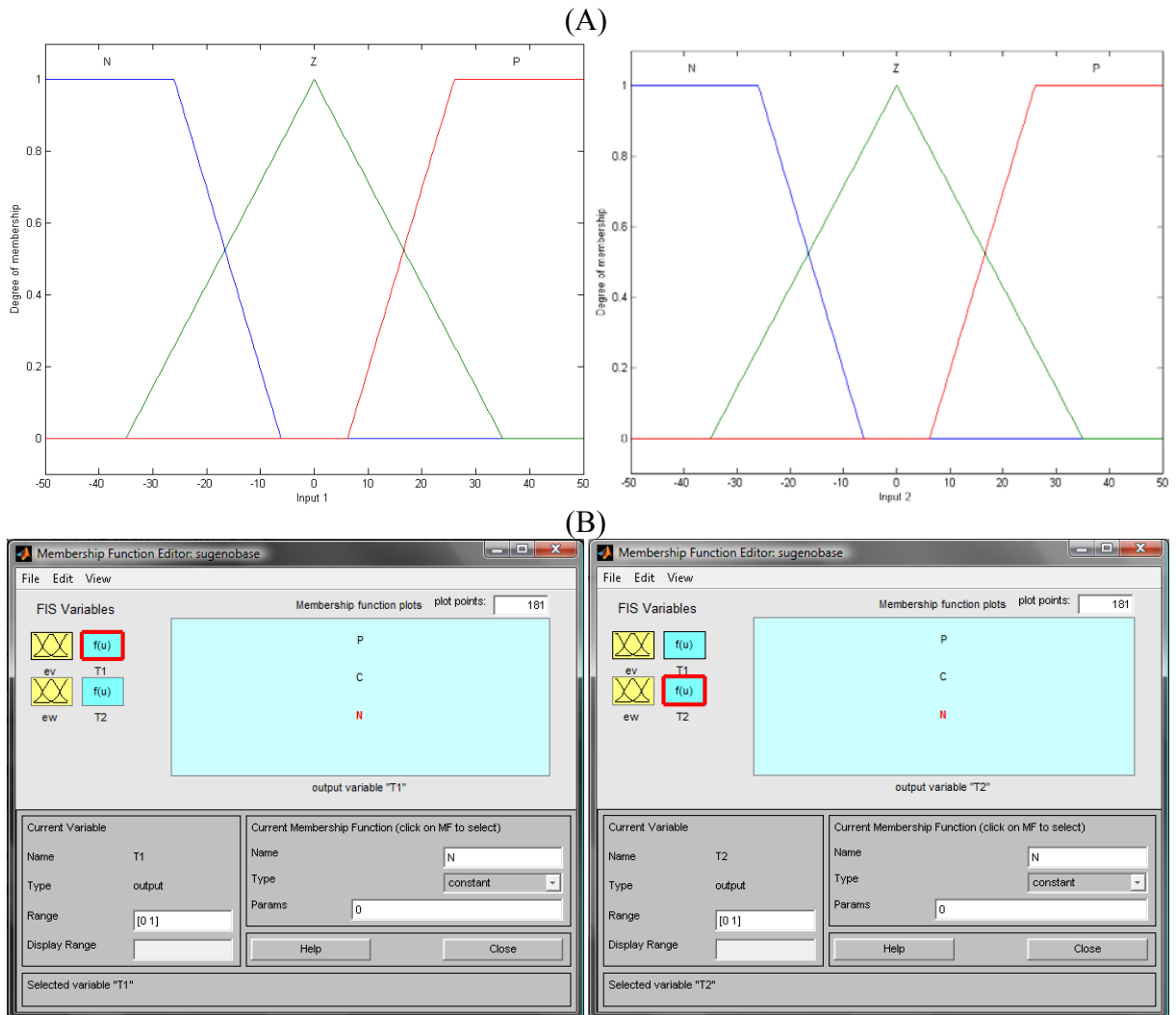


Figura 3.24: A) Error en la velocidad lineal y error de velocidad angular, B) Torque derecho y torque izquierdo.

Las 9 reglas difusas utilizadas son las siguientes:

- R1: Si V_{Lineal} es Z y $V_{angular}$ es Z, Entonces $T1$ es C y $T2$ es C
- R2: Si V_{Lineal} es Z y $V_{angular}$ es P, Entonces $T1$ es C y $T2$ es P

- R3: Si VLineal es Z y Vangular es N, Entonces $T1$ es C y $T2$ es N
- R4: Si VLineal es P y Vangular es Z, Entonces $T1$ es P y $T2$ es C
- R5: Si VLineal es P y Vangular es P, Entonces $T1$ es P y $T2$ es P
- R6: Si VLineal es P y Vangular es N, Entonces $T1$ es P y $T2$ es N
- R7: Si VLineal es N y Vangular es Z, Entonces $T1$ es N y $T2$ es C
- R8: Si VLineal es N y Vangular es P, Entonces $T1$ es N y $T2$ es P
- R9: Si VLineal es N y Vangular es N, Entonces $T1$ es N y $T2$ es N

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.17, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.17: Reglas difusas.

e_v / e_w	N	C	P
N	N/N	N/C	N/P
C	C/N	C/C	C/P
P	P/N	P/C	P/P

Simulación del robot autónomo móvil

Una vez que se diseñó el controlador, se hizo la simulación tomando en cuenta las ecuaciones (3.5),(3.6), (3.10), (3.11) y (3.12) y se diseñó el sistema de control en Simulink®. Las condiciones iniciales y la trayectoria deseada para la simulación del sistema son las mismas mencionadas anteriormente.

Configuración de PSO para el control de seguimiento

Se utilizo el método de optimización por medio de nube de partículas (PSO) para optimizar los parámetros de las funciones de membresía de entrada y salida del controlador difuso. El objetivo es que en la simulación, los errores en la velocidad lineal y angular sean cero.

De igual forma que en los otros casos de aplicación las variables que se utilizaron dentro del método de optimización son variable cognitiva y social con un rango de valores de 0.001 a 1 para los dos parámetros y un rango de 0.05 a 1 para el parámetro de inercia que de primera

instancia éstos fueron asignados manualmente para después generarlos de manera aleatoria utilizando los mismos rangos. De esta forma se encontró el mejor controlador difuso para el control de seguimiento de trayectorias del robot autónomo móvil.

La Figura 3.25 muestra los tipos de funciones de membresía utilizados para el controlador difuso.

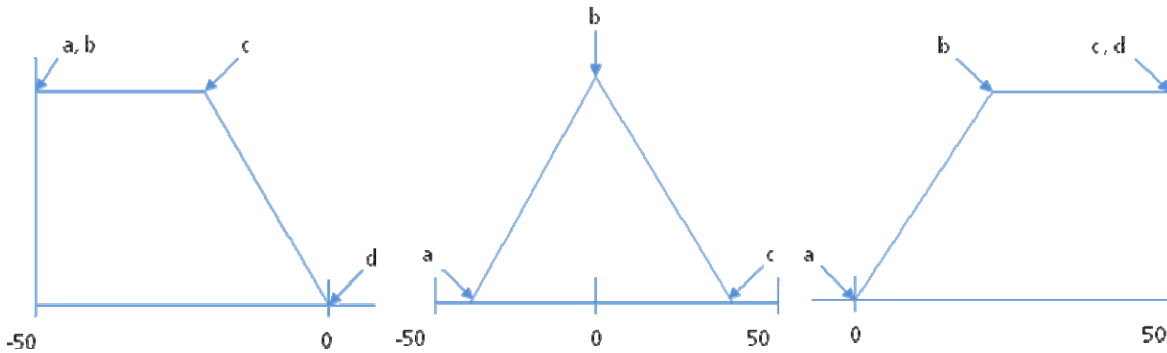


Figura 3.25 Tipos de funciones de membresía utilizados.

En la Tabla 3.18 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía.

Tabla 3.18: Límites inferiores y superiores de las funciones de membresía.

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Trapezoidal	a	-50	-50
	b	-50	-50
	c	-15	-5.05
	d	-1.5	-0.5
Triangular	a	-5	-1.75
	b	0	0
	c	1.75	5
Trapezoidal	a	0.5	1.5
	b	5.05	15
	c	50	50
	d	50	50

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con PSO y el sistema en Simulink®. El objetivo es minimizar los errores de las velocidades lineal y angular del sistema. Los valores usados para k_1 , k_2 y k_3 son 5, 24 y 3, respectivamente.

Diseño de un controlador difuso Tipo-2

Tomando como referencia el controlador difuso de Tipo-1 que fué considerado el mejor, se diseñó un controlador difuso Tipo-2, en el cual los valores de las funciones de membresía primarias fueron las mismas que las de Tipo-1.

La idea de extender el controlador difuso de Tipo-1 a Tipo-2, es compararlos entre sí agregando perturbación a la planta.

Los parámetros para la amplitud de la huella de incertidumbre fueron optimizados con el uso de PSO. Los resultados obtenidos para la optimización de los parámetros de las funciones de membresía secundarias.

3.4 Método de Optimización Híbrido (PSO-GA)

Esta sección describimos la propuesta del trabajo de tesis, la cual consiste en crear un método híbrido para la optimización de controladores difusos tipo-2. Para ello escogimos dos de los métodos con los cuales ya hemos trabajado anteriormente de forma individual que son algoritmos genéticos (GA) y optimización por medio de nube de partículas (PSO). La idea de utilizar estos dos métodos de forma híbrida es la de crear un método más robusto que nos permita resolver problemas complejos con resultados más óptimos; que en este trabajo de tesis es la de encontrar los mejores controladores difusos tipo-2 .

El método híbrido utiliza la misma población/colmena en los dos métodos (GA/PSO) en la misma iteración obteniendo el mejor individuo/partícula. PSO y GA se comunican entre ellos; cada cuatro iteraciones el mejor individuo/partícula es inyectado en la población del peor resultado interno y viceversa. Utilizamos el máximo número de iteraciones/generaciones como sistema de paro del método quedando sólo el mejor controlador difuso obtenido. El método híbrido funciona como sigue:

1. *Se genera de manera aleatoria una población/colmena (Candidatos de Solución al problema).*
2. *Asignación a las variables S y $Chrom$ la población/colmena generado.*
3. *Repetir los siguientes pasos hasta el máximo número de iteraciones/generaciones*

3.1 Inicia método PSO.

3.1.1 A cada partícula se aplica lo siguiente.

3.1.1.1 Calcula el valor del fitness.

3.1.1.2 Si el valor del fitness es mejor que el mejor valor fitness($pBest$) en memoria entonces, se toma el nuevo valor en ese momento como el nuevo $pBest$.

3.1.1.3 Tomas la partícula con el mejor valor fitness de todo el conjunto de partículas, llamado $gBest$.

3.1.1.4 Se actualizan la velocidad de todas las partículas.

3.1.1.5 Se actualiza las posiciones de todas las partículas.

3.1.1.6 Hasta "n" partículas.

3.1.1.7 Se guarda la información de la mejor partícula en la variable BM

3.2 Termina PSO.

3.3 Inicia método GA.

3.3.1 Se Calcula el fitness de cada cromosoma en la población.

3.3.2 Se repiten los siguientes pasos hasta que n descendencia ha sido creada:

3.3.2.1 Seleccione un par de cromosomas de los padres de la población actual, la probabilidad de selección es una función creciente de fitness. La selección se realiza con reemplazo, lo que significa que el mismo cromosoma se puede seleccionar más de una vez para convertirse en uno de los padres.

3.3.2.2 Con probabilidad (tasa de cruce), realiza el cruce de los padres de simple punto de manera aleatoria hasta formar dos hijos.

3.3.2.3 Se muta a los dos hijos en cada lugar con probabilidad (tasa de mutación), y reinserta los resultados del nuevo cromosoma en la nueva población.

3.3.3 Reemplaza la población actual con la nueva población.

3.3.4 Ir al paso 5.

3.3.5 Guarda la información del mejor individuo en la variable BM .

- 3.4 *Termina GA.*
4. *Si contador = 4 entonces*
 - 4.1 *Obtener el mínimo resultado de la variable BM*
 - 4.2 *Si el valor mínimo fué obtenido por el método PSO entonces*
 - 4.3 *reinserta la información de la partícula en la población del GA (Chrom)*
 - 4.3 *sino*
 - 4.4 *reinserta la información del mejor individuo del GA en la colmena de PSO*
5. *Termina la validación.*
6. *Termina el método híbrido hasta el máximo número de iteraciones/generaciones.*

El procedimiento descrito anteriormente es usado para encontrar el óptimo controlador difuso combinando PSO y GA explotando más completamente el espacio de soluciones.

En la siguiente sección se presenta los casos de estudio en donde se utilizó el método híbrido para encontrar controladores difusos óptimos.

Planta 1

Diseño del controlador difuso

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de la planta 1, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error (e) y la derivada del error (ev), fueron tomadas como variables de entrada, y una variable de Salida (S).

De acuerdo a trabajos previos con controladores difusos se decidió tomar el mismo tipo de funciones de membresía por la semejanza entre los mismos. Las funciones de membresía son para las variables de entrada son de tipo Gaussiana para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (Z). De igual forma se decidió probar con las funciones de membresía de tipo gaussiana para todos los términos lingüísticos Negativo (N), Cero (Z) y Positivo (P) de las variables de entrada.

El intervalo usado para los dos tipos de controlador difuso fué de [-22].

Las Figuras 3.26 A) muestran las variables de entrada del controlado difuso tipo-1 catalogado como *GTG* (Gaussiana-Triangular-Gaussiana) y B) muestran las variables de entrada del controlado difuso tipo-2 catalogado como *GTG*.

Diseño del controlador difuso tipo-2

Para el controlador difuso tipo-2 se utilizó el mismo diseño que el caso del controlador tipo-1 ya que se tomó el mejor controlador tipo 1, la cual sirvió como base para buscar sólo la incertidumbre y así obtener el controlador difuso tipo-2 óptimo, de igual forma se utilizaron los tipos de controladores difusos mencionados anteriormente.

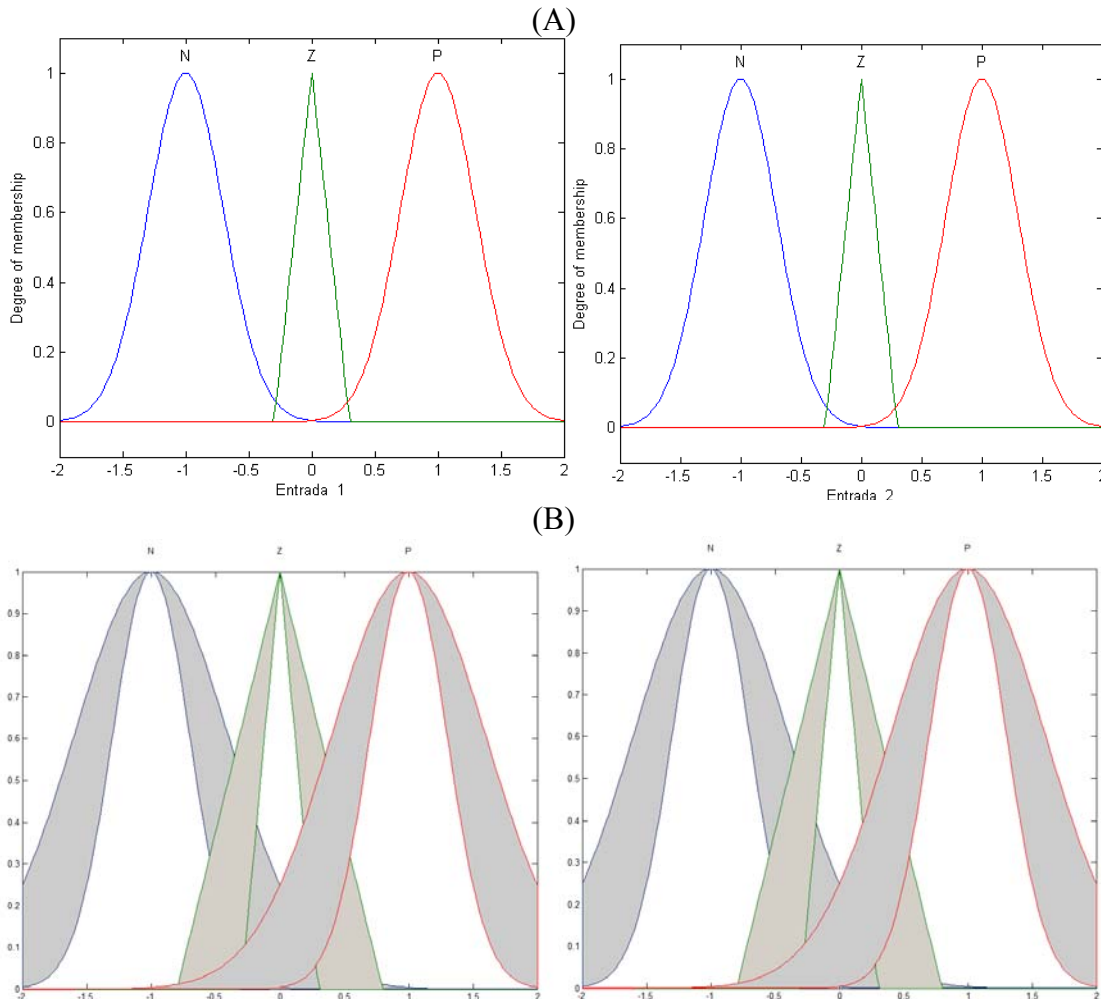


Figura 3.26: A) Entrada 1 (error "e") y entrada 2 (derivada error "ev") del controlador difuso tipo-1 *GTG* y B) 2ntrada 1 (error "e") y entrada 2 (derivada error "ev") del controlador difuso tipo-2 *GTG*.

Configuración del método híbrido PSO-GA

A continuación describimos la configuración que utilizamos internamente en el método híbrido PSO-GA.

Dentro del método híbrido tenemos los parámetros que vamos cambiando como son las siguientes: con respecto a los valores de los parámetros de PSO tenemos la $C1$ que es el parámetro del coeficiente cognitivo con valores que van desde 0.001 hasta 1.0, $C2$ que es el parámetro del coeficiente social con valores que van desde 0.001 hasta 1.0 e Inercia que es el parámetro de inercia con valores que van desde 0.005 hasta 1.0; en primera instancia estos valores fueron cambiando manualmente durante las pruebas, y después se utilizó un generador de valores aleatorios para dichas variables con rangos de 0.001 a 1.0 para las variables cognitivas y rango de 0.005 a 1.0 para la variable de inercia. En el GA se cambian los valores de mutación y cruce de simple punto y con método de selección de la ruleta.

Dentro del proceso de búsqueda del controlador óptimo aplicamos una evaluación del error promedio obtenido de la prueba del controlador durante la ejecución del algoritmo para evitar que quede estancado en un mínimo local, de manera que cada 4 iteraciones evaluamos los resultados internos del controlador de cada método y del controlador que tenga mejor resultado su información es reinsertada en la población del peor método y viceversa. Esta evaluación interna nos sirve para mejorar el proceso de encontrar el controlador difuso más óptimo para el problema.

Límites inferiores y superiores de los parámetros de las funciones de membresía

Tabla 3.19 se presenta los límites inferiores y superiores utilizados para la búsqueda de los parámetros de las funciones de membresía de los controladores difusos para la planta 1. La Figura 3.27 muestra los tipos de funciones de membresía utilizados para el controlador difuso *GTG*.

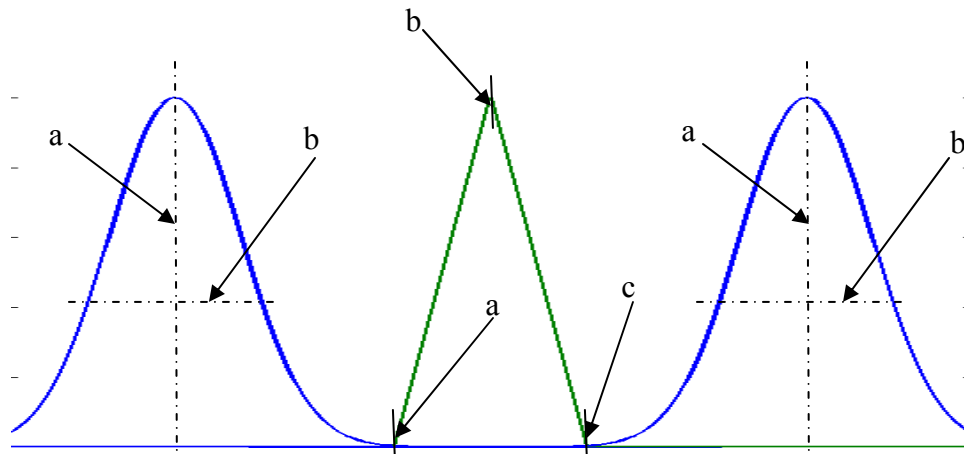


Figura 3.27 Tipos de funciones de membresía utilizados.

En la Tabla 3.19 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía.

Tabla 3.19: Límites inferiores y superiores de las funciones de membresía.

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	0.3	0.6
	b	-1	-1
Triangular	a	-0.3	-0.8
	b	0	0
	c	0.3	0.8
Gaussiana	a	0.3	0.6
	b	1	1

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con el método híbrido PSO-GA y la planta desarrollada en Simulink®. El objetivo es minimizar el error obtenido por la diferencia entre la referencia y el resultado del controlador.

Reglas difusas del controlador difuso

Se utilizaron 9 reglas difusas para este controlador debido a las 3 funciones de membresía por cada entrar que tenemos y estas fueron usadas en base a la experiencia en otros controladores similares estudiados anteriormente.

Las 9 reglas difusas IF- THEN utilizadas son las siguientes:

- R1: Si error es N y derivadaerror es N, Entonces S es N
- R2: Si error es N y derivadaerror es Z, Entonces S es N
- R3: Si error es N y derivadaerror es P, Entonces S es Z
- R4: Si error es Z y derivadaerror es N, Entonces S es N
- R5: Si error es Z y derivadaerror es Z, Entonces S es Z
- R6: Si error es Z y derivadaerror es P, Entonces S es P
- R7: Si error es P y derivadaerror es N, Entonces S es Z
- R8: Si error es P y derivadaerror es Z, Entonces S es P
- R9: Si error es P y derivadaerror es P, Entonces S es P

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.20, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.20: Reglas difusas.

e / e_v	N	Z	P
N	N/N	N/Z	N/P
Z	Z/N	Z/Z	Z/P
P	P/N	P/Z	P/P

Planta 2

Diseño del controlador difuso

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de la planta 2, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error (e) y la derivada del error (e_v), fueron tomadas como variables de entrada, y una variable de Salida (S).

Las funciones de membresía de las variables de entrada son de tipo Gaussiana para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (Z). De igual forma se decidió probar con las funciones de membresía de tipo gaussiana para todos los términos lingüísticos Negativo (N), Cero (Z) y Positivo (P) de las variables de entrada. El intervalo usado para los dos tipos de controlador difuso fué de $[-10,10]$.

Las Figuras 3.28 muestran las variables de entrada del controlador difuso A) tipo-1 catalogado como *GTG* (Gaussiana-Triangular-Gaussiana) y B) tipo2 *GTG*.

Diseño del controlador difuso tipo-2

Para el controlador difuso tipo-2 se utilizó el mismo diseño que el caso del controlador tipo-1 ya que se tomó el mejor controlador tipo 1, la cual sirvió como base para buscar sólo la incertidumbre y así obtener el controlador difuso tipo-2 óptimo, de igual forma se utilizaron los tipos de controladores difusos mencionados anteriormente.

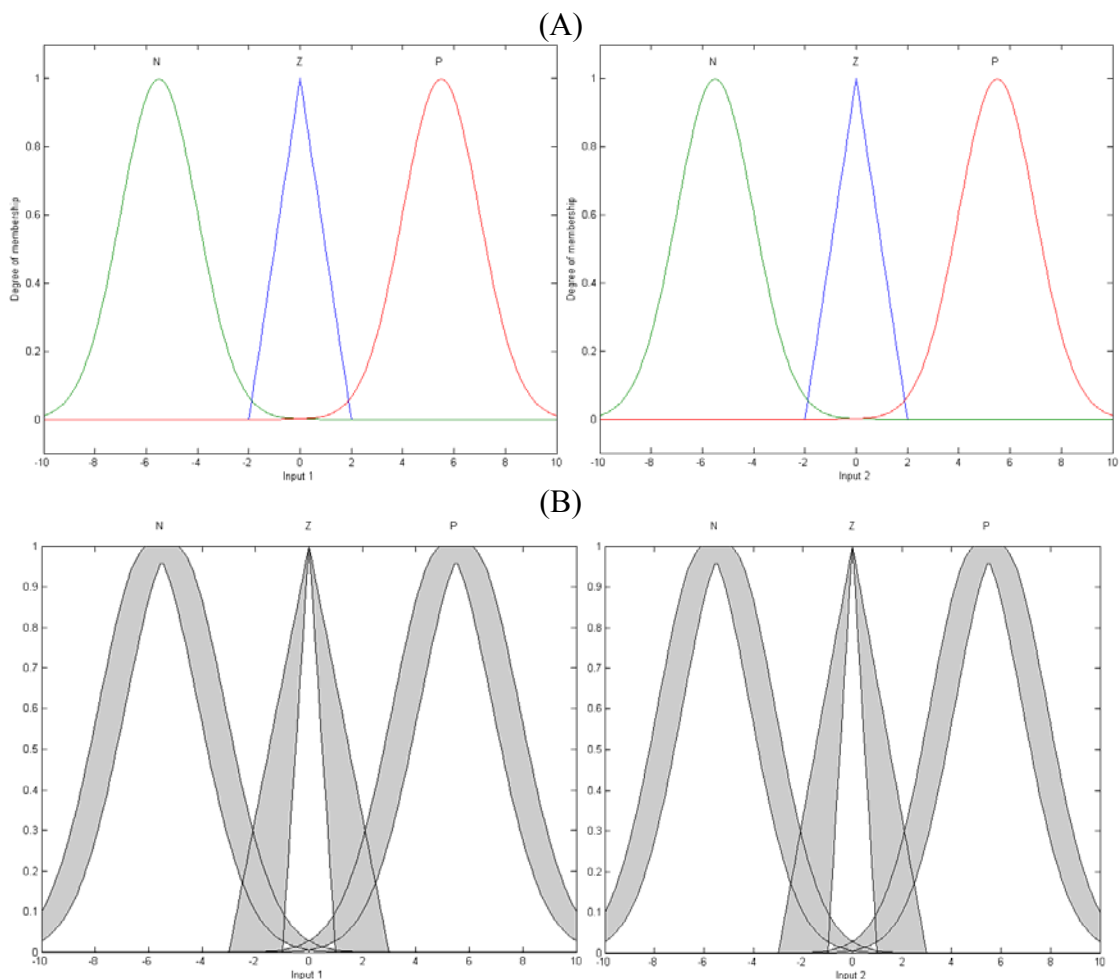


Figura 3.28: A) Entrada 1 (error "e") y entrada 2 (derivada de error "ev") del controlador difuso tipo-1 *GTG* y B) entrada 1 (error "e") y entrada 2 (derivada de error "ev") del controlador difuso tipo-2 *GTG*.

Configuración del método híbrido PSO-GA

A continuación describimos la configuración que utilizamos internamente en el método híbrido PSO-GA [65], [69].

Dentro del método híbrido tenemos los parámetros que vamos cambiando como son las siguientes: con respecto a los valores de los parámetros de PSO tenemos la $C1$ que es el parámetro del coeficiente cognitivo con valores que van desde 0.001 hasta 1.0, $C2$ que es el parámetro del coeficiente social con valores que van desde 0.001 hasta 1.0 e Inercia que es el parámetro de inercia con valores que van desde 0.005 hasta 1.0; en primera instancia estos valores fueron cambiando manualmente durante las pruebas, y después se utilizó un generador de valores aleatorios para dichas variables con rangos de 0.001 a 1.0 para las variables cognitivas y rango de 0.005 a 1.0 para la variable de inercia. En el GA se cambian los valores de mutación y cruce de simple punto y con método de selección de la ruleta.

Límites inferiores y superiores de los parámetros de las funciones de membresía

Tabla 3.21 se presenta los límites inferiores y superiores utilizados para la búsqueda de los parámetros de las funciones de membresía de los controladores difusos para la planta 2. La Figura 3.29 muestra los tipos de funciones de membresía utilizados para el controlador difuso GTG.

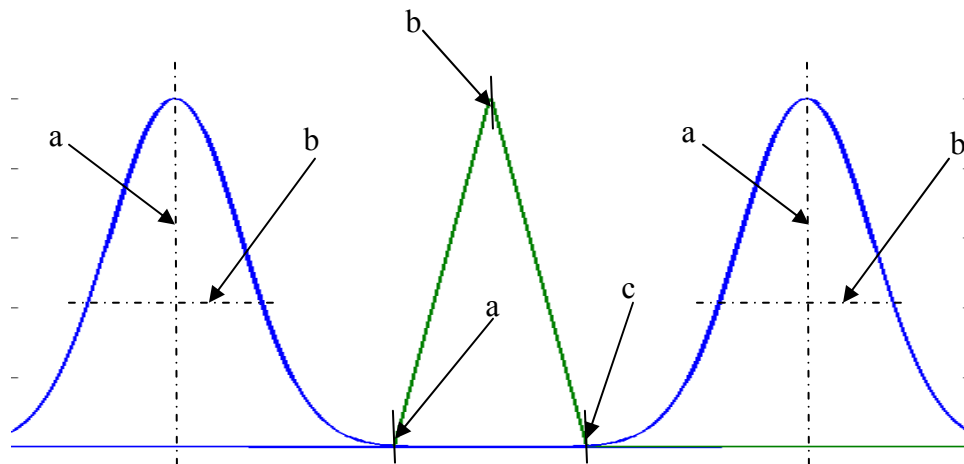


Figura 3.29: Tipos de funciones de membresía utilizados.

En la Tabla 3.21 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía.

Tabla 3.21: Límites inferiores y superiores de las funciones de membresía.

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Gaussiana	a	1.8	2.8
	b	-5	-5
Triangular	a	-0.5	-4
	b	0	0
	c	0.5	4
Gaussiana	a	1.8	2.8
	b	5	5

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con el método híbrido PSO-GA y la planta desarrollada en Simulink®. El objetivo es minimizar el error obtenido por la diferencia entre la referencia y el resultado del controlador.

Reglas difusas del controlador

Se utilizaron 9 reglas difusas para este controlador debido a las 3 funciones de membresía por cada entrar que tenemos y éstas fueron usadas en base a la experiencia en otros controladores similares estudiados anteriormente.

Las 9 reglas difusas IF- THEN utilizadas son las siguientes:

- R1: Si error es N y derivadaerror es N, Entonces S es N
- R2: Si error es N y derivadaerror es Z, Entonces S es N
- R3: Si error es N y derivadaerror es P, Entonces S es Z
- R4: Si error es Z y derivadaerror es N, Entonces S es N
- R5: Si error es Z y derivadaerror es Z, Entonces S es Z
- R6: Si error es Z y derivadaerror es P, Entonces S es P
- R7: Si error es P y derivadaerror es N, Entonces S es Z
- R8: Si error es P y derivadaerror es Z, Entonces S es P
- R9: Si error es P y derivadaerror es P, Entonces S es P

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.22, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.22: Reglas difusas.

e / e_v	N	Z	P
N	N/N	N/Z	N/P
Z	Z/N	Z/Z	Z/P
P	P/N	P/Z	P/P

Robot autónomo móvil.

Diseño del controlador difuso

Se diseñó un controlador difuso de tipo Takagi-Sugeno para el control de seguimiento del robot autónomo móvil, en el cual, las variables de entrada fueron representadas por variables lingüísticas y la salida por una constante. El error de las velocidades lineal (v_d) y angular (w_d), que no son más que la diferencia entre la velocidad deseada y la velocidad real del móvil, fueron tomadas como variables de entrada, y como variables de salida se tomó el par derecho (τ_1) e izquierdo (τ_2). Las funciones de membresía de las variables de entrada son de tipo trapezoidal para los términos lingüísticos Negativo (N) y Positivo (P), y triangular para el término lingüístico Cero (C). El intervalo usado para el controlador difuso fue de [-50 50].

Las Figuras 3.30 muestran A) las variables de entrada y B) las variables de salida.

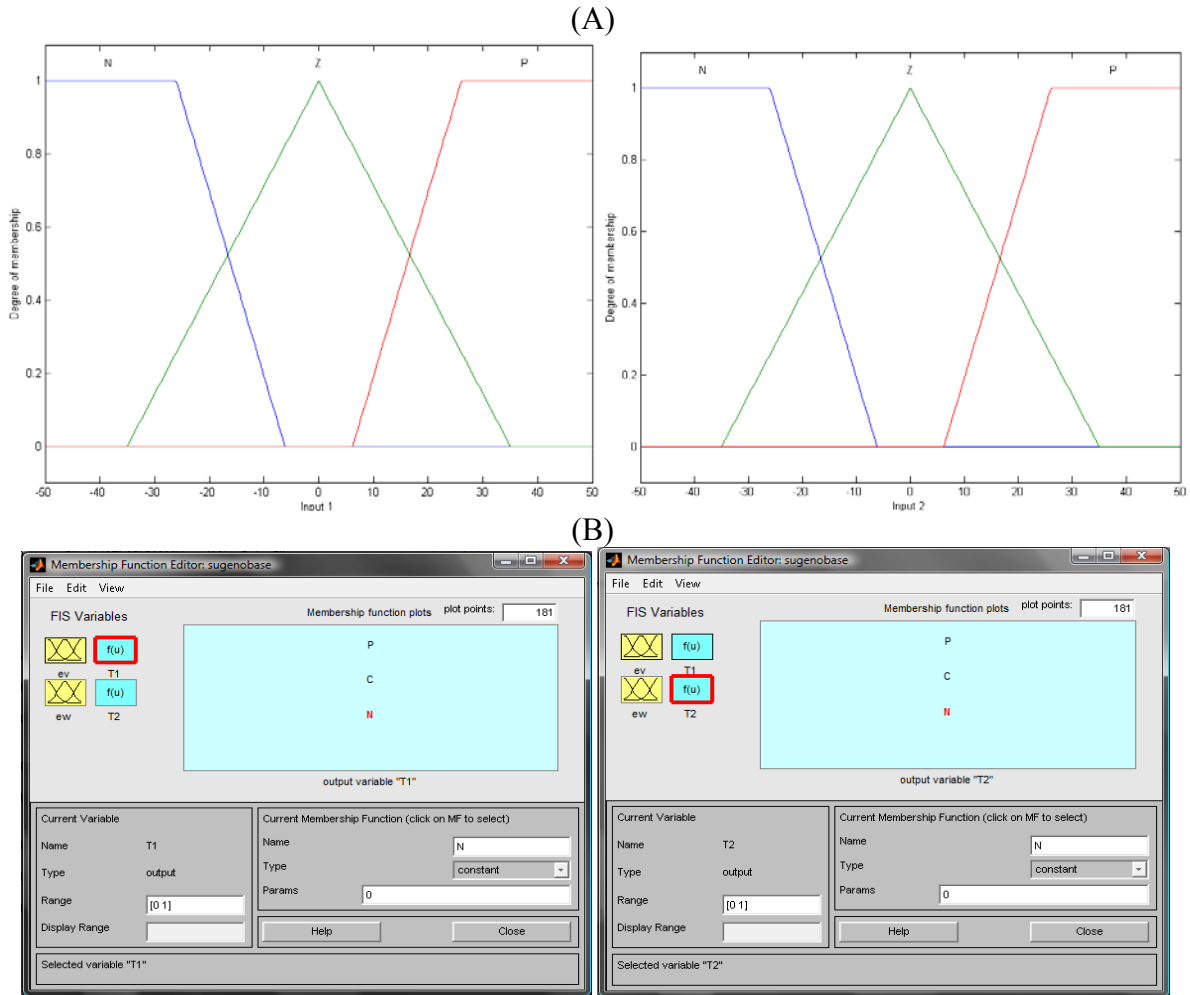


Figura 3.30: A) Error en la velocidad lineal y error de velocidad angular, B) Torque derecho y torque izquierdo.

Las 9 reglas difusas utilizadas son las siguientes:

- R1: Si V_{Lineal} es Z y $V_{angular}$ es Z, Entonces $T1$ es C y $T2$ es C
- R2: Si V_{Lineal} es Z y $V_{angular}$ es P, Entonces $T1$ es C y $T2$ es P
- R3: Si V_{Lineal} es Z y $V_{angular}$ es N, Entonces $T1$ es C y $T2$ es N
- R4: Si V_{Lineal} es P y $V_{angular}$ es Z, Entonces $T1$ es P y $T2$ es C
- R5: Si V_{Lineal} es P y $V_{angular}$ es P, Entonces $T1$ es P y $T2$ es P
- R6: Si V_{Lineal} es P y $V_{angular}$ es N, Entonces $T1$ es P y $T2$ es N
- R7: Si V_{Lineal} es N y $V_{angular}$ es Z, Entonces $T1$ es N y $T2$ es C
- R8: Si V_{Lineal} es N y $V_{angular}$ es P, Entonces $T1$ es N y $T2$ es P
- R9: Si V_{Lineal} es N y $V_{angular}$ es N, Entonces $T1$ es N y $T2$ es N

Los términos lingüísticos de las variables de entrada se muestran en la primera fila y columna de la Tabla 3.23, el resto del contenido corresponde a los términos lingüísticos de las variables de salida.

Tabla 3.23: Reglas difusas.

e_v / e_w	N	C	P
N	N/N	N/C	N/P
C	C/N	C/C	C/P
P	P/N	P/C	P/P

Simulación del robot autónomo móvil

Una vez que se diseñó el controlador, se hizo la simulación tomando en cuenta las ecuaciones (3.5),(3.6), (3.10), (3.11) y (3.12) y fué diseñado en Simulink®, fueron utilizadas las mismas condiciones iniciales mencionadas anteriormente.

Configuración del método híbrido PSO-GA

Dentro del método híbrido tenemos los parámetros que vamos cambiando como son las siguientes: con respecto a la los valores de los parámetros de PSO tenemos la C1 que es el parámetros del coeficiente cognitivo con valores que van desde 0.001 hasta 1.0, C2 que es el parámetro del coeficiente social con valores que van desde 0.001 hasta 1.0 e Inertia que es el parámetro de inercia con valores que van desde 0.005 hasta 1.0; en primera instancia estos valores fueron cambiando manualmente durante las pruebas, y después se utilizo un generador de valores aleatorios para dichas variables con rangos de 0.001 a 1.0 para las variables cognitiva y social, y un rango de 0.005 a 1.0 para la variable de inercia. En el GA se cambian los valores de mutación y cruce de simple punto y con método de selección de la ruleta. Una vez preparado el método se realizaron un conjunto de pruebas para obtener el mejor controlador difuso para el robot autónomo móvil.

La Figura 3.31 muestra los tipos de funciones de membresía utilizados para el controlador difuso.

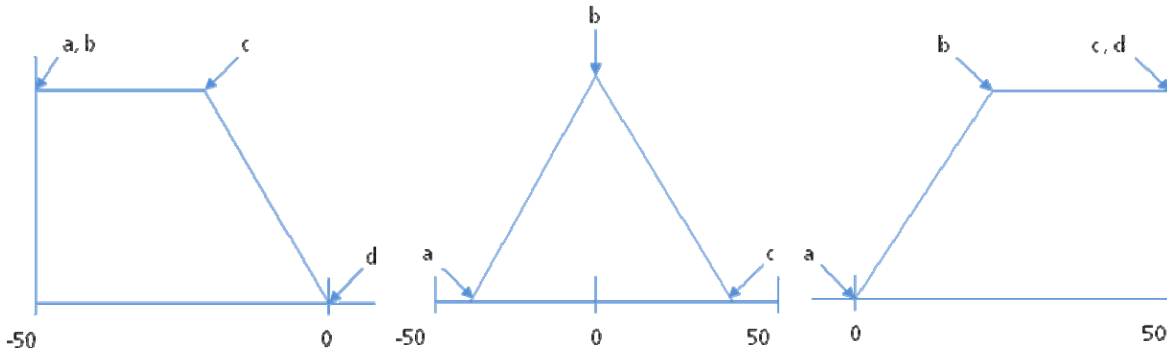


Figura 3.31: Tipos de funciones de membresía utilizados.

En la Tabla 3.24 se observan los límites inferiores y superiores que pueden tomar los diferentes puntos de las funciones de membresía.

Tabla 3.24: Límites inferiores y superiores de las funciones de membresía.

TIPO DE FM	PUNTO	VALOR MÍNIMO	VALOR MÁXIMO
Trapezoidal	a	-50	-50
	b	-50	-50
	c	-15	-5.05
	d	-1.5	-0.5
Triangular	a	-5	-1.75
	b	0	0
	c	1.75	5
Trapezoidal	a	0.5	1.5
	b	5.05	15
	c	50	50
	d	50	50

Una vez definidos los límites superiores e inferiores de las funciones de membresía de entrada y salida del controlador difuso, se hicieron pruebas con el método híbrido PSO-GA y el diseño del robot autónomo móvil utilizando Simulink®. El objetivo es minimizar los errores de las velocidades lineal y angular del sistema. Los valores usados para k_1 , k_2 y k_3 son 5, 24 y 3, respectivamente.

Diseño de un controlador difuso Tipo-2

Tomando como referencia el controlador difuso de Tipo-1 que fué considerado el mejor, se diseñó un controlador difuso Tipo-2, en el cual los valores de las funciones de membresía primarias fueron las mismas que las de Tipo-1.

La idea de extender el controlador difuso de Tipo-1 a Tipo-2, es compararlos entre sí agregando perturbación a la planta.

Los parámetros para la amplitud de la huella de incertidumbre fueron optimizados con el uso del método híbrido PSO-GA. Los resultados obtenidos para la optimización de los parámetros de las funciones de membresía secundarias.

4. Resultados

En este capítulo se presentan los resultados obtenidos de las simulaciones realizadas para la Planta 1, Planta 2, Péndulo de rueda de inercia y Robot autónomo móvil. Las herramientas utilizadas son Matlab® versión 2008b, Simulink® versión 6.6 y Laboratory inertial wheel pendulum de Mechatronics Control Kit manufactured por Quanser Inc.

4.1 Resultados del uso de los Algoritmos Genéticos

En esta sección se presentan los resultados obtenidos haciendo uso de los algoritmos genéticos para los casos de aplicación descritos a continuación.

4.1.1 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 1

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando la (3.1) que representa la planta 1.

La Tabla 4.1 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por el algoritmo genético.

En la Tabla 4.1, se muestra los resultados de las pruebas realizadas con algoritmos genéticos para la planta 1 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

Tabla 4.1: Resultados del controlador difuso tipo-1 *GTG* para planta 1.

GA GTG Planta 1 Tipo-1								
No.	Individuos	Generaciones	% Reemplazo	Cruce.	Mutación	Método de Selección	Tiempo del GA	Average Error
1	90	35	0.7	0.6	0.3	Ruleta	0:18:06	0.05087
2	150	80	0.7	0.5	0.2	Ruleta	1:19:13	0.04431
3	80	50	0.7	0.5	0.2	Ruleta	0:23:01	0.071366
4	45	60	0.7	0.6	0.3	Ruleta	0:16:03	0.068477
5	75	50	0.7	0.6	0.1	Ruleta	0:21:37	0.068158
6	100	40	0.7	0.6	0.1	Ruleta	0:24:08	0.067052
7	65	35	0.7	0.7	0.2	Ruleta	0:15:35	0.069994
8	200	70	0.7	0.4	0.1	Ruleta	1:30:02	0.072356
9	25	15	0.7	0.8	0.3	Ruleta	0:02:58	0.129872
10	50	45	0.7	0.5	0.2	Ruleta	0:13:25	0.068855
11	90	35	0.7	0.6	0.2	Ruleta	0:19:15	0.06529
12	40	25	0.7	0.7	0.4	Ruleta	0:06:48	0.175755
13	120	45	0.7	0.4	0.1	Ruleta	0:29:51	0.065761

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para la planta 1 con funciones de membresía *GTG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.2 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.2: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	80
	Generaciones	50
	% Reemplazo	0.7
	Cruce	0.5
	Mutación	0.2
	Método de Selección	Ruleta
Tiempo del GA		0:23:01
Error Alcanzado		0.071366

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.071366.

La Figura 4.1 muestra A) la evolución del algoritmo genético y su mejor individuo, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el algoritmo genético y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 1 obteniendo el controla antes de los 10 segundos de manera estable.

4.1.2 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-2 con funciones de membresía *GTG* para la Planta 1

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-2 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 1.

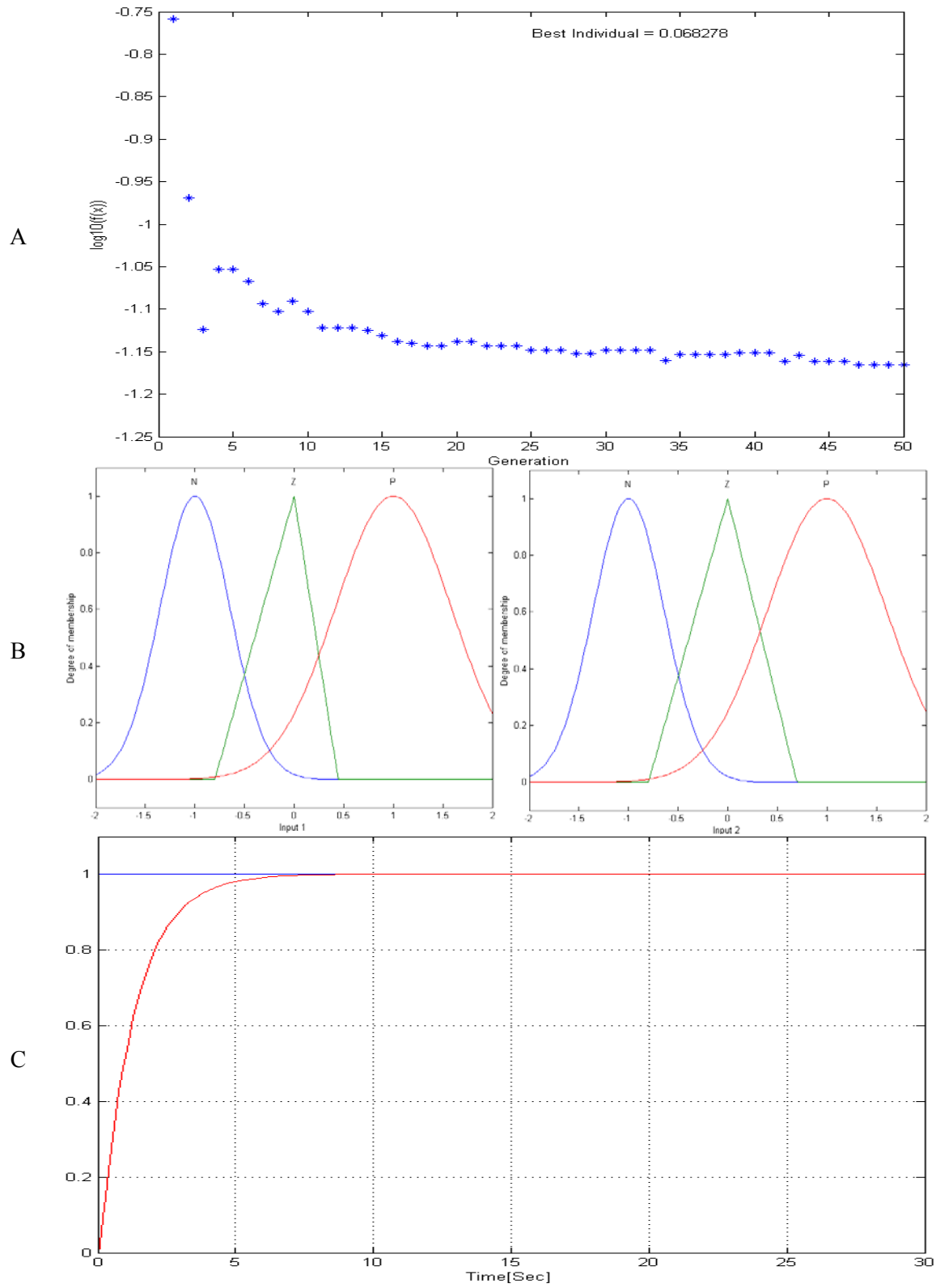


Figura 4.1: A) Evolución del algoritmo genético B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 1.

La Tabla 4.3 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el algoritmo genético.

En la Tabla 4.3, se muestra los resultados de las pruebas realizadas con algoritmos genéticos para la planta 1 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

Tabla 4.3: Resultados del controlador difuso tipo-2 *GTG* para planta 1.

GA <i>GTG</i> Planta 1 Tipo-2								
No.	Individuos	Generaciones	% Reemplazo	Cruce.	Mutación	Método de Selección	Tiempo del GA	Average Error
1	55	75	0.7	0.8	0.3	Ruleta	5:56:25	0.067651
2	90	65	0.7	0.7	0.2	Ruleta	8:33:15	0.067995
3	50	80	0.7	0.8	0.2	Ruleta	5:41:16	0.068208
4	90	85	0.7	0.7	0.2	Ruleta	11:18:23	0.068311
5	120	85	0.7	0.7	0.2	Ruleta	14:42:40	0.068749
6	30	55	0.7	0.8	0.3	Ruleta	2:21:10	0.069307
7	60	40	0.7	0.7	0.2	Ruleta	3:18:27	0.069967
8	45	40	0.7	0.9	0.3	Ruleta	2:26:12	0.073498
9	40	25	0.7	0.8	0.3	Ruleta	1:26:55	0.095847
10	65	35	0.7	0.8	0.2	Ruleta	3:08:03	0.74535

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para la planta 1 con funciones de membresía *GTG* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.4 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.4: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	55
	Generaciones	75
	% Reemplazo	0.7
	Cruce	0.8
	Mutación	0.3
	Método de Selección	Ruleta
Tiempo del GA		5:56:25
Error Alcanzado		0.067651

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.067651.

La Figura 4.2 muestra A) la evolución del algoritmo genético y su mejor individuo, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el algoritmo genético y C) el seguimiento de trayectoria del controlador difuso tipo-2 para la planta 1 obteniendo el control antes de los 5 segundos con un pequeño sobre pico al alcanzar la referencia.

4.1.3 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía *GTG* para la Planta 2

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.2) que representa la planta 2.

La Tabla 4.5 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el algoritmo genético.

En la Tabla 4.5, se muestra los resultados de las pruebas realizadas con algoritmos genéticos para la planta 2 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

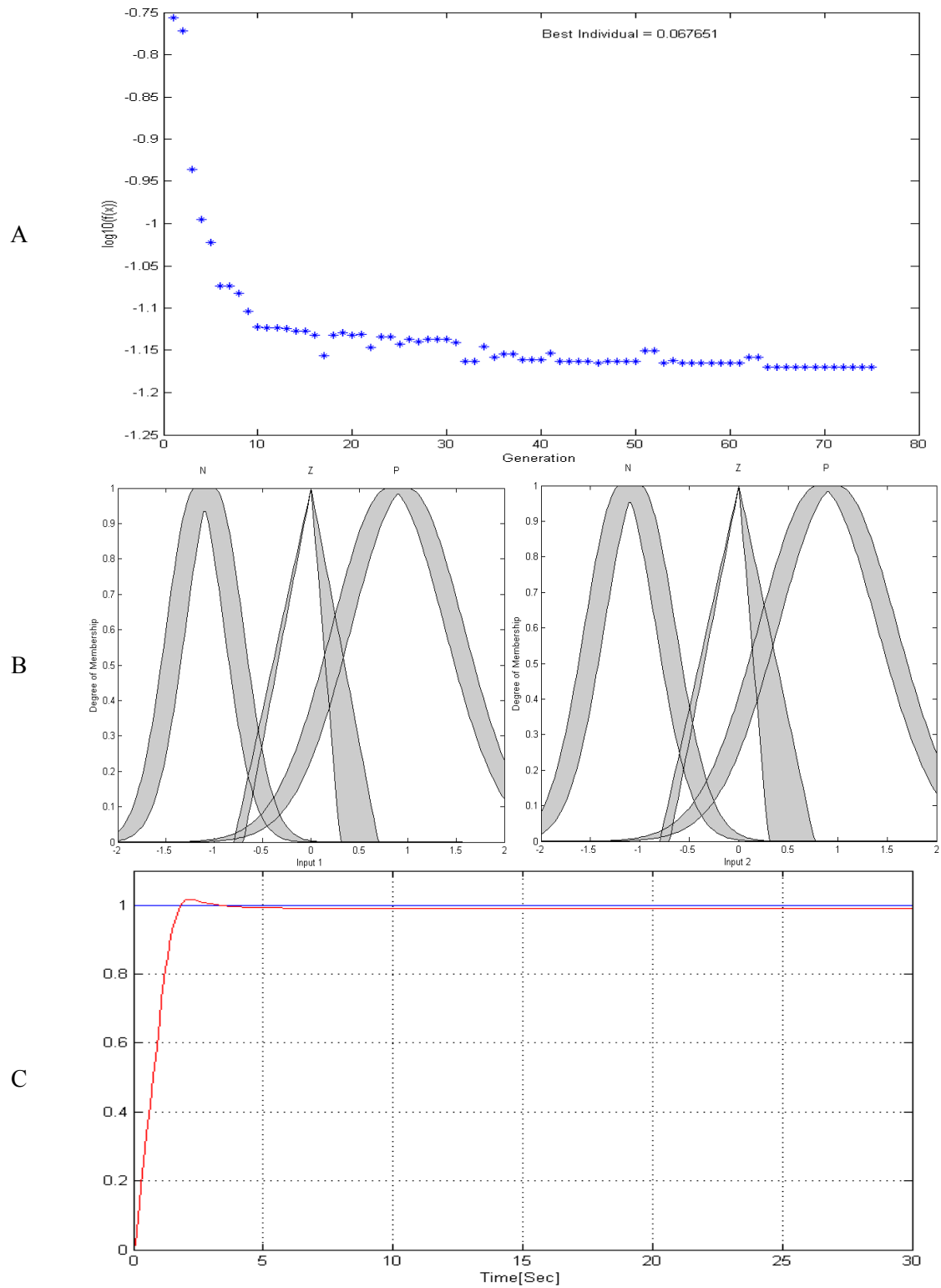


Figura 4.2: A) Evolución del algoritmo genético B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 1.

Tabla 4.5: Resultados del controlador difuso tipo-1 *GTG* para planta 2.

GA <i>GTG</i> Planta2 Tipo-1								
No.	Individuos	Generaciones	% Reemplazo	Cruce.	Mutación	Método de Selección	Tiempo del GA	Average Error
1	85	65	0.5	0.5	0.1	Ruleta	0:29:11	0.066594
2	80	50	0.5	0.6	0.1	Ruleta	0:22:18	0.076842
3	55	45	0.5	0.6	0.1	Ruleta	0:39:01	0.077597
4	100	40	0.5	0.3	0.1	Ruleta	0:20:41	0.078355
5	65	35	0.5	0.4	0.2	Ruleta	0:11:53	0.076711
6	200	90	0.5	0.4	0.1	Ruleta	1:33:19	0.065268
7	25	15	0.5	0.8	0.3	Ruleta	0:02:15	0.261787
8	90	35	0.5	0.5	0.2	Ruleta	0:16:26	0.074058
9	40	25	0.5	0.7	0.1	Ruleta	0:05:12	0.134507
10	120	85	0.5	0.4	0.1	Ruleta	0:52:13	0.070636

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para la planta 2 con funciones de membresía *GTG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.6 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.6: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	120
	Generaciones	85
	% Reemplazo	0.5
	Cruce	0.4
	Mutación	0.1
	Método de Selección	Ruleta
Tiempo del GA		0:52:13
Error Alcanzado		0.070636

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.070636.

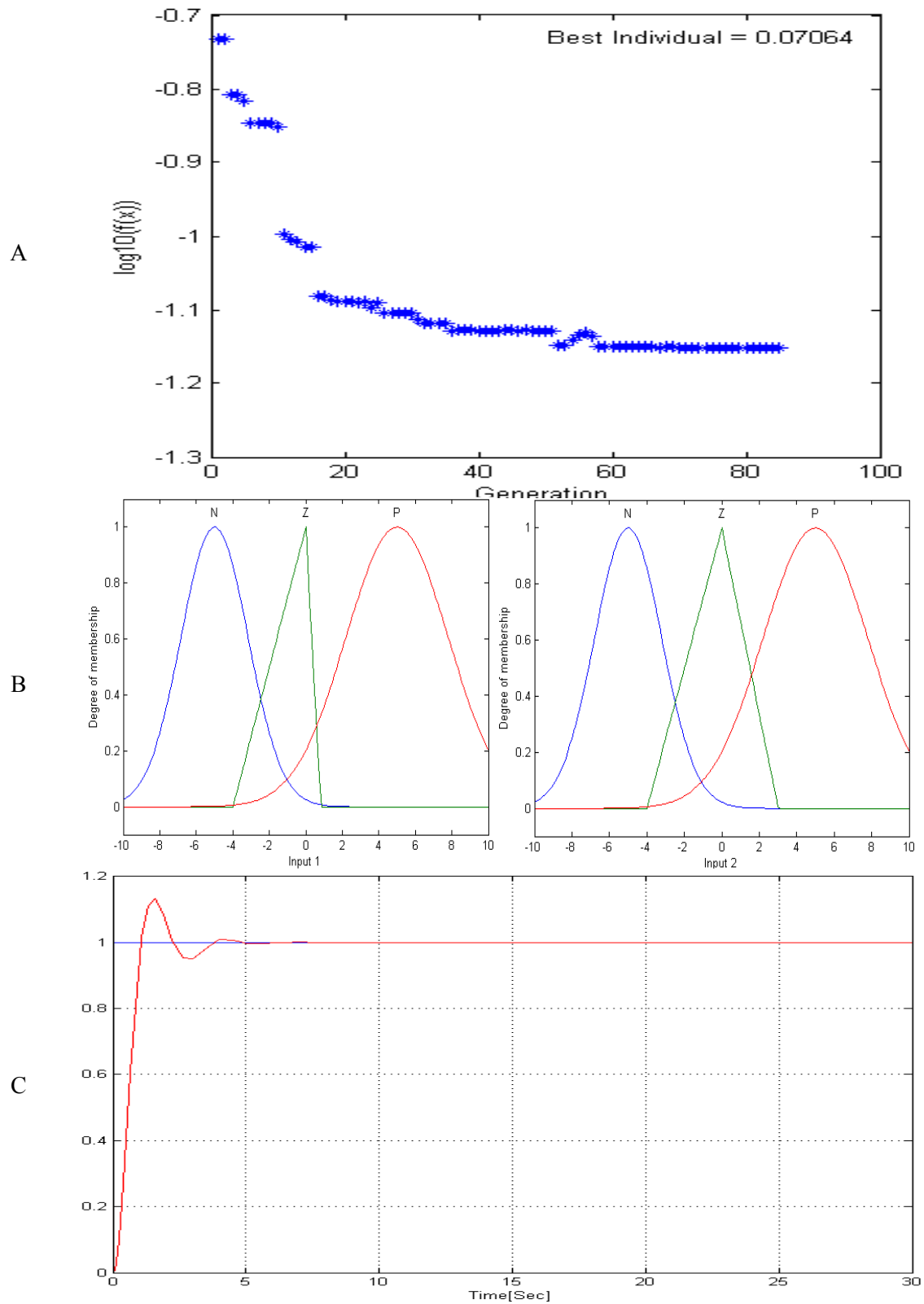


Figura 4.3: A) Evolución del algoritmo genético B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 2.

La Figura 4.3 muestra A) la evolución del algoritmo genético y su mejor individuo, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el algoritmo genético y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 2 obteniendo el control antes de los 5 segundos con un pequeño sobre pico al alcanzar la referencia.

4.1.4 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-2 con funciones de membresía *GTG* para la Planta 2

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-2 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.2) que representa la planta 2.

La Tabla 4.7 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el algoritmo genético.

En la Tabla 4.7, se muestra los resultados de las pruebas realizadas con algoritmos genéticos para la Planta 2 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

Tabla 4.7: Resultados del controlador difuso tipo-2 *GTG* para planta 2.

GA <i>GTG</i> Planta2 Tipo-2								
No.	Individuos	Generaciones	% Reemplazo	Cruce.	Mutación	Método de Selección	Tiempo del GA	Average Error
1	90	85	0.5	0.7	0.2	Ruleta	7:59:16	0.071904
2	90	65	0.5	0.7	0.2	Ruleta	5:48:37	0.072523
3	120	85	0.5	0.7	0.2	Ruleta	11:07:54	0.073215
4	45	40	0.5	0.9	0.3	Ruleta	1:51:16	0.076852
5	50	80	0.5	0.6	0.1	Ruleta	4:08:36	0.07786
6	30	55	0.5	0.8	0.3	Ruleta	1:26:11	0.079192
7	55	75	0.5	0.8	0.3	Ruleta	3:35:16	0.079204
8	40	25	0.5	0.7	0.3	Ruleta	1:04:20	0.079878
9	65	35	0.5	0.6	0.1	Ruleta	2:29:15	0.082048
10	60	40	0.5	0.7	0.2	Ruleta	2:08:08	0.089937

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para la planta 2 con funciones de membresía *GTG* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.8 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.8: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	30
	Generaciones	55
	% Reemplazo	0.5
	Cruce	0.8
	Mutación	0.3
	Método de Selección	Ruleta
Tiempo del GA		1:26:11
Error Alcanzado		0.079192

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.079192.

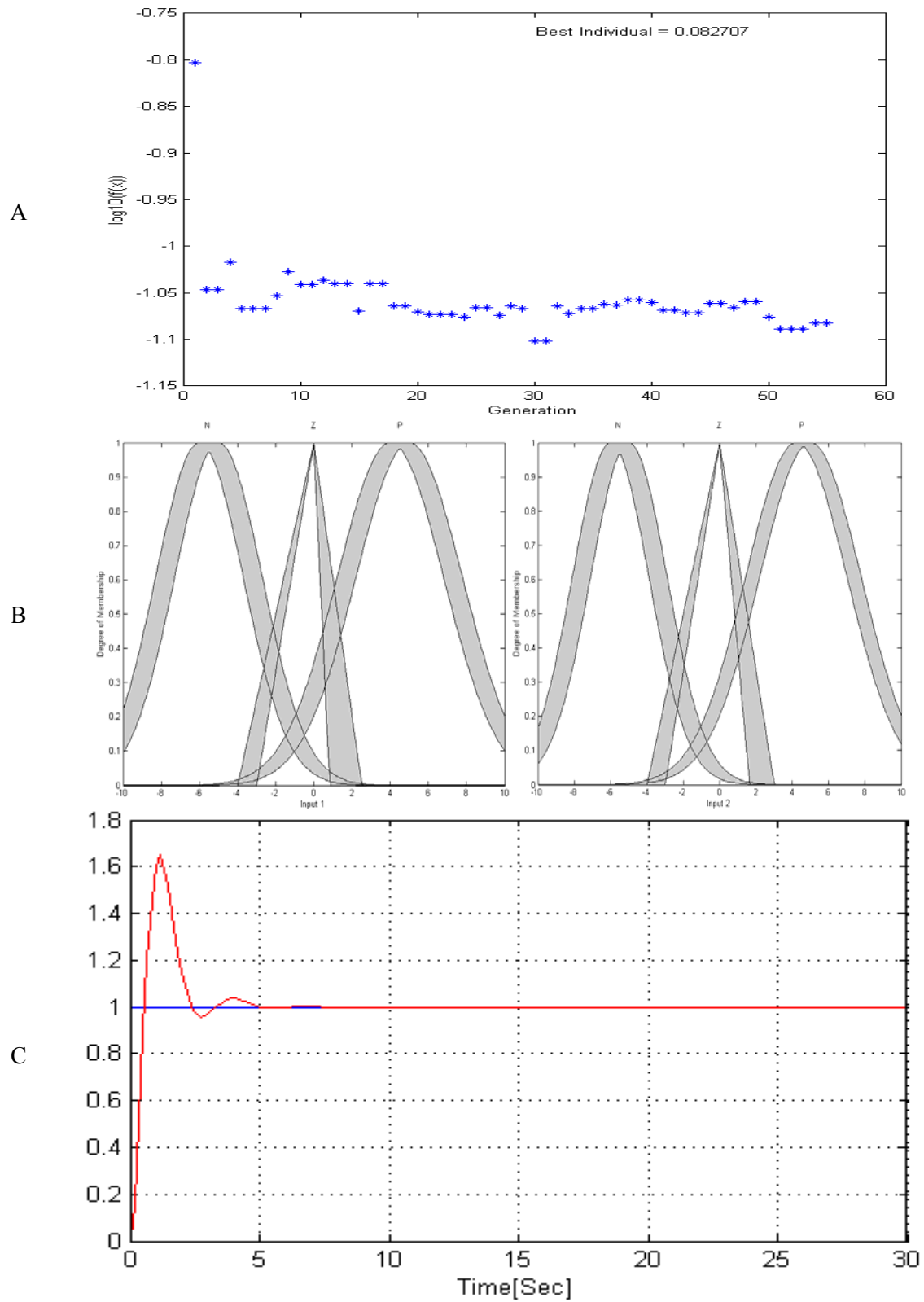


Figura 4.4: A) Evolución del algoritmo genético B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 2.

La Figura 4.4 muestra A) la evolución del algoritmo genético y su mejor individuo, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el algoritmo genético y C) el seguimiento de trayectoria del controlador difuso tipo-2 para la planta 2 obteniendo el control antes de los 5 segundos con un alto sobre pico al alcanzar la referencia.

4.1.5 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía *G**G**G* para la Planta 1

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *G**G**G* (Gaussiana-Gaussiana-Gaussiana) utilizando (3.1) que representa la planta 1.

La Tabla 4.9 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el algoritmo genético.

Tabla 4.9: Resultados del controlador difuso tipo-1 *G**G**G* para Planta 1.

GA <i>G</i> <i>G</i> <i>G</i> Planta1 Tipo-1								
No.	Individuos	Generaciones	% Reemplazo	Cruce.	Mutación	Método de Selección	Tiempo del GA	Average Error
1	90	75	0.7	0.5	0.2	Ruleta	0:19:48	0.036667
2	120	70	0.7	0.5	0.1	Ruleta	0:23:36	0.042363
3	90	75	0.7	0.5	0.2	Ruleta	0:20:25	0.07037
4	90	35	0.7	0.5	0.2	Ruleta	0:09:10	0.070813
5	50	45	0.7	0.7	0.1	Ruleta	0:06:45	0.076823
6	40	25	0.7	0.7	0.4	Ruleta	0:03:00	0.081118
7	45	30	0.7	0.6	0.2	Ruleta	0:04:03	0.082603
8	70	45	0.7	0.5	0.1	Ruleta	0:10:08	0.087568
9	150	80	0.7	0.7	0.1	Ruleta	0:33:25	0.088684
10	20	15	0.7	0.8	0.3	Ruleta	0:00:54	0.091511

En la Tabla 4.9, se muestra los resultados de las pruebas realizadas con algoritmos genéticos para la planta 1 con funciones de membresía *GGG*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para la planta 1 con funciones de membresía *GGG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.10 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.10: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	150
	Generaciones	80
	% Reemplazo	0.7
	Cruce	0.7
	Mutación	0.1
	Método de Selección	Ruleta
Tiempo del GA		0:33:25
Error Alcanzado		0.088684

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fue de 0.088684.

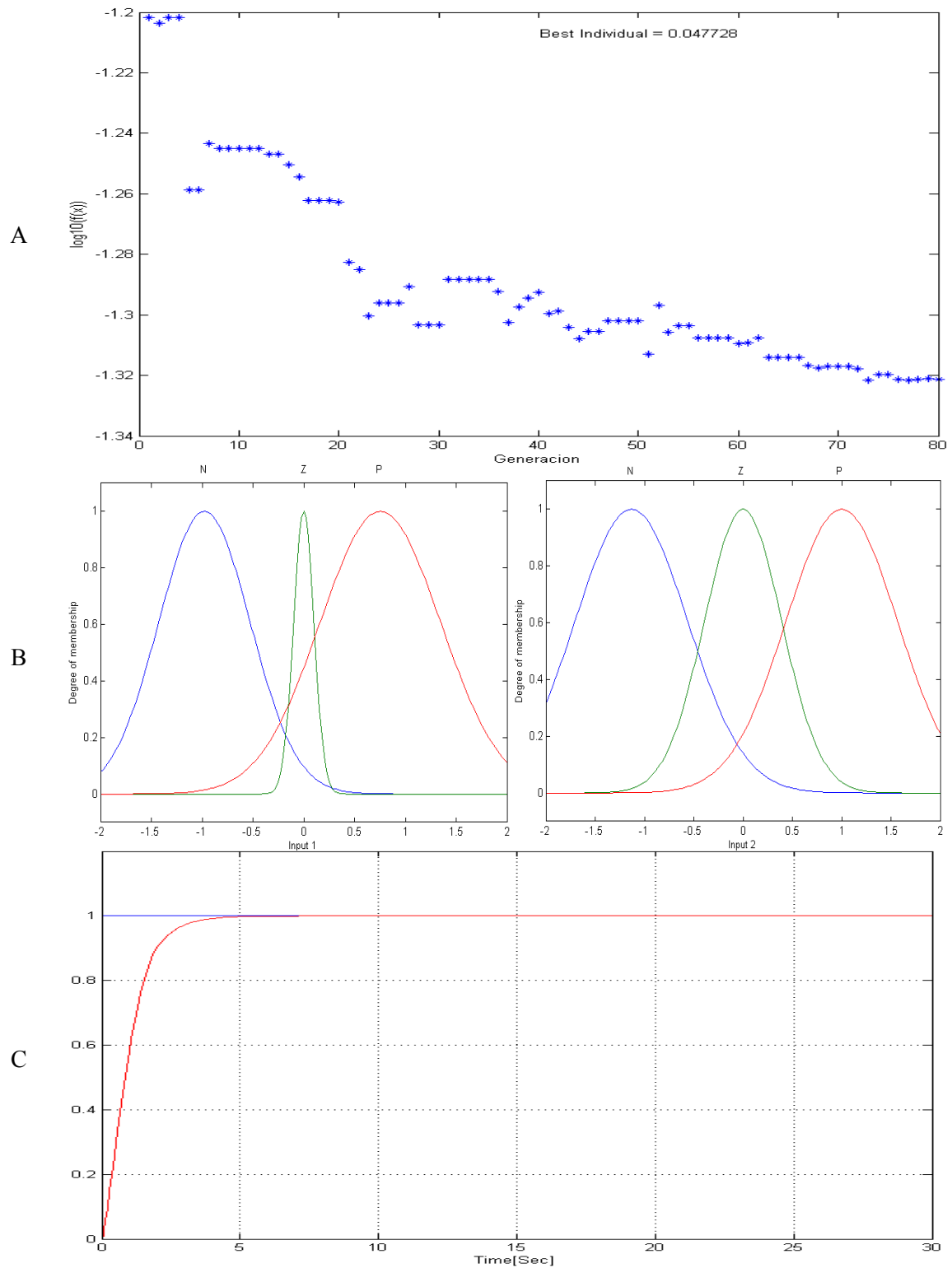


Figura 4.5: A) Evolución del algoritmo genético B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la Planta 1.

La Figura 4.5 muestra A) la evolución del algoritmo genético y su mejor individuo, B) las funciones de membresía tipo GGG de la variable de entrada 1 (“ e ” error) y la variable de entrada 2 (“ ev ” derivada del error) obtenidas por el algoritmo genético y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 1 obteniendo el control antes de los 10 segundos de manera estable.

4.1.6 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-2 con funciones de membresía GGG para la Planta 1

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-2 óptimo con funciones de membresía de tipo *GGG* utilizando (3.1) que representa la planta 1.

La Tabla 4.11 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el algoritmo genético.

En la Tabla 4.11, se muestra los resultados de las pruebas realizadas con algoritmos genéticos para la planta 1 con funciones de membresía *GGG*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

Tabla 4.11: Resultados del controlador difuso tipo-2 *GGG* para planta 1.

GA <i>GGG</i> Planta1 Tipo-2								
No.	Individuos	Generaciones	% Reemplazo	Cruce.	Mutación	Método de Selección	Tiempo del GA	Average Error
1	150	80	0.7	0.7	0.1	Ruleta	10:12:49	0.072923
2	90	75	0.7	0.5	0.2	Ruleta	6:12:19	0.075891
3	120	70	0.7	0.5	0.1	Ruleta	7:27:09	0.076167
4	70	45	0.7	0.5	0.1	Ruleta	3:01:00	0.077719
5	90	35	0.7	0.5	0.2	Ruleta	3:19:59	0.078192
6	50	45	0.7	0.7	0.1	Ruleta	2:02:14	0.07821
7	40	25	0.7	0.7	0.4	Ruleta	0:57:41	0.078347
8	45	30	0.7	0.6	0.2	Ruleta	1:11:25	0.079164
9	90	75	0.7	0.5	0.2	Ruleta	6:11:03	0.081346
10	20	15	0.7	0.8	0.3	Ruleta	0:28:03	0.083894

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para la planta 1 con funciones de membresía *GGG* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.12 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.12: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	70
	Generaciones	45
	% Reemplazo	0.7
	Cruce	0.5
	Mutación	0.1
	Método de Selección	Ruleta
Tiempo del GA		3:01:00
Error Alcanzado		0.077719

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.077719.

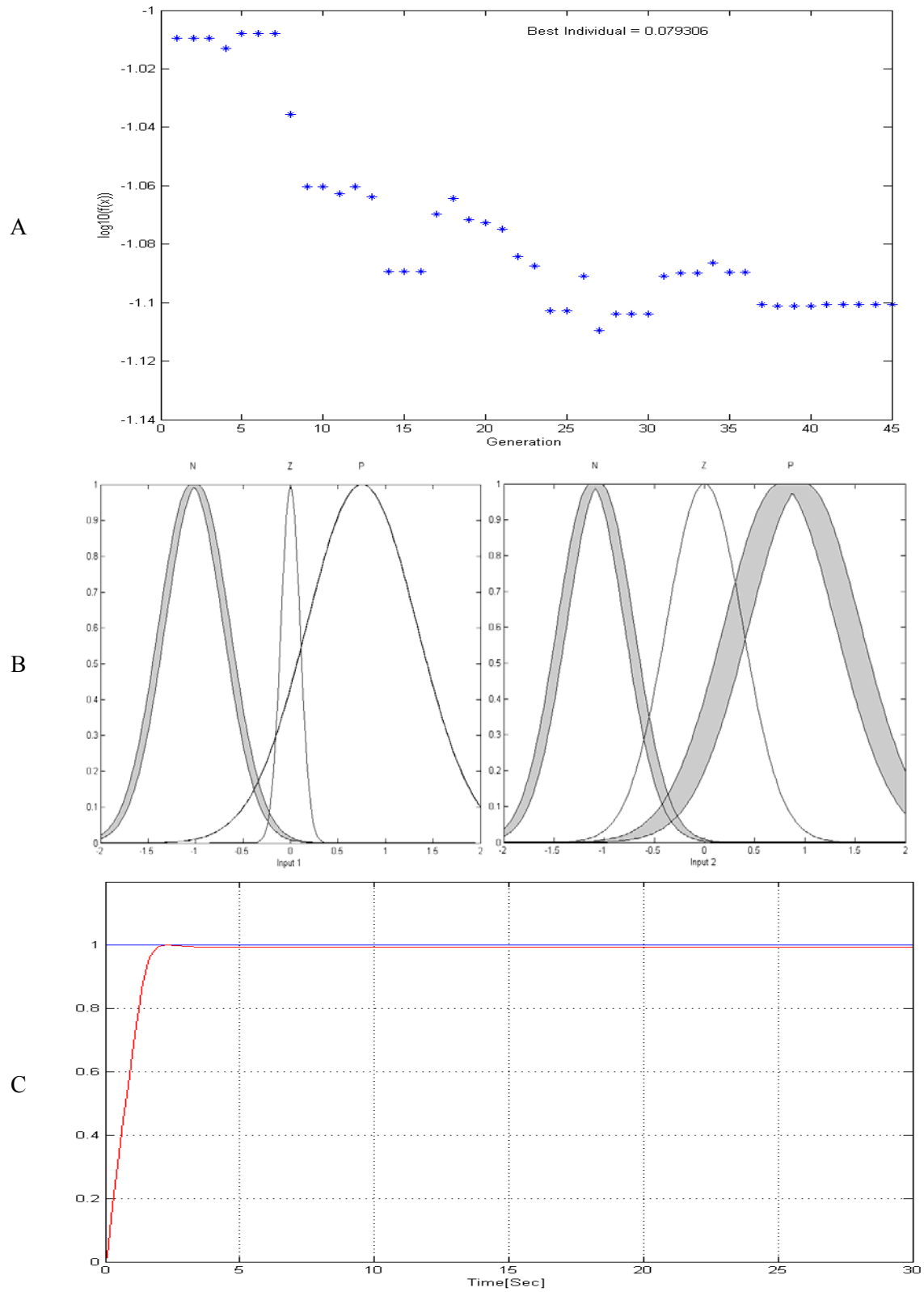


Figura 4.6: A) Evolución del algoritmo genético B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 1.

La Figura 4.6 muestra A) la evolución del algoritmo genético y su mejor individuo, B) las funciones de membresía tipo *GGG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el algoritmo genético y C) el seguimiento de trayectoria del controlador difuso tipo-2 para la planta 1 obteniendo el control antes de los 5 segundos de manera estable.

4.1.7 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía *GGG* para la Planta 2

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GGG* (Gaussiana-Gaussiana-Gaussiana) utilizando (3.2) que representa la planta 2.

La Tabla 4.13 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el algoritmo genético.

En la Tabla 4.13, se muestra los resultados de las pruebas realizadas con algoritmos genéticos para la planta 2 con funciones de membresía *GGG*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

Tabla 4.13: Resultados del controlador difuso tipo-1 *GGG* para planta 2.

GA <i>GGG</i> Planta2 Tipo-1								
No.	Individuos	Generaciones	% Reemplazo	Cruce.	Mutación	Método de Selección	Tiempo del GA	Average Error
1	120	70	0.7	0.5	0.1	Ruleta	0:22:30	0.106761
2	70	45	0.7	0.5	0.1	Ruleta	0:10:42	0.106811
3	150	80	0.7	0.7	0.1	Ruleta	0:31:09	0.107225
4	90	35	0.7	0.5	0.2	Ruleta	0:10:43	0.109182
5	90	75	0.7	0.5	0.2	Ruleta	0:23:01	0.109346
6	90	75	0.7	0.5	0.2	Ruleta	0:17:57	0.109694
7	45	30	0.7	0.6	0.2	Ruleta	0:04:32	0.111932
8	40	25	0.7	0.7	0.4	Ruleta	0:03:17	0.117842
9	50	45	0.7	0.7	0.1	Ruleta	0:07:51	0.118312
10	20	15	0.7	0.8	0.3	Ruleta	0:01:02	0.123145

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para la planta 2 con funciones de membresía *GGG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.14 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.14: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	45
	Generaciones	30
	% Reemplazo	0.7
	Cruce	0.6
	Mutación	0.2
	Método de Selección	Ruleta
Tiempo del GA		0:04:32
Error Alcanzado		0.111932

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.111932.

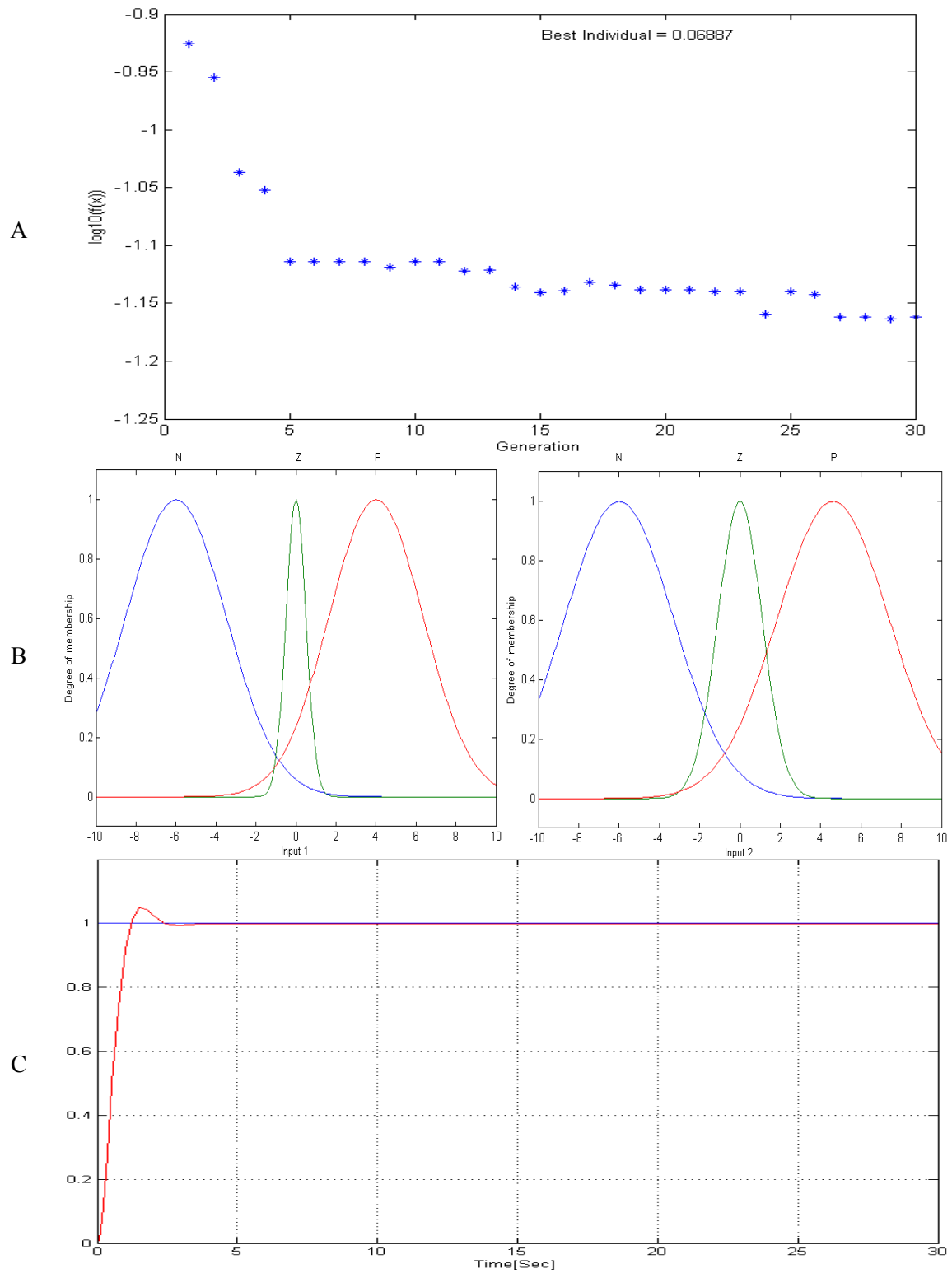


Figura 4.7: A) Evolución del algoritmo genético B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 1.

La Figura 4.7 muestra A) la evolución del algoritmo genético y su mejor individuo, B) las funciones de membresía tipo *GGG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el algoritmo genético y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 2 obteniendo el control antes de los 5 segundos con un pequeño sobre pico con respecto a la referencia.

4.1.8 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-2 con funciones de membresía *GGG* para la Planta 2

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-2 óptimo con funciones de membresía de tipo *GGG* utilizando (3.2) que representa la planta 2.

La Tabla 4.15 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el algoritmo genético.

En la Tabla 4.15, se muestra los resultados de las pruebas realizadas con algoritmos genéticos para la planta 2 con funciones de membresía *GGG*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

Tabla 4.15: Resultados del controlador difuso tipo-2 *GGG* para planta 2.

GA GGG Planta2 Tipo-2								
No.	Individuos	Generaciones	% Reemplazo	Cruce.	Mutación	Método de Selección	Tiempo del GA	Average Error
1	90	75	0.7	0.5	0.2	Ruleta	4:31:51	0.090249
2	120	70	0.7	0.5	0.1	Ruleta	5:15:38	0.091166
3	90	35	0.7	0.5	0.2	Ruleta	2:37:24	0.093232
4	150	80	0.7	0.7	0.1	Ruleta	7:36:03	0.13825
5	40	25	0.7	0.7	0.4	Ruleta	0:58:01	0.13825
6	90	75	0.7	0.5	0.2	Ruleta	4:12:43	0.1435
7	70	45	0.7	0.5	0.1	Ruleta	2:01:49	0.146866
8	45	30	0.7	0.6	0.2	Ruleta	0:58:34	0.147688
9	20	15	0.7	0.8	0.3	Ruleta	0:13:18	0.165094
10	50	45	0.7	0.7	0.1	Ruleta	1:43:13	0.179101

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para la planta 2 con funciones de membresía *GGG* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.16 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.16: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	150
	Generaciones	80
	% Reemplazo	0.7
	Cruce	0.7
	Mutación	0.1
	Método de Selección	Ruleta
Tiempo del GA		7:36:03
Error Alcanzado		0.13825

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.13825.

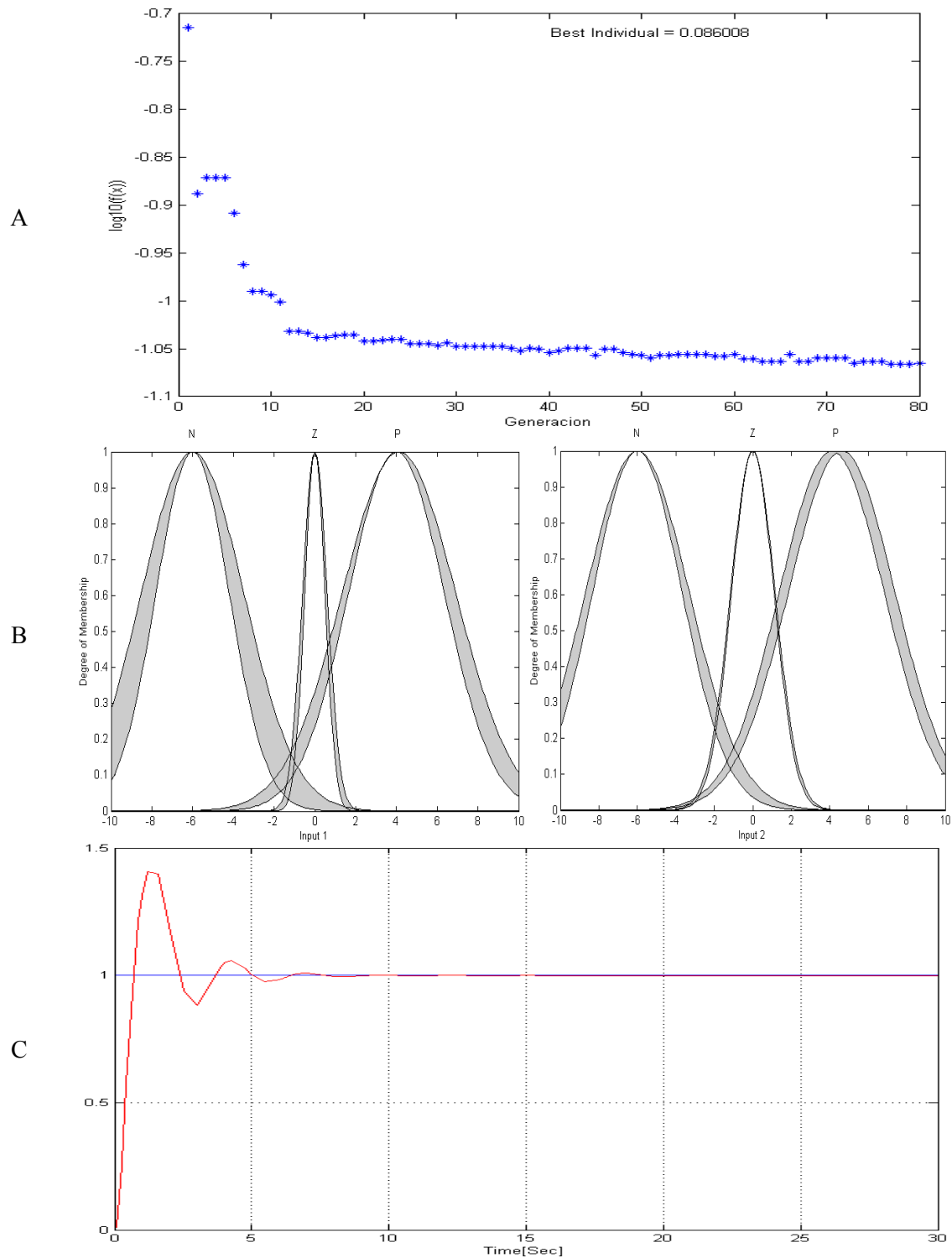


Figura 4.8: A) Evolución del algoritmo genético B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 2.

La Figura 4.8 muestra A) la evolución del algoritmo genético y su mejor individuo, B) las funciones de membresía tipo *GGG* de la variable de entrada $1e$ (“ e ” error) y la variable de entrada 2 (“ ev ” derivada del error) obtenidas por el algoritmo genético y C) el seguimiento de trayectoria del controlador difuso tipo-2 para la planta 2 obteniendo el control antes de los 10 segundos con un alto sobre pico con respecto a la referencia.

4.1.9 Resultados del uso de algoritmos genéticos para encontrar las ganancias k_1 , k_2 y k_3 del controlador PID del péndulo de rueda de inercia

Se utilizaron algoritmos genéticos para encontrar las ganancias del controlador PID para el péndulo de rueda de inercia. Se utilizó el modelo del péndulo de rueda de inercia en Matlab (3.3), para encontrar las ganancias obteniendo un error promedio de simulación, así mismo también se probó los valores encontrados en el sistema físico del péndulo de rueda de inercia.

La Tabla 4.17 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA, los valores obtenidos de las ganancias del controlador PID y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. Todas las pruebas de simulación fueron satisfactorias, pero la prueba de los valores de las ganancias en el sistema físico marco cuales eran las mejores de todo el conjunto de pruebas.

Tabla 4.17: Resultados para el controlador PID del Péndulo de rueda de inercia.

Parámetros del GA						Simulación				Experimento				
						Result.	Ganancias			Resultados Sin Perturbación			Resultados con Perturbación	
N	Pop	Max Gene	Cruce	Mut	Tiempo del GA	Sim. Error Prom.	K ₁	K ₂	K ₃	Error Prom	max tau	Tiempo de Estabilización(S)	Error Prom	max tau
1	35	40	0.7	0.3	10:14:04	0.00034	999.99	29.09	0.095	∞	∞	∞	∞	∞
2	35	40	0.7	0.3	10:11:45	0.00032	1000	22.75	0.007	∞	∞	∞	∞	∞
3	30	70	0.7	0.2	15:27:31	0.00025	1000	15.24	0.001	∞	∞	∞	∞	∞
4	30	70	0.7	0.2	18:50:42	0.00025	1000	15.11	0.001	∞	∞	∞	∞	∞
5	60	45	0.7	0.2	2:41:21	0.00025	1000	15.02	0.001	∞	∞	∞	∞	∞
6	30	50	0.7	0.2	8:57:06	0.00047	500	25.75	0.094	0.237	17.83	4	∞	∞
7	35	85	0.8	0.2	16:56:35	0.00047	500	26.45	0.097	0.31	13.77	4	∞	∞
8	35	85	0.8	0.2	10:45:58	0.00057	249.99	5	0.001	∞	∞	∞	∞	∞
9	40	65	0.8	0.2	10:38:09	0.00053	250	7.14	0.001	∞	∞	∞	∞	∞
10	25	100	0.8	0.1	10:12:16	0.00063	250	21.85	0.084	0.221	3.761	4.5	0.2741	13.126
11	30	45	0.5	0.2	6:02:57	0.00053	250	7.09	0.001	∞	∞	∞	∞	∞
12	25	60	0.8	0.3	4:40:01	0.00053	250	7.13	0.001	∞	∞	∞	∞	∞
13	35	55	0.8	0.3	5:22:01	0.00053	250	7.12	0.001	∞	∞	∞	∞	∞
14	65	85	0.8	0.2	15:49:12	0.00057	223.55	7.92	0.001	∞	∞	∞	∞	∞
15	20	30	0.8	0.1	2:18:56	0.00063	250	23.05	0.089	0.251	3.960	5	23.489	0.1729

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para las ganancias óptimas del controlador PID del Péndulo de rueda de inercia.

Experimento (Mejor Simulación)

La Tabla 4.18 muestra los parámetros utilizados para la simulación y el experimento físico del péndulo de rueda de inercia y los resultados obtenidos con las mejores ganancias.

Tabla 4.18 Parámetros utilizados en el GA, tiempo de GA y errores promedio del Péndulo.

Parámetros del A.G.	Individuos	150
	Generaciones	80
	% Reemplazo	0.7
	Cruce	0.7
	Mutación	0.1
	Método de Selección	Ruleta
	Tiempo del GA	7:36:03
Ganancias	k₁	250
	k₂	21.85
	k₃	0.0841
Simulación	Error Promedio	0.0006
Experimento	Error Promedio	0.2206
	Max Tau	3.7614
	Tiempo de Estabilización (S)	4.5
	Error Promedio/Perturbado	0.2742
	Max Tau/Perturbado	13.127

Las siguientes figuras, muestran los resultados obtenidos en la simulación y el experimento físico del péndulo de rueda de inercia con y sin perturbación.

La Figura 4.9 A) muestra la evolución del algoritmo genético y el resultado de Simulación con un error promedio de 0.00063 rad, estabilizándose antes de 1 segundo de ejecución, B) la primera imagen muestra el comportamiento del péndulo utilizando las ganancias óptimas: $k_1 = 250$, $k_2 = 21.849774$, and $k_3 = 0.0841414$; obteniendo un error promedio = 0.22061 [rad], max T = 3.7613548 [Nm] and tss = 4.5 segundos en estabilizarse y la segunda imagen muestra el comportamiento del péndulo utilizando las ganancias óptimas aplicando perturbación, obteniendo un error promedio en estado estable = 0.27417 [rad] y max T= 13.12699 [Nm].

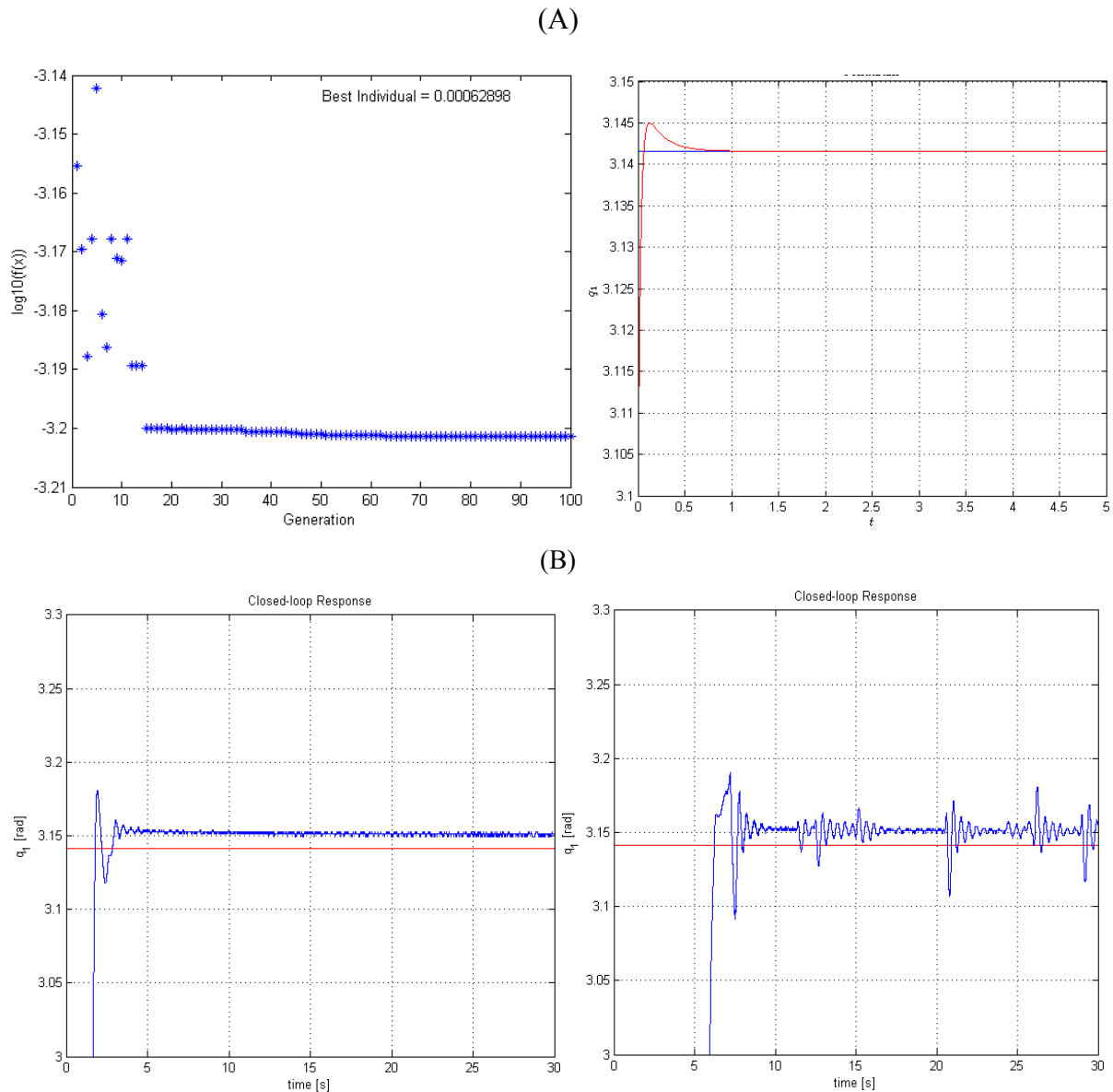


Figura 4.9: A) Evolución del algoritmo genético y resultado de simulación, B) comportamiento del péndulo físico utilizando las ganancias óptimas y C) comportamiento del péndulo físico aplicando perturbación.

4.1.10 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 para un robot autónomo móvil

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *TTT* (Trapezoidal-Triangular-Trapezoidal) utilizando (3.5) y (3.6) que representa al robot autónomo móvil.

La Tabla 4.19 contiene los valores utilizados en la configuración del algoritmo genético, el tiempo de ejecución del GA y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron el número de individuos y generaciones, usando diferentes valores de cruce y mutación y ordenada en forma ascendente. La novena columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el algoritmo genético.

En la Tabla 4.19 se muestra los resultados de las pruebas realizadas con algoritmos genéticos para el robot autónomo móvil con funciones de membresía *TTT*, mostrando los parámetros utilizados en el algoritmo genético y los mejores y peores errores promedio.

Tabla 4.19: Resultados del controlador difuso tipo-1 *TTT* para el robot autónomo móvil.

No.	Indiv.	Gener.	% Reempl	Cruce	Mut.	Método de Selección	Tiempo del GA	Error Promedio de Velocidad	Error Promedio de Posición
1	150	50	0.7	0.6	0.3	Ruleta	0:51:23	1.890014991	0.154186634
2	50	60	0.7	0.6	0.4	Ruleta	0:21:30	1.887862579	0.154245362
3	200	100	0.7	0.4	0.1	Ruleta	2:19:46	1.887844486	0.154342535
4	100	80	0.7	0.8	0.3	Ruleta	0:55:04	1.893962567	0.154346986
5	70	60	0.7	0.8	0.4	Ruleta	0:29:23	1.888086839	0.154396096
6	90	60	0.7	0.9	0.4	Ruleta	0:37:37	1.907234497	0.15448439
7	70	50	0.7	0.8	0.3	Ruleta	0:24:16	1.889902211	0.154608583
8	100	25	0.7	0.8	0.3	Ruleta	0:17:47	1.894477161	0.155028553
9	80	80	0.7	0.9	0.3	Ruleta	0:26:53	1.894309034	0.155034839
10	30	40	0.7	0.7	0.2	Ruleta	0:08:27	1.894309034	0.155034839
11	20	50	0.7	0.6	0.2	Ruleta	0:07:07	1.894309034	0.155034839
12	50	30	0.7	0.7	0.2	Ruleta	0:10:27	1.894309913	0.155034922
13	65	40	0.7	0.8	0.2	Ruleta	0:17:50	1.894309913	0.155034922
14	35	40	0.7	0.7	0.3	Ruleta	0:09:41	1.89432485	0.155036056
15	30	60	0.7	0.8	0.5	Ruleta	0:12:32	1.894418663	0.155043997
16	25	40	0.7	0.8	0.6	Ruleta	0:07:02	1.895086127	0.155066243
17	80	25	0.7	0.7	0.4	Ruleta	0:13:59	1.907962499	0.155135393
18	90	60	0.7	0.9	0.4	Ruleta	0:37:33	1.915751143	0.155779531
19	30	25	0.7	0.9	0.5	Ruleta	0:05:16	1.923661138	0.15684664
20	80	20	0.7	0.8	0.6	Ruleta	0:11:05	1.961330828	0.15939811
21	80	20	0.7	0.8	0.6	Ruleta	0:11:07	1.972232244	0.160377445
22	10	25	0.7	0.8	0.2	Ruleta	0:01:42	2.249256952	0.180101042
23	10	20	0.7	0.8	0.2	Ruleta	0:01:18	2.356856489	0.190371141
24	20	15	0.7	0.8	0.2	Ruleta	0:02:02	2.467624538	0.198579022

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para el robot autónomo móvil con funciones de membresía *TTT* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.20 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

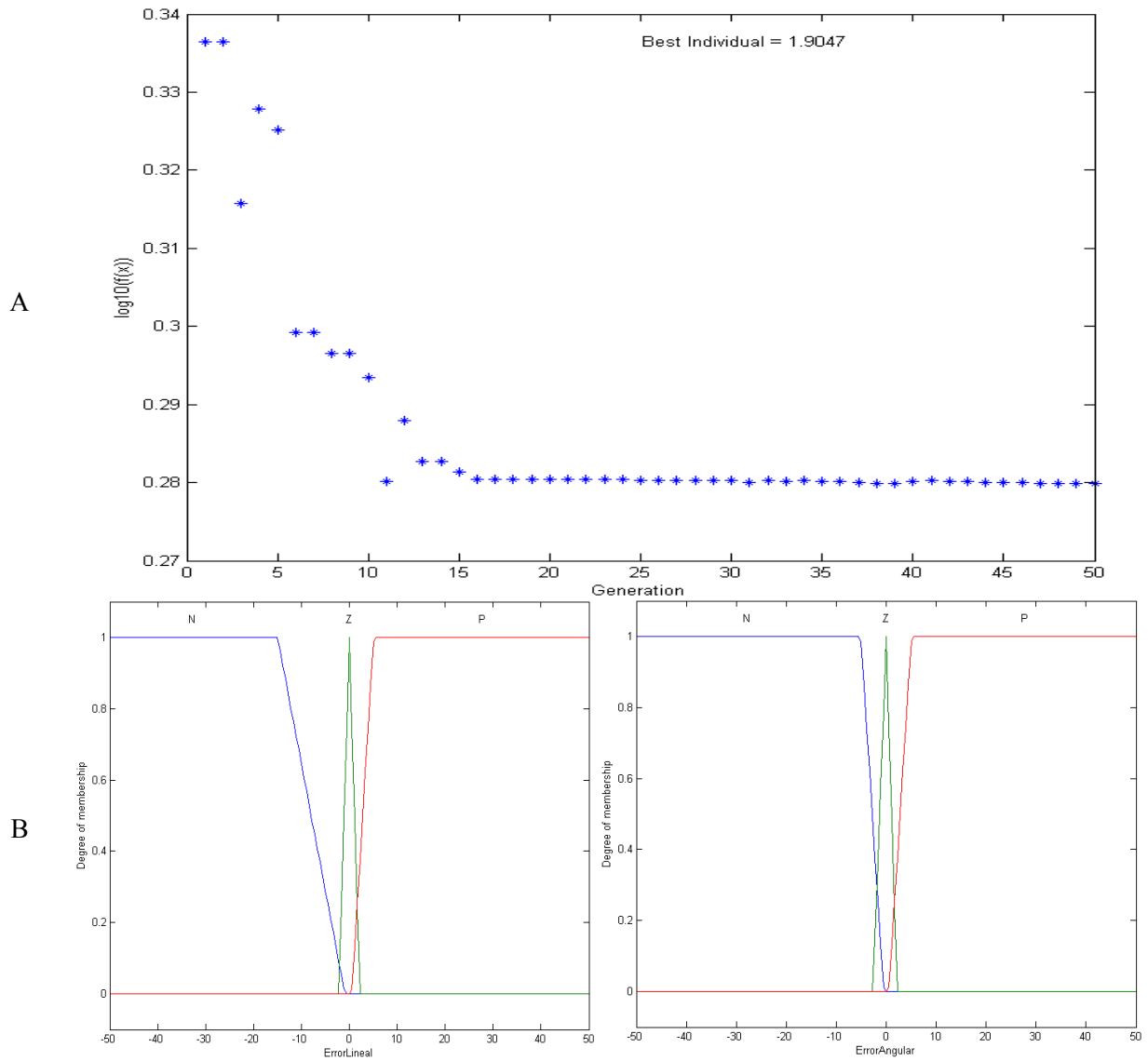


Figura 4.10: A) Evolución del algoritmo genético y B) funciones de membresía de la entrada 1 y entrada 2 del controlador difuso tipo-1 óptimo.

Tabla 4.20: Parámetros utilizados en el GA, tiempo de GA y error promedio para el experimento.

Parámetros del A.G.	Individuos	150
	Generaciones	50
	% Reemplazo	0.7
	Cruce	0.6
	Mutación	0.3
	Método de Selección	Ruleta
Tiempo del GA		0:51:23
Error Alcanzado		0.154186634

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.154186634.

La Figura 4.10 A) muestra la evolución del algoritmo genético y el mejor individuo y B) muestra las funciones de membresía la variable de entrada 1 (error de velocidad lineal) y la variable de entrada 2 (error de velocidad angular) del controlador difuso tipo-1 óptimo obtenido por el algoritmo genético.

La figura 4.11 A) muestra la respuesta los errores de velocidad lineal y la velocidad angular y los errores de torque derecho y torque izquierdo y B) muestra los errores de posición $e(x)$, $e(y)$ y $e(\theta)$ y se puede observar como tienden a cero de manera estable y C) el seguimiento de trayectoria del robot autónomo móvil con el mejor controlador difuso tipo-1 obtenido con el algoritmo genético.

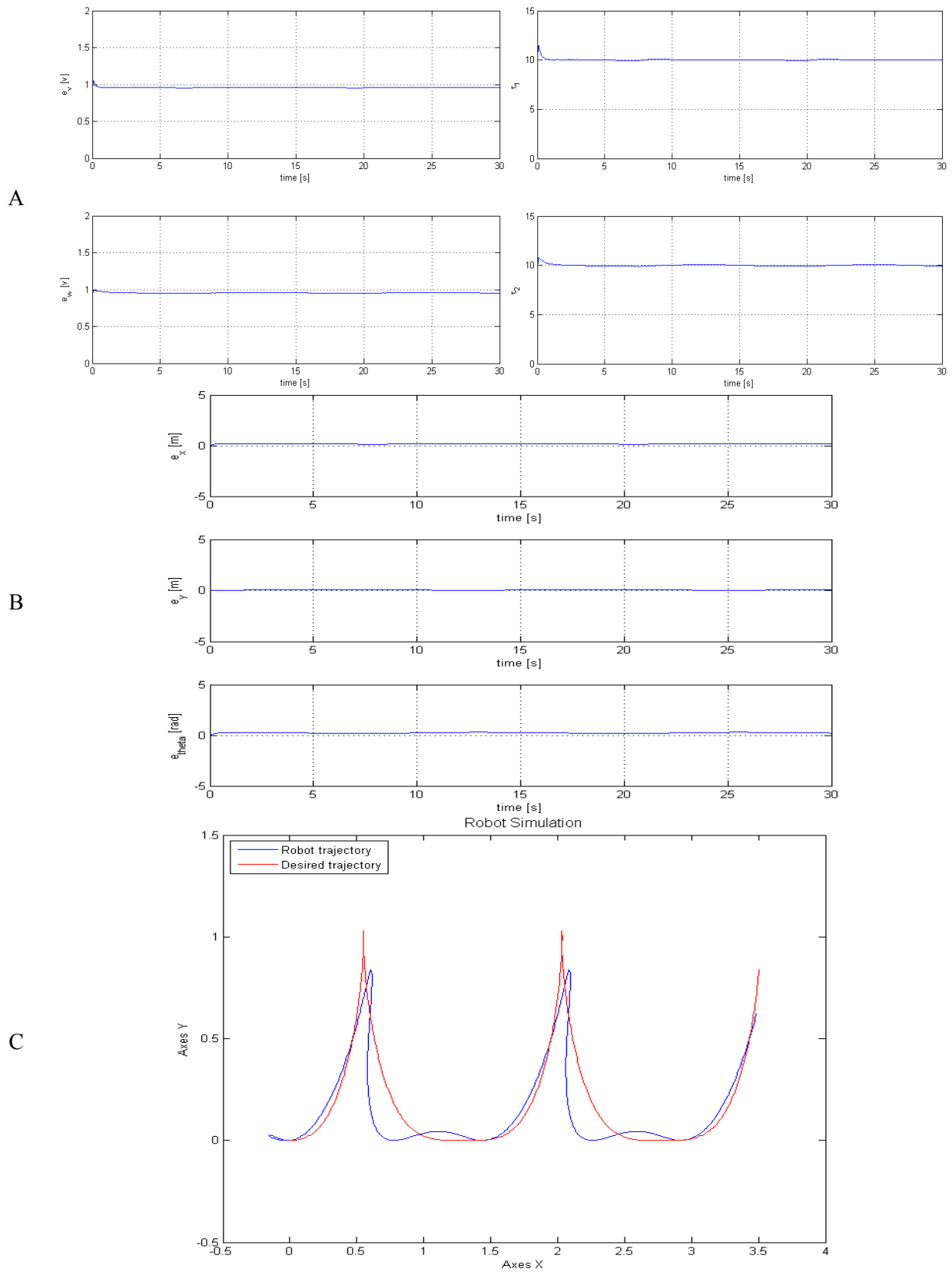


Figura 4.11: Resultados de simulación del robot autónomo móvil utilizando el controlador difuso tipo-1 optimizado.

4.2 Resultados del uso de Optimización por medio de Partículas (PSO)

En esta sección se presentan los resultados obtenidos haciendo uso de la optimización por medio de partículas para los casos de aplicación descritos a continuación.

4.2.1 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados manualmente

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 1.

La Tabla 4.21 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución del PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1) y social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO.

En la Tabla 4.21, se muestra los resultados de las pruebas realizadas con PSO para la Planta 1 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el PSO, los mejores y peores errores promedio.

Tabla 4.21: Resultados del controlador difuso tipo-1 *GTG* para planta 1.

PSO GTG Planta 1 Tipo-1							
No.	Partículas (Swarm)	Iteraciones	C1	C2	Inercia	Tiempo de PSO	Error Promedio
1	200	70	1	1	1	0:05:27	0.06321
2	200	70	0.5	0.5	1	0:07:35	0.05276
3	200	70	0.25	0.25	1	0:15:35	0.05576
4	200	70	0.15	0.15	1	0:19:15	0.09717
5	200	70	0.05	0.05	1	0:26:10	0.094
6	200	70	0.005	0.005	1	0:26:56	0.11377

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para la planta 1 con funciones de membresía *GTG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.22 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.22: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	200
	Iteraciones	70
	C1	0.15
	C2	0.15
	Inercia	1
Tiempo del GA		0:19:15
Error Alcanzado		0.09717

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.09717.

La Figura 4.12 muestra A) el comportamiento de las partículas de PSO, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por PSO y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 1 obteniendo el control antes de los 10 segundos de manera estable.

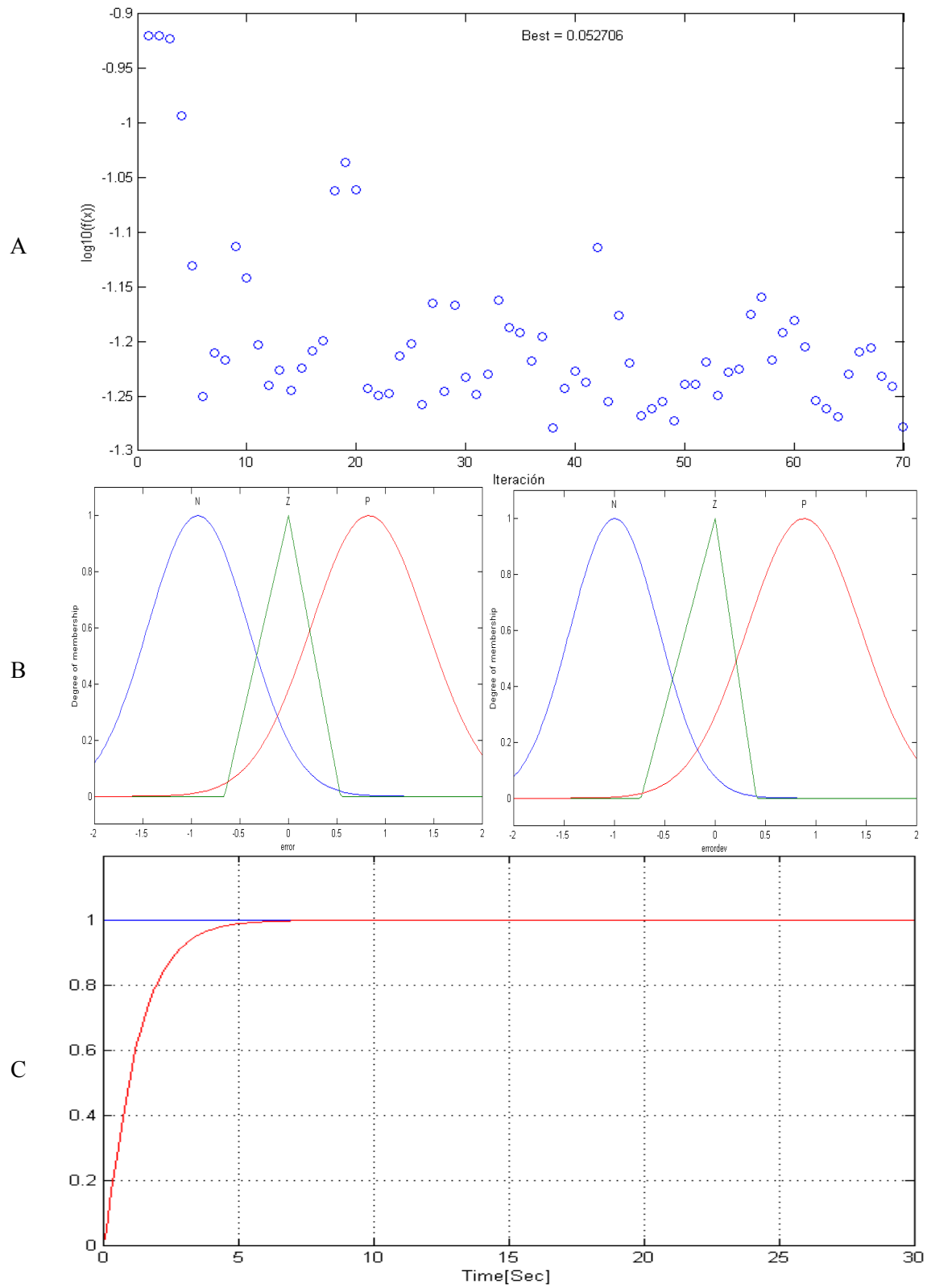


Figura 4.12: A) Comportamiento de las partículas de PSO B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 1.

4.2.2 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados manualmente

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* utilizando (3.2) que representa la planta 2.

La Tabla 4.23 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución del PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO.

En la Tabla 4.23, se muestra los resultados de las pruebas realizadas con PSO para la planta 2 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el PSO, los mejores y peores errores promedio.

Tabla 4.23: Resultados del controlador difuso tipo-1 *GTG* para planta 2.

PSO GTG Planta 1 Tipo-1							
No.	Partículas (Swarm)	Iteraciones	C1	C2	Inercia	Tiempo de PSO	Error Promedio
1	200	70	1	1	1	0:04:59	0.14286
2	200	70	0.5	0.5	1	0:07:11	0.12527
3	200	70	0.25	0.25	1	0:10:21	0.14508
4	200	70	0.15	0.15	1	0:16:25	0.15090
5	200	70	0.05	0.05	1	0:17:56	0.13431
6	200	70	0.005	0.005	1	0:21:18	0.14513

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para la planta 2 con funciones de membresía *GTG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.24 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.24: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	200
	Iteraciones	70
	C1	0.05
	C2	0.05
	Inercia	1
Tiempo del GA		0:17:56
Error Alcanzado		0.13431

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.13431.

La Figura 4.13 muestra A) el comportamiento de las partículas de PSO, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por PSO y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 2 obteniendo el control antes de los 5 segundos con un pequeño sobre pico al alcanzar la referencia.

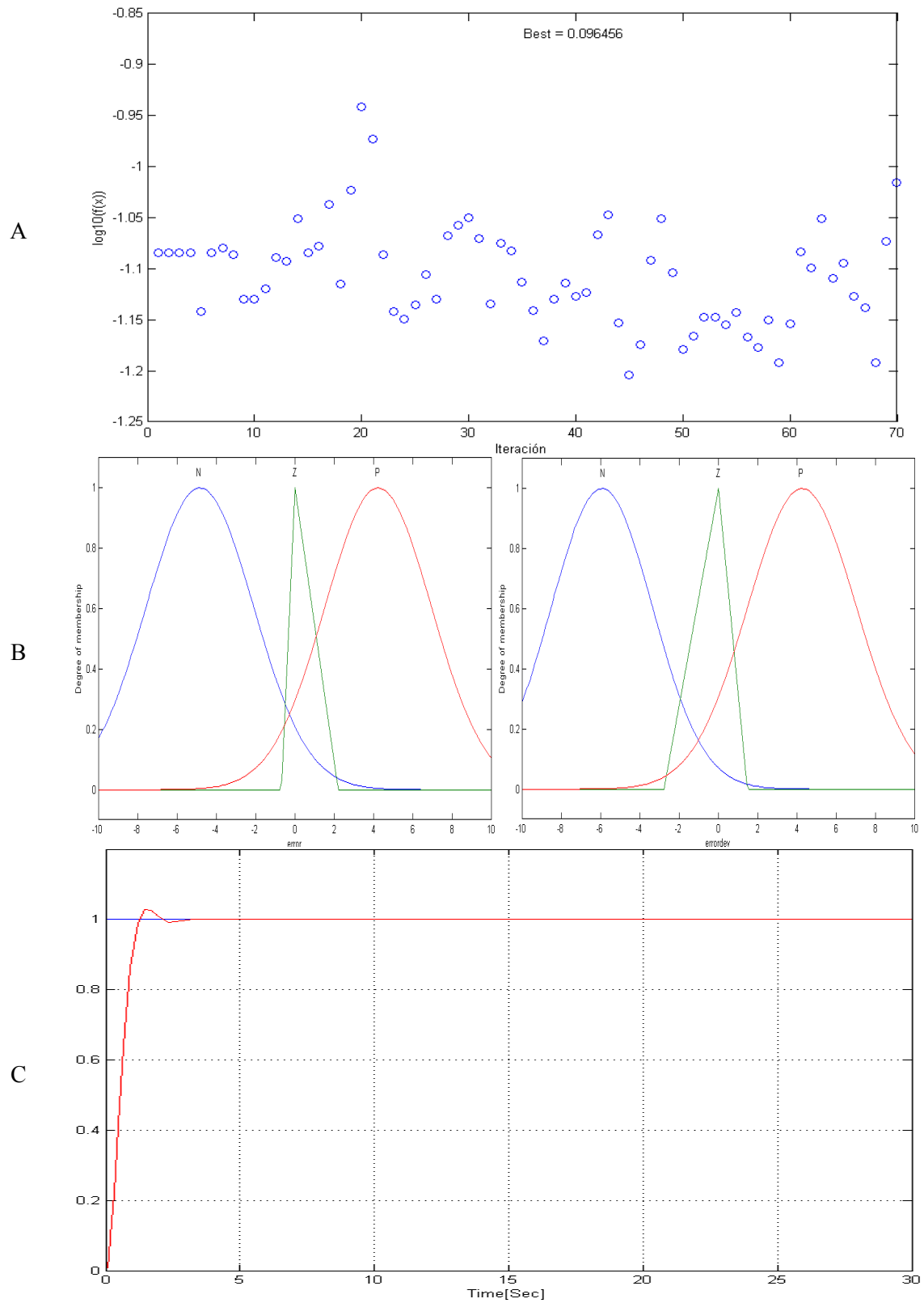


Figura 4.13: A) Comportamiento de las partículas de PSO B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 2.

4.2.3 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GGG para la Planta 1 con parámetros de PSO asignados manualmente

Se utilizó PSO para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GGG* (Gaussiana-Gaussiana-Gaussiana) utilizando (3.1) que representa la planta 1.

La Tabla 4.25 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución del PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1) y social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO.

En la Tabla 4.25, se muestra los resultados de las pruebas realizadas con PSO para la planta 1 con funciones de membresía *GGG*, mostrando los parámetros utilizados en el PSO, los mejores y peores errores promedio.

Tabla 4.25: Resultados del controlador difuso tipo-1 *GGG* para planta 1.

PSO GGG Planta 1 Tipo-1							
No.	Partículas (Swarm)	Iteraciones	C1	C2	Inercia	Tiempo de PSO	Error Promedio
1	200	70	1	0.1	1	0:20:37	0.07505
2	200	70	0.5	0.5	1	0:10:16	0.08003
3	200	70	0.25	0.25	1	0:15:34	0.08023
4	200	70	0.15	0.15	1	0:16:42	0.07996
5	200	70	0.05	0.05	1	0:19:56	0.07300
6	200	70	0.005	0.005	1	0:25:35	0.07842
7	200	70	1	1	1	0:06:26	0.08629
8	200	70	0.1	1	1	0:02:41	0.08111

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para la planta 1 con funciones de membresía *GGG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.26 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.26: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	200
	Iteraciones	70
	C1	1
	C2	0.1
	Inercia	1
Tiempo del GA		0:20:37
Error Alcanzado		0.07505

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.07505.

La Figura 4.14 muestra A) el comportamiento de las partículas de PSO, B) las funciones de membresía tipo *GGG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por PSO y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 1 obteniendo el control antes de los 10 segundos de manera estable.

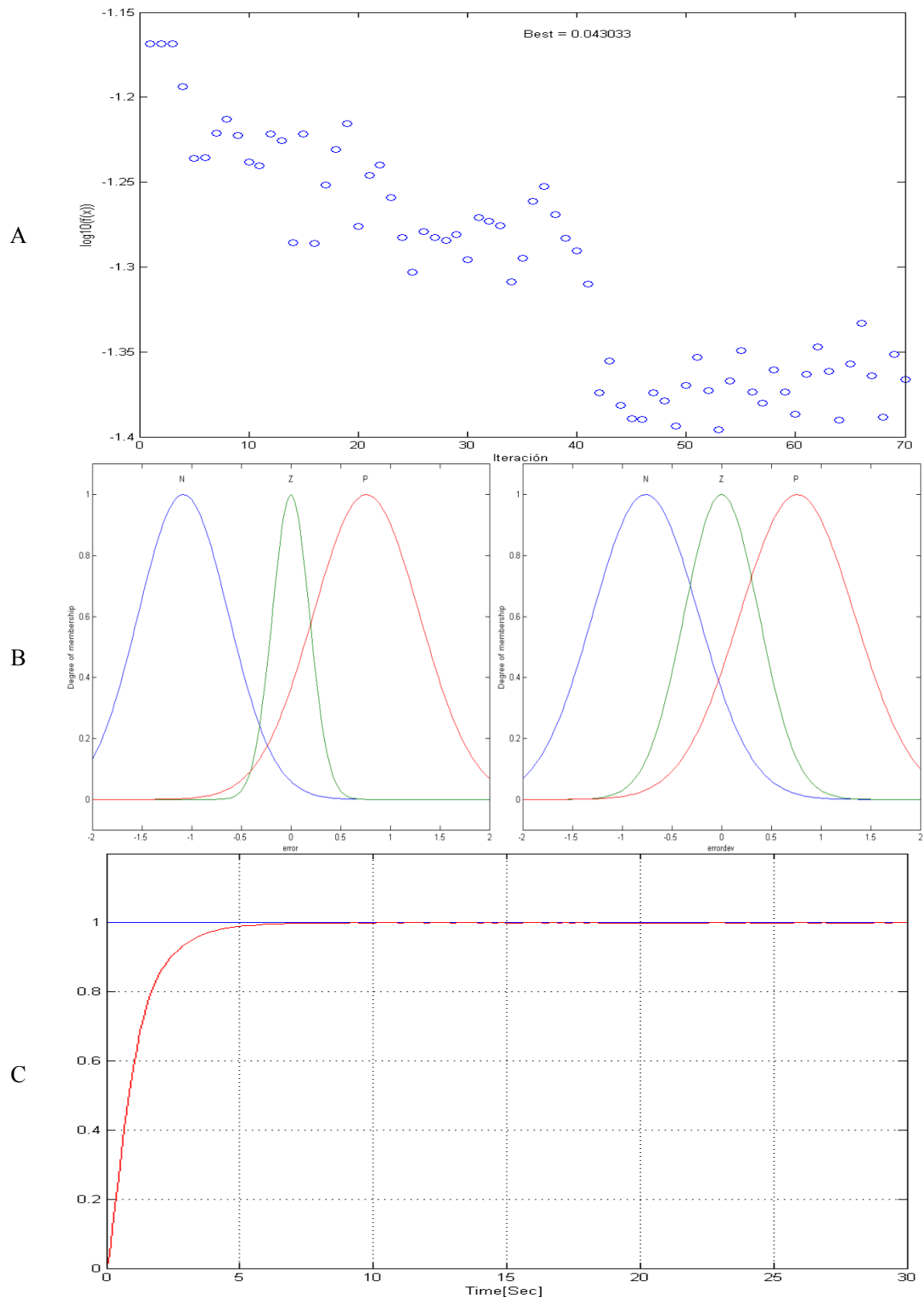


Figura 4.14: A) Comportamiento de las partículas de PSO, B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 1.

4.2.4 Resultados del uso de algoritmos genéticos para el controlador difuso tipo-1 con funciones de membresía GGG para la Planta 2 con parámetros de PSO asignados manualmente

Se utilizó PSO para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GGG* (Gaussiana-Gaussiana-Gaussiana) utilizando (3.2) que representa la planta 2.

La Tabla 4.27 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución del PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO.

En la Tabla 4.27, se muestra los resultados de las pruebas realizadas con PSO para la planta 2 con funciones de membresía *GGG*, mostrando los parámetros utilizados en el PSO, los mejores y peores errores promedio.

Tabla 4.27: Resultados del controlador difuso tipo-1 *GGG* para planta 2.

PSO GGG Planta2 Tipo-1							
No.	Partículas (Swarm)	Iteraciones	C1	C2	Inercia	Tiempo de PSO	Error Promedio
1	200	70	1	0.1	1	0:17:11	0.13957
2	200	70	0.5	0.5	1	0:16:56	0.13841
3	200	70	0.25	0.25	1	0:21:20	0.14725
4	200	70	0.15	0.15	1	0:27:13	0.13390
5	200	70	0.05	0.05	1	0:22:56	0.13367
6	200	70	0.005	0.005	1	0:24:05	0.13684
7	200	70	1	1	1	0:08:12	0.14095

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para la planta 2 con funciones de membresía *GGG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.28 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.28: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	200
	Iteraciones	70
	C1	0.25
	C2	0.25
	Inercia	1
Tiempo del GA		0:21:20
Error Alcanzado		0.14725

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.14725.

La Figura 4.15 muestra A) el comportamiento de las partículas de PSO, B) las funciones de membresía tipo *GGG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por PSO y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 2 obteniendo el control antes de los 5 segundos con un pequeño sobre pico al alcanzar la referencia.

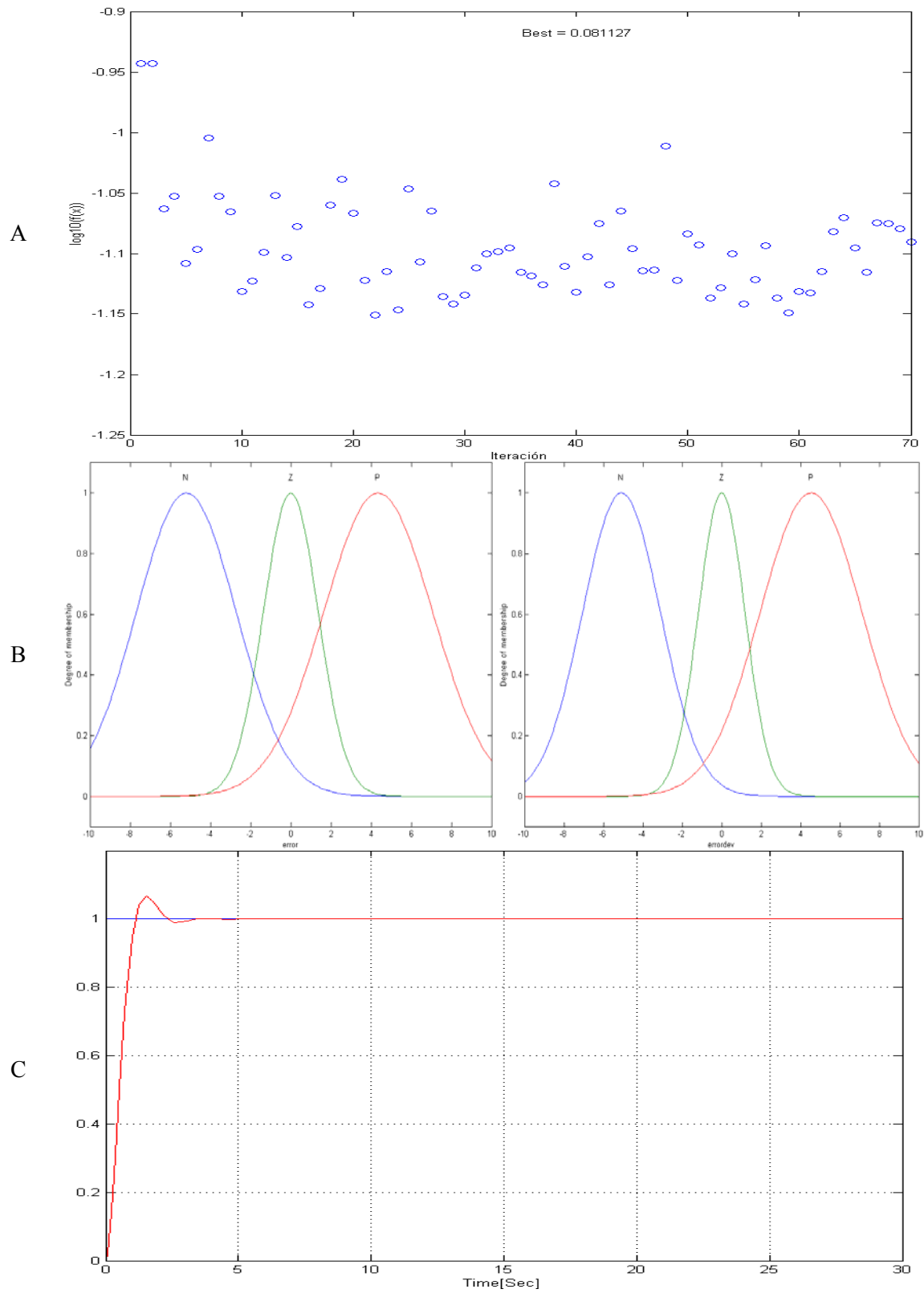


Figura 4.15: A) Comportamiento de las partículas de PSO, B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 2.

4.2.5 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados de forma aleatoria

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 1.

La Tabla 4.29 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución del PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO.

En la Tabla 4.29, se muestra los resultados de las pruebas realizadas con PSO para la planta 1 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el PSO, los mejores y peores errores promedio.

Tabla 4.29: Resultados del controlador difuso tipo-1 *GTG* para planta 1.

PSO GTG Planta 1 Tipo-1							
No.	Partículas (Swarm)	Iteraciones	C1	C2	Inercia	Tiempo de PSO	Error Promedio
1	200	70	0.4542	0.5052	0.9038	0:26:26	0.081179
2	200	70	0.9667	0.3883	0.8501	0:44:46	0.086795
3	200	70	0.183	0.8722	0.9289	0:09:23	0.087324
4	200	70	0.6798	0.7371	0.6119	0:54:22	0.089289
5	200	70	0.0016	0.8994	0.6983	0:45:59	0.090177
6	200	70	0.484	0.704	0.7194	0:49:16	0.092952
7	200	70	0.7046	0.9151	0.9988	0:05:59	0.095280
8	200	70	0.1344	0.3959	0.2534	0:58:29	0.096134
9	200	70	0.1223	0.5346	0.0681	0:57:48	0.103938
10	200	70	0.1589	0.0366	0.6223	1:04:45	0.109765

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para la planta 1 con funciones de membresía *GTG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.30 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.30: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	200
	Iteraciones	70
	C1	0.4542
	C2	0.5052
	Inercia	0.9038
Tiempo del GA		0:26:26
Error Alcanzado		0.081179

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.081179.

La Figura 4.16 muestra A) el comportamiento de las partículas de PSO, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por PSO y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 1 obteniendo el control antes de los 10 de manera estable.

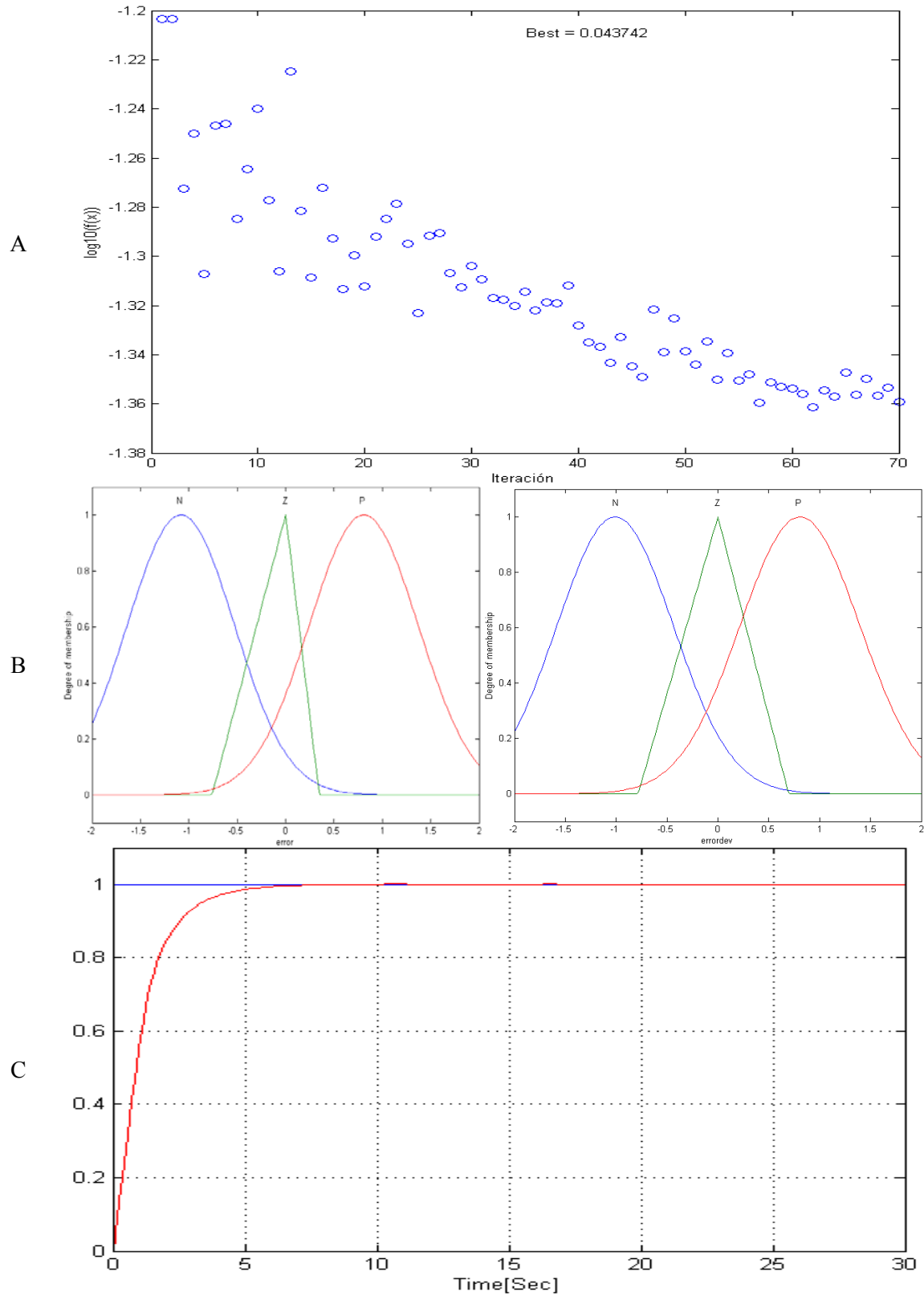


Figura 4.16: A) Comportamiento de las partículas de PSO B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 1.

4.2.6 Resultados del uso de PSO para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados de forma aleatoria

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 1.

La Tabla 4.31 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución del PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO.

En la Tabla 4.31, se muestra los resultados de las pruebas realizadas con PSO para la planta 1 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el PSO, los mejores y peores errores promedio.

Tabla 4.31: Resultados del controlador difuso tipo-2 *GTG* para planta 1.

PSO GTG Planta 1 Tipo-2							
No.	Partículas (Swarm)	Iteraciones	C1	C2	Inercia	Tiempo de PSO	Error Promedio
1	200	90	0.5149	0.3317	0.6808	7:05:20	0.08028
2	200	70	0.8149	0.9059	0.1706	12:44:19	0.08794
3	200	70	0.8129	0.8159	0.1906	10:52:09	0.08894
4	200	70	0.7646	0.9229	0.1096	9:50:45	0.12521
5	200	70	0.8168	0.9359	0.4806	11:05:43	0.12630

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para la planta 1 con funciones de membresía *GTG* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.32 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.32: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	200
	Iteraciones	90
	C1	0.5149
	C2	0.3317
	Inercia	0.6808
Tiempo del GA		7:05:20
Error Alcanzado		0.08028

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fue de 0.08028.

La Figura 4.17 muestra A) el comportamiento de las partículas de PSO, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por PSO y C) el seguimiento de trayectoria del controlador difuso tipo-2 para la planta 1 obteniendo el control antes de los 10 segundos con un pequeño sobre pico al alcanzar la referencia.

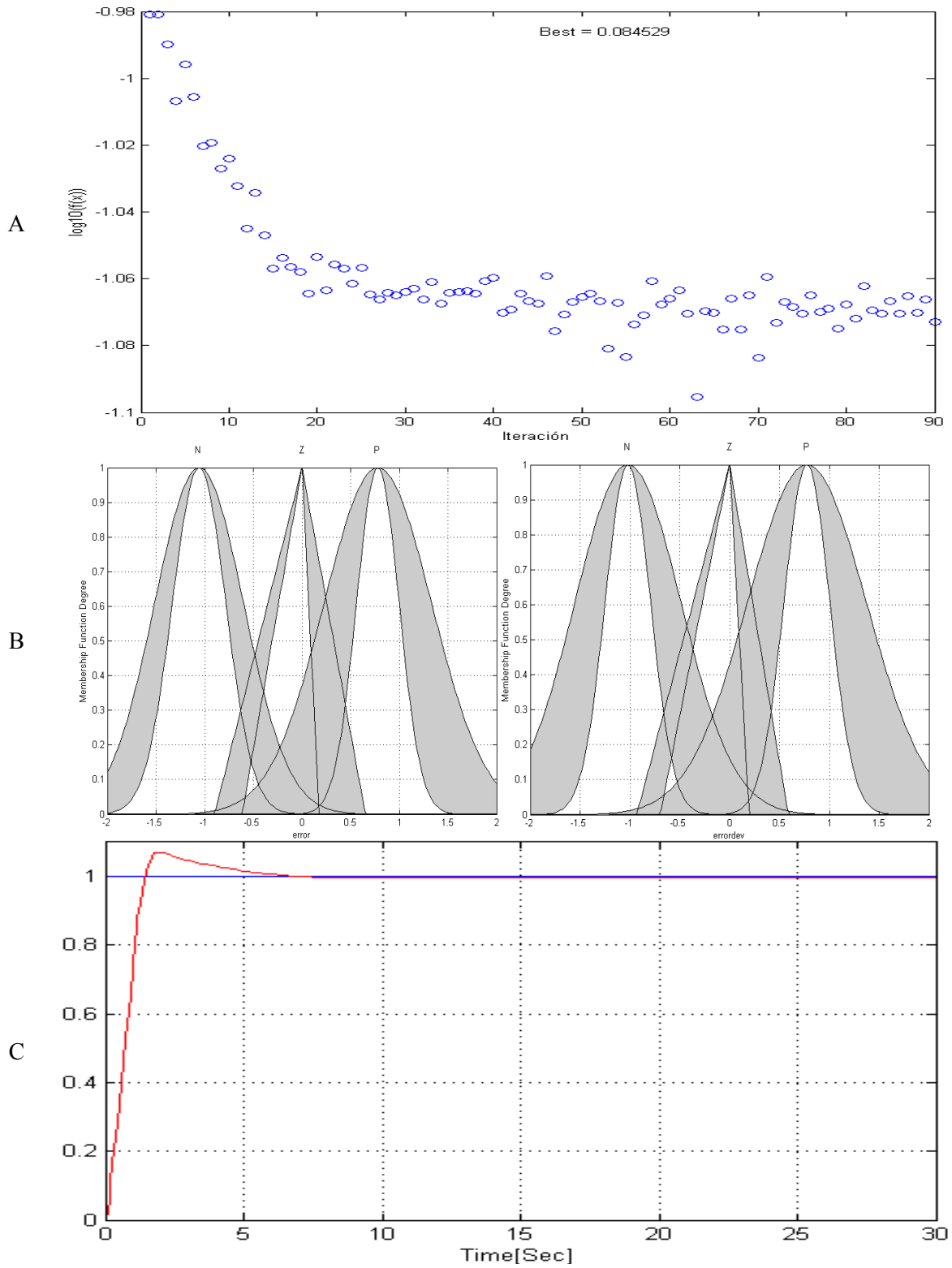


Figura 4.17: A) Comportamiento de las partículas de PSO, B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 1.

4.2.7 Resultados del uso de PSO para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados de forma aleatoria

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 2.

La Tabla 4.33 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución del PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO.

En la Tabla 4.33, se muestra los resultados de las pruebas realizadas con PSO para la planta 2 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el PSO, los mejores y peores errores promedio.

Tabla 4.33: Resultados del controlador difuso tipo-1 *GTG* para planta 2.

PSO GTG Planta2 Tipo-1							
No.	Partículas (Swarm)	Iteraciones	C1	C2	Inercia	Tiempo de PSO	Error Promedio
1	200	70	0.0429	0.6746	0.8477	0:34:32	0.131933
2	200	70	0.7580	0.1716	0.5573	0:49:17	0.136293
3	200	70	0.7488	0.0884	0.9051	0:39:05	0.141524
4	200	70	0.8149	0.9059	0.1706	0:49:38	0.148655
5	200	70	0.0660	0.7696	0.5558	0:44:32	0.150427
6	200	70	0.8660	0.1036	0.5554	1:05:25	0.151731
7	200	70	0.5543	0.6067	0.4974	0:50:25	0.152046
8	200	70	0.7154	0.2479	0.5130	0:55:45	0.154302
9	200	70	0.0158	0.0882	0.2155	1:02:43	0.160239
10	200	70	0.4458	0.9566	0.3233	0:51:22	0.164745

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para la planta 2 con funciones de membresía *GTG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.34 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.34: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	200
	Iteraciones	70
	C1	0.0429
	C2	0.6746
	Inercia	0.8477
Tiempo del GA		0:34:32
Error Alcanzado		0.131933

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fue de 0.131933.

La Figura 4.18 muestra A) el comportamiento de las partículas de PSO, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por PSO y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 2 obteniendo el control antes de los 5 segundos de manera estable.

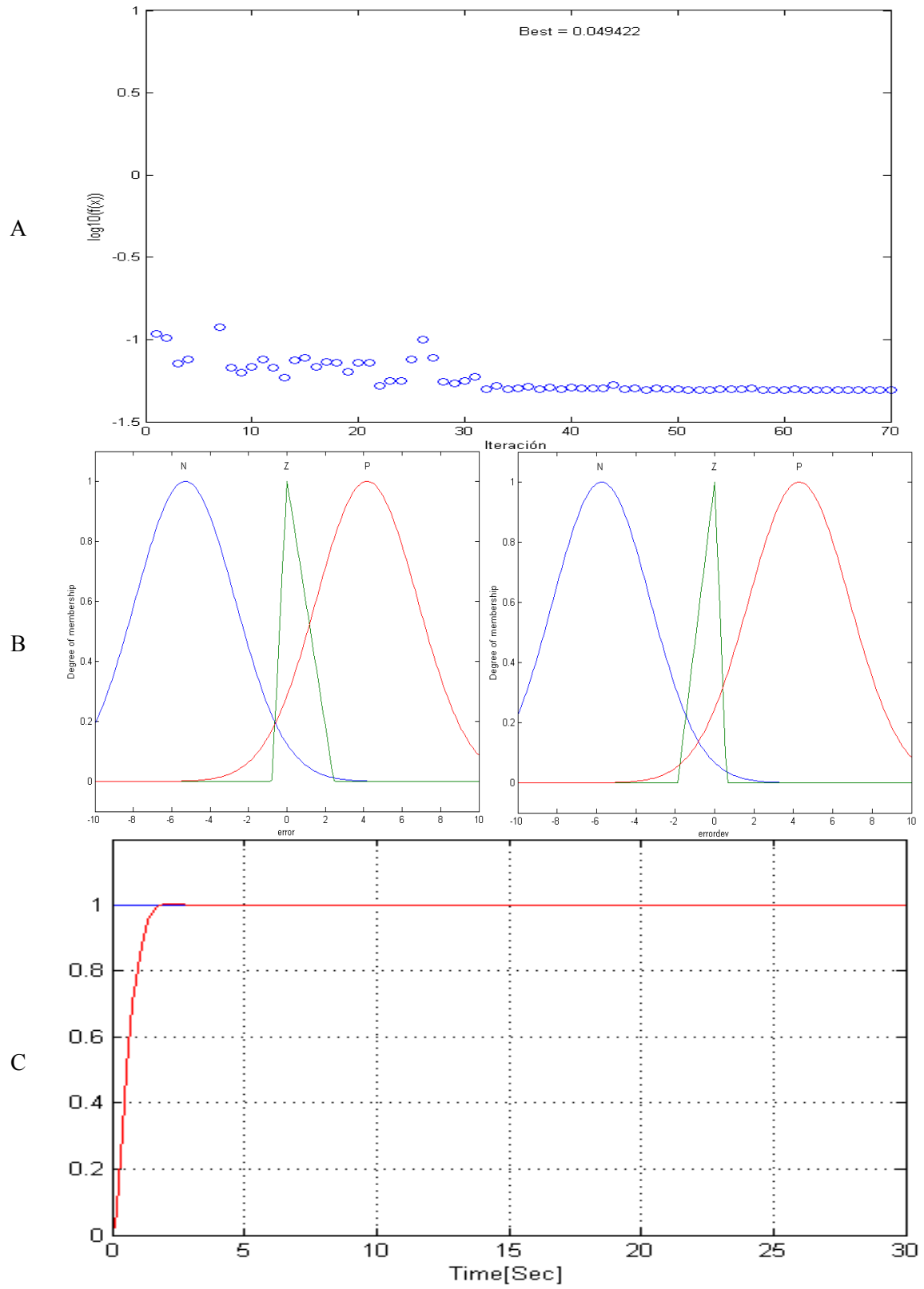


Figura 4.18: A) Comportamiento de las partículas de PSO B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 2.

4.2.8 Resultados del uso de PSO para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados de forma aleatoria

Se utilizaron algoritmos genéticos para encontrar el controlador difuso tipo-2 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 2.

La Tabla 4.35 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución del PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO.

En la Tabla 4.35, se muestra los resultados de las pruebas realizadas con PSO para la planta 2 con funciones de membresía *GTG*, mostrando los parámetros utilizados en el PSO, los mejores y peores errores promedio.

Tabla 4.35: Resultados del controlador difuso tipo-2 *GTG* para planta 2.

PSO GTG Planta2 Tipo-2							
No.	Partículas (Swarm)	Iteraciones	C1	C2	Inercia	Tiempo de PSO	Error Promedio
1	200	70	0.3179	0.5411	0.7012	8:51:28	0.083388
2	200	70	0.7690	0.2082	0.8755	8:09:51	0.090310
3	200	70	0.3889	0.6394	0.7304	8:50:16	0.096099
4	200	70	0.9512	0.8014	0.6141	9:15:29	0.101980
5	200	70	0.7870	0.6322	0.5724	10:00:28	0.102010
6	200	70	0.3787	0.8772	0.7288	8:19:46	0.103560
7	200	70	0.8781	0.4266	0.6770	7:53:11	0.106043
8	200	70	0.8553	0.8576	0.9592	0:53:38	0.107292
9	200	70	0.4413	0.1860	0.3604	9:20:45	0.143626
10	200	70	0.4043	0.1232	0.3977	9:18:30	0.182395

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para la planta 2 con funciones de membresía *GTG* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.36 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.36: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	200
	Iteraciones	70
	C1	0.3179
	C2	0.5411
	Inercia	0.7012
Tiempo del PSO		8:51:28
Error Alcanzado		0.083388

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fue de 0.083388.

La Figura 4.19 muestra A) el comportamiento de las partículas de PSO, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por PSO y C) el seguimiento de trayectoria del controlador difuso tipo-2 para la planta 2 obteniendo el control antes de los 5 segundos de manera estable.

4.2.9 Resultados del uso de PSO para encontrar las ganancias k_1 , k_2 y k_3 del controlador PID del péndulo de rueda de inercia

Se utilizaron algoritmos genéticos para encontrar las ganancias del controlador PID para el péndulo de rueda de inercia. Se utilizó el modelo del péndulo de rueda de inercia en Matlab (3.3), para encontrar las ganancias obteniendo un error promedio de simulación, así mismo también se probó los valores encontrados en el sistema físico del péndulo de rueda de inercia.

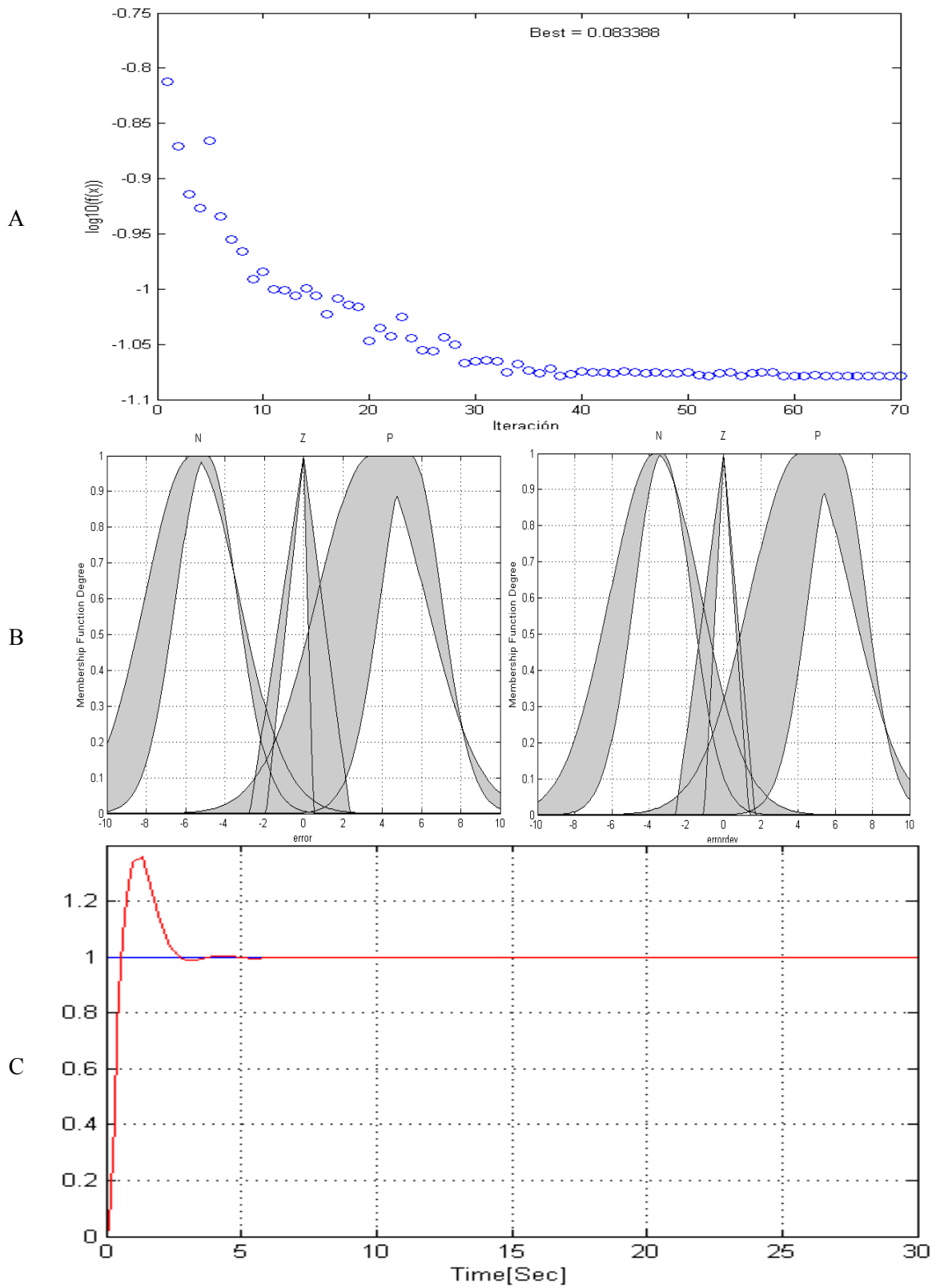


Figura 4.19: A) Comportamiento de las partículas de PSO, B) Funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 2.

La Tabla 4.37 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución de PSO, los valores obtenidos de las ganancias del controlador PID y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1) y social (C2) y la variable de inercia. La octava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por PSO. Todas las pruebas de simulación fueron satisfactorias, pero la prueba de los valores de las ganancias en el sistema físico marco cuales eran las mejores de todo el conjunto de pruebas.

Tabla 4.37: Resultados para el controlador PID del Péndulo de rueda de inercia.

Parámetros del PSO							Simulación			Experimento					
							Result.	Ganancias			Resultados Sin Perturbación			Resultados Con Perturbación	
N	Part	Max Iter	C1	C2	Inercia	Tiempo del PSO	Sim. Error Prom.	K ₁	K ₂	K ₃	Error Prom.	max tau	Tiempo de Estabilización (S)	Error Prom.	max tau
1	40	70	0.815	0.906	0.1706	12:17:32	0.00034	999.99	24.26	0.069	0.0148	∞	∞	∞	∞
2	40	70	0.056	0.693	0.0668	13:39:31	0.0003	798.47	12.88	0.004	0.0161	∞	∞	∞	∞
3	50	90	0.778	0.607	0.3444	19:41:35	0.00034	1000	29.14	0.095	0.0084	∞	∞	∞	∞
4	30	50	0.453	0.489	0.7602	5:20:54	0.00034	999.96	29.47	0.097	0.0147	∞	∞	∞	∞
5	60	80	0.162	0.698	0.479	22:56:21	0.00025	997.69	15.46	0.001	∞	∞	∞	∞	∞
6	30	50	0.815	0.906	0.1706	6:41:44	0.00034	999.92	27.46	0.086	∞	∞	∞	0.4364	46.3976
7	55	70	0.25	0.376	0.4429	13:33:58	0.00047	499.95	19.08	0.059	0.0072	6.892	4.4	0.4130	50.3305
8	40	50	0.685	0.515	0.7706	6:04:14	0.00036	498.4	10.56	0.001	∞	∞	∞	∞	∞
9	30	60	0.039	0.584	0.2618	5:57:40	0.0005	435.67	25.63	0.096	0.3339	5.443	18	0.2408	46.8569
10	20	80	0.041	0.933	0.1686	4:53:25	0.00053	377.34	24.72	0.094	0.0082	8.633	2.79	0.2101	35.9896
11	35	40	0.102	0.524	0.9704	3:57:43	0.00047	498.5	26.95	0.097	0.3999	13.46	6.5	0.2918	64.9792
12	45	70	0.27	0.817	0.4454	9:14:12	0.00047	499.99	19.8	0.062	0.3340	10.96	4.25	0.3061	29.9547
13	60	40	0.65	0.799	0.5154	8:25:19	0.00047	499.99	22.49	0.077	0.2982	12.84	2.69	0.2525	39.1294
14	25	70	0.777	0.812	0.2383	5:44:17	0.00047	499.95	17.11	0.048	1.6213	11.52	4	0.2200	21.1737
15	55	100	0.215	0.077	0.7116	20:53:28	0.00047	495.55	23.76	0.082	0.3260	22.73	5.4	0.2438	43.6512
16	40	50	0.768	0.392	0.9891	6:18:56	0.00037	494.76	12.14	0.001	∞	∞	∞	∞	∞
17	25	50	0.258	0.389	0.2483	4:30:36	0.0005	453.17	15.5	0.042	0.3494	7.400	5.5	0.1910	30.6645
18	80	100	0.805	0.951	0.7239	20:45:24	0.00047	499.99	26.94	0.1	0.2660	13.66	8	0.2586	889231.4
19	20	50	0.737	0.722	0.1434	4:32:07	0.00043	378.13	11.18	0.001	∞	∞	∞	∞	∞
20	35	90	0.383	0.107	0.5693	11:57:01	0.0005	440.96	14.56	0.038	0.3363	7.193	6	0.3026	22.8146
21	80	90	0.941	0.764	0.3209	2:49:13	0.00036	500	10.24	0.001	∞	∞	∞	∞	∞

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando algoritmos genéticos para las ganancias óptimas del controlador PID del Péndulo de rueda de inercia.

Experimento (Mejor Simulación)

La Tabla 4.38 muestra los parámetros utilizados para la simulación y el experimento físico del péndulo de rueda de inercia y los resultados obtenidos con las mejores ganancias.

Tabla 4.38: Parámetros utilizados en el PSO, tiempo de PSO y errores promedio del Péndulo.

Parámetros del PSO	Partículas	55
	Iteraciones	70
	C1	0.2499
	C2	0.3757
	Inercia	0.4429
	Tiempo del PSO	13:33:58
Ganancias	k₁	499.9494
	k₂	19.0849
	k₃	0.0587
Simulación	Error Promedio	0.00047
Experimento	Error Promedio	0.00723
	Max Tau	6.8924
	Tiempo de Estabilización (S)	4.4
	Error Promedio/Perturbado	0.41301
	Max Tau/Perturbado	50.33051

Las siguientes figuras, muestran los resultados obtenidos en la simulación y el experimento físico del péndulo de rueda de inercia con y sin perturbación.

La Figura 4.20 A) muestra el comportamiento de las partículas de PSO y el resultado de Simulación con un error promedio de 0.00047 rad, estabilizándose antes de 1 segundo de ejecución, B) la primera imagen muestra el comportamiento del péndulo utilizando las ganancias óptimas: $k_1 = 499.94936$, $k_2 = 19.08493$, y $k_3 = 0.05867$; obteniendo un error promedio = 0.0072278 [rad], max T = 6.8924 [Nm] y tss = 4.4 segundos en estabilizarse y la segunda imagen muestra el comportamiento del péndulo utilizando las ganancias óptimas aplicando perturbación, obteniendo un error promedio en estado estable = 0.41301 [rad] and max T = 50.33051 [Nm].

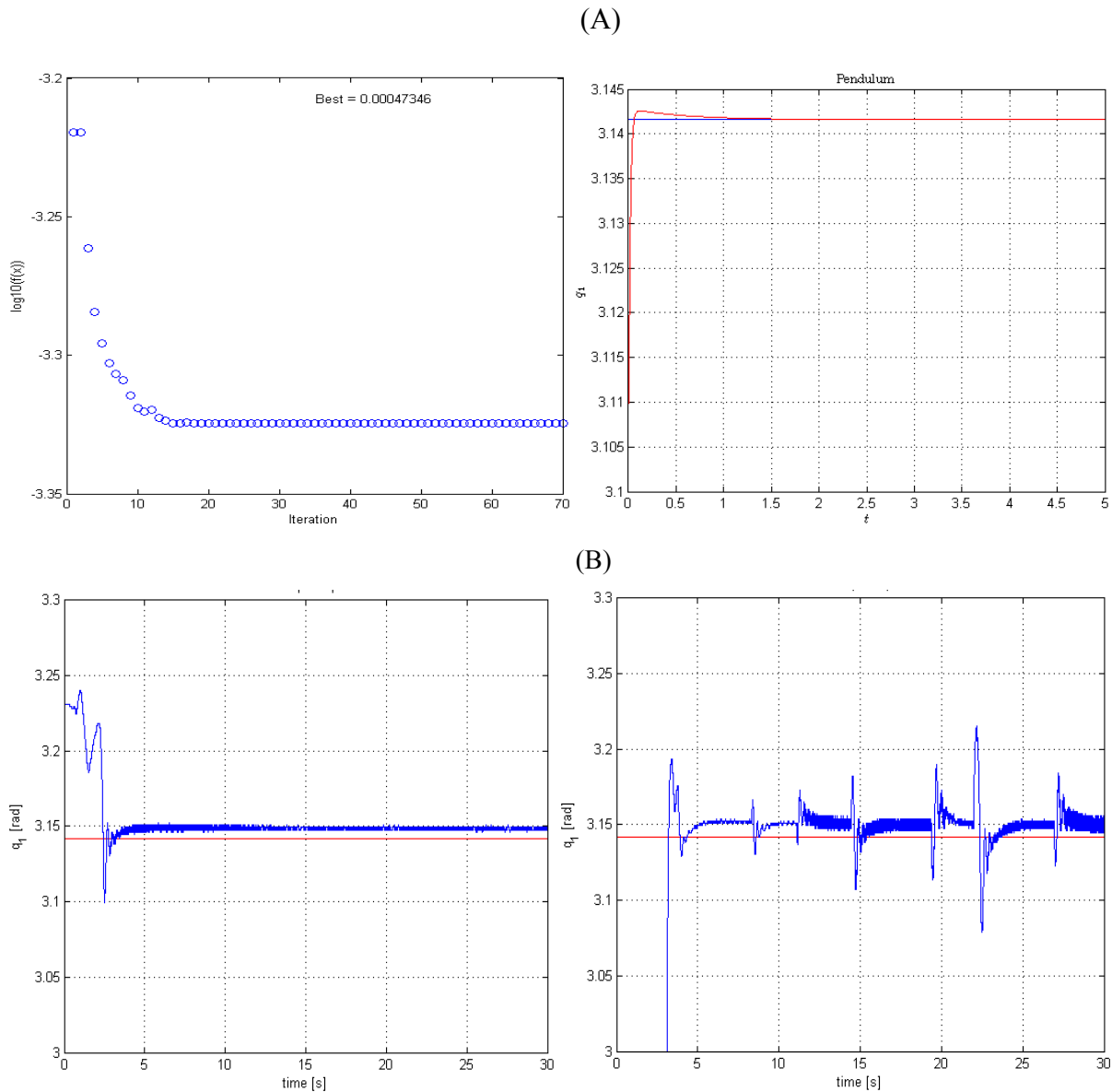


Figura 4.20: A) Comportamiento de las partículas de PSO y resultado de simulación, B) comportamiento del péndulo físico utilizando las ganancias óptimas y comportamiento del péndulo físico aplicando perturbación.

4.2.10 Resultados del uso de PSO para el controlador difuso tipo-1 para un robot autónomo móvil

Se utilizó PSO para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *TTT* (Trapezoidal-Triangular-Trapezoidal) utilizando (3.5) y (3.6) que representa al robot autónomo móvil.

La Tabla 4.39 contiene los valores utilizados en la configuración de PSO, el tiempo de ejecución de PSO y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1) y social (C2) y la variable de inercia. La décima columna de ésta tabla, indica el mejor error promedio alcanzado con el controlador obtenido por el PSO.

Tabla 4.39: Resultados del controlador difuso tipo-1 *TTT* para el robot autónomo móvil.

No.	Partic.	Itera.	C1	C2	Inercia	Tiempo de PSO	Error Promedio de Velocidad	Error Promedio de Posición
1	100	50	0.8612	0.2107	0.4291	0:23:59	2.624143156	0.209954806
2	100	150	0.7548	0.896	0.8498	0:19:22	1.980805424	0.16190013
3	80	250	0.5093	0.5838	0.795	0:59:30	1.983634755	0.161329487
4	110	220	0.3935	0.1791	0.7216	1:45:28	2.035741258	0.166472432
5	90	150	0.7799	0.8207	0.3702	1:04:45	2.050600565	0.164740399
6	150	80	0.8137	0.8331	0.8847	0:15:14	1.996473172	0.163146492
7	150	200	0.1062	0.3237	0.629	2:13:04	2.076681385	0.172415214
8	60	200	0.9517	0.8052	0.7738	0:35:57	1.983893659	0.162053652
9	80	100	0.8347	0.1461	0.2874	0:44:12	2.385216633	0.195082195
10	50	130	0.7374	0.1772	0.4664	0:40:25	2.675882645	0.213524262
11	50	130	0.9431	0.2375	0.2394	0:32:01	2.211079314	0.180839016
12	25	130	0.8166	0.9067	0.1706	0:17:09	2.222589091	0.180230817
13	200	70	0.5128	0.1831	0.438	1:01:53	2.41192882	0.202799842
14	200	80	0.2957	0.5469	0.6863	0:41:34	1.982409075	0.161788315
15	200	50	0.455	0.6108	0.9261	0:12:00	2.025932982	0.165476219
16	200	50	0.8149	0.9059	0.1314	0:48:14	2.27043938	0.180384739
17	150	180	0.3694	0.299	0.6094	2:04:15	1.984853292	0.162465607
18	90	150	0.4652	0.7426	0.3185	1:03:52	2.157676439	0.173411787
19	90	150	0.0261	0.136	0.1486	1:03:41	2.700780054	0.214611702
20	90	250	0.7311	0.051	0.0565	1:47:46	2.59371008	0.214386261
21	200	60	0.2353	0.745	0.0962	0:57:36	2.38594898	0.192155094
22	200	70	0.8153	0.8883	0.8321	0:20:07	1.986597724	0.162268408
23	200	70	0.936	0.0787	0.1922	1:13:33	2.61004479	0.222032724

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando PSO para el robot autónomo móvil con funciones de membresía *TTT* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.40 muestra los parámetros utilizados para las simulaciones y el error promedio del experimento.

Tabla 4.40: Parámetros utilizados en el PSO, tiempo de PSO y error promedio para el experimento.

Parámetros de PSO	Partículas	80
	Iteraciones	250
	C1	0.5093
	C2	0.5838
	Inercia	0.7950
Tiempo de PSO		0:59:30
Error Alcanzado		0.161329487

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fue de 0.161329487.

La Figura 4.21 A) muestra el comportamiento de las partículas de PSO y B) muestra las funciones de membresía la variable de entrada 1 (error de velocidad lineal) y la variable de entrada 2 (error de velocidad angular) del controlador difuso tipo-1 óptimo obtenido por PSO.

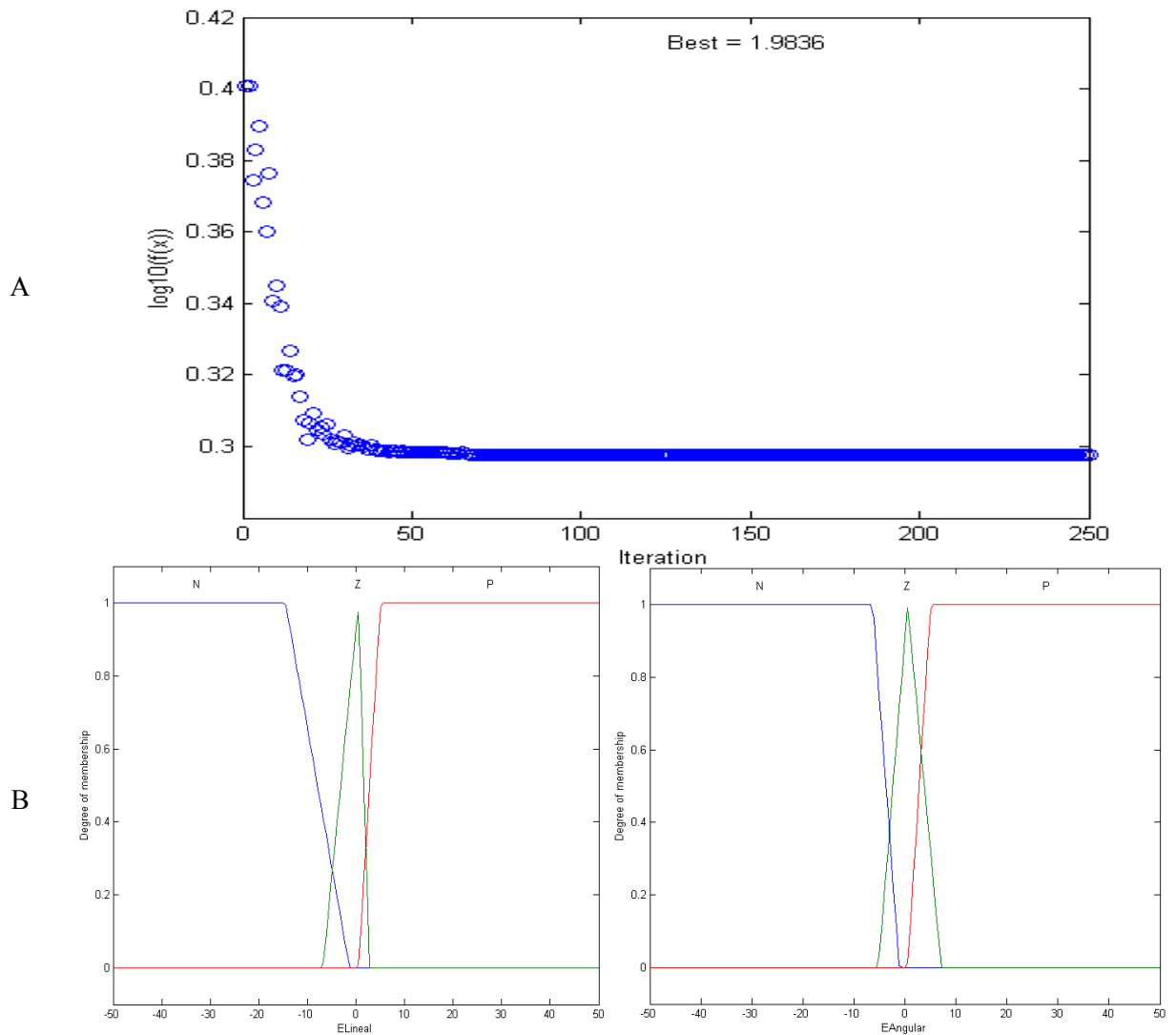


Figura 4.21: A) Comportamiento de las partículas de PSO y B) funciones de membresía de la entrada 1 y entrada 2 del controlador difuso tipo-1 óptimo.

La figura 4.22 A) muestra la respuesta los errores de velocidad lineal y la velocidad angular y los errores de torque derecho y torque izquierdo y B) muestra los errores de posición $e(x)$, $e(y)$ y $e(\theta)$ y se puede observar como tienden a cero de manera estable y C) el seguimiento de trayectoria del robot autónomo móvil con el mejor controlador difuso tipo-1 obtenido con PSO.

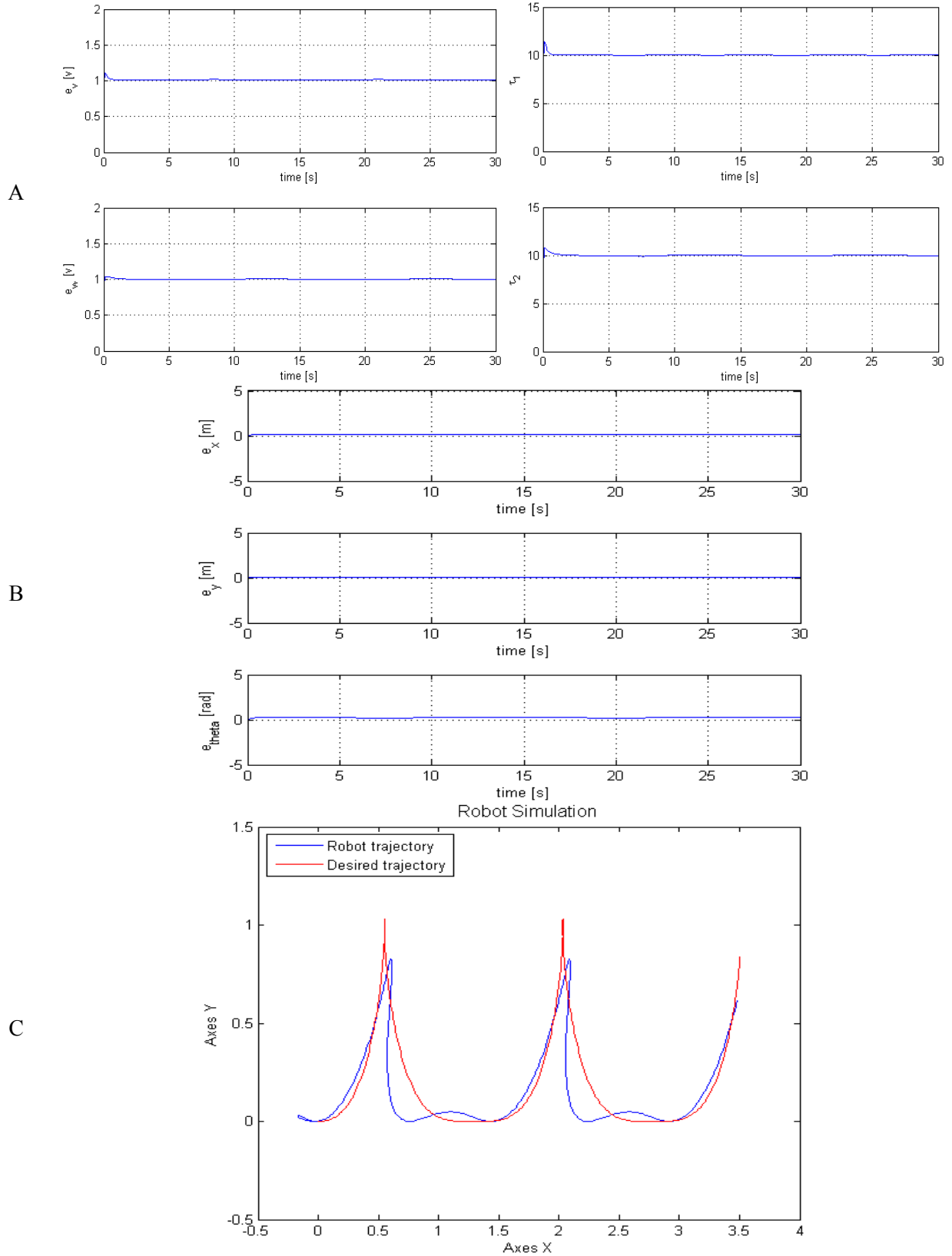


Figura 4.22: Resultados de simulación del robot autónomo móvil utilizando el controlador difuso tipo-1 optimizado.

4.3 Resultados del uso del Método de Optimización Híbrido PSO-GA

En esta sección se presentan los resultados obtenidos haciendo uso del método de optimización Híbrido PSO-GA para los casos de aplicación descritos a continuación.

4.3.1 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 1 con parámetros de PSO asignados manualmente

Se utilizó el método híbrido propuesto PSO-GA para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 1.

La Tabla 4.41 contiene los valores utilizados en la configuración del método Híbrido PSO-GA, el tiempo de ejecución y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (*C1*), social (*C2*), la variable de inercia, variable de cruce y variable de mutación. La onceava columna de esta tabla, indica el error promedio alcanzado con el controlador obtenido por el método híbrido.

Experimento (Mejor Simulación)

La Tabla 4.42 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.42: Parámetros utilizados en el método Híbrido PSO-GA, tiempo y error promedio para el experimento.

Parámetros del Híbrido PSO-GA	Partículas/Población	70
	Iteraciones/Generaciones	200
	C1	0.2706
	C2	0.0403
	Inercia	0.3254
	% Reemplazo	0.7
	Cruce	0.7
	Mutación	0.2
Tiempo del Híbrido PSO-GA		1:46:07
Error Alcanzado		0.10956

Tabla 4.41: Resultados del controlador difuso tipo-1 *GTG* para planta 1.

No.	Partículas/ Población	Iteraciones / Generaciones	C1	C2	Inercia	% Remp	Cruce	Mut	Tiempo Método	Error Promedio
1	70	200	0.2706	0.0403	0.3254	0.7	0.7	0.2	1:46:07	0.10956
2	200	70	0.1780	0.4863	0.5381	0.7	0.5	0.2	1:20:14	0.11375
3	200	70	0.8963	0.3534	0.6738	0.7	0.7	0.1	1:18:21	0.11517
4	200	70	0.4743	0.5077	0.5163	0.7	0.6	0.2	1:28:50	0.11559
5	200	70	0.4587	0.6339	0.1024	0.7	0.5	0.5	1:42:23	0.11584
6	200	70	0.1780	0.4863	0.5381	0.7	0.6	0.1	1:23:22	0.11595
7	200	70	0.5159	0.0732	0.1455	0.7	0.5	0.2	1:36:49	0.11616
8	200	70	0.5279	0.5336	0.6763	0.7	0.7	0.4	1:25:52	0.11785
9	200	70	0.6608	0.1696	0.8927	0.7	0.8	0.3	1:29:15	0.11833
10	150	80	0.7293	0.1387	0.8532	0.7	0.7	0.1	1:15:23	0.11971
11	90	75	0.2521	0.3490	0.9708	0.7	0.5	0.2	0:25:14	0.12240
12	200	70	0.8149	0.9059	0.1314	0.7	0.5	0.1	1:33:46	0.12325
13	80	100	0.5159	0.0732	0.1455	0.7	0.6	0.1	0:48:21	0.12488
14	90	35	0.2110	0.9645	0.1126	0.7	0.5	0.2	0:17:52	0.12853
15	70	45	0.7269	0.6954	0.8965	0.7	0.5	0.1	0:13:37	0.13191
16	45	30	0.0937	0.9671	0.8263	0.7	0.6	0.2	0:05:52	0.14190
17	50	45	0.7571	0.8877	0.1899	0.7	0.7	0.1	0:13:24	0.15041
18	150	80	0.0050	0.0050	1	0.7	0.7	0.1	0:14:44	0.15275
19	200	70	1	1	1	0.7	0.5	0.1	0:48:30	0.15546
20	40	25	0.4570	0.2620	0.5709	0.7	0.7	0.4	0:06:44	0.15637
21	200	70	0.15	0.15	1	0.7	0.7	0.1	1:00:12	0.15722
22	200	70	0.8149	0.9059	0.1314	0.7	0.6	0.1	1:16:30	0.15762
23	90	75	0.25	0.25	1	0.7	0.5	0.2	0:24:37	0.15805
24	20	15	0.5981	0.1583	0.4188	0.7	0.8	0.3	0:02:13	0.15867
25	200	70	0.25	0.25	1	0.7	0.5	0.2	0:52:06	0.15898
26	200	70	0.6608	0.1696	0.8927	0.7	0.5	0.2	1:04:59	0.15931
27	200	70	1	0.1	1	0.7	0.5	0.1	0:59:33	0.15957
28	200	70	70	0.1	1	0.7	0.8	0.3	0:45:04	0.15980
29	120	70	0.5	0.5	1	0.7	0.5	0.1	0:29:09	0.16013
30	200	70	0.05	0.05	1	0.7	0.7	0.4	0:53:54	0.16114
31	200	70	0.005	0.005	1	0.7	0.6	0.2	0:58:54	0.16196
32	90	75	1	1	1	0.7	0.5	0.2	0:30:26	0.16295
33	200	70	0.5	0.5	1	0.7	0.5	0.2	1:01:03	0.16388
34	70	45	0.15	0.15	1	0.7	0.5	0.1	0:06:13	0.16558
35	90	35	0.15	0.15	1	0.7	0.5	0.2	0:14:36	0.16942
36	50	45	0.05	0.05	1	0.7	0.7	0.1	0:48:23	0.17233
37	40	25	0.005	0.005	1	0.7	0.7	0.4	0:08:46	0.17264
38	45	30	0.05	0.05	1	0.7	0.6	0.2	0:06:01	0.17558
39	20	15	0.25	0.25	1	0.7	0.8	0.3	0:01:17	0.19144

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando el método Híbrido PSO-GA para la planta 1 con funciones de membresía *GTG* tipo-1.

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.10956.

La Figura 4.23 muestra A) el comportamiento del método Híbrido PSO-GA, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el método Híbrido PSO-GA y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 1 obteniendo el control antes de los 10 segundos de manera estable.

4.3.2 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-2 con funciones de membresía *GTG* para la Planta 1 con parámetros de PSO asignados manualmente

Se utilizó el método híbrido propuesto PSO-GA para encontrar el controlador difuso tipo-2 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 1.

La Tabla 4.43 contiene los valores utilizados en la configuración del método Híbrido PSO-GA, el tiempo de ejecución y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2), la variable de inercia, variable de cruce y variable de mutación. La onceava columna de ésta tabla, indica el error promedio alcanzado con el controlador obtenido por el método híbrido.

Tabla 4.43: Resultados del controlador difuso tipo-2 *GTG* para planta 1.

No.	Partículas/ Población	Iteraciones / Generaciones	C1	C2	Inercia	% Remp	Cruce	Mut	Tiempo Método	Error Promedio
1	25	50	0.8149	0.9059	0.1706	0.7	0.8	0.3	11:46:36	0.132233
2	25	50	0.0873	0.4457	0.6676	0.7	0.8	0.3	16:08:47	0.140007
3	45	30	0.9118	0.6906	0.7163	0.7	0.6	0.1	19:08:49	0.147934
4	25	50	0.4928	0.1730	0.7160	0.7	0.8	0.3	20:36:40	0.148900
5	20	15	0.9458	0.3435	0.6244	0.7	0.8	0.3	7:18:33	0.151596
6	40	25	0.8149	0.9059	0.1706	0.7	0.7	0.4	1:19:48	0.157026
7	20	15	0.0910	0.2991	0.8237	0.9	0.8	0.3	2:46:26	0.159745
8	40	25	0.8149	0.9059	0.1706	0.7	0.7	0.4	14:28:04	0.163500
9	15	60	0.8149	0.9059	0.1706	0.7	0.7	0.1	14:05:55	0.178253
10	25	50	0.8149	0.9059	0.1706	0.7	0.8	0.3	11:31:26	0.179500

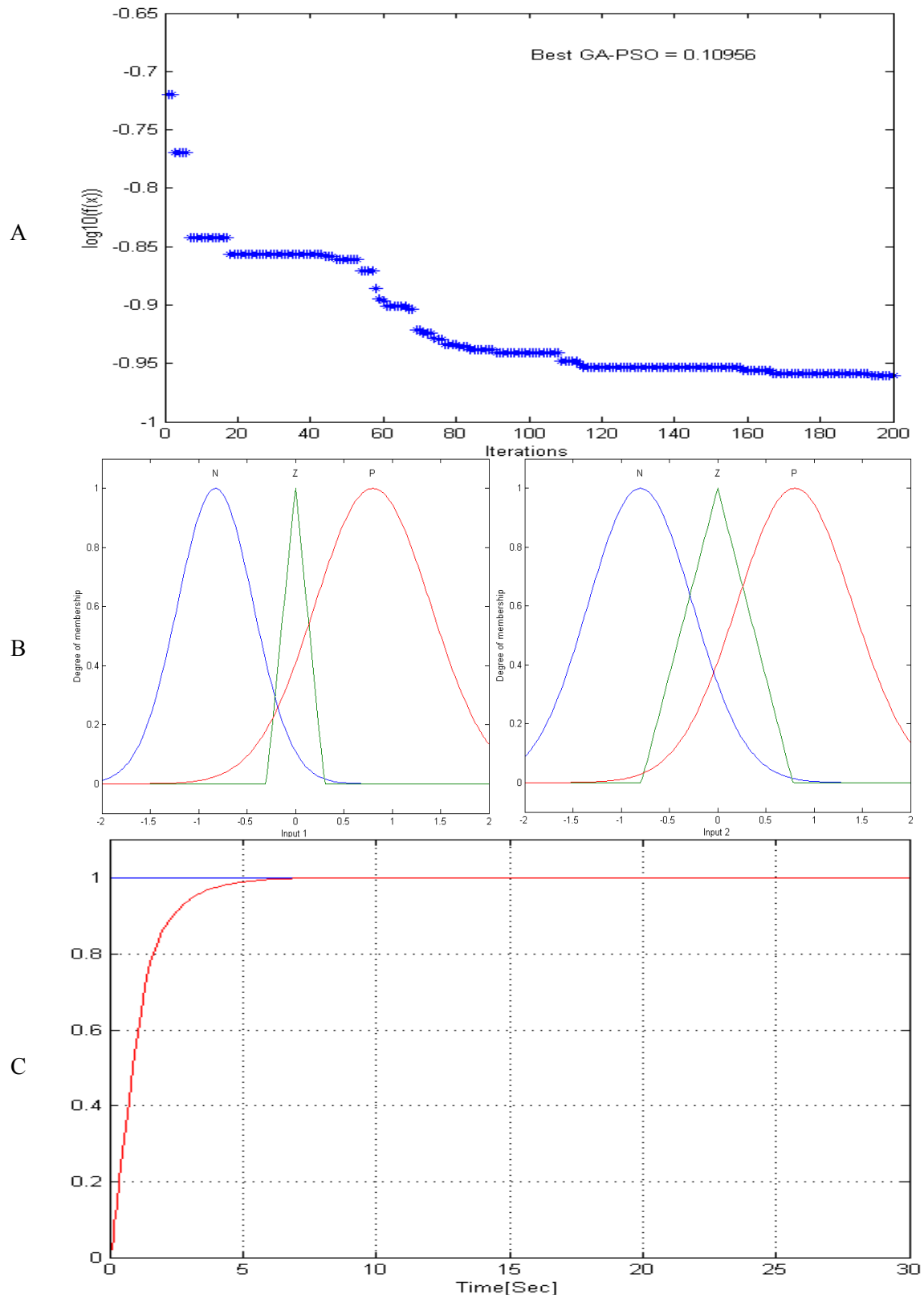


Figura 4.23: A) Comportamiento del método Híbrido PSO-GA B) funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 1.

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando el método Híbrido PSO-GA para la planta 1 con funciones de membresía *GTG* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.44 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.44: Parámetros utilizados en el método Híbrido PSO-GA, tiempo y error promedio para el experimento.

Parámetros del Híbrido PSO-GA	Partículas/Población	25
	Iteraciones/Generaciones	50
	C1	0.8149
	C2	0.9059
	Inercia	0.1706
	%Reemplazo	0.7
	Cruce	0.8
	Mutación	0.3
Tiempo del Híbrido PSO-GA		11:46:36
Error Alcanzado		0.132233

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.132233.

La Figura 4.24 muestra A) el comportamiento del método Híbrido PSO-GA, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el método Híbrido PSO-GA y C) el seguimiento de trayectoria del controlador difuso tipo-2 para la planta 1 llegando a estabilizarse antes de los 10 segundo pero sin haber llegado a la referencia.

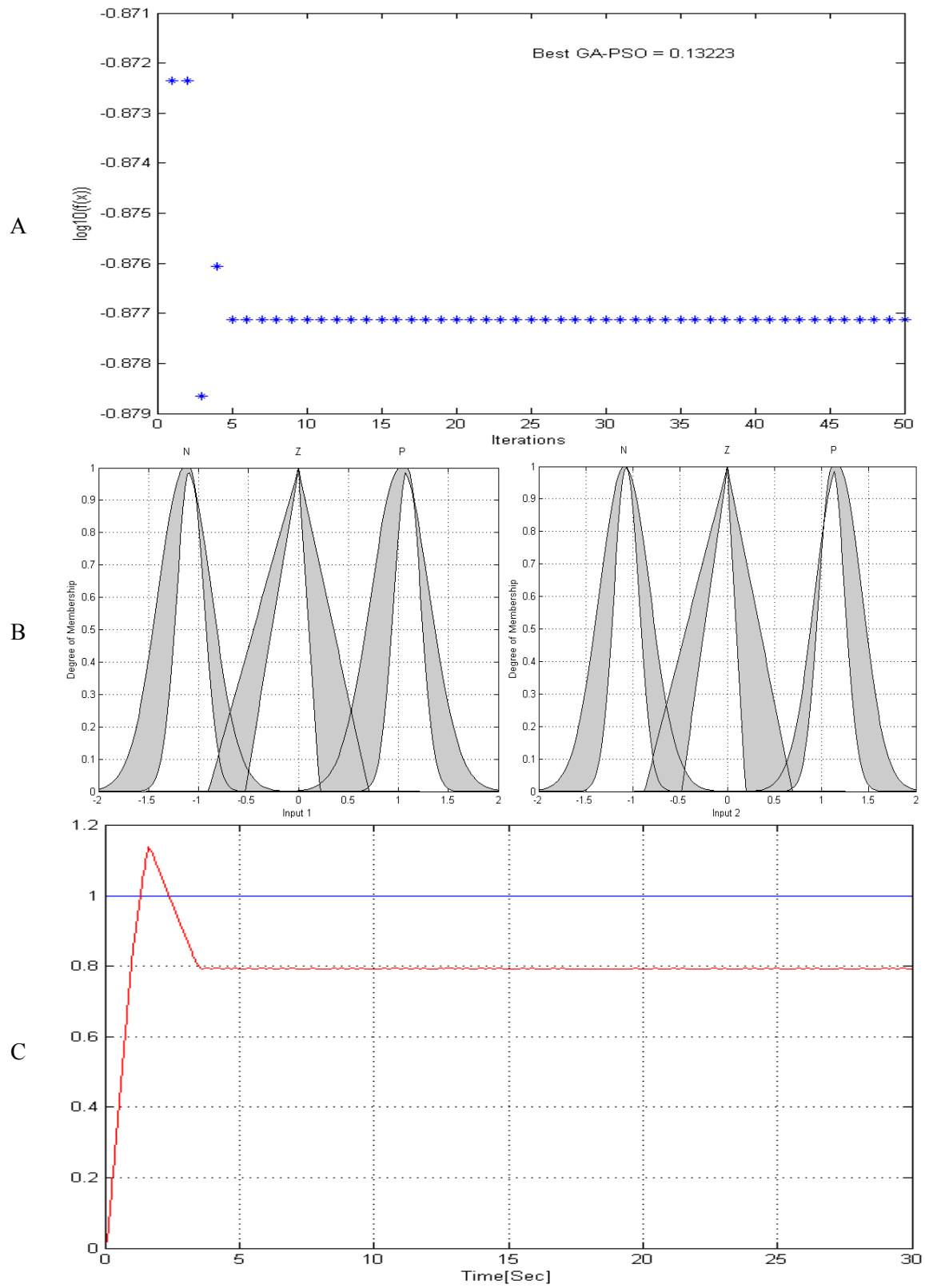


Figura 4.24: A) Comportamiento del método Híbrido PSO-GA B) funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 1.

4.3.3 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-1 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados manualmente

Se utilizó el método híbrido propuesto PSO-GA para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 2.

La Tabla 4.45 contiene los valores utilizados en la configuración del método Híbrido PSO-GA, el tiempo de ejecución y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2), la variable de inercia, variable de cruce y variable de mutación. La onceava columna de esta tabla, indica el error promedio alcanzado con el controlador obtenido por el método híbrido.

Tabla 4.45: Resultados del controlador difuso tipo-1 *GTG* para planta 2.

No.	Partículas/ Población	Iteraciones / Generaciones	C1	C2	Inercia	% Remp	Cruce	Mut	Tiempo Método	Error Promedio
1	200	70	0.05	0.05	1	0.7	0.5	0.2	1:10:15	0.045568
2	200	70	1	1	1	0.7	0.5	0.1	0:50:02	0.045656
3	150	80	0.5171	0.5210	0.8254	0.7	0.7	0.1	1:11:57	0.046050
4	80	120	0.7372	0.8160	0.7541	0.7	0.5	0.2	1:00:36	0.046147
5	100	150	0.6608	0.1696	0.8976	0.7	0.5	0.2	1:41:48	0.046674
6	200	70	0.5	0.5	1	0.7	0.5	0.1	0:52:53	0.046756
7	120	70	0.8149	0.9059	0.1706	0.7	0.5	0.1	0:53:02	0.050639
8	200	70	0.15	0.15	1	0.7	0.5	0.2	1:07:10	0.054020
9	200	70	0.005	0.005	1	0.7	0.6	0.2	1:05:40	0.054236
10	200	70	0.25	0.25	1	0.7	0.7	0.1	1:53:35	0.054476
11	200	70	0.8149	0.9059	0.1706	0.7	0.8	0.3	1:32:57	0.055365
12	50	45	0.8192	0.0754	0.8559	0.7	0.7	0.1	0:13:43	0.055597
13	50	150	0.1934	0.4693	0.7081	0.7	0.6	0.2	0:55:20	0.057434
14	30	80	0.3163	0.2037	0.5234	0.7	0.8	0.3	0:16:05	0.057775
15	70	45	0.9156	0.1470	0.6697	0.7	0.5	0.1	0:21:06	0.058017
16	45	30	0.9910	0.0706	0.0596	0.7	0.6	0.2	0:08:38	0.062210
17	90	35	0.8149	0.9059	0.1706	0.7	0.5	0.2	0:19:53	0.062464
18	40	25	0.7249	0.3464	0.7198	0.7	0.7	0.4	0:05:52	0.064338
19	90	75	0.4201	0.8464	0.3230	0.7	0.5	0.2	0:42:03	0.551320
20	20	15	0.9988	0.3483	0.2185	0.7	0.8	0.3	0:01:57	0.632490

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando el método Híbrido PSO-GA para la planta 2 con funciones de membresía *GTG* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.46 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.46: Parámetros utilizados en el método Híbrido PSO-GA, tiempo y error promedio para el experimento.

Parámetros del Híbrido PSO-GA	Partículas/Población	200
	Iteraciones/Generaciones	70
	C1	1
	C2	1
	Inercia	1
	%Reemplazo	0.7
	Cruce	0.5
	Mutación	0.1
Tiempo del Híbrido PSO-GA		0:50:02
Error Alcanzado		0.045656

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.045656.

La Figura 4.25 muestra A) el comportamiento del método Híbrido PSO-GA, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el método Híbrido PSO-GA y C) el seguimiento de trayectoria del controlador difuso tipo-1 para la planta 2 controlando antes de los 5 segundos de manera estable.

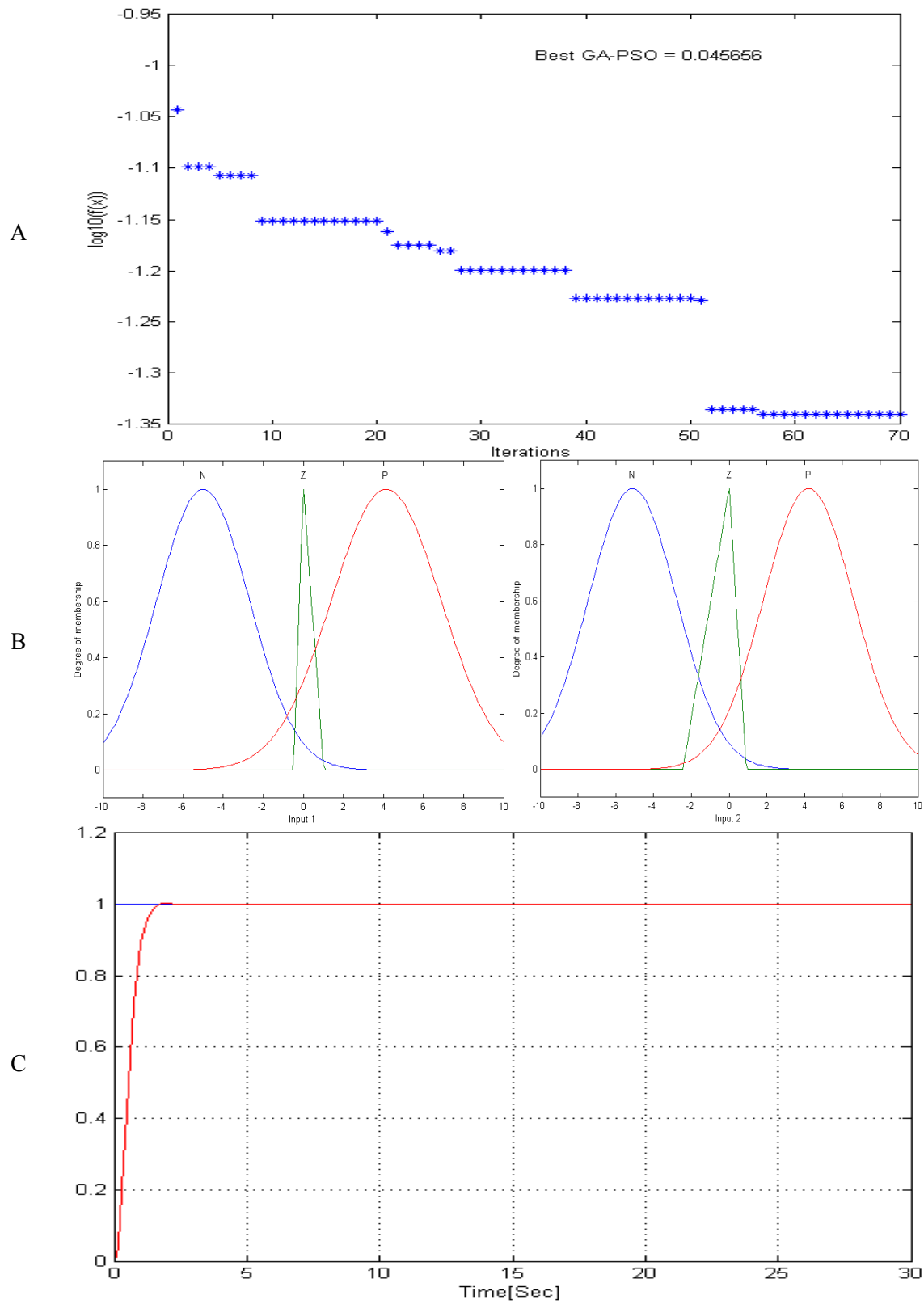


Figura 4.25: A) Comportamiento del método Híbrido PSO-GA B) funciones de membresía óptimas y C) respuesta del controlador difuso tipo-1 para la planta 2.

4.3.4 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-2 con funciones de membresía GTG para la Planta 2 con parámetros de PSO asignados manualmente

Se utilizó el método híbrido propuesto PSO-GA para encontrar el controlador difuso tipo-2 óptimo con funciones de membresía de tipo *GTG* (Gaussiana-Triangular-Gaussiana) utilizando (3.1) que representa la planta 2.

La Tabla 4.47 contiene los valores utilizados en la configuración del método Híbrido PSO-GA, el tiempo de ejecución y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2), la variable de inercia, variable de cruce y variable de mutación. La onceava columna de esta tabla, indica el error promedio alcanzado con el controlador obtenido por el método híbrido.

Tabla 4.47: Resultados del controlador difuso tipo-2 *GTG* para planta 2.

No.	Partículas/ Población	Iteraciones / Generaciones	C1	C2	Inercia	% Remp	Cruce	Mut	Tiempo Método	Error Promedio
1	200	70	0.9197	0.1524	0.9224	0.9	0.7	0.1	23:26:29	0.052994
2	200	70	0.6769	0.0633	0.5805	0.7	0.8	0.3	23:18:31	0.053131
3	90	75	0.6423	0.5047	0.2825	0.9	0.5	0.1	11:46:50	0.054486
4	100	150	0.8585	0.5916	0.6241	0.7	0.7	0.2	23:52:43	0.055428
5	45	30	0.7364	0.2531	0.2844	0.7	0.6	0.2	1:52:47	0.055463
6	50	45	0.7788	0.2838	0.3317	0.7	0.7	0.1	3:36:40	0.055628
7	50	150	0.0660	0.1742	0.7760	0.7	0.6	0.2	13:24:31	0.056913
8	120	70	0.4290	0.7808	0.4866	0.7	0.5	0.1	14:38:23	0.057324
9	150	80	0.2078	0.5113	0.6797	0.7	0.7	0.1	23:47:11	0.057343
10	80	120	0.6423	0.6964	0.7362	0.7	0.5	0.2	16:18:49	0.057476
11	70	45	0.1723	0.9135	0.9778	0.7	0.5	0.1	5:59:03	0.059703
12	90	35	0.4734	0.0731	0.0842	0.7	0.5	0.2	5:30:13	0.059950
13	30	80	0.1119	0.3513	0.4500	0.7	0.8	0.3	3:53:44	0.064661
14	40	25	0.0510	0.3279	0.3148	0.7	0.7	0.4	2:10:26	0.067949

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando el método Híbrido PSO-GA para la planta 2 con funciones de membresía *GTG* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.48 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.48: Parámetros utilizados en el método Híbrido PSO-GA, tiempo y error promedio para el experimento.

Parámetros del Híbrido PSO-GA	Partículas/Población	90
	Iteraciones/Generaciones	75
	C1	0.6423
	C2	0.5047
	Inercia	0.2825
	%Reemplazo	0.9
	Cruce	0.5
	Mutación	0.1
Tiempo del Híbrido PSO-GA		11:46:50
Error Alcanzado		0.054486

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.054486.

La Figura 4.26 muestra A) el comportamiento del método Híbrido PSO-GA, B) las funciones de membresía tipo *GTG* de la variable de entrada 1 (“*e*” error) y la variable de entrada 2 (“*ev*” derivada del error) obtenidas por el método Híbrido PSO-GA y C) el seguimiento de trayectoria del controlador difuso tipo-2 para la planta 2 controlando antes de los 10 segundos con alto sobre pico antes de alcanzar la referencia.

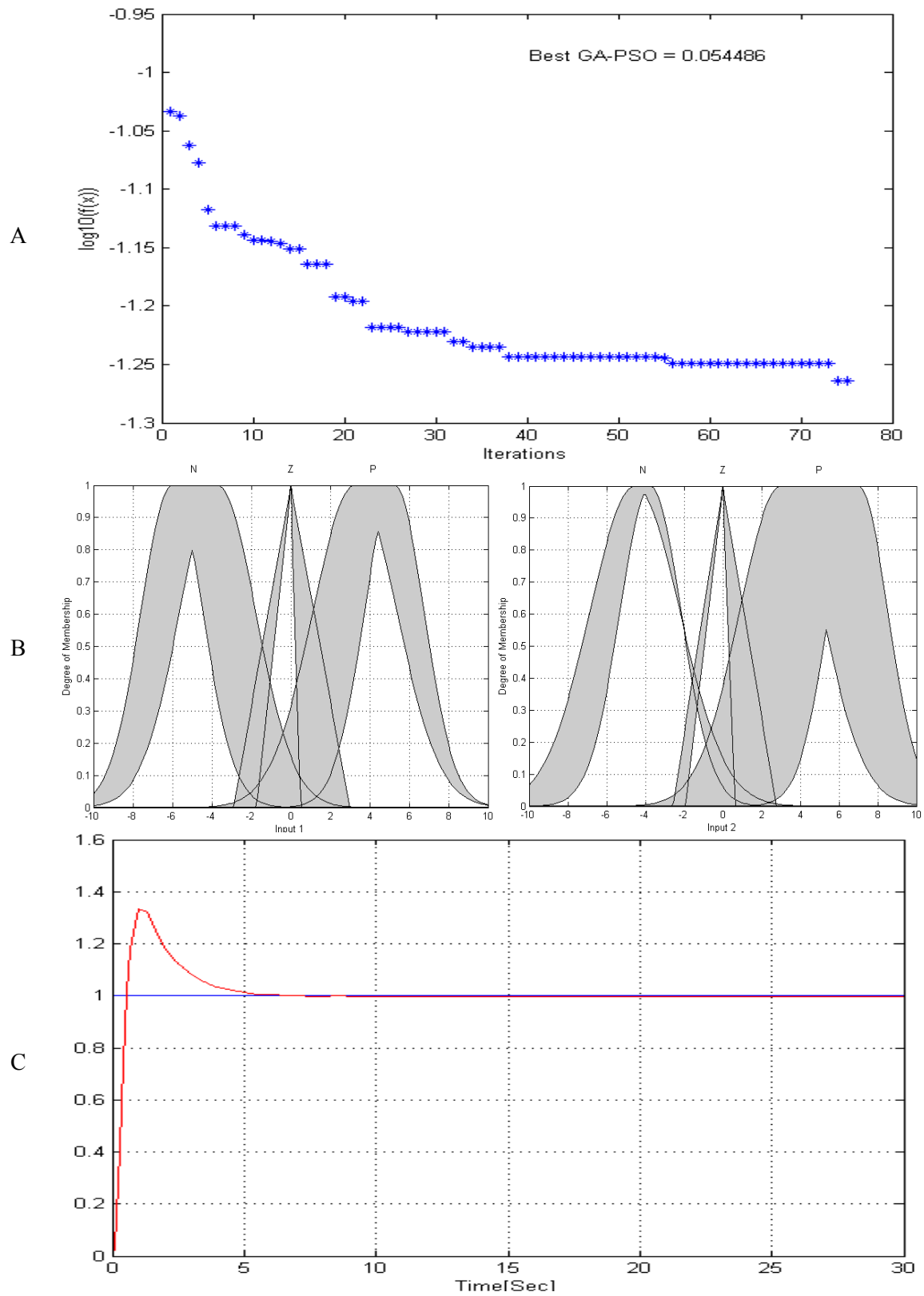


Figura 4.26: A) Comportamiento del método Híbrido PSO-GA B) funciones de membresía óptimas y C) respuesta del controlador difuso tipo-2 para la planta 2.

4.3.5 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-1 para un robot autónomo móvil

Se utilizó el método híbrido propuesto PSO-GA para encontrar el controlador difuso tipo-1 óptimo con funciones de membresía de tipo *TTT* (Trapezoidal-Triangular-Trapezoidal) utilizando (3.5) y (3.6) que representa al robot autónomo móvil.

La Tabla 4.49 contiene los valores utilizados en la configuración del método Híbrido PSO-GA, el tiempo de ejecución y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2), la variable de inercia, variable de cruce y variable de mutación. La onceava columna de esta tabla, indica el error promedio alcanzado con el controlador obtenido por el método híbrido.

Tabla 4.49: Resultados del controlador difuso tipo-1 *TTT* para robot autónomo móvil.

No	Part/ Pobl.	Iter / Gener	C1	C2	Inercia	% Rem	Cruce	Mut	Tiempo del PSO-GA	Error Promedio de Velocidad	Error Promedio de Posición
1	70	60	0.17713	0.69816	0.11483	0.9	0.8	0.3	1:09:06	1.672285	0.13601414
2	25	100	0.53013	0.52094	0.37772	0.7	0.7	0.1	0:35:11	1.672254	0.13608301
3	30	60	0.25575	0.88815	0.94136	0.7	0.8	0.1	0:40:50	1.672321	0.13608640
4	200	70	0.48252	0.99690	0.31332	0.7	0.8	0.1	3:15:53	1.672257	0.13609641
5	80	80	0.31352	0.95637	0.71256	0.7	0.9	0.3	1:28:09	1.672263	0.13610438
6	60	80	0.82947	0.97647	0.43004	0.7	0.8	0.2	1:06:33	1.672254	0.13616807
7	70	200	0.94346	0.53077	0.86140	0.9	0.8	0.1	6:15:59	1.672254	0.13618371
8	90	60	0.53408	0.12473	0.78395	0.7	0.9	0.4	1:55:47	1.672357	0.13618799
9	200	100	0.10588	0.54691	0.38577	0.9	0.8	0.1	20:02:35	1.672241	0.13619885
10	45	200	0.60367	0.74069	0.65106	0.7	0.7	0.1	5:34:11	1.672261	0.13644928
11	65	40	0.89286	0.63552	0.66415	0.7	0.8	0.2	0:38:15	1.674017	0.13647978
12	150	300	0.81370	0.55015	0.11792	0.9	0.8	0.1	11:48:40	1.672250	0.13650342
13	150	50	0.29833	0.60411	0.31905	0.7	0.6	0.3	1:57:02	1.672252	0.13657426
14	50	30	0.75411	0.80523	0.16201	0.7	0.6	0.1	0:22:52	1.674017	0.13664966
15	70	200	0.28406	0.75732	0.29247	0.7	0.6	0.1	5:26:42	1.672257	0.13665137
16	20	50	0.01124	0.21224	0.77619	0.9	0.6	0.2	0:14:09	1.672764	0.13665202
17	50	60	0.97575	0.14565	0.99751	0.9	0.6	0.1	0:39:58	1.673727	0.13666762
18	70	50	0.08383	0.88655	0.85119	0.7	0.8	0.3	0:46:14	1.672325	0.13670122
19	35	40	0.23548	0.15421	0.43283	0.7	0.7	0.3	0:22:18	1.673844	0.13675621
20	30	40	0.04685	0.67511	0.86956	0.9	0.7	0.2	0:15:10	1.672324	0.13723912
21	80	20	0.83700	0.54022	0.50676	0.9	0.8	0.4	0:26:18	1.673160	0.13731085
22	80	25	0.49064	0.44229	0.87195	0.7	0.7	0.1	0:34:20	1.674483	0.14290034
23	80	20	0.16038	0.43585	0.42489	0.9	0.8	0.2	0:25:30	1.686742	0.14538182
24	125	40	0.88974	0.50529	0.59005	0.9	0.8	0.2	1:15:35	1.672257	0.14785814

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando el método Híbrido PSO-GA para el robot autónomo móvil funciones de membresía *TTT* tipo-1.

Experimento (Mejor Simulación)

La Tabla 4.50 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.50: Parámetros utilizados en el método Híbrido PSO-GA, tiempo y error promedio para el experimento.

Parámetros del Híbrido PSO-GA	Partículas/Población	70
	Iteraciones/Generaciones	60
	C1	0.17713
	C2	0.69816
	Inercia	0.11483
	%Reemplazo	0.9
	Cruce	0.8
	Mutación	0.3
Tiempo del Híbrido PSO-GA		1:09:06
Error Alcanzado		0.13601414

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.13601414.

La Figura 4.27 A) muestra el comportamiento del método Híbrido PSO-GA y B) muestra las funciones de membresía la variable de entrada 1 (error de velocidad lineal) y la variable de entrada 2 (error de velocidad angular) del controlador difuso tipo-1 óptimo obtenido por el método Híbrido PSO-GA.

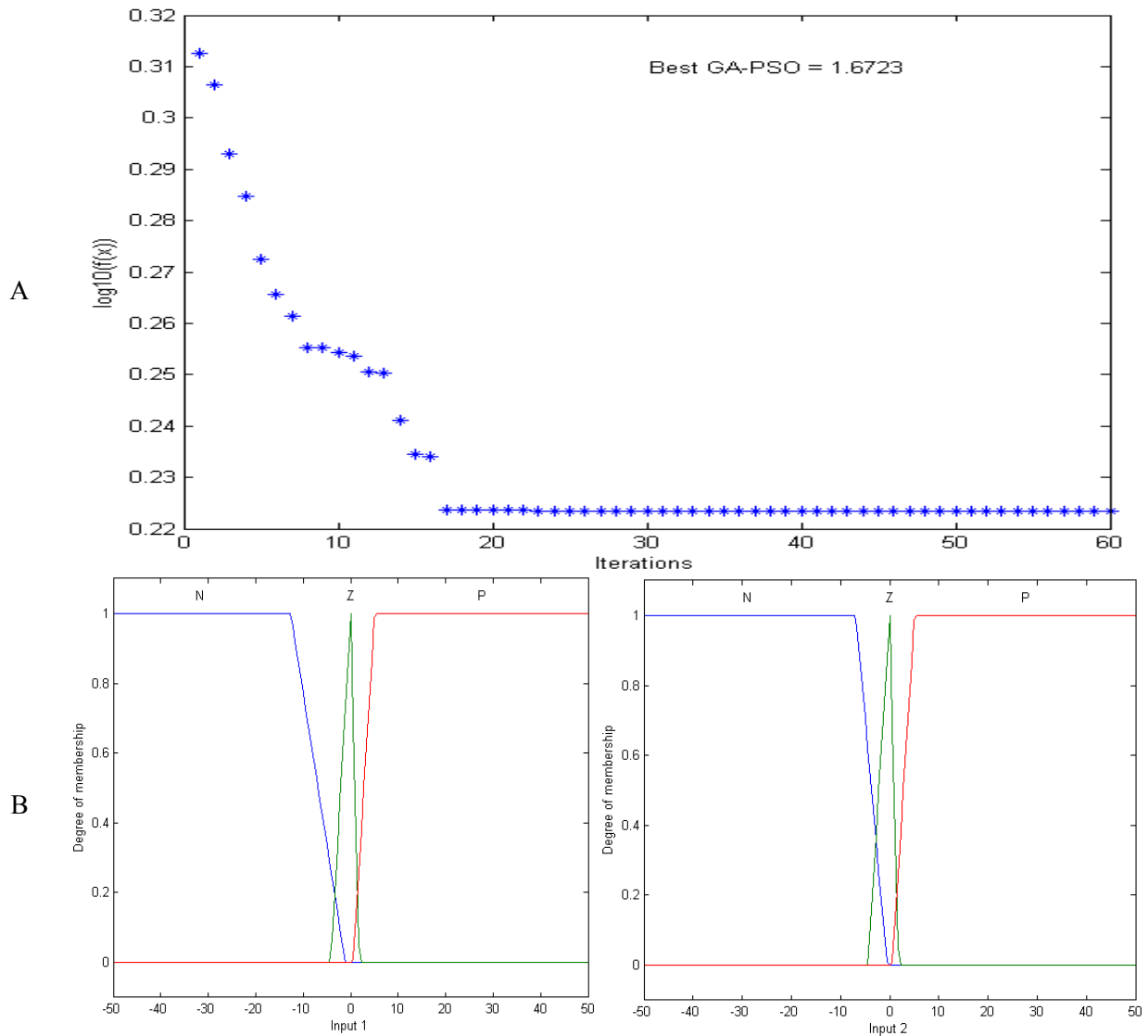


Figura 4.27: A) Comportamiento del método Híbrido PSO-GA y B) funciones de membresía de la entrada 1 y entrada 2 del controlador difuso tipo-1 óptimo.

La figura 4.28 A) muestra la respuesta los errores de velocidad lineal y la velocidad angular y los errores de torque derecho y torque izquierdo y B) muestra los errores de posición $e(x)$, $e(y)$ y $e(\theta)$ y se puede observar como tienden a cero de manera estable y C) el seguimiento de trayectoria del robot autónomo móvil con el mejor controlador difuso tipo-1 obtenido con método Híbrido PSO-GA.

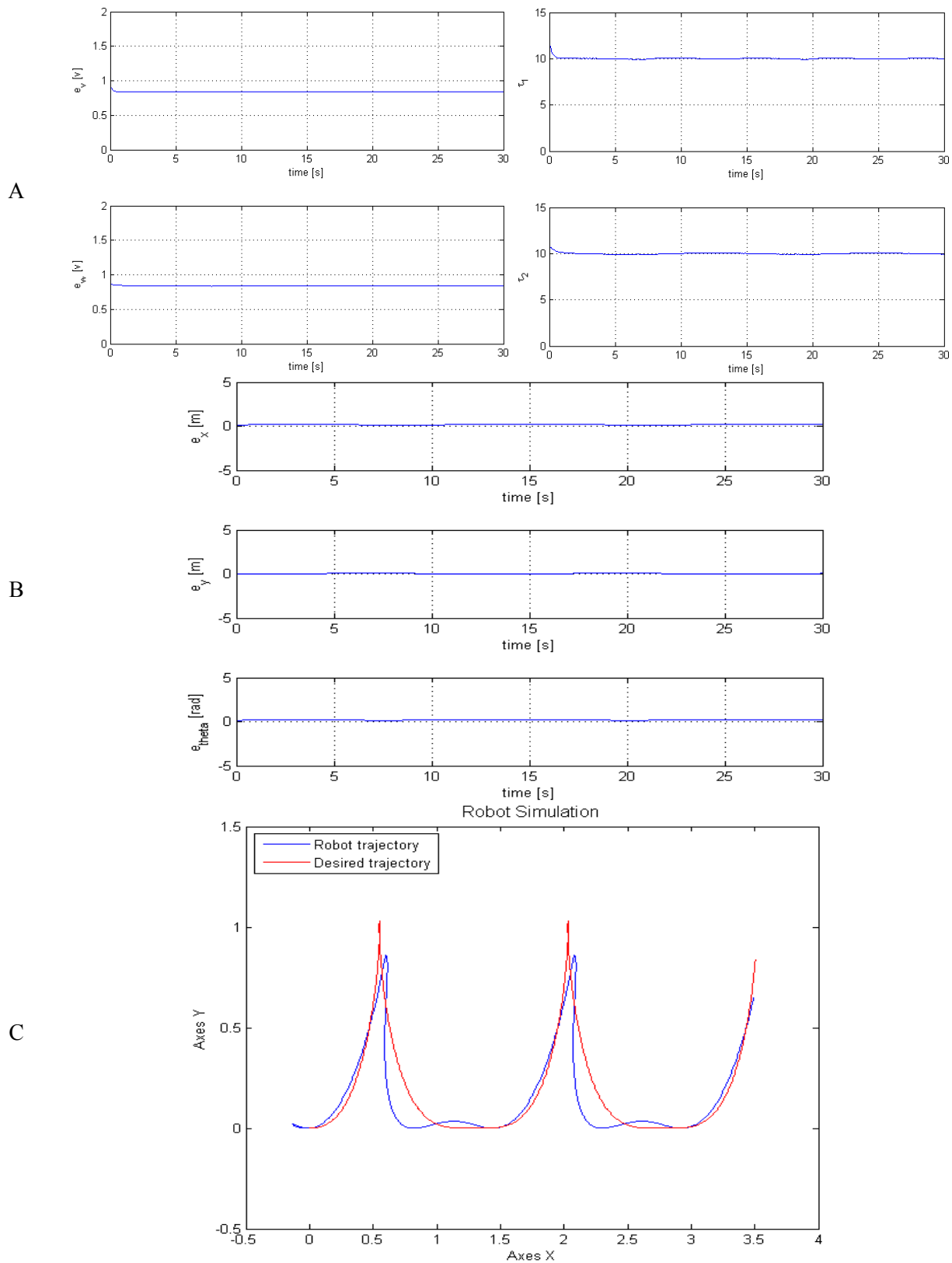


Figura 4.28: Resultados de simulación del robot autónomo móvil utilizando el controlador difuso tipo-1 optimizado.

4.3.6 Resultados del uso del método Híbrido PSO-GA para el controlador difuso tipo-2 para un robot autónomo móvil

Se utilizó el método híbrido propuesto PSO-GA para encontrar el controlador difuso tipo-2 óptimo con funciones de membresía de tipo *TTT* (Trapezoidal-Triangular-Trapezoidal) utilizando (3.5) y (3.6) que representa al robot autónomo móvil.

La Tabla 4.51 contiene los valores utilizados en la configuración del método Híbrido PSO-GA, el tiempo de ejecución y el error promedio obtenido del mejor controlador difuso para cada configuración. Se variaron los valores de la variable cognitiva (C1), social (C2), la variable de inercia, variable de cruce y variable de mutación. La onceava columna de esta tabla, indica el error promedio alcanzado con el controlador obtenido por el método híbrido.

Tabla 4.51: Resultados del controlador difuso tipo-2 *TTT* para robot autónomo móvil.

No	Part/ Pobl.	Iter / Gener	C1	C2	Inercia	% Rem	Cruce	Mut	Tiempo del PSO-GA	Error Promedio de Velocidad	Error Promedio de Posición
1	35	40	0.52374	0.50953	0.90165	0.7	0.7	0.3	80:29:37	1.676020	0.137041
2	70	50	0.81491	0.90589	0.17064	0.7	0.8	0.3	165:46:26	1.676313	0.137068
3	25	50	0.12368	0.91535	0.22321	0.7	0.7	0.1	80:01:09	1.676313	0.137068
4	30	60	0.30544	0.70235	0.69500	0.7	0.8	0.1	76:18:29	1.676313	0.137068
5	80	25	0.32473	0.56269	0.31545	0.7	0.7	0.1	159:52:46	1.676313	0.137068
6	20	50	0.70790	0.34631	0.88330	0.7	0.9	0.6	72:14:45	1.676313	0.137068
7	80	20	0.76088	0.26179	0.23559	0.9	0.8	0.4	84:43:52	1.676313	0.137068
8	80	20	0.53955	0.32518	0.63027	0.9	0.8	0.2	81:40:23	1.676770	0.137068
9	25	100	0.12368	0.91535	0.22321	0.7	0.7	0.2	165:31:37	1.676313	0.137068
10	50	30	0.97811	0.76252	0.46100	0.7	0.6	0.1	73:38:03	1.676394	0.137068
11	50	30	0.97811	0.76252	0.46100	0.7	0.6	0.1	126:05:02	1.676313	0.137068
12	30	25	0.62217	0.09218	0.19170	0.9	0.9	0.3	41:04:42	1.676313	0.137068
13	20	15	0.60960	0.63315	0.77131	0.7	0.8	0.2	14:07:14	1.676313	0.137163

A continuación se presenta el mejor resultado obtenido de las pruebas realizadas utilizando el método Híbrido PSO-GA para el robot autónomo móvil funciones de membresía *TTT* tipo-2.

Experimento (Mejor Simulación)

La Tabla 4.52 muestra los parámetros utilizados para las simulaciones y el error promedio del mejor experimento.

Tabla 4.52: Parámetros utilizados en el método Híbrido PSO-GA, tiempo y error promedio para el experimento.

Parámetros del Híbrido PSO-GA	Partículas/Población	35
	Iteraciones/Generaciones	40
	C1	0.52374
	C2	0.50953
	Inercia	0.90165
	%Reemplazo	0.7
	Cruce	0.7
	Mutación	0.3
Tiempo del Híbrido PSO-GA		80:29:37
Error Alcanzado		0.137041

Las siguientes figuras, muestran los resultados obtenidos en la simulación para el experimento, el error promedio alcanzado fué de 0.137041.

La Figura 4.29 A) muestra el comportamiento del método Híbrido PSO-GA y B) muestra las funciones de membresía la variable de entrada 1 (error de velocidad lineal) y la variable de entrada 2 (error de velocidad angular) del controlador difuso tipo-2 óptimo obtenido por el método Híbrido PSO-GA.

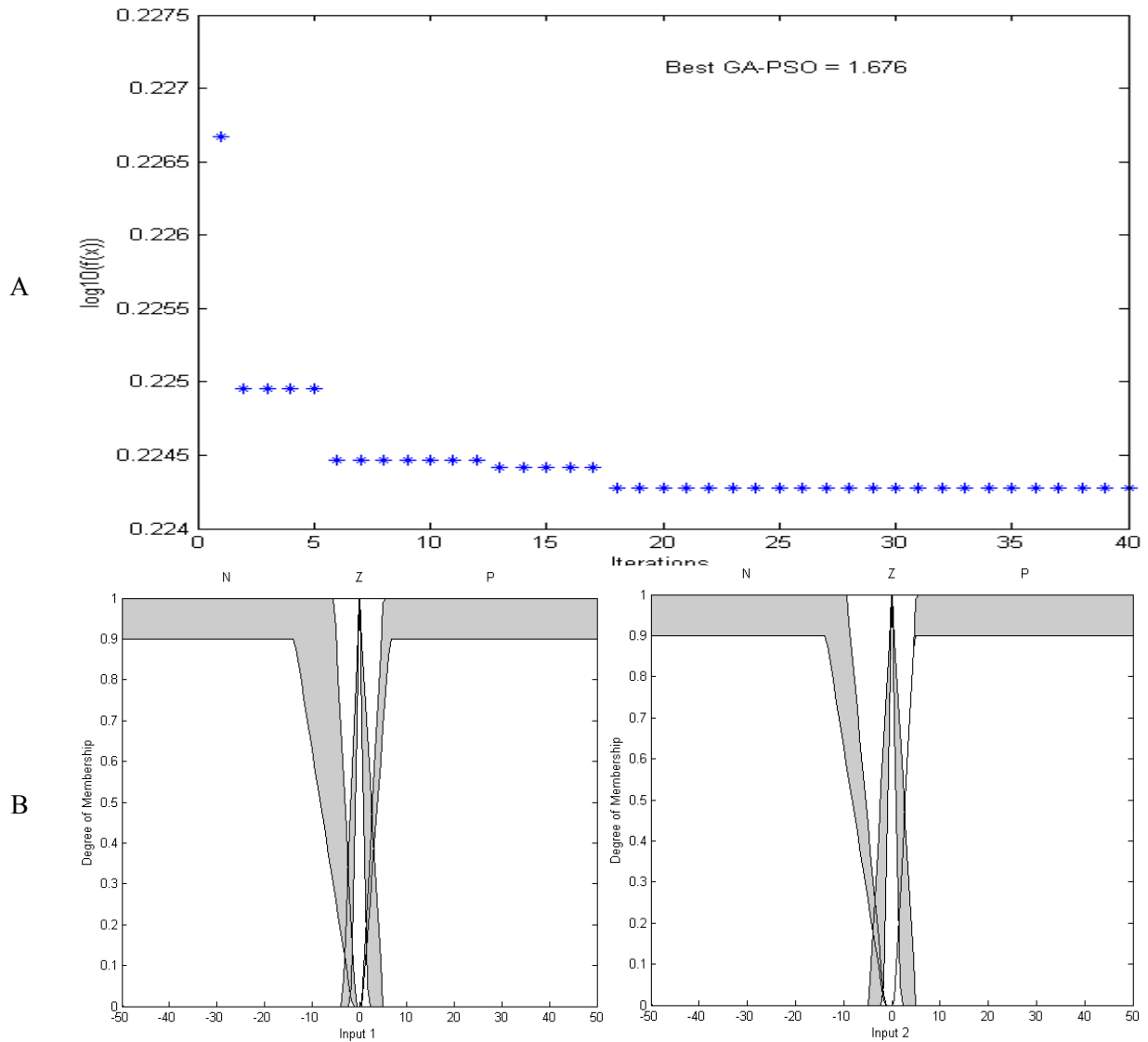


Figura 4.29: A) Comportamiento del método Híbrido PSO-GA y B) funciones de membresía de la entrada 1 y entrada 2 del controlador difuso tipo-2 óptimo.

La figura 4.30 A) muestra la respuesta los errores de velocidad lineal y la velocidad angular y los errores de torque derecho y torque izquierdo y B) muestra los errores de posición $e(x)$, $e(y)$ y $e(\theta)$ y se puede observar como tienden a cero de manera estable y C) el seguimiento de trayectoria del robot autónomo móvil con el mejor controlador difuso tipo-2 obtenido con método Híbrido PSO-GA.

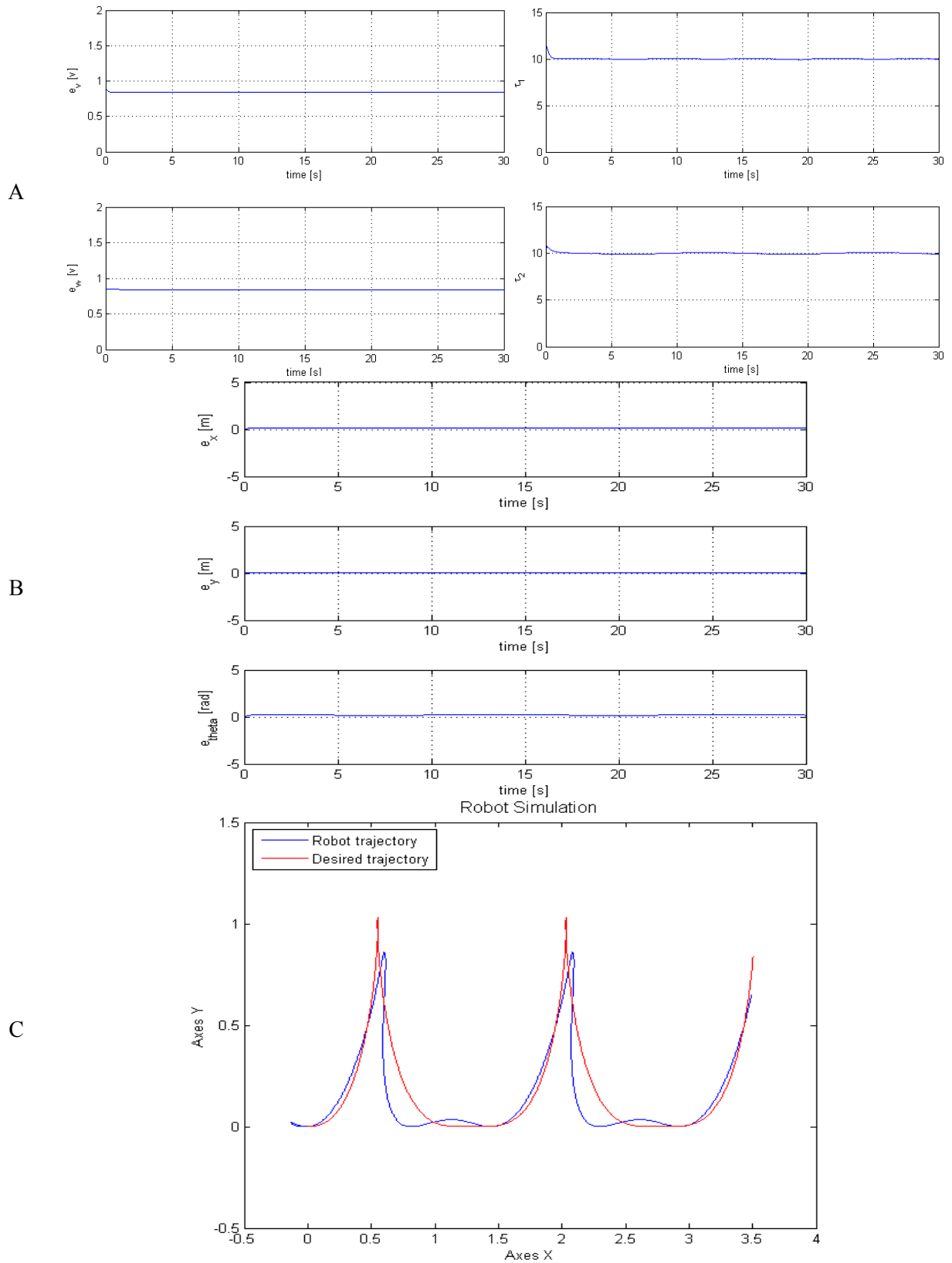


Figura 4.30: Resultados de simulación del robot autónomo móvil utilizando el controlador difuso tipo-2 optimizado.

5. Conclusiones

En este trabajo de tesis, se realizó el diseño de un método de optimización híbrido para encontrar controladores difusos óptimos. El método de optimización híbrido está compuesto por algoritmos genéticos (GA) y optimización por medio de nube de partículas (PSO). Para mostrar el desempeño del método híbrido se utilizaron diferentes casos de aplicación el cual dos de ellos se buscó el controlador difuso optimizado (Funciones de transferencia, llamadas planta 1 y planta 2, y un robot autónomo móvil) y en otro de los casos (Péndulo de rueda de Inercia) sólo se utilizó para encontrar las ganancias óptimas de un controlador PID.

Para el caso de las funciones de transferencia, planta 1 y planta 2 primeramente se hicieron pruebas haciendo uso de los algoritmos genéticos simples para encontrar el controlador optimizado para cada planta. Se tomaron dos tipo de controladores difusos, uno con funciones de membresía gaussianas y triangulares y otro controlador sólo con funciones de membresía gaussianas. Se usó un cromosoma de valores reales con 17 posiciones la cual las primeras 14 posiciones representaban la variable de entrada 1 y la variable de entrada 2 y las 3 posiciones

restantes representaban la variable de salida del controlador difuso con funciones de membresía gaussianas y triangulares; para el controlador con funciones de membresía gaussianas se utilizó un cromosoma de valores reales de 15 posiciones la cual las primeras 12 posiciones eran para la variable de entrada 1 y la variable de entrada 2 y las 3 posiciones restantes eran para la variable de salida del controlador difuso.

Para evaluar el algoritmo genético se utilizó el error promedio obtenido por la diferencia entre la referencia que debía seguir el controlador y el resultado obtenido del mismo arrojados por el sistema, el objetivo fué buscar el error más pequeño posible llegando a su referencia de manera suave y estable.

Una vez terminado el conjunto de pruebas con los controladores tipo-1 se diseñó el controlador tipo-2 para las mismas plantas y obtener cual tipo de controlador era el más eficiente para este tipo de aplicación, dando mejores resultados el controlador con funciones de membresía gaussianas y triangulares ya que llegaba a la referencia en menor tiempo y error promedio que el controlador con sólo funciones de membresía gaussianas.

Para el caso del Péndulo de rueda de inercia no se buscó un controlador difuso, sino que se buscaron las ganancias k_1 , k_2 , y k_3 de su controlador PID convencional para estabilizar el péndulo en posición vertical.

Se utilizó un cromosoma de valores reales de 3 posiciones que representaban las ganancias del controlador PID; los valores máximos que se tomaron para cada una de las ganancias fueron: para $k_1 = 250$, $k_2 = 50$ y para $k_3 = 0.1$. El péndulo fué representado de forma matemática en Matlab para encontrar los valores de las ganancias haciendo uso de los algoritmos genéticos. El resultado de las ganancias fueron de $k_1 = 250$, $k_2 = 21.849774$, y $k_3 = 0.0841414$ con un error promedio de simulación de 0.00063 rad; probando éstas misma ganancias en el modelo físico obtuvimos un error promedio de 0.22061 rad con un tiempo de estabilización de 4.5 segundos.

Para el caso del robot autónomo móvil se tomó el mismo diseño utilizado en trabajos anteriores, utilizando un algoritmo genético de 28 posiciones de valores reales que representaba a

un controlador difuso con funciones de membresía trapezoidales y triangulares en el cual las primeras 11 posiciones representaban la variable de entrada 1, las otras 11 posiciones para la variable de entrada 2 y las otras 6 posiciones representaban las variables de salida de torque 1 y torque 2. En este caso de aplicación con algoritmos genéticos sólo se hicieron pruebas con controladores difusos tipo-1, y obtuvimos un error promedio de posición de 0.154186634.

Otro de los métodos de optimización individual fué el de Optimización por medio de nube de Partículas (PSO, Particle Swarm Optimization). Se utilizaron los mismos parámetros de configuración de los controladores difusos para la planta 1 y planta 2 para encontrar el óptimo pero haciendo uso de PSO. Pudimos observar que el tiempo de ejecución del método de optimización PSO es menor y con mejores resultados.

De nueva cuenta se utilizó PSO para la búsqueda de las ganancias óptimas del controlador PID para el Péndulo de rueda de inercia. Basados en la misma configuración de los límites mínimos y máximos para la búsqueda de cada una de las ganancias obtuvimos el siguiente resultado: $k_1 = 499.95$, $k_2 = 19.08$, y $k_3 = 0.0587$ con un error promedio de simulación de 0.00047 rad; probando estas misma ganancias en el modelo físico obtuvimos un error promedio de 0.00723 rad con un tiempo de estabilización de 4.4 segundos.

Para el caso del robot autónomo móvil se tomó el mismo diseño del controlador difuso pero ahora el método de optimización fué PSO. En este caso de aplicación con PSO sólo se hicieron pruebas con controladores difusos tipo-1, y obtuvimos un error promedio de posición de 0.161329487. Haciendo una comparación entre los resultados de los controladores difusos, el encontrado con algoritmos genéticos resulto el mejor.

Finalmente se hizo uso del método de optimización propuesto, que es el método Híbrido PSO-GA. En este trabajo el método híbrido sólo fué aplicado para obtener controladores difusos óptimos utilizando los casos de la planta 1, planta 2 y robot autónomo móvil.

Al realizar las pruebas en la planta 1 observamos que no mejoró los resultados obtenidos por los métodos de optimización anteriores y el tiempo de ejecución del método híbrido

incremento en más del 50% en comparación con los otros dos métodos. Sin embargo al utilizarlo para la planta 2 mostró muchos mejores resultados que los dos métodos anteriores demostrando que el método híbrido puede llegar a mejorar los resultados en aplicaciones con mayor grado de complejidad.

En el caso del robot autónomo móvil utilizamos la misma configuración base utilizada en los otros métodos de optimización dando como resultado un error promedio de 0.13601414 utilizando el mejor controlador difuso tipo-1. Se hicieron pruebas con el método híbrido para controladores difusos tipo-2 y obtuvimos un error promedio del mejor controlador de 0.137041.

Una vez terminado con todo el conjunto de pruebas concluimos que el método híbrido PSO-GA para la búsqueda de controladores difusos funciona mejor que con los métodos de optimización individuales y además de dar muy buenos resultados cuando los casos de aplicación son más complejos.

Uno de los puntos inconvenientes del método Híbrido PSO-GA, es el tiempo de ejecución del mismo ya que el registro más largo de tiempo fue de 165:31:37 hrs. Por tal motivo como trabajo futuro sería mejorar los procesos de comunicación internos del método híbrido PSO-GA para bajar los tiempos de ejecución además de poder mejorar los resultados en la optimización de controladores difusos.

6. Referencias

- [1] Aceves, A., Aguilar J., "A Simplified Version of Mamdani's Fuzzy Controller: The Natural Logic Controller", *IEEE Transactions on Fuzzy Systems*, Vol. 14, No. 1. 2006.
- [2] Aguilar, L. T., Boiko, I., Fridman, L. and Freidovich, L., "Inducing Oscillations in an Inertia Wheel Pendulum via Two-Relays Controller: Theory and Experiments", *American Control Conference*, St. Louis, MO, USA, pp. 65-70, June 10-12, 2009.
- [3] Alcalá, R., Alcalá-Fdez, J., and Herrera, F., "A proposal for the genetic lateral tuning of linguistic fuzzy systems and its interaction with rule selection," *IEEE Transactions on Fuzzy Systems*, Vol. 15, No. 4, pp. 616-635, 2007.
- [4] Alcalá, R., Gacto, M. J., Herrera, F., and Alcalá-Fdez, J., "A multi-objective genetic algorithm for tuning and rule selection to obtain accurate and compact linguistic fuzzy rule-based systems," *Int. J. Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol.15, No. 5, pp. 539-557, 2007.
- [5] Angeline, P. J., "Evolutionary Optimization Versus Particle Swarm Optimization: Philosophy and Performance Differences", In: Porto, V.W., Waagen, D. (eds.) EP 1998. LNCS, Vol. 1447, pp. 601–610. Springer, Heidelberg, 1998.
- [6] Angeline, P. J., "Using selection to improve particle swarm optimization," In *Proceedings IEEE World Congress on Computational Intelligence*, Anchorage, Alaska, IEEE, pp. 84-89, 1998.
- [7] Astudillo, L., Castillo, O., Aguilar, L. T., Martinez, R., "Hybrid Control for an Autonomous Wheeled Mobile Robot Under Perturbed Torques", Vol. 4239, No 1 in LNAI, Cancun, Mexico, IFSA 2007, pp.

594-603, Springer Verlag, June 2007.

- [8] Bentalba, S., El Hajjaji, A., Rachid, A., "Fuzzy Control of a Mobile Robot: A New Approach", Proceedings IEEE International Conference On Control Applications, Hartford, CT, pp 69-72, October 1997.
- [9] Bloch, A. M., Drakunov, S., "Tracking in NonHolonomic Dynamic System Via Sliding Modes", Proceedings IEEE Conference on Decision & Control, Brighton, UK, pp. 1127-1132, 1991.
- [10] Brockett, R.W., "Asymptotic stability and feedback stabilization", In: Millman, R.S., Sussman, H.J. (eds.) Differential Geometric Control Theory, pp. 181–191. Birkhauser, Boston 1983.
- [11] Campion, G., Bastin, G., B. Novel, D'Andrea., "Structural Properties and Classification of Kinematic and Dynamic Models of Wheeled Mobile Robots", IEEE Transactions On Robotics and Automation, Vol. 12, No. 1, pp. 47-62, February 1996.
- [12] Carlisle, A., Dozier, G., "An off-the-self PSO", *Proceedings of the Workshop on Particle Swarm Optimization*, Indianapolis (USA), pp. 1-6, April 2001.
- [13] Casillas, J., Cordon, O., del Jesús, M. J., and Herrera, F., "Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction," *IEEE Transactions on Fuzzy Systems*, Vol. 13, No. 1, pp. 13-29, 2005.
- [14] Castillo O., Melin P., "Intelligent Systems with Interval Type-2 Fuzzy Logic", *International Journal of Innovative Computing, Information and Control (ICIC) International*. ISSN: 1349-4198. Vol. 4, No. 4: 771-783, Abril 2008.
- [15] Chen, M.-R., Li, X., Zhang, X., and Lu, Y.-Z.; "A novel particle swarm optimizer hybridized with extremal optimization," *Applied Soft Computing*, Vol. 10, pp. 367-373, 2010.
- [16] Cordon, O., Gomide, F., Herrera, F., Hoffmann, F., and Magdalena, L., "Ten years of genetic fuzzy systems: current framework and new trends," *Fuzzy Sets and Systems*, Vol. 141, No. 1, pp. 5-31, 2004.
- [17] DeJong K., "Learning with genetic algorithms: an overview", *Machine Learning* Vol. 3 No. 2-3, pp. 121–138, 1988.
- [18] Driankov D., Hellendoorn H., Reinfrank M., "An Introduction to Fuzzy Control", Springer, Berlin, 1996.
- [19] Duc Do, K., Zhong-Ping, J., Pan, J., "A global output-feedback controller for simultaneous tracking and stabilizations of unicycle-type mobile robots", *IEEE Transactions on Robotics and Automation* Vol. 20, No. 3, pp. 589-594, 2004.
- [20] Eberhart, R.C., Kennedy, J., "A new optimizer using particle swarm theory", In: Proceedings of the Sixth International Symposium on Micromachine and Human Science, Nagoya, Japan, pp. 39-43, 1995.
- [21] Eberhart, R.C., Shi, Y., "Particle swarm optimization: developments, applications and resources", *Proceedings of the 2001 Congress on Evolutionary Computation*, Seoul (South Korea), Vol. 1, pp. 81-86, May 2001.
- [22] Engelbrecht, A. P., "Fundamentals of Computational Swarm Intelligence", Wiley, pp. 86-122, England, 2005.
- [23] Estrada, A., Aguilar, L., Iriarte, R., and Fridman, L. "Two relay controller for real time trajectory generation and its application to inverted orbital stabilization of inertia wheel pendulum via quasi-

continuous HOSM,” *Asian Journal of Control*, pp. 1-9, 2010.

- [24] Fierro, R., Lewis, F.L., "Control of a Nonholonomic Mobile Robot Using Neural Networks", *IEEE Transactions on Neural Networks*, Vol. 9, No. 4, pp 589-600, July 1998.
- [25] Fogel D.B., "An introduction to simulated evolutionary optimization", *IEEE Transactions on neural networks*, Vol. 5, No. 1, pp. 3-14, January 1994.
- [26] Fukao, T., Nakagawa, H., Adachi, N., "Adaptive Tracking Control of a NonHolonomic Mobile Robot", *IEEE Transactions on Robotics and Automation* Vol. 16 No. 5, pp. 609-615, 2000.
- [27] Garibaldi, J.A., Castillo O., "Optimización de Rutas para un Robot Autónomo Móvil utilizando Algoritmos Genéticos en Paralelo, Tesis, División de Estudios de Posgrado e Investigación, ITT, México, 2005.
- [28] Hu, X., "Particle swarm optimization". (<http://www.swarmintelligence.org/>).
- [29] Hwang, J.P., Euntai, K., "Robust Tracking Control of an Electrically Driven Robot: Adaptive Fuzzy Logic Approach", *IEEE Transactions on Fuzzy Systems*, Vol. 14, No. 2, 2006.
- [30] Ishikawa, S., "A Method of Indoor Mobile Robot Navigation by Fuzzy Control", *Proceedings International Conference on Intelligent Robots and Systems*, Osaka, Japan, pp 1013-1018, 1991.
- [31] Jang, J.-S. R., Sun, C.-T., Mizutani, E., "Neuro-Fuzzy and Soft Computing: a computational approach to learning and machine intelligence", Prentice Hall, 1997.
- [32] Kanayama, Y., Kimura, Y., Miyazaki, F., Noguchi, T., "A Stable Tracking Control Method For a Non-Holonomic Mobile Robot", *Proceedings IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, Osaka, Japan, pp 1236- 1241, 1991.
- [33] Kao, Yi-T. and Zahara, E., "Hybrid genetic algorithm and particle swarm optimization for multimodal functions", *Applied Soft Computing*, Volume 8, Issue 2, pp. 849-857, March 2008.
- [34] Karnik, N.N., Mendel, J. M., "An Introduction to Type-2 Fuzzy Logic Systems", University of Southern California, Los Angeles, EEUU, June 1998.
- [35] Karnik, N.N., Mendel, J. M., "Operations on Type-2 Fuzzy Sets", *Fuzzy Sets and Systems*, No. 122, pp. 327-348, 2001.
- [36] Karnik, N.N., Mendel, J. M., Liang, Q., "Type-2 Fuzzy Logic Systems", *IEEE Transactions on Fuzzy Systems*, Vol.7, No.6, pp. 643-658, 1999.
- [37] Kennedy, J., and Mendes, R., "Population structure and particle swarm performance". *Proceedings of IEEE conference on Evolutionary Computation*, pp. 1671-1676, 2002.
- [38] Kennedy, J., Clerc, M., "The particle swarm-explosion, stability, and convergence in a multidimensional complex space", *IEEE Transactions on Evolutionary Computation* Vol. 6, No. 1, pp. 58-73, 2002.
- [39] Kennedy, J., Eberhart R.C., "Swarm intelligence", Morgan Kaufmann Publishers, San Francisco, 2001.
- [40] Kennedy, J., Eberhart, R.C., "Particle swarm optimization", *Proceedings of the IEEE International Conference on Neural Networks-ICNN'95*, Perth (Australia), Vol.4, pp. 1942-1948, December 1995.
- [41] Khalil, H., "Nonlinear systems: 3rd edition", New York, Prentice Hall, 2002.
- [42] Kim, D. H., and Hirota, K., "Vector control for loss minimization of induction motor using GA-PSO," *Applied Soft Computing*, Vol. 8, pp. 1692-1702, 2008.

- [43] Kolmanovsky, I., McClamroch, N. H., "Developments in Nonholonomic Control Problems", IEEE Control Systems, Vol. 15, No. 6, pp. 20–36, December. 1995.
- [44] Kulkarni, A., "Computer Vision and Fuzzy-Neural Systems", Prentice Hall PTR, May 2001.
- [45] Lee, T. H., Leung, F. H. F., Tam, P. K. S., "Position Control for Wheeled Mobile Robot Using a Fuzzy Controller", Proceedings IEEE on Industrial Electronics Society, Vol. 2, pp. 525-528, 1999.
- [46] Lee, T-C., Lee, C. H., Teng, C-C., "Tracking Control of Mobile Robots Using the Backstepping Technique", Proceedings 5th. International Conference on Control, Automation, Robot Vision, Singapore, pp. 1715-1719, December 1998.
- [47] Lee, T-C., Tai, K., "Tracking Control of Unicycle-Modeled Mobile robots Using a Saturation Feedback Controller", IEEE Transactions on Control Systems Technology, Vol. 9, No. 2, pp 305-318, March 2001.
- [48] Man, K.F., Tang, K.S., Kwong, S., "Genetic Algorithms: Concepts and Designs", Springer, 2000.
- [49] Marczyk, Adam "Genetic Algorithms and Evolutionary Computation", <http://www.talkorigins.org/faqs/genalg/genalg.html>, 2004.
- [50] Marinakis, Y. and Marinaki, M., "A hybrid genetic – Particle Swarm Optimization Algorithm for the vehicle routing problem", Expert Systems with Applications, Volume 37, Issue 2, pp. 1446-1455, March 2010.
- [51] Martínez, R., Castillo, O., Aguilar, L. T., Rodríguez, A., "Genetic Optimization of Interval Type-2 Fuzzy Logic Controllers Applied to Linear Plants", V Encuentro Regional Académico ERA`09 los días 11 y 12 de Noviembre del 2009, Tijuana, Baja California, Mexico, pp. 84-89, ISBN: 978-607-414-113-9.
- [52] Martinez, R., Castillo, O., Aguilar, L. T., "Optimization of Interval Type-2 Fuzzy Logic Controllers for a Perturbed Autonomous Wheeled Mobile Robot using Genetic Algorithms", Soft Computing for Hybrid Intelligence Systems, pp. 3-18, 2008.
- [53] Martinez, R., Castillo, O., Aguilar, L. T., Melin, P., "Fuzzy Logic Controller Optimization using Genetic Algorithms and Particle Swarm Optimization", 9th Mexican International Conference on Artificial Intelligence, MICAI, November 8-13, Pachuca, Mexico, pp. 475-486, 2010.
- [54] Martinez, R., Castillo, O., Aguilar, L. T., Rodríguez, A., "Bio-inspired Optimization methods of Fuzzy Logic Controllers Applied to Linear Plants", WSC14, 2009 Online World Conference on Soft Computing in Industrial Applications WWW, November 17-29, 2009. Publish: Soft Computing in Industrial Applications, Advances in Soft Computing, Vol. 75, pp. 245-252, 2010.
- [55] Martinez, R., Castillo, O., Aguilar, L. T., Rodríguez, A., "Evolutionary Optimization of Type-2 Fuzzy Logic Systems Applied to Linear Plants", Evolutionary Design of Intelligent Systems in Modeling, Simulation and Control, Springer, ISBN: 978-3-642-04513-4, pp. 17- 31, October 2009.
- [56] Martinez, R., Castillo, O., Aguilar, L. T., Rodríguez, A., "Optimization of Type-2 Fuzzy Logic Controllers for Mobile Robots Using Evolutionary Methods", Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics San Antonio, TX, USA , pp. 4909 -4914, October 2009.
- [57] Martinez, R., Castillo, O., Aguilar, L. T., Rodríguez, A., "Optimization of Type-2 Fuzzy Logic Controllers using PSO Applied to Liner Plants", Oscar Castillo, Witold Pedrycz, Janusz Kacprzyk in Soft Computing for Intelligent Control and Mobile Robotics, Springer, Alemania, pp. 181-193, 2010.
- [58] Martinez, R., Castillo, O., Aguilar, L. T., Rodríguez, A., "Type-2 Fuzzy Logic Controllers

Optimization using Genetic Algorithms and Particle Swarm Optimization”, 2010 IEEE Conference on Granular Computing (GrC 2010), Aug 14-16 , Silicon Valley USA, pp.724-727, 2010.

- [59] Martinez, R., Castillo, O., Aguilar, L.T., "Intelligent Control For A Perturbed Autonomous Wheeled Mobile Robot Using Type-2 Fuzzy Logic and Genetic Algorithms", *Journal of Automation, Mobile Robotics & Intelligent Systems* Vol. 2, 2008.
- [60] Martinez, R., Castillo, O., Aguilar, L.T., "Optimization of Interval Type-2 Fuzzy Logic Controllers for a Perturbed Autonomous Wheeled Mobile Robot using Genetic Algorithms", *Information Sciences, Informatics and Computer Science Intelligent Systems Applications and International Journal*, Vol. 179 No. 13, pp. 2158-2174, 2009.
- [61] Martinez, R., Rodriguez, A., Aguilar, L. T., "Gain Optimization for Inertia Wheel Pendulum Stabilization using a Bio-Inspired and Evolutionary Methods", *Proceeding of the World Conference on Soft Computing*, San Francisco, CA; pp. 246-251, May 23-26, 2011.
- [62] Martinez-Soto, R., Castillo, O., Aguilar, L. T., Rodriguez, A., "Gain Optimization for Inertia Wheel Pendulum Stabilization using Particle Swarm Optimization and Genetic Algorithms", *International Journal of Innovative Computing, Information & Control*, (IJICIC), 2012.
- [63] Mendel, J., "Uncertain Rule-Based Fuzzy Logic Systems, Introduction and new directions", Prentice Hall PTR, 2001.
- [64] Mendel, J., Bob, R., "Type-2 Fuzzy Sets Made Simple", *IEEE Transactions on Fuzzy Systems*, Vol.10, No.2, pp.117-127, 2002.
- [65] Muhammad, S., Saira, Z., Muddassar, F., "A Hybrid GA-PSO Fuzzy System for user Identification on Smartphones," *GECCO'09 Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, Montreal-Canada, pp. 1617-1624, July 8-12, 2009.
- [66] Mukherjee, V., and Ghoshal, S.P., "Intelligent particle swarm optimized fuzzy PID controller for AVR system", *Electric Power Systems Research*, Vol. 77, pp. 1689-1698, 2007.
- [67] Nelson, W., Cox, I., "Local Path Control for an Autonomous Vehicle", *Proceedings IEEE Conference on Robotics and Automation*, pp. 1504-1510, 1988.
- [68] Pawlowski, S., Dutkiewicz, P., Kozłowski, K., Wroblewski, W., "Fuzzy Logic Implementation in Mobile Robot Control", *2nd Workshop On Robot Motion and Control*, pp 65-70, October 2001.
- [69] Senthil Arumugam, M. and Rao, M.V.C.; "On the improved performances of the particle swarm optimization algorithms with adaptive parameters, cross-over operators and root mean square (RMS) variants for computing optimal control of a class of hybrid systems," *Applied Soft Computing*, Vol. 8, pp. 324-336, 2008.
- [70] Sepulveda, R., Castillo, O., Melin, P. and Montiel, O.; "Fundamentos de Lógica Difusa"; Ediciones ILCSA, Agosto 2002.
- [71] Song, K. T., Sheen, L. H., "Heuristic fuzzy-neural Network and its application to reactive navigation of a mobile robot", *Fuzzy Sets Systems*, Vol. 110, No. 3, pp 331-340, 2000.
- [72] Tsai, C-C., Lin, H-H., Lin, C-C., "Trajectory Tracking Control of a Laser-Guided Wheeled Mobile Robot", *Proceedings IEEE International Conference On Control Applications*, Taipei, Taiwan, pp 1055-1059, September 2004.
- [73] Valdez, F., Melin, P. and Castillo, O., "An improved evolutionary method with fuzzy logic for combining Particles Swarm Optimization and Genetic Algorithms", *Applied SoftComputing-The*

Impact of soft Computing for the Progress of Artificial Intelligence, Volume 11, Issue 2, pp. 2625-2632, March 2011.

- [74] Veeramachaneni, K., Osadciw, L., Yan, W., "Improving Classifier Fusion Using Particle Swarm Optimization", In: IEEE Fusion Conference, Italy, July 2006.
- [75] Wei, W., Jiatao, S., Zhongzxiu, Y., Zheru: Wavelet-based Illumination Compensation for Face Recognition using Eigenface Method", Intelligent Control and Automation. In: WCICA 2006, June 21-23, Vol. 2, pp. 10356-10360, 2006.
- [76] Yen, J., Langari, R., "Fuzzy Logic: Intelligence, Control, and Information", Prentice Hall.
- [77] Zadeh, L.A., "Fuzzy Sets", Information and Control, Vol. 8, pp. 338-353, 1965.
- [78] Zadeh, L.A., "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic", Fuzzy Sets and Systems, Vol. 90, pp. 117-117, Elsevier, 1997.