



INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA

**SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA**

**SISTEMA DE EXCITACIÓN DE UN GENERADOR
SÍNCRONO UTILIZANDO UN CONTROLADOR
LÓGICO DIFUSO TIPO-2**

TESIS

**QUE PARA OBTENER EL GRADO DE MAESTRO
EN CIENCIAS EN INGENIERÍA ELÉCTRICA**

PRESENTA:

EMILIO RICARDO MOO MEDINA

DIRECTORES DE TESIS:

DR. DAVID ROMERO ROMERO

DR. RAÚL CORTÉS MATEOS



MEXICO, D.F. JULIO 2015

Dedicatorias

En memoria de:

Mi Madre Silvia, Abuelos Aida & Ricardo. Porque siempre conté con su amor y cariño. Que Dios los tenga en su Santa Gloria.

A mi padre:

Santos Wilber Moo Moo quien me ha educado y enseñado valores.

A mis hermanos:

Martha Esthela Moo Moo & Rodrigo Alonso Moo Moo por su cariño y apoyo incondicional.

A mis pequeños sobrinos:

Evelyn, Josué, Saúl y Alan; quienes siempre me inspiran inocencia y ganas de vivir.

A esa persona tan especial:

Francis, porque su amistad y cariño fue decisivo en la recta final del desarrollo de la presente tesis. Y porque me ha enseñado a sonreír y tener fe en la vida. Te amo Frans.

Agradecimientos

A Dios Padre, Hijo y Espíritu Santo por el maravilloso don de la vida, la familia y por su amor y misericordia para conmigo.

A todos mis familiares, por su apoyo y comprensión en todo aspecto. Los quiero mucho.

A mi novia Francis por su incondicional cariño y apoyo. Dios permita tenerte siempre a mi lado.

A la Escuela de Pastoral, por el apoyo espiritual y amistades que de ahí se han dado.

Al Dr. David Romero por el apoyo académico consejos y amistad. Le expreso mi respeto y le debo gran parte de mi formación estos años.

Al Dr. Raúl Cortés por sus consejos, apoyo, amistad y conocimientos compartidos durante el desarrollo de la presente tesis.

A mis compañeros, amigos y demás profesores de la sección quienes hicieron agradable el estudio y estancia en esta institución.

A compañeros y amigos que desde mi ciudad natal siempre me manifestaron afecto y apoyo.

A todos muchas gracias.

RESUMEN

En este trabajo se desarrolla un regulador de voltaje para el sistema de excitación de un generador síncrono utilizando un controlador difuso tipo-2 en intervalos (IT2 FLC del inglés Interval Type-2 Fuzzy Logic Controller). Se muestra el diseño del controlador desde su arquitectura. Para el espacio de entrada, se utilizan conjuntos difusos triangulares y Gaussianos tipo-2 en intervalos y se implementan los algoritmos de reducción de tipo Karnick & Mendel (KM) y el Karnick & Mendel mejorado (EKM, del inglés Enhanced Karnik & Mendel), siendo este último más eficiente en tiempo de cómputo. La metodología de sintonización parte de un controlador difuso tipo-1 como base y se añaden incertidumbres en forma manual considerando la evolución del diagrama de fases, índices de desempeño del error y la superficie de control propia del controlador.

El sistema de prueba es un generador síncrono conectado a un bus infinito mediante una línea de transmisión corta o larga. Este sistema de prueba se somete a cortocircuito en el bus del generador con duración de 6 y 10 ciclos; esta falla se denomina gran disturbio y se consideran casos con y sin incertidumbre en el voltaje terminal. Esta incertidumbre permite simular variaciones en las mediciones del transductor de voltaje terminal. Se realizan también pruebas a pequeños disturbios como son : toma de carga del generador manipulando el mando del gobernador; y cambios en el voltaje de referencia con y sin incertidumbre en la señal de voltaje de referencia. Esta última prueba permite medir índices de comportamiento dinámico tales como: sobretiro, tiempo de respuesta, tiempo de estabilización y tiempo en alcanzar el valor pico.

Se compara el controlador tipo-2 con un controlador lógico difuso tipo-1 (T1 FLC del inglés Type-1 Fuzzy Logic Controller) y un regulador convencional proporcional ST1.

El desempeño en general de ambos controladores difusos y el regulador convencional ST1 es cuantificado mediante índices de desempeño del error en el voltaje terminal y ángulo de carga de la máquina. Estos índices permiten cuantificar el error en el sistema de control y se denominan: integral squared error (ISE); integral absolute error (IAE) e integral time absolute error (ITAE).

Los resultados en general muestran un mejor desempeño del controlador difuso tipo-2 cuando utiliza conjuntos difusos Gaussianos tipo-2 con pequeña incertidumbre y nueve reglas de conocimiento, el cual es el mínimo conocimiento posible en este trabajo. Esto sugiere que el IT2 FLC logra un mejor desempeño cuando el conocimiento del sistema - dado el número de reglas- es en cierta forma incierto.

ABSTRACT

In this thesis a voltage regulator based in an interval type-2 Fuzzy Logic Controller (IT2 FLC) for the excitation system of a synchronous generator is developed. The design of the fuzzy controller from its architecture is showed. This controller uses triangular and Gaussian interval type-2 fuzzy sets. Two type reduction algorithms were compared, the conventional Karnik-Mendel (KM) and the enhanced Karnik-Mendel (EKM). The results show a better computation time and a fewer iterations for the EKM over the conventional KM. The type-2 fuzzy controller is tuned from the type-1 controller; the difference is the amount of uncertainty in the input fuzzy sets. This tuning methodology considers the evolution of the phase diagram, the performance index and the control surface.

Tests were applied to the machine-infinite bus model with long or short line transmission. The synchronous generator is subject to a short circuit with six and ten cycles fault duration with and without uncertainty in the terminal voltage; this is called large disturbance. These uncertainties allow us to simulate variations in measurements of the transducer terminal voltage. The system was also subject to test to a small signal such as: load changes and change of voltage reference with and without uncertainty. Small signals tests allow us to measure the performance dynamic index of the transient response such as: overshoot, rise time, settling time and time to reach peak value.

An IT2 FLC is compared with a type-1 fuzzy logic controller (T1 FLC) and a proportional regulator type ST1.

The performance of both fuzzy controllers and the proportional regulator is measure in terms of integral square error (ISE), integral absolute error (IAE) and integral time absolute error (ITAE) for the error in terminal voltage and power angle.

Overall results show a better performance and improved handling to uncertainty of the interval type-2 fuzzy logic controller when uses Gaussian type-2 fuzzy sets and low number of fuzzy rules. This suggests that the IT2 FLC for this case allows a better performance when knowledge or the system is uncertain.

ÍNDICE GENERAL

DEDICATORIAS	<i>i</i>
AGRADECIMIENTOS	<i>ii</i>
RESUMEN	<i>iii</i>
ABSTRACT	<i>iv</i>
ÍNDICE	<i>v</i>
ÍNDICE DE FIGURAS	<i>x</i>
ÍNDICE DE TABLAS	<i>xv</i>
GLOSARIO DE TÉRMINOS	<i>xvii</i>
CAPÍTULO 1	1
<u>1. INTRODUCCIÓN</u>	<u>1</u>
1.1 GENERALIDADES	1
1.2 OBJETIVO	1
1.3 MOTIVACIÓN	2
1.4 LIMITACIONES Y ALCANCES	3
1.5 ESTADO DEL ARTE	4
1.6 CONTENIDO DE LA TESIS	8
CAPÍTULO 2	9
<u>2. SISTEMAS LÓGICOS DIFUSOS TIPO-2</u>	<u>9</u>
2.1 INTRODUCCIÓN	9
2.2 FUNCIONES DE MEMBRESÍA TIPO-2	11
2.3 CONCEPTOS SOBRE CONJUNTOS DIFUSOS TIPO-2	14
2.4 CONJUNTOS EMBEBIDOS TIPO 1 Y TIPO 2	18
2.5 OPERACIONES CON CONJUNTOS DIFUSO TIPO-2 EN INTERVALOS	20
2.6 INFERENCIA DE UN SISTEMA LÓGICO DIFUSO TIPO-2	20
2.7 MÉTODO DE REDUCCIÓN DE TIPO DE UN SISTEMA LÓGICO DIFUSO TIPO 2	23
2.8 MÉTODO DE REDUCCIÓN DE TIPO DE UN IT2 FS	25
2.8.1 ALGORITMO KM	26
2.8.2 ALGORITMO EKM	26
2.9 DEDIFUSIFICACIÓN	27
CAPÍTULO 3	28
<u>3. SISTEMA DE EXCITACIÓN UTILIZANDO CONTROL DIFUSO TIPO-2</u>	<u>28</u>
3.1 INTRODUCCIÓN	28
3.2 ARQUITECTURA DEL SISTEMA DE CONTROL	29
3.3 ESTRUCTURA DEL CONTROLADOR TIPO-1 Y TIPO-2	30
3.3.1 ELEMENTOS DEL CONTROLADOR TIPO-1	31
3.3.2 ELEMENTOS DEL CONTROLADOR TIPO-2	34

3.4	PARAMETRIZACIÓN DE CONJUNTOS DIFUSOS TIPO-2	35
3.4.1	PARAMETRIZACIÓN DE LOS CONJUNTOS DE ENTRADA	35
3.4.2	PARAMETRIZACIÓN DE LAS BARRAS DE SALIDA	39
3.5	SUPERFICIE DE CONTROL	40
3.6	FUENTES DE INCERTIDUMBRE A SIMULAR EN EL CONTROLADOR	41
3.7	CASO EJEMPLO DE OPERACIONES EN EL IT2 FLC	43
3.7.1	CONTROLADOR DIFUSO TIPO-2 COMO CONTROLADOR DIFUSO TIPO-1	46
3.8	METODOLOGÍA DE SINTONIZACIÓN DEL IT2 FLC	49
CAPÍTULO 4		53
4.	<u>PRUEBAS Y RESULTADOS</u>	53
4.1	INTRODUCCIÓN	53
4.2	SINTONIZACIÓN MANUAL PARA UNA LÍNEA DE TRANSMISIÓN CORTA Y GENERADOR SÍNCRONO EN VACÍO	54
4.2.1	ANÁLISIS DE RESULTADOS DE LA SINTONIZACIÓN DEL T1 FLC CASO LÍNEA CORTA	61
4.2.2	AJUSTE DE LA INCERTIDUMBRE EN LAS FUNCIONES DE MEMBRESÍA TIPO-2 CASO LÍNEA CORTA	62
4.2.3	ANÁLISIS DE RESULTADOS DE LA SINTONIZACIÓN DEL IT2 FLC CASO LÍNEA CORTA	71
4.3	SINTONIZACIÓN MANUAL PARA UNA LÍNEA DE TRANSMISIÓN LARGA CON EL GENERADOR SÍNCRONO EN VACÍO	74
4.3.1	ANÁLISIS DE LA SINTONIZACIÓN DEL T1 FLC CASO LÍNEA LARGA	78
4.3.2	AJUSTE DE LA INCERTIDUMBRE EN LAS FUNCIONES DE MEMBRESÍA TIPO-2 CASO LÍNEA LARGA	79
4.3.3	ANÁLISIS DE LA SINTONIZACIÓN DEL IT2 FLC CASO LÍNEA LARGA	83
4.4	PRUEBAS PARA UNA LÍNEA DE TRANSMISIÓN CORTA, SUJETA A UN DISTURBIO GRANDE	84
4.4.1	CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES Y 9 REGLAS DE CONOCIMIENTO	85
4.4.2	CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES Y 25 REGLAS DE CONOCIMIENTO	88
4.4.3	CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES Y 49 REGLAS DE CONOCIMIENTO	91
4.4.4	CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS Y 9 REGLAS DE CONOCIMIENTO	94
4.4.5	CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS Y 25 REGLAS DE CONOCIMIENTO	98
4.4.6	CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS Y 49 REGLAS DE CONOCIMIENTO	101
4.4.7	TIEMPO CRÍTICO DE LIBERACIÓN DE FALLA	103
4.4.8	ANÁLISIS DE RESULTADOS	105
4.5	PARA UNA LÍNEA DE TRANSMISIÓN CORTA, SUJETO A UN DISTURBIO GRANDE E INCERTIDUMBRE EN EL VOLTAJE TERMINAL	105
4.5.1	CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS E INCERTIDUMBRE EN EL VOLTAJE TERMINAL	106
4.5.2	CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES E INCERTIDUMBRE EN EL VOLTAJE TERMINAL	112
4.5.3	ANÁLISIS DE RESULTADOS	116

4.6 PRUEBAS PARA UNA LÍNEA DE TRANSMISIÓN LARGA, SUJETA A UN DISTURBIO GRANDE	117
4.6.1 CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES Y 9 REGLAS DE CONOCIMIENTO	117
4.6.2 CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES Y 25 REGLAS DE CONOCIMIENTO	120
4.6.3 CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES Y 49 REGLAS DE CONOCIMIENTO	122
4.6.4 CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS Y 9 REGLAS DE CONOCIMIENTO	124
4.6.5 CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS Y 25 REGLAS DE CONOCIMIENTO	126
4.6.6 CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS Y 49 REGLAS DE CONOCIMIENTO	128
4.6.7 ANÁLISIS DE RESULTADOS	128
4.7 PARA UNA LÍNEA DE TRANSMISIÓN CORTA SUJETA A UN PEQUEÑO DISTURBIO EN EL VOLTAJE DE REFERENCIA DE 1 A 1.05 P.U.	129
4.7.1 CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES Y 9 REGLAS DE CONOCIMIENTO	131
4.7.2 ANÁLISIS DE RESULTADOS	133
4.8 PARA UNA LÍNEA DE TRANSMISIÓN CORTA SUJETA A UN PEQUEÑO DISTURBIO EN EL VOLTAJE DE REFERENCIA DE 1 A 1.05 P.U. E INCERTIDUMBRE DE $\pm 1\%$	134
4.8.1 CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS Y 49 REGLAS DE CONOCIMIENTO	134
4.8.2 CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES Y 9 REGLAS DE CONOCIMIENTO	135
4.8.3 ANÁLISIS DE RESULTADOS	137
4.9 PARA UNA LÍNEA DE TRANSMISIÓN CORTA, TOMA DE CARGA DE 0 – 1 P.U.	138
4.9.1 CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES	138
4.9.2 CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS	140
4.9.3 ANÁLISIS DE RESULTADOS	141
4.10 PRUEBA DE TOMA DE CARGA LÍNEA LARGA DE 0 – 0.7 P.U.	142
4.10.1 CONTROLADORES CON FUNCIONES DE MEMBRESÍA TRIANGULARES	142
4.10.2 CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS	144
4.10.1 ANÁLISIS DE RESULTADOS	145
CAPÍTULO 5	146
<u>5. CONCLUSIONES Y RECOMENDACIONES</u>	<u>146</u>
5.1 CONCLUSIONES GENERALES	146
5.2 APORTACIONES	147
5.3 TRABAJOS FUTUROS	148
REFERENCIAS	149
APÉNDICE A	152
<u>A.1 ÍNDICES DE FUNCIONAMIENTO</u>	<u>152</u>

A.1.1 INTRODUCCIÓN A LOS ÍNDICES DE FUNCIONAMIENTO	152
A 1.2 ÍNDICES ISE, IAE E ITAE	152
APÉNDICE B	154
<u>B.1 MODELOS MATEMÁTICOS</u>	<u>154</u>
B 1.1 INTRODUCCIÓN	154
B 1.2 MODELO DE SÉPTIMO ORDEN EN VARIABLES DE PARK	154
B 1.3 MODELO DE 5TO ORDEN EN VARIABLES DE PARK	155
B 1.4 MODELO DE 5TO ORDEN EN EL SISTEMA NO RECÍPROCO	156
B 1.5 MODELO DE LA RED	162
B 1.6 CONCEPTO MÁQUINA MODIFICADA	163
B 1.7 MODELO DEL SISTEMA DE EXCITACIÓN	163
B 1.8 MODELO DEL REGULADOR DE VOLTAJE	163
B 1.9 MODELO DEL GOBERNADOR Y TURBINA	164
B 1.10 MODELO COMPLETO DE 5TO ORDEN DE LA MÁQUINA SÍNCRONA	165
B 1.11 DETERMINACIÓN DE LAS CONDICIONES INICIALES DEL SISTEMA MÁQUINA SÍNCRONA BUS INFINITO	167
B 1.12 CONDICIONES INICIALES DE LA RED	167
B 1.13 CONDICIONES INICIALES DEL GENERADOR	167
B 1.14 CONDICIONES INICIALES DEL SISTEMA DE EXCITACIÓN Y REGULADOR DE VOLTAJE	168
B 1.15 CONDICIONES INICIALES DEL GOBERNADOR DE VELOCIDAD Y TURBINA DE VAPOR	169
B 1.16 DATOS DEL SISTEMA	169
APÉNDICE C	170
<u>C. 1 ALGORITMO KM Y ALGORITMO EKM</u>	<u>170</u>
APÉNDICE D	173
<u>D.1 BÚSQUEDA POR TABÚ</u>	<u>173</u>
D 1.1 INTRODUCCIÓN	173
D 1.2 ESTRUCTURA DE BÚSQUEDA POR TABÚ	173
D 1.2.1 MEMORIA A CORTO PLAZO	174
D 1.2.2 MEMORIA A LARGO PLAZO	175
D 1.3 PRUEBA LÍNEA CORTA CON AUTOSINTONIZACIÓN POR BÚSQUEDA TABÚ	176
D 1.3.1 CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS, 25 REGLAS DE CONOCIMIENTO, Y AUTOZINTONIZACIÓN POR BÚSQUEDA TABÚ CON CRITERIO DE ALTA INFLUENCIA	177
D 1.3.2 CONTROLADORES CON FUNCIONES DE MEMBRESÍA GAUSSIANAS, 25 REGLAS DE CONOCIMIENTO, Y AUTOZINTONIZACIÓN POR BÚSQUEDA TABÚ CON CRITERIO DE BAJA INFLUENCIA	179
D 1.3.5 ANÁLISIS DE RESULTADOS	181

APÉNDICE E	182
<u>E.1 CONTROL DIFUSO TIPO-1 ENTRE KL25Z Y SIMULINK MEDIANTE COMUNICACIÓN SERIAL</u>	<u>182</u>
E 1.1 INTRODUCCIÓN	182
E 1.2 COMUNICACIÓN SERIAL	182
E 1.3 MODELO EN SIMULINK	183
E 1.4 IMPLEMENTACIÓN CON KINETIS FRDM-KL25Z Y CODEWARRIOR 10.4	184
E 1.5 SIMULACIONES	189
E 1.6 CÓDIGO DEL PROGRAMA	192
APÉNDICE F	198
<u>F 1 CÓDIGOS DE PROGRAMACIÓN</u>	<u>198</u>
F 1.1 INTRODUCCIÓN	198
F 1.2 DATOS DE SIMULACIÓN	198
F 1.3 PROGRAMA PRINCIPAL	202
F 1.4 FUNCIONES SECUNDARIAS	206
F 1.5 FUNCIONES PARA LOS CONTROLADORES DIFUSOS	210
F 1.6 FUNCIONES AUXILIARES PARA LOS CONTROLADORES DIFUSOS	215
F 1.7 FUNCIONES PARA BÚSQUEDA POR TABÚ	216

ÍNDICE DE FIGURAS

Figura 2.1 Conjuntos: a) clásicos; b) difusos tipo-1; c) difusos tipo 2.	10
Figura 2.2 Conjunto Difuso tipo-2, representado a partir de la incertidumbre en σ para una función de membresía gaussiana de un conjunto difuso tipo-1.	11
Figura 2.3 Conjunto Difuso tipo-2, representado a partir de la incertidumbre en m para una función de membresía gaussiana de un conjunto difuso tipo-1.	12
Figura 2.4 Vista de una función de membresía en un espacio de dimensión tres para un conjunto difuso tipo-2.	13
Figura 2.5 Huella de incertidumbre de un conjunto difuso general tipo-2.	13
Figura 2.6 Sistemas lógicos difusos: (a) singleton tipo-1; (b) no singleton tipo-1; (c) singleton tipo-2 en intervalos; (d) tipo-1 no singleton tipo-2 en intervalos; (e) tipo-2 en intervalos no singleton tipo-2 en intervalos. Tomado de [3]	14
Figura 2.7 Función de membresía para un: (a) conjunto difuso tipo 1; (b) conjunto difuso tipo 2.	15
Figura 2.8 Descripción de un conjunto difuso tipo 2.	17
Figura 2.9 Conjunto difuso general tipo-2, a partir de muestras discretas para la variable primaria.	19
Figura 2.10 Conjuntos general discreto difuso tipo-2: a) en universo de discurso discreto, b) embebido tipo-2. Tomado de [25].	19
Figura 2.11 Conjunto difuso discreto tipo-2 en intervalos: a) en un iniverso de discurso discreto; b) embebido tipo-2 en intervalos. Tomado de [26].	20
Figura 2.12 Sistema de inferencia para una simple regla difusa.	21
Figura 2.13 Sistema de inferencia para una simple regla difusa, con conjuntos antecedentes y consecuentes tipo-2.	22
Figura 2.14 Sistema de inferencia para dos reglas difusas, con conjuntos consecuentes tipo 2.	23
Figura 3.1 Sistema regulador de voltaje y gobernador de un generador. Tomado de [43].	28
Figura 3.2 Diagrama de bloques del controlador difuso como regulador de voltaje.	29
Figura 3.3 Controlador difuso tipo PI	29
Figura 3.4 Estructura general de un controlador difuso tipo-1 convencional.	30
Figura 3.5 Estructura general de un controlador difuso tipo-2 en intervalos.	31
Figura 3.6 Etiquetas del universo de entrada con tres conjuntos difusos tipo-1.	31
Figura 3.7 Etiquetas del universo de entrada con cinco conjuntos difusos tipo-1.	32
Figura 3.8 Etiquetas del universo de entrada con siete conjuntos difusos tipo-1.	32
Figura 3.9 Etiquetas del universo de entrada con siete conjuntos difusos triangulares tipo-2.	34
Figura 3.10 Etiquetas del universo de entrada con cinco conjuntos difusos triangulares tipo-2.	34
Figura 3.11 Etiquetas del universo de entrada con tres conjuntos difusos triangulares tipo-2.	34
Figura 3.12 Parametrización del universo de discurso con conjuntos difusos simétricos: a) universo del error con tres funciones de membresía; b) universo del cambio del error con tres funciones de membresía, y c) Universo de salida.	36
Figura 3.13 Parametrización g eneral de un universo de entrada.	36
Figura 3.14 Universo de discurso con funciones de membresía triangulares parametrizado en $A = 10$, y $N = 5$.	37
Figura 3.15 Universo de discurso con funciones de membresía Gaussianas parametrizado en $A = 10$, $d_i = 5$, $\sigma = 5/3$ y $N = 5$.	37
Figura 3.16 Funciones de membresía para diferentes conjuntos tipo-2 en intervalos.	39
Figura 3.17 Universo de salida con siete funciones tipo barra.	40
Figura 3.18 Incertidumbre en barra de salida.	40
Figura 3.19 Superficie de control del T1 FLC con conjuntos difusos de entrada simétricos y matriz de conocimiento difuso simétrica..	40
Figura 3.20 Supercie de control para los parámetros en Figura 3.6, Figura 3.21 y Tabla 3.4 .	41
Figura 3.21 Universo de salida no uniforme para la Figura 3.20.	41
Figura 3.22 Diagrama de bloques del sistema de control con fuentes de incertidumbre.	42
Figura 3.23 Voltaje en terminales ideal y con incertidumbre.	42
Figura 3.24 Universo de entrada del error.	43

Figura 3.25 Universo de entrada del cambio de error.	44
Figura 3.26 Datos de simulación requeridos por los controladores: (a) difuso tipo-1, (b) difuso tipo-2.	47
Figura 3.27 Comportamiento del voltaje terminal de la maquina síncrona con controladores difusos con parámetros idénticos y $\delta e = \delta \Delta e = 0$.	48
Figura 3.28 Comportamiento del voltaje de excitación de la maquina síncrona con controladores difusos con parámetros idénticos y $\delta e = \delta \Delta e = 0$.	48
Figura 3.29 Comportamiento del voltaje de campo de la maquina síncrona con controladores difusos con parámetros idénticos y $\delta e = \delta \Delta e = 0$.	48
Figura 3.30 Diagrama de flujo del proceso de ajuste de incertidumbre en conjuntos difusos tipo-2.	51
Figura 3.31 Matriz de asociación difusa.	52
Figura 4.1 Expansión y contracción de los universos de entrada: a) universo del error; b) universo del cambio del error.	55
Figura 4.2 Evolución del diagrama de fases para línea corta: a) $A = 0.5, B = 0.01$; b) $A = 0.5, B = 0.02$; c) $A = 0.5, B = 0.005$; d) $A = 0.5, B = 0.003$; e) $A = 0.7, B = 0.001$; f) $A = 0.3, B = 0.003$.	56
Figura 4.3 Evolución del voltaje terminal para línea corta: a) $A = 0.5, B = 0.01$; b) $A = 0.5, B = 0.02$; c) $A = 0.5, B = 0.005$; d) $A = 0.5, B = 0.003$; e) $A = 0.7, B = 0.001$; f) $A = 0.3, B = 0.003$.	57
Figura 4.4 Superficies de control de para un control tipo-1 con: a) funciones de membresía triangulares $A = 0.3, B = 0.003, C = 3$; b) funciones de membresía Gaussianas $A = 0.5, B = 0.01, C = 3$.	58
Figura 4.5 Superficies de control de para un control tipo-1 con: a) funciones de membresía triangulares $A = 0.5, B = 0.01, C = 3$; b) funciones de membresía Gaussianas $A = 0.3, B = 0.003, C = 3$.	59
Figura 4.6 Respuesta del voltaje terminal ante una falla de corto circuito de 10 ciclos de duración: a) y b) Universos de entrada con funciones de membresía triangulares; c) y d) universos de entrada con funciones de membresía Gaussianas; e) diagrama de fases, funciones de membresía triangulares; f) diagrama de fases, funciones de membresía Gaussianas.	59
Figura 4.7 Respuesta del voltaje terminal al modificar el número de barras de salida del controlador tipo-1 con: a) universos de entrada con funciones de membresía triangulares; b) universos de entrada con funciones de membresía Gaussianas.	61
Figura 4.8 Funciones de membresía triangulares 1 tipo-2, para una línea de transmisión corta.	62
Figura 4.9 Respuesta del sistema utilizando las funciones de membresía triangulares 1 tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.	63
Figura 4.10 Funciones de membresía triangulares 2 tipo-2, para una línea de transmisión corta.	64
Figura 4.11 Respuesta del sistema utilizando las funciones de membresía triangulares 2 tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.	65
Figura 4.12 Funciones de membresía triangulares 3 tipo-2, para una línea de transmisión corta.	66
Figura 4.13 Respuesta del sistema utilizando las funciones de membresía triangulares 3 tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.	67
Figura 4.14 Funciones de membresía triangulares 4 tipo-2, para una línea de transmisión corta.	68
Figura 4.15 Respuesta del sistema utilizando las funciones de membresía triangulares 4 tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.	69
Figura 4.16 Funciones de membresía Gaussianas tipo-2, para una línea de transmisión corta.	70
Figura 4.17 Respuesta del sistema utilizando las funciones de membresía Gaussianas tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.	71
Figura 4.18 Superficie de control resultante del IT2 FLC con los parámetros según Tabla 4.18 y 9 reglas de conocimiento difuso.	73
Figura 4.19 Superficie de control resultante del IT2 FLC con los parámetros según Tabla 4.18 y 25 reglas de conocimiento difuso.	73
Figura 4.20 Superficie de control resultante del IT2 FLC con los parámetros según Tabla 4.18 y 49 reglas de conocimiento difuso.	73
Figura 4.21 Expansión y contracción de los universos de entrada: a) Universo del error; b) Universo del cambio del error.	74
Figura 4.22 Evolución del diagrama de fases para línea corta: a) $A = 0.5, B = 0.01$; b) $A = 0.5, B = 0.005$; c) $A = 0.7, B = 0.005$; d) $A = 1.0, B = 0.005$; e) $A = 0.3, B = 0.005$; f) $A = 0.3, B = 0.003$.	75

Figura 4.23 Evolución del voltaje terminal para línea corta: a) $A = 0.5, B = 0.01$; b) $A = 0.5, B = 0.005$; c) $A = 0.7, B = 0.005$; d) $A = 1.0, B = 0.005$; e) $A = 0.3, B = 0.005$; f) $A = 0.3, B = 0.003$.	76
Figura 4.24 Respuesta del voltaje terminal al variar el parámetro C en el universo de salida del T1 FLC con : a) universos de entrada con funciones de membresía triangulares; b) universos de entrada con funciones de membresía Gaussianas.	77
Figura 4.25 Respuesta del voltaje terminal al modificar el número de barras de salida del controlador tipo-1 con: a) universos de entrada con funciones de membresía triangulares; b) universos de entrada con funciones de membresía Gaussianas.	78
Figura 4.26 Funciones de membresía triangulares 1 tipo-2, para una línea de transmisión corta.	79
Figura 4.27 Funciones de membresía triangulares 2 tipo-2, para una línea de transmisión corta.	80
Figura 4.28 Funciones de membresía triangulares 3 tipo-2, para una línea de transmisión corta.	81
Figura 4.29 Funciones de membresía triangulares 4 tipo-2, para una línea de transmisión corta.	82
Figura 4.30 Funciones de membresía Gaussianas tipo-2, para una línea de transmisión corta.	83
Figura 4.31 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.	85
Figura 4.32 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.	86
Figura 4.33 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.	88
Figura 4.34 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.	89
Figura 4.35 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.	91
Figura 4.36 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.	92
Figura 4.37 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.	94
Figura 4.38 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.	96
Figura 4.39 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.	98
Figura 4.40 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.	99
Figura 4.41 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.	101
Figura 4.42 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.	102
Figura 4.43 Respuesta del sistema sujeto a corto circuito con 11 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.	104
Figura 4.44 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 9 reglas de conocimiento y $\pm 3\%$ de incertidumbre en la lectura del voltaje terminal.	106
Figura 4.45 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 9 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.	107
Figura 4.46 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.	109
Figura 4.47 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 49 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.	110
Figura 4.48 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 9 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.	112

Figura 4.49 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 25 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.	113
Figura 4.50 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 49 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.	115
Figura 4.51 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.	117
Figura 4.52 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.	118
Figura 4.53 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.	120
Figura 4.54 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.	121
Figura 4.55 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.	122
Figura 4.56 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.	123
Figura 4.57 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.	124
Figura 4.58 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.	125
Figura 4.59 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.	126
Figura 4.60 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.	127
Figura 4.61 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.	128
Figura 4.62 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.	129
Figura 4.63 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.	131
Figura 4.64 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento e incertidumbre de $\pm 2\%$ en el valor del voltaje de referencia.	134
Figura 4.65 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía triangulares, 9 reglas de conocimiento e incertidumbre de $\pm 2\%$ en el valor del voltaje de referencia.	135
Figura 4.66 Respuesta del sistema sujeto a cambios de carga; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.	138
Figura 4.67 Respuesta del sistema sujeto a cambios de carga; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.	140
Figura 4.68 Respuesta del sistema sujeto a cambios de carga; caso línea larga; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.	142
Figura 4.69 Respuesta del sistema sujeto a cambios de carga; caso línea larga; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.	144
Figura B.1 Sistema máquina bus infinito.	154
Figura B.2 Diagrama de bloques de la ecuación de oscilación de la máquina síncrona.	162
Figura B.3 Diagrama de bloques del regulador tipo ST1.	164
Figura B.4 Diagrama de bloques del gobernador de velocidad.	164
Figura B.5 Función de transferencia en lazo abierto del gobernador de velocidad.	164
Figura B.6 Diagrama vectorial. Tomado de [9].	168
Figura C.1 Iteraciones ocupadas por KM y EKM ante distintos casos: a) funciones de membresía triangulares tipo-2 y 9 reglas; b)) funciones de membresía Gaussianas tipo-2 y 9 reglas; c)) funciones de membresía triangulares tipo-2 y 25 reglas; d)) funciones de membresía Gaussianas tipo-2 y 25 reglas.	170

Figura D.1 Respuesta del sistema sujeto a falla trifásica en el bus del generador con duración de 10 ciclos; controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento y autosintonización por búsqueda tabú bajo el criterio de alta influencia.	177
Figura D.2 Respuesta del sistema sujeto a falla trifásica en el bus del generador con duración de 10 ciclos; controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento y autosintonización por búsqueda tabú bajo el criterio de baja influencia.	179
Figura E.1 Comunicación serial asíncrona.	183
Figura E.2 Bloques para comunicación serial.	183
Figura E.3 Configuración de bloques de envío y recepción en Simulink: a) configuración del hardware del bloque de envío; b) configuración del hardware del bloque de recepción; c) configuración del comando de consulta del bloque de recepción.	183
Figura E.4 Variable Ts en espacio de trabajo.	184
Figura E.5 Lazo de control para el modelo de un motor DC.	184
Figura E.6 Subsistema del modelo para un motor DC.	184
Figura E.7 Ubicación de pines PTA1 y PTA2 en conectores del Kinetis KL25Z.	185
Figura E.8 Arreglos de tarea 3. (a) variable del error, (b) variable del cambio del error.	186
Figura E.9 Arreglos: a) normT_min de tarea 4, b) Tnorm de la tarea 5.	186
Figura E.10 Diagrama de secuencia del algoritmo de control implementado en el microcontrolador.	188
Figura E.11 Parámetros de ajuste de conjuntos difusos de las variables de entrada.	188
Figura E.12 Respuesta del modelo con control difuso con Ts = 1/64 seg.	189
Figura E.13 Diagrama de fase, Ts = 1/64 seg.	189
Figura E.14 Respuesta del modelo con Ts=1/128 seg (0.0078 seg).	190
Figura E.15 Respuesta del modelo con Ts = 1/256 seg. (0.0039 seg).	190
Figura E.16 Respuesta final en el tiempo con Ts = 1/256 (0.0039 seg).	191
Figura E.17 Respuesta en el tiempo a una perturbación de 2 N.m, de carga mecánica.	191

ÍNDICE DE TABLAS

Tabla 2.1 Algoritmo KM para y	26
Tabla 2.2 Algoritmo KM para y	26
Tabla 2.3 Algoritmo EKM para y	27
Tabla 2.4 Algoritmo EKM para y	27
Tabla 3.1 Matriz de 9 reglas difusas	33
Tabla 3.2 Matriz de 9 reglas difusas	33
Tabla 3.3 Matriz de 9 reglas difusas	33
Tabla 3.4 Matriz de conocimiento difuso para la superficie de control de la Figura 3.20	41
Tabla 3.5 Parámetros del Controlador tipo-2.	43
Tabla 3.6 Inferencia del controlador tipo-2, operador mínimo.	44
Tabla 3.7 Inferencia del controlador tipo-2, operador máximo.	44
Tabla 3.8 Parámetros de Controladores	47
Tabla 3.9 Salida de Controladores	49
Tabla 4.1 Datos de líneas de transmisión.	53
Tabla 4.2 Índices para sintonización del T1 FLC, línea corta.	58
Tabla 4.3 Índices del error ante diversos valores para C .	59
Tabla 4.4 Índices del error ante diferentes números de barras de salida.	61
Tabla 4.5 Parámetros para el controlador Tipo-1.	62
Tabla 4.6 Índices al añadir incertidumbre en los conjuntos triangulares 1 tipo-2, caso línea corta.	62
Tabla 4.7 Índices al añadir incertidumbre en las barras de salida, caso triangular 1 y línea corta.	62
Tabla 4.8 Índices al añadir incertidumbre en los conjuntos triangulares 2 tipo-2, caso línea corta.	64
Tabla 4.9 Índices al añadir incertidumbre en las barras de salida, caso triangular 2 y línea corta.	64
Tabla 4.10 Índices al añadir incertidumbre en los conjuntos triangulares 3 tipo-2. Caso línea corta	66
Tabla 4.11 Índices al añadir incertidumbre en las barras de salida, caso triangular 3 y línea corta.	66
Tabla 4.12 Índices al añadir incertidumbre en los conjuntos triangulares 4 tipo-2. Caso línea corta.	68
Tabla 4.13 índices al añadir incertidumbre en las barras de salida, caso triangular 4 y línea corta.	68
Tabla 4.14 Índices al añadir incertidumbre en los conjuntos Gaussianos tipo-2. Caso línea corta.	70
Tabla 4.15 Índices al añadir incertidumbre en las barras de salida. Caso Gaussianas y línea corta.	70
Tabla 4.16 Resumen mejores parámetros, caso línea corta.	72
Tabla 4.17 Resumen de mejores índices según Tabla 4.16, caso línea corta.	72
Tabla 4.18 Parámetros finales para el IT2 FLC, caso línea corta.	72
Tabla 4.19 Resumen de índices para la sintonización del T1 FLC, caso línea larga.	77
Tabla 4.20 Índices del error ante distintos valores del parámetro C , caso línea larga.	77
Tabla 4.21 Índices ante diferentes números de barras de salida, caso línea larga.	78
Tabla 4.22 Parámetros del T1 FLC, caso línea larga.	79
Tabla 4.23 Índices al añadir incertidumbre en los conjuntos triangulares 1 tipo-2, caso línea larga.	79
Tabla 4.24 Índices al añadir incertidumbre en las barras de salida, caso triangular 1 y línea larga.	79
Tabla 4.25 Índices al añadir incertidumbre en los conjuntos triangulares 2 tipo-2, caso línea larga.	80
Tabla 4.26 Índices al añadir incertidumbre en las barras de salida, caso triangular 3 y línea larga.	80
Tabla 4.27 Índices al añadir incertidumbre en los conjuntos triangulares 3 tipo-2, caso línea larga.	81
Tabla 4.28 Índices al añadir incertidumbre en las barras de salida, caso triangular 3 y línea larga.	81
Tabla 4.29 Índices al añadir incertidumbre en los conjuntos triangulares 4 tipo-2, caso línea larga.	82
Tabla 4.30 Índices al añadir incertidumbre en las barras de salida, caso Gaussianas y línea larga.	82
Tabla 4.31 Índices al añadir incertidumbre en los conjuntos Gaussianos tipo-2, caso línea larga.	83
Tabla 4.32 Índices al añadir incertidumbre en las barras de salida, caso Gaussianas y línea larga.	83
Tabla 4.33 Resumen de mejores parámetros del IT2 FLC, caso línea larga.	84
Tabla 4.34 Mejores índices según Tabla 4.33.	84
Tabla 4.35 Parámetros para el controlador tipo-2, caso línea larga.	84
Tabla 4.36 Índices de comportamiento dinámico de detalle 1 voltaje terminal Figura 4.62.	130
Tabla 4.37 Índices de comportamiento dinámico de detalle 1 voltaje terminal Figura 4.62.	132
Tabla B.1 Datos del Sistema bajo prueba.	169
Tabla C.1 Comparación del desempeño de algoritmos reducción de tipo para el IT2 FLC.	172

Tabla D.1 Parámetros para la búsqueda Tabú.	177
Tabla D.2 Matriz de 9 reglas difusas inicial para el T1FLC y el IT2 FLC, criterio baja y alta influencia.	180
Tabla D.3 Matriz de 9 reglas difusas final para el T1FLC a los 2.5s de simulación, criterio baja influencia	180
Tabla D.4 Matriz de 9 reglas difusas final para IT2 FLC a los 2.5s de simulación, criterio baja influencia	180
Tabla D.5 Matriz de 9 reglas difusas final para el T1FLC a los 2.5s de simulación, criterio baja influencia	180
Tabla D.6 Matriz de 9 reglas difusas final para el T1FLC a los 2.5s de simulación, criterio baja influencia	180
Tabla E.1 Eventos utilizados en el programa.	185
Tabla E.2 Eventos utilizados en el programa.	185
Tabla E.3 Matriz de conocimiento difuso para la superficie de control de la Figura 3.20	188

GLOSARIO DE TÉRMINOS

Símbolo	Descripción
A	Parámetro del universe del error.
B	Parámetro de universe del cambio del error.
C	Parametro del universe de salida.
\tilde{A}	Conjunto difuso tipo-1.
\tilde{A}_G	Conjunto general difuso tipo-2.
\tilde{A}_I	Conjunto difuso tipo-2 en intervalos.
IT2- FLC	Controlador difuso tipo-2 en intervalos.
T1 FLC	Controlador convencional tipo 1.
ISE	Integral del cuadrado del error.
IAE	Integral del valor absolute del error.
$ITAE$	Integral de tiempo por el valor absolute del error.
ST1	Regulador proporcional tipo ST1.
e	Error.
Δe	Cambio del error.
u	Entrada de control en la excitatriz.
\dot{u}	Salida incremental del controlador.
e''_d	Voltaje subtransitorio en el eje d.
e'_d	Voltaje transitorio en el eje q.
e''_q	Voltaje subtransitorio en el eje q.
KM	Algoritmo de Karnik & Mendel
EKM	Algoritmo de Mejorado Karnik & Mendel
F_{HP}	Fracción de la potencia total generada por la turbina para la sección de alta presión..
FOU	Huella de incertidumbre.
i_d	Corriente en el eje d.
i_q	Corriente en el eje q.
K_A	Ganancia del regulador.
K_d	Constante de amortiguamiento en e eje d.
K_f	Ganancia del circuito estabilizador del regulador.
M	Constante de inercia.
MW	Mega Watts.
$MVar$	Mega Volt-Ampere reactive.
P_A	Potencia aparente.
P_{GV}	Potencia en la compuerta de salida en la válvula del gobernador.
P_{HP}	Potencia de alta presión.
P_R	Potencia reactiva.
$P.U$	Por unidad.
r_a	Resistencia de armadura.
r_e	Resistecia externa de la línea de transmission.
s	Deslizamiento del rotor.
seg	Segundo.
SL	Desplazamiento a la izquierda de C .
SR	Desplazamiento a la derecha de C .
T_A	Constante de tiempo del amplificador del regulador.
T_{CH}	Constante de tiempo de volúmenes en la entrada principal y vasija de vapor.
T_e	Par eléctrico.
T_f	Constante de tiempo del circuito estabilizador del regulador.
T_m	Par mecánico.

T_{SM}	Constante de tiempo de recalentamiento.
T_x	Constante de tiempo en la excitatriz.
T'_{d0}	Constante de tiempo transitoria a circuito abierto en el eje d.
T''_{d0}	Constante de tiempo subtransitoria a circuito abierto en el eje d.
T''_{q0}	Constante de tiempo subtransitoria a circuito abierto en el eje q.
U_g	Entrada de control del gobernador.
V_{bus}	Voltaje en el bus infinito.
V_d	Voltaje en el eje d.
V_{EXC}	Voltaje del circuito estabilizador del regulador.
V_f	Voltaje de campo.
V_q	Voltaje en el eje q.
V_{ref}	Voltaje de referencia.
V_{ter}	Voltaje terminal.
ω_0	Velocidad síncrona.
X_d	Reactancia síncrona en el eje d.
X_q	Reactancia síncrona en el eje q.
X'_d	Reactancia transitoria en el eje d.
X''_d	Reactancia subtransitoria en el eje d.
X''_q	Reactancia subtransitoria en el eje q.
X_e	Impedancia externa de la línea de transmisión.
δ_e	Incertidumbre en los conjuntos difusos tipo-2 en el universe del error.
$\delta_{\Delta e}$	Incertidumbre en los conjuntos difusos tipo-2 en el universe del cambio del error.
δ_r	Ángulo de carga.
θ	Ángulo del voltaje terminal.
$\mu_A(x)$	Función de membresía de conjunto difuso tipo-1.
$\mu_A(x')$	Grado de membresía para $x = x'$.
$\underline{\mu}_{\bar{A}}(x)$	Función de membresía inferior.
$\overline{\mu}_{\bar{A}}(x)$	Función de membresía superior.
$\underline{\mu}_{\bar{A}}(x')$	Grado de membresía inferior para $x = x'$.
$\overline{\mu}_{\bar{A}}(x')$	Grado de membresía superior para $x = x'$.
$\mu_{\bar{A}}(x, u)$	Función de membresía secundaria.
$\mu_{\bar{A}}(x', u)$	Grado de membresía secundario para $x = x'$.
J_{x_i}	Intervalo Intervalo $[\underline{\mu}_{\bar{A}_i}(x') \ \overline{\mu}_{\bar{A}_i}(x')] con i = 1, \dots, N$.

CAPÍTULO 1

El hombre prudente oculta su ciencia, el corazón del necio pregonera su ignorancia. Pr. 12,23.

1. INTRODUCCIÓN

1.1 Generalidades

El objetivo fundamental del control de excitación de un generador síncrono, es realizar el ajuste en forma automática del voltaje de campo, y de esta forma, la cantidad de corriente al devanado de campo del generador; de modo que se mantengan los niveles de voltaje en las terminales del generador en los valores deseados. Además el control de excitación debe ser capaz de responder ante disturbios que pudieran presentarse en el sistema, sin alterar el funcionamiento normal del generador [1].

Hoy en día, se cuenta con diferentes tipos de controladores que utilizan técnicas de control robustas. Dichos sistemas de control miden y evalúan: las variables de interés y las condiciones de funcionamiento del sistema, y de esta forma, determinan las acciones de control adecuadas. Una práctica alternativa dentro del control inteligente que puede ser utilizada para este fin es la lógica difusa. La cual satisface el diseño de sistemas sofisticados de control, ya que con la información almacenada en sus reglas difusas de control definidas mediante información heurística, es capaz de hacer frente a los sistemas con escenarios de incertidumbre e imprecisiones, mediante algoritmos que imitan el pensamiento, percepción y conocimiento humano, para hallar soluciones a problemas y situaciones particulares.

No obstante en algunos sistemas, el conocimiento que se tiene para construir las reglas de control en un sistema lógico difuso es incierto. Esto conduce a incertidumbres en los antecedentes y consecuentes de las reglas difusas. Con lo que los tradicionales conjuntos difusos, - los llamados tipo 1 - no pueden manejar directamente estas incertidumbres. Una nueva clase de conjuntos lógicos difusos de tipo 2 fueron introducidos por Lofti A. Zadeh en 1975 [2], como una extensión de los ordinarios conjuntos difusos. Con lo que dio lugar a sistemas lógicos difusos tipo-2 [3], los cuales pueden manejar en mejor forma las incertidumbres presentes, debido a que sus grados de pertenencia son así mismos conjuntos difusos de tipo 1.

En el presente trabajo se desarrolla un controlador difuso tipo 2 que utiliza, conjuntos difusos de tipo 2 y se aplica como regulador de voltaje para el control de excitación de una máquina síncrona.

1.2 Objetivo

Diseñar un control de excitación para un generador síncrono, utilizando un controlador lógico difusos tipo-2, el cual utiliza conjuntos difusos tipo-2 en intervalos.

Analizar las ventajas con respecto a los conjuntos difusos de 1er tipo, evaluar sus características; así como desarrollar una metodología de sintonización.

1.3 Motivación

Incertidumbre en los sistemas eléctricos de potencia

Los sistemas eléctricos de potencia son extensos y complejos sistemas dinámicos del tipo electromecánico [4]. Estos sistemas operan en forma continua y se ven influenciados por eventos inesperados.

Existen muchos conceptos imprecisos e inciertos asociados a los sistemas eléctricos de potencia [5]. Por ejemplo si se habla de la estabilidad absoluta en un sistema eléctrico de potencia, se puede decir que el sistema es estable o inestable y en este caso no existe incertidumbre en el concepto estabilidad. Sin embargo si nos referimos a estabilidad relativa, se debe definir el grado de estabilidad usando términos como: ligeramente amortiguado; altamente amortiguado; sobre amortiguado; etcétera. Si el sistema es inestable pudiera referirse a: inestabilidad a pequeños disturbios; inestabilidad dinámica; inestabilidad transitoria; inestabilidad en el voltaje. Es obvio que el concepto de inestabilidad relativa es difuso y puede tratarse como una variable lingüística y describirse mediante funciones de membresía.

En cuestiones de operación en los sistemas eléctricos de potencia, existe incertidumbre debido a un gran número de causas [4][5], a manera de ejemplo tenemos:

1. Cambios en las condiciones de operación del sistema de potencia: carga, generación, y topología.
2. Falta de exactitud en los valores de los parámetros del sistema utilizados en el modelo del sistema, así como aproximaciones y simplificaciones hechas a la estructura del modelo, utilizado para fines de planeación y operación del sistema.
3. Información incompleta acerca del estado del sistema de potencia.
4. Variación en las mediciones realizadas por, transductores de voltaje y corriente; adquisición por sistemas SCADA¹, y ruido por interferencia electromagnética.
5. Inexactitud en los pronósticos de carga.
6. Eventos impredecibles, tales como, fallas en equipos, y condiciones climáticas adversas.
7. Variabilidad en aplicaciones de operación, control y planeación al sistema, debido a la intervención humana.

Pero ante esto surge la pregunta: ¿Qué es la incertidumbre?.

Una de las mejores fuentes hechas acerca de la incertidumbre en general fue hecha por Klir y Wierman [6]. En este se afirma: "Cuando se trata con problemas del mundo real, raramente se puede evitar tratar con incertidumbre. A niveles empíricos, la incertidumbre es un inseparable compañero de la mayoría de las mediciones, y es el resultado de

¹ SCADA es el acrónimo de Supervisory Control And Data Acquisition (Supervisión, Control y Adquisición de Datos). Un sistema SCADA es un sistema basado en computadoras que permite supervisar y controlar a distancia un sistema físico en forma remota.

inevitables errores en la medición y límites en la resolución de los equipos de medición. A niveles cognitivos, es debido a la vaguedad y ambigüedad inherente en el lenguaje natural. A nivel social, la incertidumbre ha sido estrategia utilizada y a menudo creada por personas para diferentes propósitos (privacidad, propiedad, etc).”

Estas razones justifican y motivan el uso de controladores difusos, que mediante conocimiento almacenado, permiten desempeñarse como sistemas expertos en el control de los sistemas eléctricos de potencia.

Incertidumbre en los sistemas lógicos difusos

Los sistemas difusos tipo-1 permiten diseñar en forma amigable, algoritmos de control de la mano con el pensamiento y percepción humana. Estos controles son del tipo no lineales y permiten cierta incertidumbre en sus variables lingüísticas de entrada. Sin embargo una vez definidas las funciones de membresía de sus conjuntos difusos toda incertidumbre en sus variables de entrada desaparece [3]. Por su parte los controladores difusos tipo-2 permiten mantener cierta incertidumbre en sus conjuntos difusos descritos por funciones de membresía con huella de incertidumbre. Este hecho permite al controlador tipo-2 un nuevo grado de libertad en el manejo de la incertidumbre asociada a sus variables de entrada por encima de los controladores tipo-1 [3].

Además siempre existe incertidumbre presente en diferentes formas en los sistemas difusos convencionales. Existen al menos cuatro fuentes de incertidumbre presentes en estos sistemas [3]:

- Incertidumbre asociada a las palabras utilizadas en las reglas.
- Incertidumbre asociada al consecuente para una regla en particular.
- Incertidumbre asociada en las mediciones de las variables físicas de entrada al sistema difuso.
- Incertidumbre asociada en los datos utilizados para la sintonización de los parámetros del sistema difuso.

Por otra parte la incertidumbre en un sistema puede ser evaluada mediante la media y varianza. Esta última proporciona una medida de dispersión alrededor de la media y a menudo se utilizan intervalos de confianza [3]. En este sentido, puede interpretarse el procesamiento de salida de un controlador difuso tipo-2 que involucra un algoritmo de reducción de tipo, el cual proporcionará una medida de dispersión alrededor de su salida defusificada mediante intervalos de confianza lingüísticos.

Estas razones justifican investigar el uso de controladores lógicos difusos tipo-2 en el control de los sistemas eléctricos de potencia, que en los últimos años han reportado el uso de sistemas difusos tipo-2 en diferentes áreas [7].

1.4 Limitaciones y Alcances

En esta tesis se estudian las características y propiedades de los conjuntos difusos tipo-2. Se desarrollará la aplicación y sintonización de un control lógico difuso de tipo-2 en intervalos para el control de excitación de un generador síncrono. Se analiza su desempeño, limitaciones y ventajas con respecto a un controlador tipo-1 y un regulador convencional tipo ST1 en escenarios ideales y con incertidumbre.

Limitaciones

- El sistema bajo prueba es el modelo de 5to orden de un generador de 645 MVA [8] conectado a un bus infinito mediante una línea de transmisión corta o larga [8][9][12][17][18]. El paso de integración en la simulación del modelo es igual al tiempo de muestreo de los controladores difusos.
- La sintonización del controlador disuso tipo-2 es realizado a partir del controlador difuso tipo-1 [19].
- El ajuste de la incertidumbre en los conjuntos difusos tipo-2 es realizada en forma manual, no se utiliza ninguna técnica de optimización inteligente.

Alcances

- Se sujeta el sistema bajo prueba antes pequeños y grandes disturbios. Cambios en el voltaje de referencia y pruebas de toma de carga son considerados pequeños disturbio mientras que una falla de cortocircuito en el bus infinito es considerado un disturbio grande.
- Para el controlador tipo-2, se desarrollaran pruebas con 5 diferentes conjuntos difusos tipo-2 en el proceso de sintonización. Cuatro conjuntos triangulares y un conjunto tipo Gaussiano.
- Para el controlador difuso tipo-1, se realizarán pruebas con conjuntos difusos triangulares y Gaussianos.
- Se desarrollará el algoritmo convencional y mejorado de Karnik y Mendel, y se compararán resultados entre ellos.
- Se analizarán pruebas con incertidumbre en dos situaciones: incertidumbre en la lectura del voltaje terminal e incertidumbre en la señal de voltaje de referencia.
- El desempeño de los controladores difusos y el regulador ST1 es cuantificado mediante índices de desempeño del error [10] e índices de comportamiento dinámico ante pequeños disturbios [11].

1.5 Estado del Arte

En el razonamiento lógico, se utilizan únicamente los valores *uno* y *cero*, mediante los cuales se realizan inferencias, a través reglas de inferencia² que permiten pasar de *premisas* dadas (antecedentes), a una conclusión propuesta (consecuente) [13].

En 1965 Lofti A. Zadeh presentó los fundamentos de la lógica difusa [14], y propuso la idea de que las clases de los objetos en el mundo físico real no tienen un criterio preciso definido para sus miembros.

Ejemplo:

La clase de *animales* claramente incluye miembros como: perros; caballos, pájaros. Y de la misma forma excluye miembros tales como: plantas, fluidos, rocas, etc. Sin embargo miembros como: peces y bacterias, mantienen un criterio ambiguo con respecto a la clase *animales*.

Lofti A. Zadeh definió de igual forma el concepto de variable lingüística [2], como una variable en la cual los valores de la misma son a su vez palabras o sentencias en un lenguaje natural o artificial. Este concepto tiene una importante área de aplicación dentro

² Modus Tollendo Tollens, Modus Ponendo Ponens, etc. Ref [13].

del razonamiento aproximado³. Por ejemplo la variable *edad* es lingüística, desde que sus valores son: *joven, muy joven, no muy joven, viejo, no muy viejo, etc.* Una variable lingüística formalmente esta caracterizada por cinco parámetros [2], esto es $(\xi, T(\xi), U, S, N)$ en donde ξ es el nombre de la variable; $T(\xi)$ es la colección de los valores lingüísticos de la variable ξ ; U es el universo de discusión o universo de discurso; S es una regla sintáctica para los términos en $T(\xi)$; N es una regla semántica que asocia cada valor lingüístico X con su significado. $N(X)$ representa un subconjunto difuso en U . Además el significado de un valor lingüístico X es caracterizado por una función de compactibilidad, $f: U \rightarrow [0, 1]$, la cual asocia la compactibilidad de cada valor de u en U con X . De esta forma Lofti A. Zadeh propuso los conjuntos difusos, los cuales son caracterizados por una función de membresía, que permite a los miembros incluidos dentro del conjunto difuso, tener un grado de membresía o pertenencia a dicho conjunto, con un valor dentro del intervalo $[0, 1]$.

Las primeras aplicaciones exitosas de la lógica difusa en el control de procesos fueron desarrollados por Mamdani en 1974 [15][16]. Años más tarde Takagi y Sugeno propusieron una estructura con diferente descripción lingüística en los conjuntos de salida. De esta forma hay sistemas que operan mediante lógica difusa, los cuales han sido desarrollados satisfactoriamente en muchas aplicaciones, sin embargo, los conjuntos difusos de tipo-1 tienen capacidades limitadas para manejar directamente datos con incertidumbre.

En 1975 fue publicada por Lofti Zadeh, una extensión de los conceptos de los primeros conjuntos difusos [2]. Así los sistemas desarrollados con los primeros conjuntos difusos, fueron renombrados como sistema lógico difuso tipo-1 (por sus siglas en inglés "*Type-1 Fuzzy Logic System*", *T1 FLS*), mientras que estos sistemas con conjuntos difusos con un orden mayor fueron denominados sistemas lógicos difusos de tipo-2 (por sus siglas en inglés "*Type-2 Fuzzy Logic Systems*", *T2 FLS*). Estos últimos conjuntos difusos añaden una nueva dimensión en sus conjuntos, lo que implica un mayor grado de libertad, en el manejo de la incertidumbre.

Sin embargo desde 1975, nadie había extendido el uso de los sistemas lógicos difusos utilizando conjuntos difusos tipo-2. Los principales obstáculos fueron: la caracterización de funciones de membresía tipo 2; realizar operaciones con conjuntos difusos tipo 2; el sistema de inferencia al tratar con este tipo de conjuntos; así como el procesamiento de salida que implica: reducción de tipo y dedifusificación, entre otros [3]. Debido a esto, surgieron nuevos conceptos asociados a los conjuntos difusos tipo-2, los cuales resultan de extensiones de los convencionales conjuntos difusos tipo-1.

Mizumoto y Tanaka [21] analizaron las operaciones de los conjuntos difusos tipo-2 y las propiedades de los grados de membresía de tales conjuntos. Karnik y Mendel publicaron el estudio de operaciones con conjuntos difusos tipo-2, propiedades de los grados de membresía con conjuntos tipo-2 basados en los trabajos de Mizumoto y Tanaka, las relaciones tipo-2, composiciones así como el proceso de reducción de tipo y dedifusificación.

³ El razonamiento aproximado es un modo de razonamiento inexacto. Este tipo de lógica proporciona un marco más realista del razonamiento humano que los tradicionales valores lógicos [2].

Karnick, Mendel y Liang en [22][23] presentan el procesamiento de salida que involucra un sistema difuso tipo-2, mediante el proceso de reducción de tipo y dedifusificación que obtuvieron al extender los procesos tradicionales de dedifusificación en los sistemas difusos tipo-1. Muestran además, aplicaciones de un sistema difuso tipo-2 para un canal de ecualización de tiempo variable [22]. En 1999 El concepto de huella de incertidumbre que caracteriza a un conjunto intervalo difuso tipo 2 fue propuesto por primera vez por Mendel y Liang. Posteriormente en 2006, Jerry Mendel, Robert John y Liu publican en [26], conceptos con el fin de facilitar la comprensión de los conjuntos intervalos difusos tipo-2.

Trabajos internacionales de controladores tipo-1 en sistemas de excitación han sido reportados. En 1997 [29] se implementó en un DSP un control lógico difuso adaptable en línea para el control de excitación de un generador síncrono conectado a un bus infinito. La sintonización en línea permitió buena eficiencia y robustez ante diferentes condiciones de operación.

En [30] se presenta un control lógico difuso tipo-1 para el control de excitación de un generador síncrono. El control difuso utiliza como entradas la velocidad instantánea y la aceleración del error. Se realizaron simulaciones y comparaciones entre el control difuso tipo-1, un control PID y un control auto ajustable, ante disturbios de cortocircuito y variación de torque. El control difuso mejora la respuesta dinámica con respecto a los otros dos controladores.

Otra implementación de un control difuso tipo-1 como regulador de voltaje y estabilizador es presentada en [31]. Se prueba el control ante una falla severa de cortocircuito en las tres fases del generador y se comparan resultados contra un control de voltaje PI y un PSS convencional. En este caso el control difuso mejora el control y estabilidad de voltaje del generador síncrono.

En [32] utilizando el software comercial Simulink y el Fuzzy Logic Toolbox se simula un control difuso tipo-1 como regulador y estabilizador de voltaje. El modelo del generador síncrono conectado a un bus infinito se sujeta a una falla trifásica a tierra. Analizando la respuesta de la potencia activa, se observan resultados satisfactorios y una mejor amortiguación de la potencia sobre a los tradicionales PSS.

En la sección de estudios de posgrado SEPI-ESIME Zacatenco se han aplicado técnicas de control inteligente utilizando lógica difusa. En 1997 C. Hernández, presenta en su tesis de maestría que lleva por título "*Control difuso de velocidad de un motor de corriente directa*" [27], la aplicación de un sistema de control de velocidad difuso tipo Mamdani aplicado al modelo lineal y no lineal de una máquina de corriente directa operando en régimen motor.

En lo que respecta a controladores difusos tipo-1 aplicados a sistemas de excitación de generadores síncronos, en [28] R. Cortés en 1997, presenta en su tesis de doctorado "*Control de excitación difuso de un generador síncrono*", el diseño e implementación de un control difuso tipo-1 en un microcontrolador de 8 bits para el control de excitación de un generador síncrono, obteniendo resultados de control satisfactorios.

Existen también otras tesis que reportan el uso de lógica difusa tipo-1 en conjunto con otras técnicas de control inteligente para el sistema de excitación de un generador síncrono conectado a un bus infinito tales como: controladores neuro difusos [9]; control difuso con matriz de reglas difusas autosintonizable por búsqueda tabú [12], búsqueda por tabú y algoritmos genéticos aplicado a las funciones de membresía [17], entre otros.

En lo que respecta a controladores difusos tipo-2, se han reportado algunas aplicaciones específicas en el área de sistemas de potencia [7]. En [33] se presenta la

aplicación de un control difuso proporcional derivativo tipo-2 en intervalos como PSS al sistema de prueba Kundur [1], el cual consiste en un sistema de dos áreas unidas mediante dos líneas de transmisión, y cada área con dos generadores síncronos. El control tipo-2 se compara contra un control tipo-1 como PSS. Se analiza el sistema sujeto ante una falla a tierra en la mitad de una de las líneas de transmisión. Los resultados en la respuesta de la potencia, muestran mejor tiempo de estabilización y sobretiro utilizando el controlador tipo-2.

También un control difuso tipo-2 en intervalos como PSS para un sistema máquina bus infinito y un sistema multimáquinas fue hecho en [33][34]. El sistema multimáquinas de dos áreas cuenta con dos generadores por área en cada uno de los cuales se coloca el control PSS difuso tipo-2. El sistema se sujeta a fallas a tierra y pequeños cambios en el torque mecánico. El control tipo-2 muestra menor sobretiro, mejor tiempo de estabilización y menos oscilaciones en el sistema en las pruebas realizadas. Las comparaciones fueron hechas contra un PSS convencional y un control difuso tipo-1 como PSS.

Un controlador difuso Takagi Sugeno tipo-2 en intervalos fue propuesto en [35], para el control de voltaje de un sistema con dos generadores conectados a un bus infinito. El controlador tiene la habilidad para mejorar la estabilidad transitoria y regular el voltaje en forma simultánea en el sistema multimáquinas. Además se muestra el análisis matemático de estabilidad para facilitar el diseño de controladores Takagi Sugeno tipo-2.

Otras aplicaciones exitosas de control difusos que utilizan conjuntos difusos tipo-2 en intervalos han sido reportadas. En [37], se muestra un método sencillo para aplicaciones de control. Se propone que un sistema difuso tipo-2 puede ser sustituido mediante dos sistemas tipo-1. Es decir un conjunto difuso tipo-2 puede ser representado por dos funciones de membresía denominadas superior e inferior. El proceso de inferencia tipo-2, se sustituye por dos sistemas de inferencia tipo-1, y el proceso de reducción de tipo se sustituye por la defusificación de cada sistema tradicional tipo-1, promediando el resultado de ambos.

Una de las primeras aplicaciones robustas en la industria fue realizada en [38], en la cual se diseñó un controlador difuso tipo-2 en intervalos, para el control de velocidad de un motor diésel marino. La principal dificultad radicó en la precisión y robustez que el control del gobernador requiere para evitar daños a la máquina diésel, y variaciones en la frecuencia del voltaje de salida. Los resultados en el desempeño de la velocidad también fueron satisfactorios. Otros ejemplos son: controlador difuso tipo-2 para el control de nivel de un proceso con dos contenedores de agua interconectados [19]; control de velocidad difuso PI tipo-2 para un motor CD de baja capacidad [36]; controlador tipo-2 en intervalos para el control de robots móviles [38]; controlador tipo-2 en intervalos para el control de altitud de un aeroplano [40]; entre otros.

A la fecha no existe trabajo realizado en SEPI-ESIME Zacatenco, con respecto a un controlador tipo-2 para el control de voltaje. Por lo que en el presente trabajo se desarrolla la simulación de un controlador difuso tipo-2 con conjuntos difusos tipo-2 en intervalos para el control de excitación de un modelo de generador síncrono de 5to orden [8]. Se desarrolla el controlador con una base de conocimiento estática (matriz de reglas difusas constante). Se prueban además algunas variantes que la estructura misma del controlador permite como son: número de reglas difusas, tipo de funciones de membresía tipo-2, valor de incertidumbre asociada a los conjuntos difusos de entrada y algoritmo de reducción de tipo.

Referencia bibliográfica acerca de libros y artículos en relación a conjuntos difusos tipo-2 y Sistemas Lógicos Difusos puede ser consultado en: <http://sipi.usc.edu/~mendel/>

1.6 Contenido de la Tesis

El contenido por capítulos es el siguiente:

En el capítulo 2 se aborda teoría y términos necesarios para la comprensión de los conjuntos difusos tipo-2 en intervalos. Se muestra en forma gráfica el sistema de inferencia y los algoritmos de reducción de tipo Karnick & Mendel y Karnick & Mendel Mejorado.

En el capítulo 3, se explica a detalle el controlador difuso tipo-2 como regulador de voltaje para el sistema de excitación de un generador síncrono. Se muestra la parametrización de los espacios de entrada y salida para minimizar el número de parámetros requeridos por ambos controladores. Además se indican las fuentes de incertidumbres a simular en el sistema de control del presente trabajo. Se ilustra un caso ejemplo de las operaciones realizadas por el controlador tipo-2. Por último se muestra la metodología de sintonización utilizada en esta tesis.

En el capítulo 4, se muestran los resultados de sintonización del T1 FLC y el IT2 FLC con el generador en vacío para una línea de transmisión corta y larga. Posteriormente se muestran simulaciones con el generador síncrono sujeto a carga. El sistema se prueba ante grandes y pequeños disturbios y también se presentan resultados al añadir incertidumbre en ciertas variables dentro del sistema de control.

El capítulo 5 presenta las conclusiones generales del trabajo así como las recomendaciones a trabajos futuros.

Finalmente se presenta en la sección de apéndices se presenta: los índices de desempeño integral squared error (ISE), integral absolute error (IAE) e integral time absolute error (ITAE) (apéndice A) utilizados en este trabajo para la medición de desempeño ante grandes disturbios; el desarrollo del modelo matemático del generador síncrono (apéndice B); la comparación entre los algoritmos Karnick & Mendel y Karnick & Mendel Mejorado (apéndice C); teoría básica sobre el algoritmo de búsqueda por Tabú y resultados al aplicarlo sobre la matriz de conocimiento difuso del T1 FLC y el IT2 FLC (apéndice D); una aplicación de control difuso tipo-1 mediante un microcontrolador y comunicación serial para controlar un modelo de un motor de corriente directa en simulik (apéndice E), y finalmente se anexa el código MatLab utilizado en el desarrollo de este trabajo (apéndice F).

CAPÍTULO 2

*Me lo contaron y lo olvidé; lo vi y lo entendí; lo hice y lo aprendí.
Confucio, filósofo chino.*

2. SISTEMAS LÓGICOS DIFUSOS TIPO-2

2.1 Introducción

El concepto de conjunto difuso de tipo-2 fue introducido por L. A. Zadeh en 1975 [2], como una extensión de los primeros conjuntos difusos tipo-1⁴ definidos por él mismo en 1965 [14]. En los sistemas lógicos difusos tipo-1 (*por sus siglas en inglés T1 FLS*), el grado de membresía o pertenencia de una variable difusificada a dicho conjunto, es un número real en $[0, 1]$, por lo que una vez definido T1 FLC sus funciones de membresía ya tienen una forma definida y no existe incertidumbre en sus conjuntos [20]. Por su parte, una poderosa característica en los sistemas lógicos difusos tipo-2 (*T2 FLC por sus siglas en inglés*), es que el grado de pertenencia de una variable difusificada, es a su vez un conjunto difuso tipo-1 [3] [17]. De esta forma los conjuntos difusos tipo-2, modelan en mejor forma la incertidumbre presente al momento de definir los conjuntos difusos dentro del universo de discurso para una variable lingüística de entrada dada. Se puede pensar en los conjuntos difusos tipo-1, como conjuntos difusos con aproximación de primer orden para la representación de la incertidumbre presente en situaciones de mundo real, mientras que los conjuntos difusos tipo-2 son una aproximación de segundo[3]. Siguiendo esta idea, un conjunto difuso de orden n cuando $n \rightarrow \infty$, representaría en forma completa la incertidumbre en las situaciones reales, sin embargo la dificultad en los sistemas difusos aumenta en forma significativa conforme aumenta el orden de sus conjuntos difusos[3].

Suponga que se cuestiona acerca de una variable en particular por ejemplo la temperatura ciertamente es una variable lingüística desde que podemos decir temperatura fría, caliente, muy fría etc. Si quisiéramos definirlos mediante los conjuntos clásicos tendríamos que definir intervalos en donde la temperatura - por ejemplo - es considerada fría o caliente, de esta forma imaginemos definir un límite en 20°C y se concluiría que 19.9999 es temperatura fría y 20.0001 es temperatura caliente, lo cual a simple vista parece poco razonable (ver **Figura 2.1a**). Utilizando conjuntos difusos tipo-1 la representación a esta variable lingüística es más apropiada. Tomando el valor de 25°C , en los conjuntos clásicos tendríamos que decir que esa temperatura es caliente al 100% de veracidad, mientras que para los conjuntos difusos tipo-1 existe una transición la afirmación fría o caliente, para este caso, 25°C tiene una pertenencia de 0.2 fría y 0.8 caliente, lo cual resulta más apropiado ante la inexactitud de los términos frío y caliente.

⁴ Con la publicación de Zadeh en 1975[2], los conjuntos difusos tradicionales fueron renombrados a conjuntos difusos tipo 1.

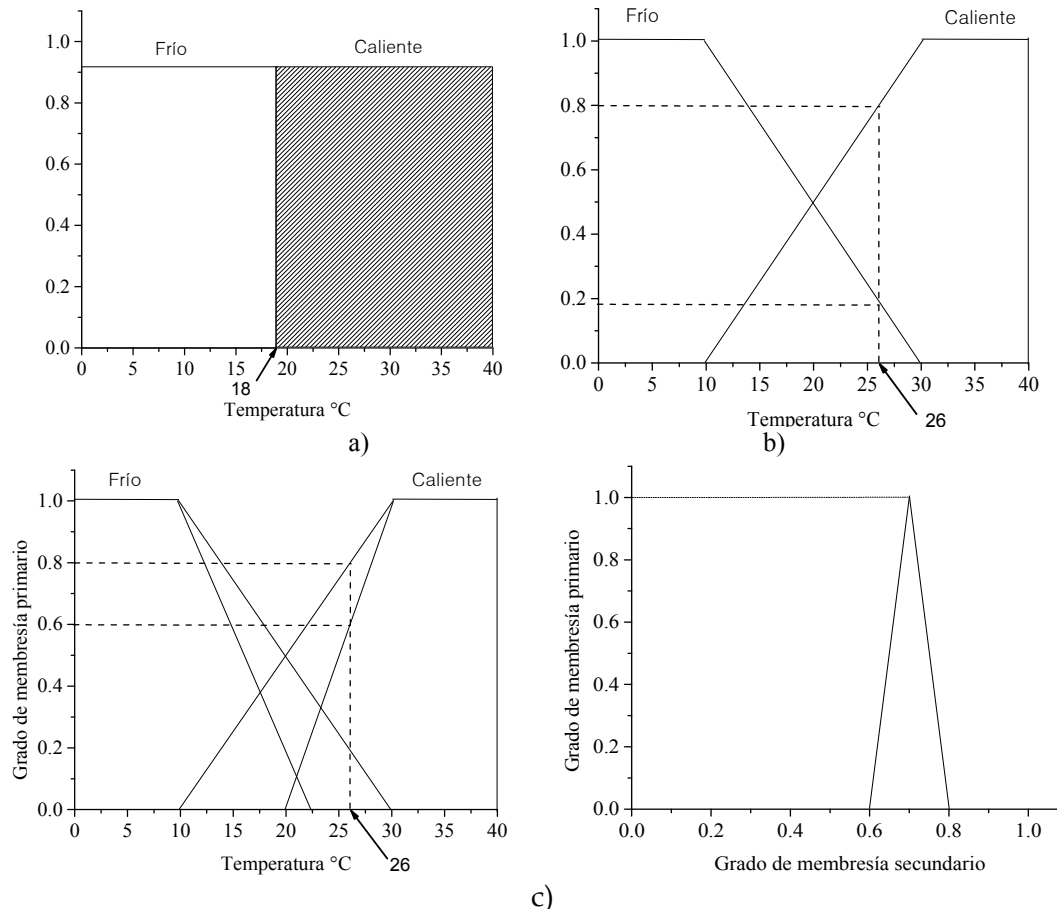


Figura 2.1 Conjuntos: a) clásicos; b) difusos tipo-1; c) difusos tipo 2.

Ahora bien, en **Figura 2.1b** para un valor dado en la variable de temperatura, su grado de membresía es un número como tal. La función de membresía es bien conocida, esto no permite ninguna incertidumbre[20].

Por otra parte al utilizar conjuntos difusos de tipo-2, cada valor de membresía es así mismo un conjunto difuso en el intervalo $[0, 1]$, así en la **Figura 2.1c** se tiene una función que toma valores desde 0.6 a 0.8 en el dominio del grado de membresía primario, y enseguida se tiene un mapeo al grado de membresía secundario con una función de membresía también tipo triangular con valor unitario en el valor medio entre $[0.6, 0.8]$ (ver **Figura 2.1d**, es claro que también son posibles otro tipo de funciones), y cuando todos estos grados de membresía secundarios son unitarios se tiene un conjunto tipo-2 en intervalos caracterizados por la huella de incertidumbre como se verá mas adelante (ejemplo tomado de [20]).

Los sistemas difusos tipo-1 han sido aplicados en forma satisfactoria en múltiples aplicaciones. Sin embargo, los conjuntos difusos tipo-1, tienen capacidades limitadas para manejar directamente información con incertidumbre. No obstante, surge la pregunta al momento de diseñar un controlador difuso; *¿Cuándo utilizar conjuntos difusos tipo-2?* La respuesta inmediata a esto sería: cuando existe dificultad en determinar la forma exacta y el dominio para las funciones de membresía dentro de un universo de discurso dado; cuando la información presente para las reglas difusas es incierta, ya que generalmente no existe información para los antecedentes y consecuentes acerca de dónde deben ser

localizados los centros de las funciones de membresía en los universos de discurso correspondientes.

Para una mejor comprensión, en lo siguiente se presentan los conceptos y definiciones, claves para los sistemas lógicos difusos tipo-2.

2.2 Funciones de Membresía Tipo-2

Un conjunto difuso tipo-2, se encuentra caracterizado por una función de membresía difusa con incertidumbre. El decir el grado de membresía para cada elemento dentro del conjunto difuso tipo-2 es a su vez un conjunto difuso tipo-1. Para ejemplificar los conjuntos difusos tipo-2 en forma gráfica, consideremos primero un conjunto difuso tipo-1 \bar{A} caracterizado por una función de membresía⁵ Gaussiana definida mediante:

$$\mu_{\bar{A}}(x) = e^{\{-1/2[(x-m)/\sigma]^2\}}; \sigma \in [\sigma_1 \sigma_2] \quad (2.1)$$

Donde m (media) y σ (desviación estándar) [37] son los parámetros que definen el dominio de la función de membresía dentro del universo de discurso dado. Desde que σ toma valores dentro del intervalo cerrado $[\sigma_1 \sigma_2]$ el conjunto difuso tendrá una forma mostrada en la **Figura 2.2**.

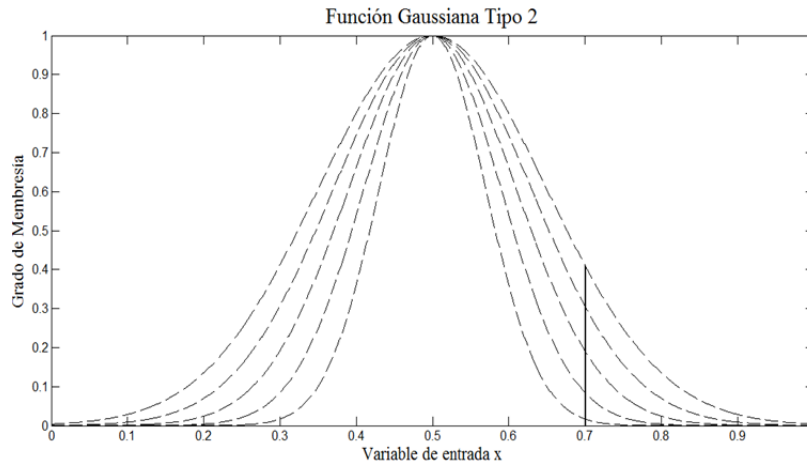


Figura 2.2 Conjunto Difuso tipo-2, representado a partir de la incertidumbre en σ para una función de membresía gaussiana de un conjunto difuso tipo-1.

Si ahora tomamos la incertidumbre para la media m ; es decir

$$\mu_{\bar{A}}(x) = e^{\{-1/2[(x-m)/\sigma]^2\}}; m \in [m_1 m_2] \quad (2.2)$$

Es decir m toma valores dentro del intervalo cerrado $[m_1 m_2]$, tendremos:

⁵ Función de membresía es la función que genera la forma del conjunto difuso $\mu_{\bar{A}}(x)$. Grado de membresía es el grado de verdad para un determinado $x = x'$, es decir $\mu_{\bar{A}}(x')$

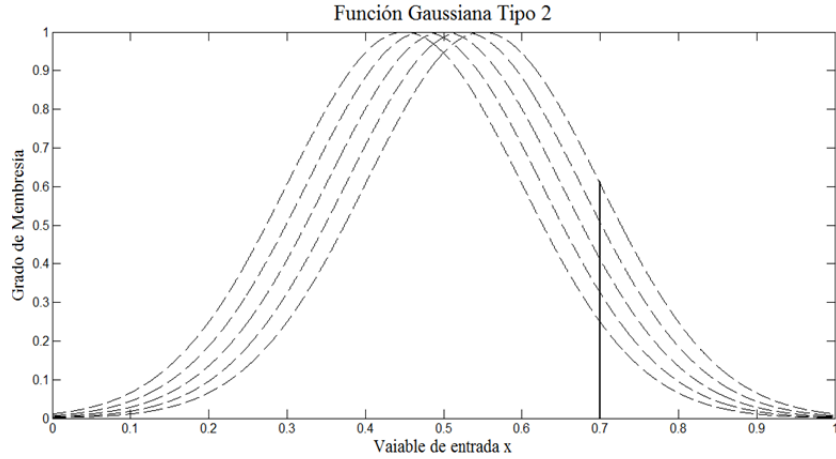


Figura 2.3 Conjunto Difuso tipo-2, representado a partir de la incertidumbre en m para una función de membresía gaussiana de un conjunto difuso tipo-1.

Es claro ver que para ambos casos (**Figura 2.2** y **Figura 2.3**), la variable mapeada a los conjuntos difusos tipo-2, no tienen un único grado de membresía, sino un intervalo de valores que representan un conjunto difuso tipo-1[37].

Por otra parte si consideramos un conjunto difuso tipo-1 \bar{A} caracterizado por una función de membresía tipo Gaussiana con media m y desviación estándar σ_x , el cual proporciona un valor numérico como grado de membresía para cada $x \in X$ tal que

$$\mu_{\bar{A}}(x) = e^{\{-1/2[(x-m)/\sigma_x]^2\}} \quad (2.3)$$

Consideremos la función de membresía de x es decir $\mu_{\bar{A}}(x)$, un conjunto difuso tipo-1, cuyo dominio (conjunto de todos los valores de $\mu_{\bar{A}}(x)$), es conocido como función de membresía primaria de x , la cual denotaremos para este caso como μ_p , en el mismo sentido, denotaremos al conjunto de todos los grados de membresía para todos los valores posibles de μ_p , como función de membresía secundaria, $\mu_s(x, \mu_p)$.

En otras palabras, para un valor fijo de $x = x'$ el conjunto de los grados de membresía forman un intervalo considerado como el dominio de un nuevo conjunto difuso tipo-1 (**Figura 2.2** y **Figura 2.3**), y sus correspondientes grados de membresía forman la función de membresía secundaria para el valor particular x' . Para el caso de una función de membresía tipo Gaussiana con media en $\mu_{\bar{A}}(x)$ y desviación estándar en σ_m se tiene como resultado la **Figura 2.4**.

$$\mu(x, \mu_p) = e^{\{-1/2[(\mu_p - \mu_{\bar{A}}(x))/\sigma_m]^2\}} \quad (2.4)$$

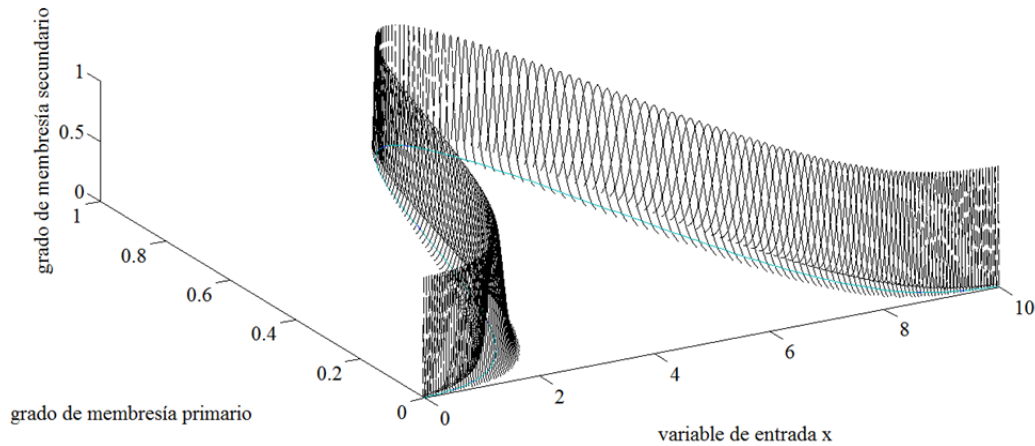


Figura 2.4 Vista de una función de membresía en un espacio de dimensión tres para un conjunto difuso tipo-2.

La dimensión extra que adicionan los conjuntos difusos tipo 2, es representada en el gráfico por el grado de membresía secundario (eje z), el cual mantiene como restricción el intervalo de valores $[0, 1]$.

Como se verá más adelante, cuando todos los valores de los grados de membresía secundarios son unitarios, el conjunto difuso tipo 2 se reduce a un caso especial denominado, conjunto intervalo tipo 2 [3]. El cual equivale para **Figura 2.4**, hacer los grados de membresía secundarios iguales a uno, y en esta forma despreciar el grado de membresía secundario, desde que ya no ofrece información adicional, no obstante en el plano x,y aún se mantiene incertidumbre para el conjunto difuso tipo 2.

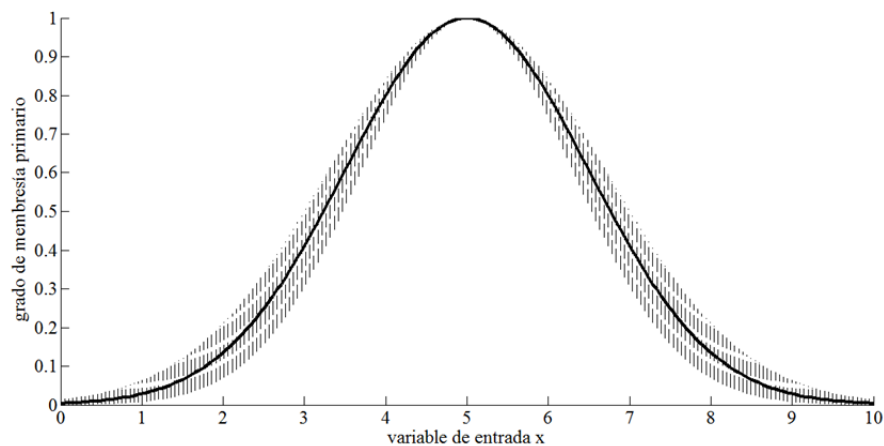


Figura 2.5 Huella de incertidumbre de un conjunto difuso general tipo-2.

La línea de mayor grosor que se muestra, es la función de membresía dada en (2. 3). De esta forma cuando toda incertidumbre desaparece, todo conjunto difuso tipo-2 se reduce al caso conjunto difuso tipo-1.

La idea hasta el momento acerca de los conjuntos difusos tipo-2 ha sido en forma general y en particular a las funciones de membresía tipo gaussianas. Estas ideas pueden de igual forma aplicarse a las funciones de membresía tipo triangulares tratadas también en este trabajo, y las cuales son mostradas y definidas mediante parámetros en la sección 3.4.

2.3 Conceptos sobre conjuntos difusos tipo-2

Los sistemas lógicos difusos tipo-1 y tipo-2, pueden ser clasificados cinco tipos en base al modelado de las entradas al sistema [3] es decir, el sistema difuso si considera incertidumbre en los antecedentes y consecuentes pero no en la(s) variables de entrada:

- Sistemas singleton tipo-1: en donde la(s) entrada(s) al sistema es un (son) número(s) como tal.
- Sistemas no singleton tipo-1: en donde la(s) entrada(s) al sistema es (son) modelada(s) con incertidumbre, mediante un conjunto difuso tipo-1.
- Sistema singleton tipo-2 en intervalos: en donde la(s) entrada(s) al sistema es un (son) número(s) como tal.
- Sistema tipo-1 no singleton tipo-2 en intervalos: en donde la(s) entrada(s) al sistema tipo-2 en intervalos, es (son) modelada(s) con incertidumbre, mediante un conjunto difuso tipo-1.
- Sistema tipo-2 en intervalos no singleton tipo-2 en intervalos: en donde la(s) entrada(s) al sistema tipo-2 en intervalos, es (son) modelada(s) con incertidumbre, mediante un conjunto difuso tipo-2 en intervalos.

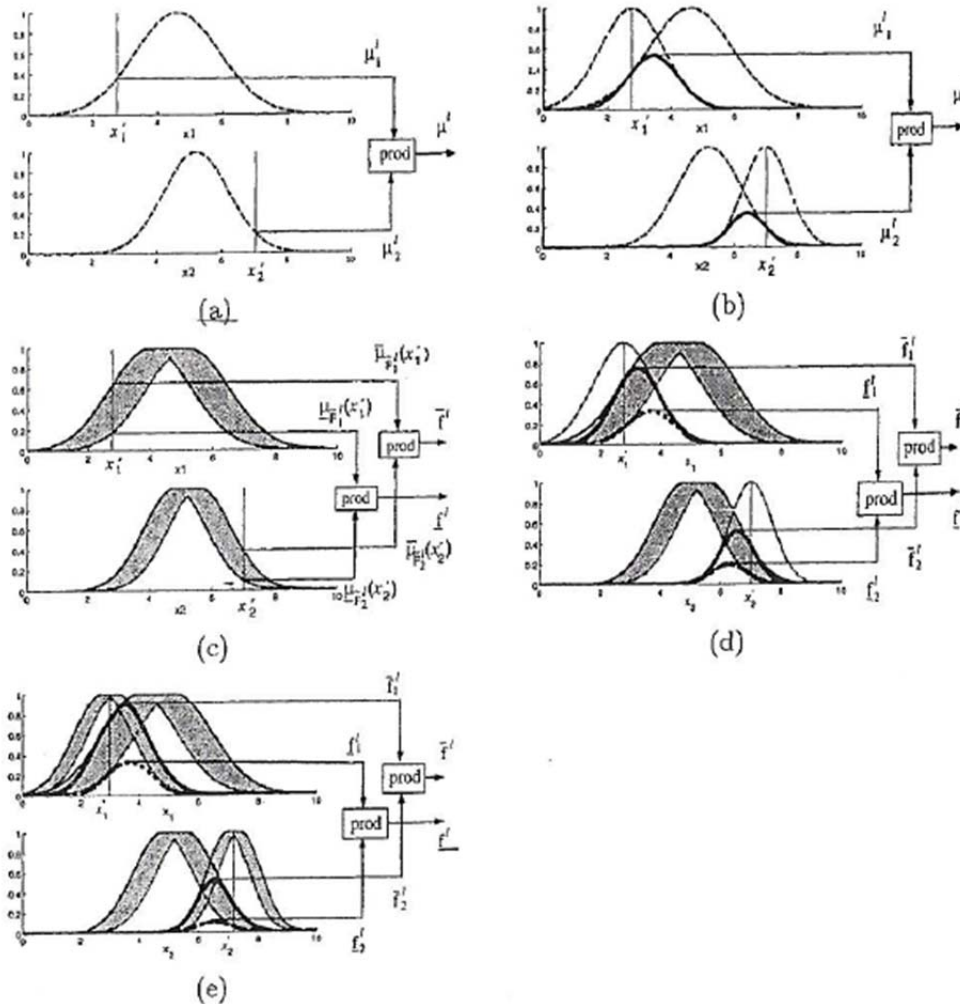


Figura 2.6 Sistemas lógicos difusos: (a) singleton tipo-1; (b) no singleton tipo-1; (c) singleton tipo-2 en intervalos; (d) tipo-1 no singleton tipo-2 en intervalos; (e) tipo-2 en intervalos no singleton tipo-2 en intervalos. Tomado de [3]

En este trabajo se tratan sistemas singleton tipo-1 y singleton en intervalos tipo-2.

Conjunto general lógico difuso tipo-2

Sea A_i un conjunto difuso de 1er tipo, el cual está representado por la función de membresía mostrada en la **Figura 2.7(a)**. Si se desplazaran los puntos base hacia la izquierda y derecha, no necesariamente en la misma cantidad [20], obtendríamos un conjunto en la forma mostrada en la **Figura 2.7(b)**. Entonces para un valor específico x' de la variable x , se tendría un conjunto de valores o grados de membresía sobre la línea vertical dentro del área sombreada, **Figura 2.7(b)**.

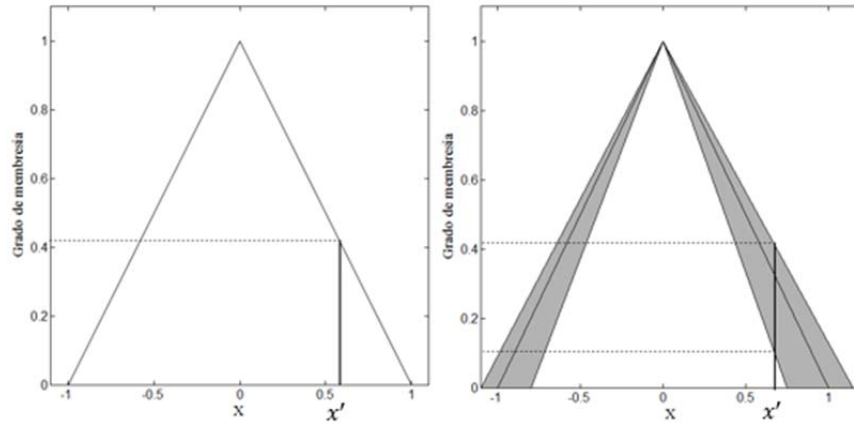


Figura 2.7 Función de membresía para un: (a) conjunto difuso tipo 1; (b) conjunto difuso tipo 2.

Estos puntos no tienen el mismo peso o valor, de aquí que es posible asignar una amplitud distribuida para cada uno de estos puntos. Haciendo esto para toda $x \in X$, se obtiene una función de membresía de tres dimensiones, lo cual caracteriza a los conjuntos difusos de tipo-2.

Un conjunto general difuso tipo-2 (*Type-2 Fuzzy Set - T2 FS*), queda representado por una función de membresía en un espacio de dimensión tres.

$$\tilde{A}_G = \left\{ (x, u), \mu_{\tilde{A}_G}(x, u) \mid \forall x \in X, \forall u \in J_x \subseteq [0, 1] \right\} \quad (2.5)$$

En donde el grado de membresía secundario toma valores entre cero y uno; es decir $0 \leq \mu_{\tilde{A}_G}(x, u) \leq 1$. (ver **Figura 2.4**). \tilde{A}_G también puede ser expresado como:

$$\tilde{A}_G = \int_{x \in X} \int_{u \in J_k} \mu_{\tilde{A}_G}(x, u) / (x, u) \quad J_x \subseteq [1, 0] \quad (2.6)$$

Donde la doble integral indica unión sobre todos los valores admisibles para x y u . Para universos discretos el signo integral se reemplaza por el signo sumatoria.

Conjunto lógico difuso tipo-2 en intervalos

Cuando todas los grados de membresía secundarios son unitarios, es decir: $\mu_{\tilde{A}_I}(x, u) = 1$, en (2. 6) entonces el conjunto \tilde{A}_I es llamado conjunto difuso tipo-2 en intervalos [37] (por sus silas en inglés *Interval Type-2 Fuzzy Set - IT2 FS*).

Así la tercera dimensión de la función de membresía no es ya necesaria, debido a que ya no proporciona información extra. Un IT2 FS es un caso especial de un T2 FS, y se expresa como:

$$\tilde{A}_I = \int_{x \in X} \int_{u \in J_x} 1/(x, u) J_x \subseteq [1, 0] \quad (2. 7)$$

Variable Primaria

Es la variable física de interés dentro del sistema, es decir presión, temperatura, posición etc. Para este caso, las variables primarias son: error y cambio del error del voltaje terminal en el generador, definido en sección 3.2.

Función de membresía primaria

Es cada uno de los valores de membresía que, dada variable primaria $x = x'$ forman un intervalo $J_{x'} = [J_{x'1}(x'), J_{x'N}(x')] = [\underline{\mu}_{\tilde{A}_I}(x'), \bar{\mu}_{\tilde{A}_I}(x')]$. En donde $\underline{\mu}_{\tilde{A}_I}(x')$ y $\bar{\mu}_{\tilde{A}_I}(x')$ indican los grados de membresía primarios inferior y superior respectivamnte. En otras palabras es el dominio de una función de membresía secundaria.

Variable secundaria

Es cada elemento de la función de membresía primaria J_x . Es decir para $x = x'$, cada elemento de $J_{x'i=1, \dots, N} = [J_{x'1}(x'), J_{x'N}(x')]$.

Función de membresía secundaria

Para cada valor de $x = x'$, el conjunto de valores en el plano $u, \mu_{\tilde{A}_I}(x, u)$ es llamado corte vertical de $\mu_{\tilde{A}_I}(x, u)$. Una función de membresía secundaria es cada corte vertical en $\mu_{\tilde{A}_I}(x', u)$ debido a $x = x'$ es decir cada $J_{x'i}$ con $i = 1, \dots, N$ (ver Figura 2.8).

Grado de membresía secundario

Dada una x' se tiene un intervalo $J_{x'i=1, \dots, N} = [J_{x'1}(x'), J_{x'P}(x')]$. Cada grado de membresía o pertenencia de cada elemento de $J_{x'}$ se denomina grado de membresía secundario, y se exprea mediante $\mu_{\tilde{A}_I}(x', u)$.

Huella de incertidumbre de \tilde{A}_I

$FOU(\tilde{A}_I)$ (por sus siglas en inglés *Footprint Uncertainty*), es la unión de todas las funciones de membresía primarias, en el plano x, u ; es decir el área entre el $UMF(\tilde{A}_I)$ y

$LMF(\tilde{A}_I)$, ver **Figura 2.8**. Puede entenderse de igual forma como la incertidumbre en la función de membresía primaria de un conjunto difuso tipo 2.

$$FOU(\tilde{A}_I) = \bigcup_{x \in X} J_x \tag{2.8}$$

Función de membresía superior de \tilde{A}_I , $UMF(\tilde{A}_I)$

Es el límite superior de $FOU(\tilde{A}_I)$, (ver **Figura 2.8**).

Función de membresía inferior de \tilde{A}_I , $LMF(\tilde{A}_I)$

Es el límite inferior de $FOU(\tilde{A}_I)$, (ver **Figura 2.8**).

Conjunto difuso tipo-2 en intervalos

Es un T2 FS en el cual todos los grados de membresía secundarios son unitarios, y es descrito completamente por la huella de incertidumbre. $\tilde{A}_I = \frac{1}{FOU(\tilde{A}_I)}$, es decir el grado de membresía secundario para todos los elementos de $FOU(\tilde{A}_I)$ son unitarios (ver **Figura 2.5**).

Conjunto Embebido Difuso tipo 1, \bar{A}_e

Cualquier conjunto difuso tipo-1 contenido en \tilde{A}_I que abarca $\forall x \in X$. También llamado dominio de la función de membresía secundaria o del conjunto difuso tipo-2.

Conjunto Difuso tipo 2 embebido, \tilde{A}_e

Cada conjunto difuso tipo-1 considerando los grados de membresía secundarios de cada uno de sus elementos (ver sección 2.4).

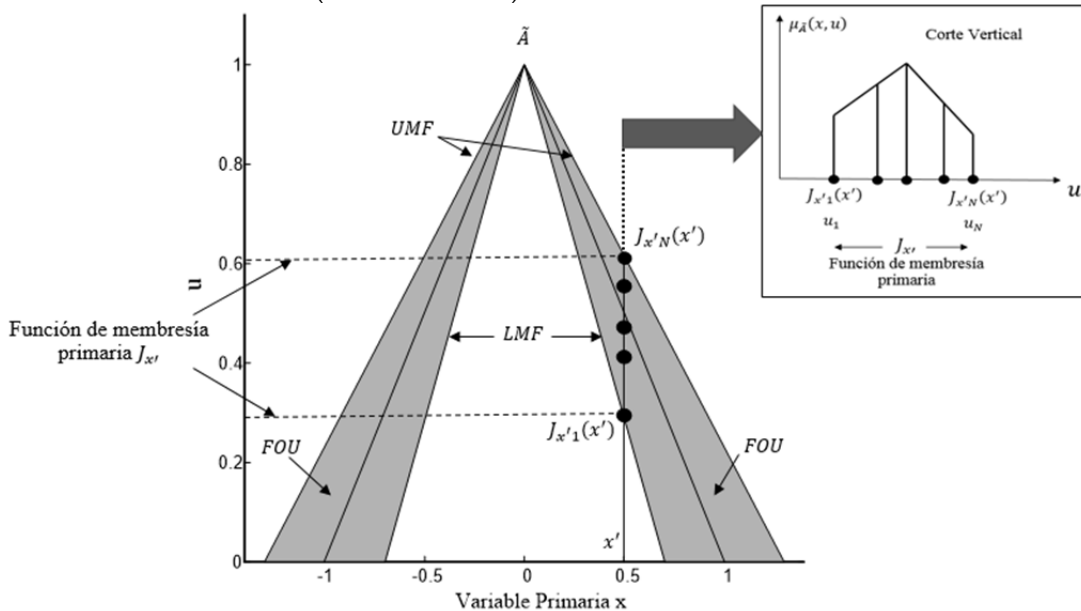


Figura 2.8 Descripción de un conjunto difuso tipo 2.

2.4 Conjuntos embebidos tipo 1 y tipo 2

Para comprender mejor los conjuntos difusos embebidos tipo-1 y 2 [3][22], en lo siguiente se extienden ambos conceptos.

Conjunto Embebido Difuso tipo 1

Para un universo continuo X y U , un conjunto difuso tipo-1 embebido se define mediante:

$$\bar{A}_e = \int_{x \in X} \theta/x \quad \theta \in J_x \subseteq U = [0 \ 1] \quad (2.9)$$

Es decir para cada valor de x , la función de membresía de \bar{A}_e tiene únicamente una función de membresía primaria denominada θ , donde $\theta \in J_x$.

Para un universo discreto X y U , un conjunto difuso tipo-1 embebido \bar{A}_e en \tilde{A}_e , contiene N elementos, uno de cada J_{x_i} con $i = 1, \dots, N$ denominado $\theta_1, \theta_2, \dots, \theta_N$.

$$\bar{A}_e = \sum_{i=1}^N \theta_i/x_i \quad \theta \in J_{x_i} \subseteq U = [0 \ 1] \quad (2.10)$$

Conjunto Difuso tipo-2 embebido

Un conjunto difuso tipo 2 embebido, es el resultado de considerar los grados de membresía secundarios de los N elementos de un conjunto difuso tipo-1 embebido \bar{A}_e en \tilde{A}_e . De esta forma, para un universo de discurso continuo (X, U) , un conjunto difuso tipo-2 embebido, puede expresarse mediante:

$$\tilde{A}_e = \int_{x \in X} [f_x(\theta)/\theta]/x \quad \theta \in J_x \subseteq U = [0 \ 1] \quad (2.11)$$

Es decir para cada valor de x , la función de membresía de \tilde{A}_e tiene únicamente una función de membresía primaria denominada θ , donde $\theta \in J_x$ y su correspondiente grado de membresía secundario asociado.

Para un universo de discurso discreto X y U , un conjunto difuso tipo-2 embebido \tilde{A}_e , tiene N elementos, donde \bar{A}_e contiene únicamente un elemento de cada J_{x_i} , con $i = 1, \dots, N = J_{x_1}, J_{x_2}, \dots, J_{x_N}$ denominados $\theta_1, \theta_2, \dots, \theta_N$ cada cual se encuentra asociado con su respectivo grado de membresía secundario $f_{x_1}(\theta_1), f_{x_2}(\theta_2), \dots, f_{x_N}(\theta_N)$.

$$\tilde{A}_e = \sum_{i=1}^N [f_{x_i}(\theta_i)/\theta_i]/x_i \quad \theta \in J_{x_i} \subseteq U = [0 \ 1] \quad (2.12)$$

Para ejemplificar gráficamente los conceptos anteriores, suponga que se tiene un conjunto difuso general tipo-2 \tilde{A}_G , visto únicamente desde el plano $X-U$.

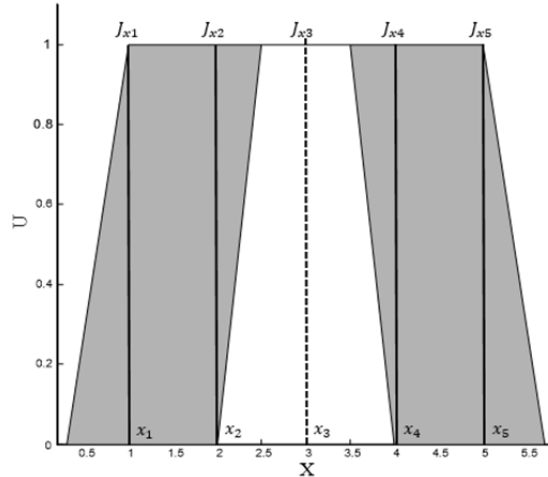


Figura 2.9 Conjunto difuso general tipo-2, a partir de muestras discretas para la variable primaria.

Si se considera un conjunto discreto con valores de x tal que x_1, x_2, \dots, x_N , se tendrán J_{x_N} funciones de membresía primarias. Y tomando u_k puntos para cada J_{x_N} , donde k no necesariamente es igual para cada J_{x_N} entonces el conjunto difuso tipo-2 discretizado resulta:

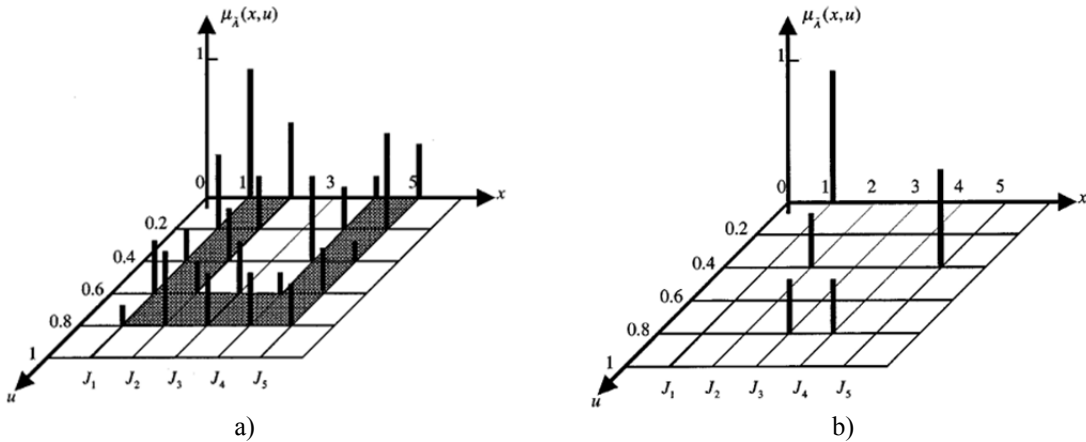


Figura 2.10 Conjuntos general discreto difuso tipo-2: a) en universo de discurso discreto, b) embebido tipo-2. Tomado de [25].

De donde se pueden obtener $\prod_{i=1}^N M_i$ conjuntos \bar{A}_e y \tilde{A}_e . M_i indica el número de elementos en cada J_{x_i} donde $i = 1, \dots, N$. Para la Figura 2.9(a), se tienen $5 \times 5 \times 2 \times 5 \times 5 = 1250$ conjuntos embebidos \bar{A}_e y \tilde{A}_e . La Figura 2.9(b), muestra un conjunto difuso tipo-2 embebido, y por lo tanto se tiene un conjunto difuso tipo-1 embebido dado por: $0/1 + 0.4/2 + 0.8/3 + 0.8/4 + 0.4/5$. Note que el grado de membresía secundario es diferente para cada elemento de \tilde{A}_e . Para este caso discreto, el área sombreada en Figura 2.9(a), representa el concepto de huella de incertidumbre para un conjunto difuso tipo-2.

El caso de un conjunto difuso tipo-2 en intervalos \tilde{A}_i y su conjunto difuso embebido tipo-2 en intervalos se muestra en la Figura 2.11.

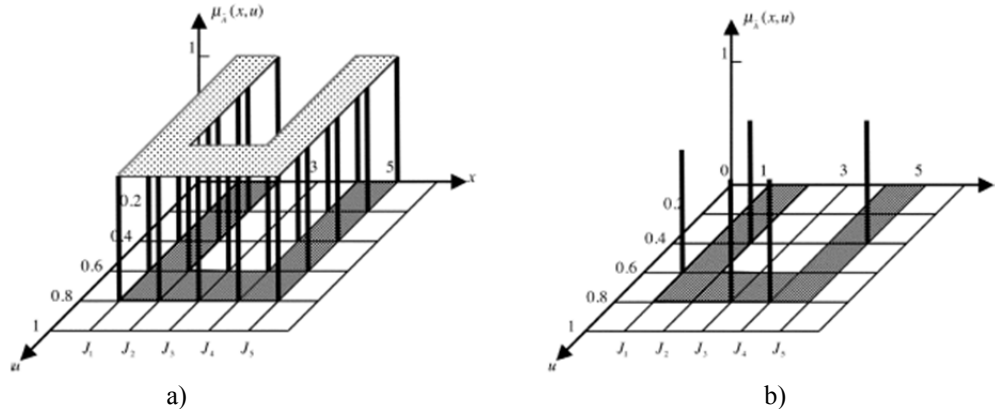


Figura 2.11 Conjunto difuso discreto tipo-2 en intervalos: a) en un universo de discurso discreto; b) embebido tipo-2 en intervalos. Tomado de [26].

Para el caso de un universo de discurso continuo, el número de \bar{A}_e y \tilde{A}_e es incontable [26].

2.5 Operaciones con Conjuntos Difuso Tipo-2 en Intervalos

El conjunto de operaciones de unión, intersección y complemento son ampliamente utilizadas en los sistemas lógicos difusos. Estas operaciones son deducidas a partir del principio de extensión [14]. En este trabajo se hace uso de conjuntos difusos tipo-2 en intervalos. Por lo que a continuación se presentan las operaciones de unión, intersección y complemento para dichos conjuntos.

Unión de dos conjuntos difusos tipo-2 en intervalos:

$$\tilde{A} \cup \tilde{B} = 1/[\underline{\mu}_{\tilde{A}_l}(x) \vee \underline{\mu}_{\tilde{B}_l}(x), \bar{\mu}_{\tilde{A}_l}(x) \vee \bar{\mu}_{\tilde{B}_l}(x)] \quad \forall x \in X \quad (2.13)$$

Donde \vee indica máximo.

La intersección de dos conjuntos difusos tipo-2 en intervalos es:

$$\tilde{A} \cap \tilde{B} = 1/[\underline{\mu}_{\tilde{A}_l}(x) \wedge \underline{\mu}_{\tilde{B}_l}(x), \bar{\mu}_{\tilde{A}_l}(x) \wedge \bar{\mu}_{\tilde{B}_l}(x)] \quad \forall x \in X \quad (2.14)$$

Donde \wedge indica mínimo.

El complemento de un conjunto difuso tipo-2 en intervalos, viene dado por:

$$\bar{\tilde{A}} = 1/[1 - \bar{\mu}_{\tilde{A}_l}(x), 1 - \underline{\mu}_{\tilde{A}_l}(x)] \quad \forall x \in X \quad (2.15)$$

2.6 Inferencia de un Sistema Lógico Difuso Tipo-2

Considerando un sistema lógico difuso con p entradas $x_1 \in X_1, x_2 \in X_2, \dots, x_p \in X_p$, y una sola salida, el cual utiliza reglas difusas de tipo *SI-ENTONCES*, las cuales son de la forma siguiente [3][22]:

$$R^l: \text{Si } x_1 \text{ es } \tilde{A}_1^l \text{ "y" } x_2 \text{ es } \tilde{A}_2^l \text{ "y" } \dots \text{ "y" } x_p \text{ es } \tilde{A}_p^l \text{ entonces y es } \tilde{C}^l \quad (2.16)$$

Esta regla representa una relación difusa tipo-2, entre el espacio de las variables de entrada y la variable de salida. En donde \tilde{A}_1^l , expresa un conjunto difuso tipo-2, en el

universo de discurso de la variable x_1 . El subíndice l hace referencia a la l – ésima regla. Mientras que \tilde{C}^l se refiere al conjunto de salida difuso tipo-2 para la l – ésima regla.

Es de notar que la estructura de las reglas tiene la misma forma que las reglas en un sistema lógico difuso convencional, con la particularidad de utilizar conjuntos difusos tipo-2 en los universos de discurso para las variables de entrada. Un sistema de inferencia lógico difuso que utiliza al menos un conjunto difuso tipo-2 para alguna de sus variables de entrada, es llamado sistema lógico difuso tipo-2[3].

Enseguida se muestra el análisis en forma gráfica del sistema de inferencia para una sola regla difusa, esto con objeto de facilitar su comprensión. Se muestra primero el caso tradicional, sistema de inferencia lógico difuso tipo-1, para luego extender el proceso a un sistema de inferencia lógico difuso tipo-2.

En la **Figura 2.12** se muestra el sistema de inferencia para una simple regla difusa con dos antecedentes y un solo consecuente, utilizando el operador norma-t mínimo.

Primero el proceso de difusificación permite mapear las variables físicas en los conjuntos difusos tipo-1 con su respectiva etiqueta (valor lingüístico). Este paso proporciona el grado de membresía de la variable física al respectivo conjunto difuso. Segundo, teniendo dichos grados de membresía, se relacionan las variables de entrada mediante un operador norma-t (operación mínimo para este caso). Finalmente se tiene como resultado un único grado de pertenencia el cual es a su vez el grado de pertenencia de disparo para el conjunto consecuente correspondiente a la variable de salida de la regla en cuestión.

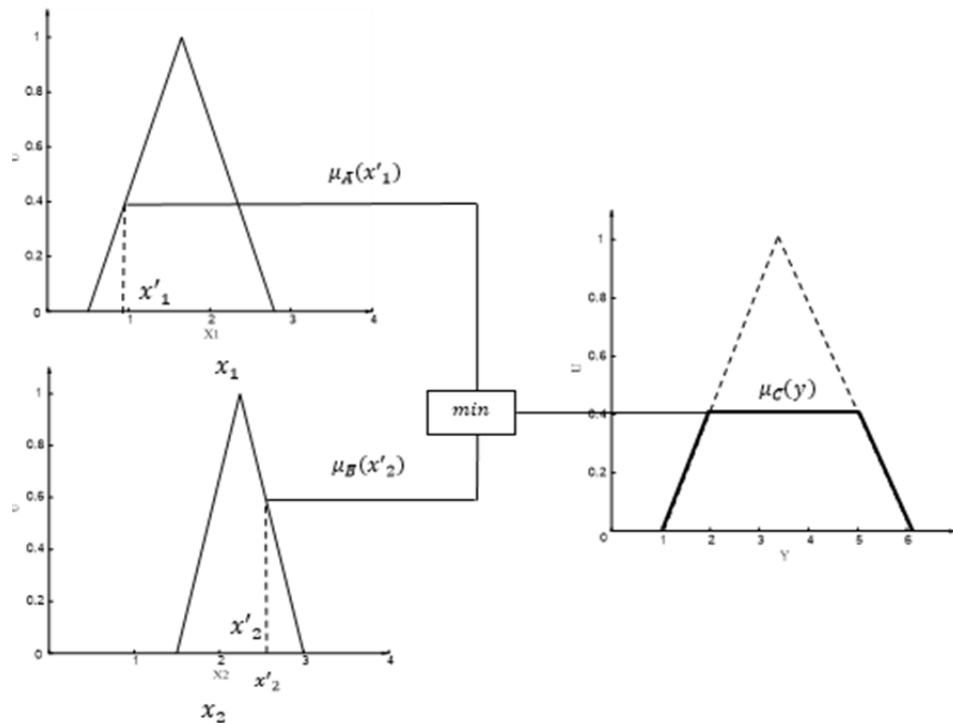


Figura 2.12 Sistema de inferencia para una simple regla difusa.

Este análisis es ahora extendido para un sistema de inferencia con conjuntos difusos tipo-2 en los consecuentes. Primero se tiene el proceso de difusificación para cada variable de entrada, lo que resulta en un intervalo de valores de u , lo que se conoce como función de membresía primaria, desde que se analizan conjuntos intervalos difusos tipo-2, los

grados secundarios de membresía son todos unitarios. Se toman únicamente los valores de membresía superior UMF y el inferior LMF para cada variable de entrada. Por lo que para la variable x_1 , se tiene una función de membresía primaria de $u = [\underline{\mu}_{\tilde{A}_1}, \bar{\mu}_{\tilde{A}_1}] = [LMF_{\tilde{A}_1}, UMF_{\tilde{A}_1}]$ y para x_2 se tiene $u = [\underline{\mu}_{\tilde{B}_1}, \bar{\mu}_{\tilde{B}_1}] = [LMF_{\tilde{B}_1}, UMF_{\tilde{B}_1}]$. Segundo, se relacionan los UMF y LMF de cada variable de entrada en forma separada mediante el operador norma-t (operación mínimo para este caso). Finalmente se tendrán dos valores correspondientes a UMF y LMF para el conjunto difuso tipo-2 consecuente de la regla difusa en cuestión.

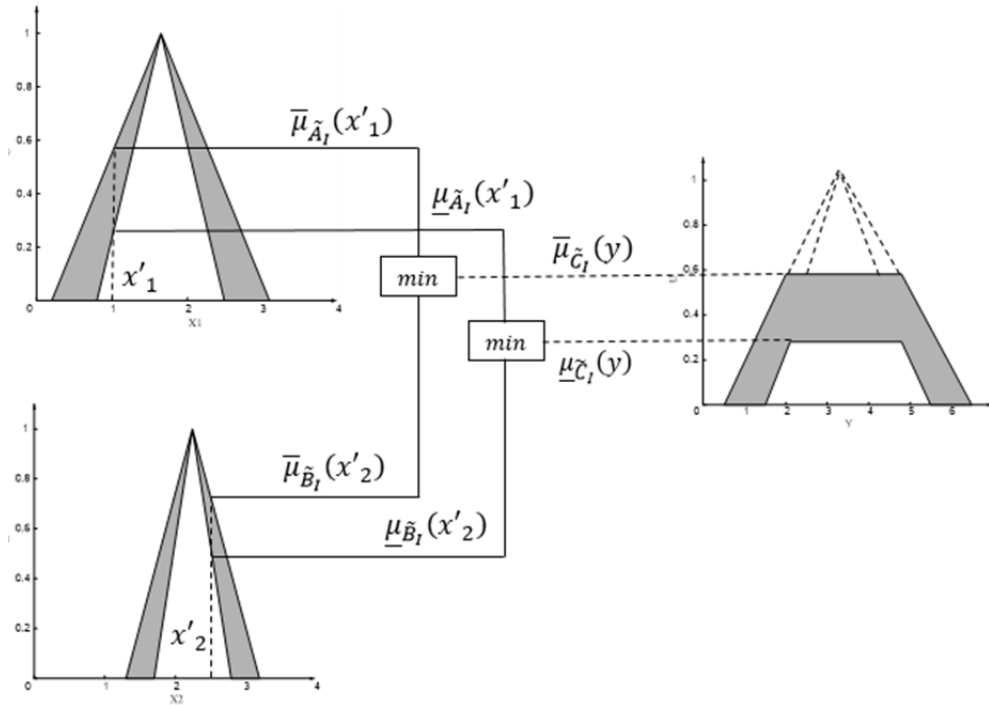


Figura 2.13 Sistema de inferencia para una simple regla difusa, con conjuntos antecedentes y consecuentes tipo-2.

Este análisis realizado para un sistema de inferencia con conjuntos difusos tipo-2. Es ahora extendido para el caso de activación de dos reglas difusas. En donde el grado de membresía inferior LMF y superior UMF del conjunto de salida $\tilde{C}_{1\vee 2}$ resultado de la unión de \tilde{C}_1 y \tilde{C}_2 , es determinado en mediante el operador máximo conorma-t.

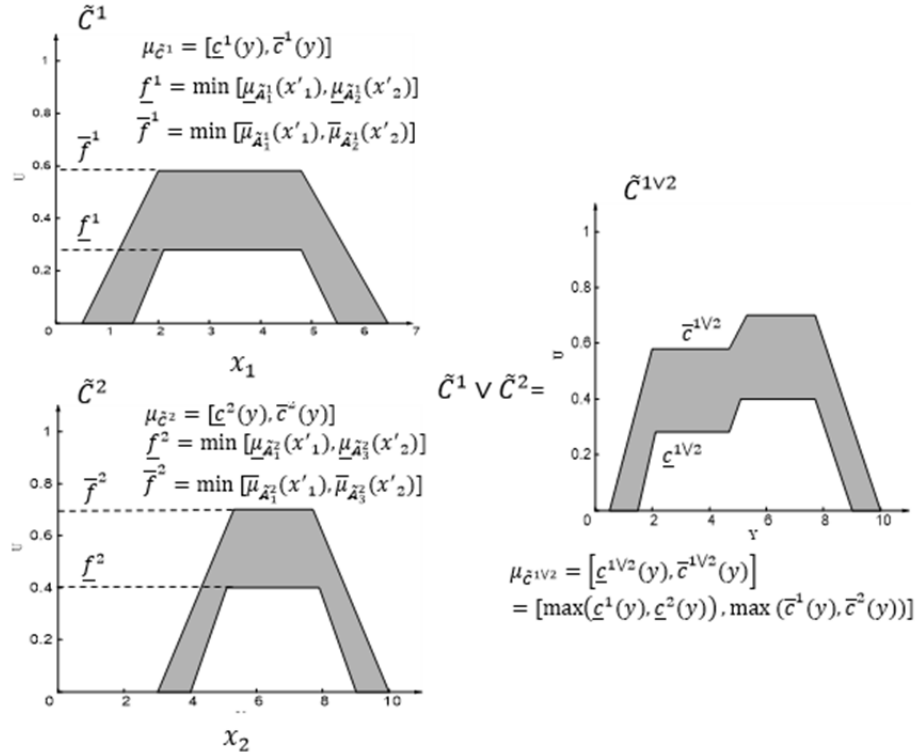


Figura 2.14 Sistema de inferencia para dos reglas difusas, con conjuntos consecuentes tipo 2.

De la figura anterior se observa, que cada conjunto de salida \tilde{C}_l donde l es la regla difusa activa, se encuentra definido mediante funciones de membresía inferior y superior de la forma $\mu_{\tilde{B}}^l = [\underline{b}^l(y), \bar{b}^l(y)]$. $\underline{b}^l(y)$ es la función de membresía inferior, es decir el trapecio interno del conjunto difuso tipo 2 \tilde{C}_l ; y $\bar{b}^l(y)$ es la función de membresía superior, es decir el trapecio externo de \tilde{C}_l . Por lo que la unión de \tilde{C}_1 y \tilde{C}_2 se obtiene mediante el operador conorma-t máximo; es decir:

$$\begin{aligned} \mu_{\tilde{C}^1 \vee \tilde{C}^2} &= [\underline{c}^{1\vee 2}(y), \bar{c}^{1\vee 2}(y)] = [\underline{c}^1(y) \vee \underline{c}^2(y), \bar{c}^1(y) \vee \bar{c}^2(y)] \\ &= [\max(\underline{c}^1(y), \underline{c}^2(y)), \max(\bar{c}^1(y), \bar{c}^2(y))] \end{aligned} \quad (2.17)$$

Donde \vee indica máximo.

Dado las operaciones mínimo máximo utilizadas en el proceso de inferencia para dos o más reglas difusas, el proceso de inferencia se conoce como min-max [3][5][16][45].

2.7 Método de Reducción de tipo de un Sistema Lógico Difuso Tipo 2

El conjunto de salida correspondiente a cada regla difusa en un T2 FLS es un conjunto difuso tipo-2 [3]. Para el procesamiento de la salida tal como lo muestra la **Figura 3.5**, consta de un bloque de reducción de tipo, el cual tiene como objetivo combinar todos los conjuntos difusos tipo-2 de salida en alguna forma y desarrollar el cálculo del centroide [3][22] el cual dará como resultado un conjunto difuso de 1er tipo al cual se le denomina

conjunto de tipo o grado reducido⁶ [22], para luego realizar el proceso de dedifusificación basado en un promedio de los centroides obtenidos. El centroide de un conjunto difuso tipo-1 para un dominio discreto en N puntos x_1, x_2, \dots, x_N , está definido como:

$$C_{\bar{A}} = \frac{\sum_{i=1}^N x_i \mu_{\bar{A}}(x_i)}{\sum_{i=1}^N \mu_{\bar{A}}(x_i)} \quad (2.18)$$

Sin pérdida de generalidad, es posible extender esta definición utilizando el principio de extensión [3], para un conjunto difuso tipo-2, de forma tal que:

$$C_{\bar{A}_G} = \int_{\theta_i \in J_{x_i}} \dots \int_{\theta_N \in J_{x_N}} [f_{x_1}(\theta_1) * \dots * f_{x_N}(\theta_N)] / \frac{\sum_{i=1}^N x_i \theta_i}{\sum_{i=1}^N \theta_i} \quad (2.19)$$

En donde $*$ indica la operación norma-t seleccionada. Además para este caso $f_{x_1}(\theta_1)$ indica el grado de membresía secundario de las funciones de membresía primarias J_{x_i} , con $i = 1, \dots, N = J_{x_1}, J_{x_2}, \dots, J_{x_N} = \theta_1, \theta_2, \dots, \theta_N$

Sea

$$\begin{aligned} a(\theta) &= \frac{\sum_{i=1}^N x_i \theta_i}{\sum_{i=1}^N \theta_i} \\ b(\theta) &= f_{x_1}(\theta_i) * \dots * f_{x_N}(\theta_N) \end{aligned} \quad (2.20)$$

En forma compacta (2.19) puede ser reescrita como:

$$C_{\bar{A}_G} = \int_{\theta_i \in J_{x_i}} \dots \int_{\theta_N \in J_{x_N}} b(\theta)/a(\theta) \quad (2.21)$$

Cada combinación $\theta_1, \dots, \theta_N$, y sus correspondientes grados de membresía secundarios, $f_{x_1}(\theta_i) * \dots * f_{x_N}(\theta_N)$ forman un conjunto difuso tipo-2 embebido (\tilde{A}_e , ver **Figura 2.10(b)**). Entonces cada elemento de $C_{\bar{A}_G}$ puede ser calculado mediante $a(\theta)$, lo que implica el cálculo de cada centroide de cada conjunto difuso tipo-1 embebido \bar{A}_e el cual se encuentra asociado con \tilde{A}_e . Así, hallar el centroide consiste en cada posible combinación $\{\theta_i, \dots, \theta_N\}$, tal que $\theta_i \in J_{x_i}$. Para cada combinación es necesario realizar el cálculo del centroide mediante $a(\theta)$; y a cada punto en el centroide se asigna un grado de membresía igual a la norma-t de los grados de membresía de $\theta_i \in J_{x_i}$. Si existe más de una combinación de θ_i genera un mismo punto en el centroide para $a(\theta)$, se conserva el grado de membresía $b(\theta)$ de valor mayor. El cálculo completo del centroide es determinado haciendo este proceso para todos los conjuntos difusos tipo-2 embebidos \tilde{A}_e en \tilde{A}_G .

En suma, el resultado de $C_{\bar{A}_G}$ es un conjunto difuso tipo-1, cuyos elementos son los centroides de todos los conjuntos tipo-1 embebidos \bar{A}_e en \tilde{A}_G , con un grado de membresía dado a partir de aplicar la norma-t a los grados de membresía secundarios correspondientes a los puntos $\{\theta_i, \dots, \theta_N\}$ que forman \tilde{A}_e .

Es de notar que el uso de un sistema general difuso de tipo-2, es complicado debido a la cantidad de operaciones realizadas para el cálculo del bloque *reducción de tipo*, las operaciones son simplificadas cuando los grados de membresía secundarios son unitarios,

⁶ Conjunto de grado reducido hace referencia a obtener un conjunto difuso tipo-1 en el procesamiento de salida partiendo de conjuntos difusos tipo-2 en intervalos.

que como se ha dicho, se reduce al caso conjuntos difuso tipo-2 en intervalos. Diferentes métodos para la *reducción de tipo* de los conjuntos de salida tipo 2 han sido desarrollados⁷ [22], desde que son una extensión del centroide de un conjunto difuso tipo 1. Algunos métodos son:

1. Reducción de tipo por centroide.
2. Reducción de tipo por pesos.
3. Reducción de tipo por centro de conjuntos.

2.8 Método de reducción de tipo de un IT2 FS

El grado de membresía secundario es siempre unitario para un conjunto intervalo difuso tipo-2. Por lo que el caso general (2.19) se reduce a

$$C_{\tilde{A}_I} = \int_{\theta_i \in J_{x_i}} \dots \int_{\theta_N \in J_{x_N}} 1 / \frac{\sum_{i=1}^N x_i \theta_i}{\sum_{i=1}^N \theta_i} \quad (2.22)$$

En donde \tilde{B} indica un conjunto intervalo difuso tipo-2. Y la ecuación anterior implica únicamente realizar el cálculo de $a(\theta)$.

Utilizando la representación corte vertical, es posible definir el centroide $C_{\tilde{A}_I}$, de un conjunto intervalo tipo-2, como una colección de todos los centroides de los conjuntos difusos embebidos de 2do tipo \tilde{A}_e dentro de los cuales existirá un centroide de mayor valor y uno de valor inferior que los demás. Definiendo a los centroides como $C_l(\tilde{A}_I) \equiv \underline{y}$ y $C_r(\tilde{B}) \equiv \bar{y}$, como los centroides de menor y mayor valor respectivamente. Esto significa que

$$C_{\tilde{A}_I} = 1/\{\underline{y}, \dots, \bar{y}\} \quad (2.23)$$

De donde

$$\underline{y} = \min_{\forall \theta_i \in [\underline{\mu}_{\tilde{A}_I}(x), \bar{\mu}_{\tilde{A}_I}(x)]} \frac{\sum_{i=1}^N \underline{x}_i \theta_i}{\sum_{i=1}^N \theta_i} \quad (2.24)$$

$$\bar{y} = \max_{\forall \theta_i \in [\underline{\mu}_{\tilde{A}_I}(x), \bar{\mu}_{\tilde{A}_I}(x)]} \frac{\sum_{i=1}^N \bar{x}_i \theta_i}{\sum_{i=1}^N \theta_i} \quad (2.25)$$

O lo es igual a⁸:

$$y_l = \frac{\sum_{i=1}^L \underline{x}_i \bar{\mu}_{\tilde{A}_I}(x_i) + \sum_{i=L+1}^N \underline{x}_i \underline{\mu}_{\tilde{A}_I}(x_i)}{\sum_{i=1}^L \bar{\mu}_{\tilde{A}_I}(x_i) + \sum_{i=L+1}^N \underline{\mu}_{\tilde{A}_I}(x_i)} \quad (2.26)$$

$$y_r = \frac{\sum_{i=1}^R \bar{x}_i \underline{\mu}_{\tilde{A}_I}(x_i) + \sum_{i=R+1}^N \bar{x}_i \bar{\mu}_{\tilde{A}_I}(x_i)}{\sum_{i=1}^R \underline{\mu}_{\tilde{A}_I}(x_i) + \sum_{i=R+1}^N \bar{\mu}_{\tilde{A}_I}(x_i)} \quad (2.27)$$

⁷ Para mayor detalle consultar ref [3] y [22].

⁸ El desarrollo de esta igualdad es hecho en [3]

L y R , son los puntos de conmutación y satisfacen:

$$\begin{aligned} \underline{x}_L &\leq \underline{y} \leq \underline{x}_{L+1} \\ \bar{x}_R &\leq \bar{y} \leq \bar{x}_{R+1} \end{aligned} \quad (2.28)$$

2.8.1 Algoritmo KM

Karnik y Mandel desarrollaron un algoritmo iterativo para el cálculo de estos centroides de los conjuntos generales difusos tipo-2 [3]. El algoritmo converge en forma exponencial y en los desarrollos requiere entre 2 y 6 iteraciones para obtener el resultado, lo que lo hace muy útil para aplicaciones de control. Este algoritmo es llamado Algoritmo KM [3][24][26][41][42] (en honor a sus autores). Este algoritmo permite hallar L y R de (2.26) y (2.27). El desarrollo del algoritmo es el siguiente [3][24]:

Tabla 2.1 Algoritmo KM para \underline{y}	
Cálculo de \underline{y}	
1.-Hacer discreto cada conjunto de salida en N puntos x_i es decir	$\{\underline{x}_1, \dots, \underline{x}_N\}$. Tal que $\underline{x}_1 < \underline{x}_2 < \dots < \underline{x}_N$
2.-Inicializar $\theta_i = \left[\frac{\underline{\mu}_{\bar{A}_i}(x_i) + \bar{\mu}_{\bar{A}_i}(x_i)}{2} \right], i = 1, \dots, N$	
3.-Calcular $\underline{y}'_l = \frac{\sum_{i=1}^N \bar{x}_i \theta_i}{\sum_{i=1}^N \theta_i}$	
4.-Hallar K ($K = 1, \dots, N - 1$), tal que $\bar{x}_k < \underline{y}'_l < \bar{x}_{k+1}$	
5.-Hacer $\theta_i = \begin{cases} \bar{\mu}_{\bar{A}_i}(x_i), & i \leq k \\ \underline{\mu}_{\bar{A}_i}(x_i), & i > k \end{cases}$, donde ($i = 1, \dots, N$).	
6.- Con θ_i corregido, calcular de nuevo $\underline{y}_l = \frac{\sum_{i=1}^N \bar{x}_i \theta_i}{\sum_{i=1}^N \theta_i}$	
7.- Si $\underline{y}_l = \underline{y}'_l$, entonces $\underline{y} = \underline{y}_l$, y $L = k$; Si no $\underline{y}'_l = \underline{y}_l$, y regresar a 4.	

Tabla 2.2 Algoritmo KM para \bar{y}	
Cálculo de \bar{y}	
1.-Hacer discreto cada conjunto de salida en N puntos x_i es decir	$\{\bar{x}_1, \dots, \bar{x}_N\}$. Tal que $\bar{x}_1 < \bar{x}_2 < \dots < \bar{x}_N$
2.-Inicializar $\theta_i = \left[\frac{\underline{\mu}_{\bar{A}_i}(x_i) + \bar{\mu}_{\bar{A}_i}(x_i)}{2} \right], i = 1, \dots, N$	
3.-Calcular $\bar{y}'_r = \frac{\sum_{i=1}^N \bar{x}_i \theta_i}{\sum_{i=1}^N \theta_i}$	
4.-Hallar K ($K = 1, \dots, N - 1$), tal que $\bar{x}_k < \bar{y}'_r < \bar{x}_{k+1}$	
5.-Hacer $\theta_i = \begin{cases} \underline{\mu}_{\bar{A}_i}(x_i), & i \leq k \\ \bar{\mu}_{\bar{A}_i}(x_i), & i > k \end{cases}$, donde ($i = 1, \dots, N$).	
6.- Con θ_i corregido, calcular de nuevo $\bar{y}_r = \frac{\sum_{i=1}^N \bar{x}_i \theta_i}{\sum_{i=1}^N \theta_i}$	
7.- Si $\bar{y}_r = \bar{y}'_r$, entonces $\bar{y} = \bar{y}_r$, y $R = k$; Si no $\bar{y}'_r = \bar{y}_r$, y regresar a 4.	

2.8.2 Algoritmo EKM

El algoritmo mejorado de Karnik & Mendel (*Enhanced Karnik and Mendel*), fue propuesto por sus autores Karnik y Mendel [42], para reducir el costo computacional del primer algoritmo KM propuesto por ellos. En forma similar, EKM consiste de dos partes; calcular \underline{y} y \bar{y} . Pero con la diferencia de que EKM propone mejoras sobre KM para reducir dicho costo computacional. Primero se reduce el número de iteraciones al realizar una mejor inicialización con respecto a KM; segundo, la condición de paro del algoritmo es cambiada, para reducir iteraciones innecesarias.

Karnik y Mendel, se percataron de que la inicialización de $\theta_i = [\underline{\mu}_{\bar{A}_i}(x_i) + \bar{\mu}_{\bar{A}_i}(x_i)]/2$, en el original KM requería una mejor inicialización [42], para hallar L y R . Por lo que en base a extensas simulaciones realizadas, hallaron que el algoritmo presentaba mejor desempeño al hacer $N/2.4$ para \underline{y} , $N/1.7$ para \bar{y} .

El desarrollo del algoritmo es el siguiente [41][42]:

Tabla 2.3 Algoritmo EKM para \underline{y}	
Cálculo de \underline{y}	
1.- Hacer discreto cada conjunto de salida en N puntos x_i es decir $\{\underline{x}_1, \dots, \underline{x}_N\}$. Tal que $\underline{x}_1 < \underline{x}_2 < \dots < \underline{x}_N$	
2.- Hacer $k = N/2.4$ (redondear al entero más cercano).	
3.- Calcular $a = \sum_{i=1}^k \underline{x}_i \bar{\mu}_{\bar{A}_i}(x_i) + \sum_{i=k+1}^N \underline{x}_i \mu_{\bar{A}_i}(x_i)$	
4.- Calcular $b = \sum_{i=1}^k \bar{\mu}_{\bar{A}_i}(x_i) + \sum_{i=k+1}^N \underline{\mu}_{\bar{A}_i}(x_i)$	
5.- Hallar k' tal que $x_{k'} \leq a/b < x_{k'+1}$	
6.- Mientras $k' \neq k$ hacer:	
7.- Hacer $s = \text{signo}(k' - k)$	
8.- Actualiza $a \leftarrow a + s \sum_{i=\min\{k, k'\}+1}^{\max\{k, k'\}} \underline{x}_i (\bar{\mu}_{\bar{A}_i}(x_i) - \underline{\mu}_{\bar{A}_i}(x_i))$	
9.- Actualiza $b \leftarrow b + s \sum_{i=\min\{k, k'\}+1}^{\max\{k, k'\}} \bar{\mu}_{\bar{A}_i}(x_i) - \underline{\mu}_{\bar{A}_i}(x_i)$	
10.- Actualiza $k \leftarrow k'$	
11.- Hallar k' tal que $x_{k'} \leq a/b < x_{k'+1}$	
12.- Retorna $y_l = a/b$; y $L = k$	

Tabla 2.4 Algoritmo EKM para \bar{y}	
Cálculo de \bar{y}	
1.- Hacer discreto cada conjunto de salida en N puntos x_i es decir $\{\bar{x}_1, \dots, \bar{x}_N\}$. Tal que $\bar{x}_1 < \bar{x}_2 < \dots < \bar{x}_N$	
2.- Hacer $k = N/1.7$ (redondear al entero más cercano).	
3.- Calcular $a = \sum_{i=1}^k \bar{x}_i \underline{\mu}_{\bar{A}_i}(x_i) + \sum_{i=k+1}^N \bar{x}_i \bar{\mu}_{\bar{A}_i}(x_i)$	
4.- Calcular $a = \sum_{i=1}^k \underline{\mu}_{\bar{A}_i}(x_i) + \sum_{i=k+1}^N \bar{\mu}_{\bar{A}_i}(x_i)$	
5.- Hallar k' tal que $x_{k'} \leq a/b < x_{k'+1}$	
6.- Mientras $k' \neq k$ hacer:	
7.- Hacer $s = \text{signo}(k' - k)$	
8.- Actualiza $a \leftarrow a - s \sum_{i=\min\{k, k'\}+1}^{\max\{k, k'\}} \bar{x}_i (\bar{\mu}_{\bar{A}_i}(x_i) - \underline{\mu}_{\bar{A}_i}(x_i))$	
9.- Actualiza $b \leftarrow b - s \sum_{i=\min\{k, k'\}+1}^{\max\{k, k'\}} \bar{\mu}_{\bar{A}_i}(x_i) - \underline{\mu}_{\bar{A}_i}(x_i)$	
10.- Actualiza $k \leftarrow k'$	
11.- Hallar k' tal que $x_{k'} \leq a/b < x_{k'+1}$	
12.- Retorna $y_r = a/b$; y $R = k$	

2.9 Dedifusificación

Finalmente e independientemente del método de reducción de tipo utilizado, el proceso de dedifusificación consiste en el promedio de los valores de los centroides extremos \underline{y} y \bar{y} , es decir:

$$y = (\underline{y} + \bar{y})/2 \quad (2.29)$$

Siendo este último valor, la salida final numérica de un sistema lógico difuso con conjuntos tipo-2.

CAPÍTULO 3

*La historia universal es una lucha entre dos formas de amor: entre el amor a sí mismo –hasta la destrucción del mundo- y el amor al otro –hasta la renuncia de sí mismo-
San Agustín. Doctor de la Iglesia.*

3. SISTEMA DE EXCITACIÓN UTILIZANDO CONTROL DIFUSO TIPO-2

3.1 Introducción

El regulador de voltaje para el sistema de excitación de una máquina síncrona ajusta la potencia de salida necesaria para controlar la magnitud de la tensión en las terminales del generador. Esto significa que, ante un cambio en el voltaje de referencia (V_{ref}), el voltaje en la salida del regulador (V_{reg}) provoca un cambio en el voltaje del excitador (V_{fd}), el cual es aplicado al devanado de campo de la máquina síncrona para ajustar el voltaje en terminales [43]. Por lo que el regulador de voltaje debe presentar robustez y confiabilidad para asegurar un correcto funcionamiento en el control del voltaje en terminales ante cualquier disturbio que se presente.

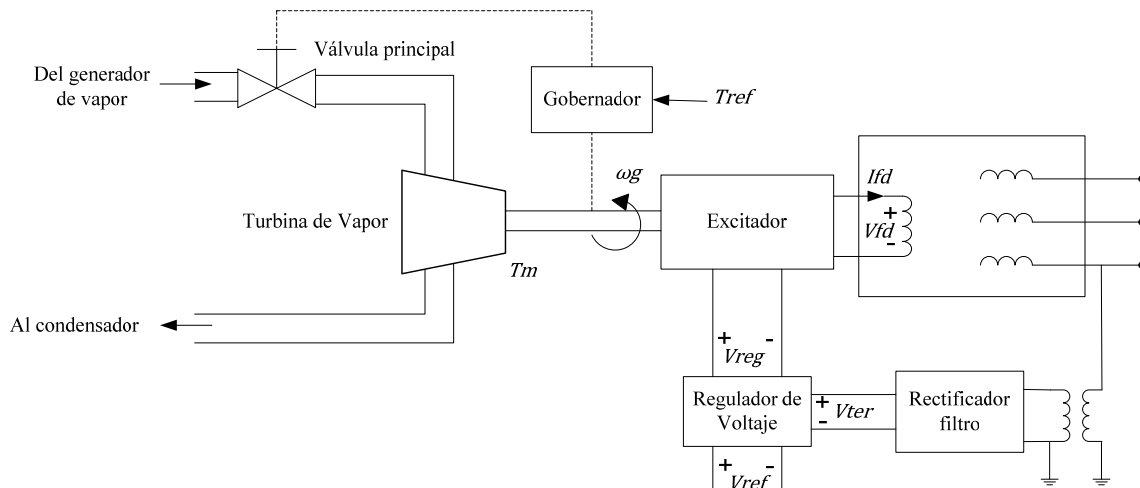


Figura 3.1 Sistema regulador de voltaje y gobernador de un generador. Tomado de [43].

En este trabajo se utiliza para la simulación de la máquina síncrona, el modelo de quinto orden, el cual es un modelo aproximado y desprecia los transitorios en el estator y la red (Ver apéndice B). El control del sistema de excitación se realiza mediante un controlador difuso tipo-2, el cual utiliza conjuntos difusos tipo-2 en intervalos. Y por sus siglas en inglés, se ha denominado IT2 FLC. Su desempeño se compara con su predecesor el controlador difuso tipo-1 y el regulador ST1.

3.2 Arquitectura del Sistema de Control

Para poder utilizar el control difuso como regulador de voltaje del sistema de excitación, simplemente se implementa un lazo de control del voltaje en terminales de la máquina junto con un nodo sumador y un valor de referencia con lo que se obtiene una señal de error. La siguiente figura muestra en forma general la arquitectura del sistema de control para el control de voltaje en terminales de la máquina síncrona.

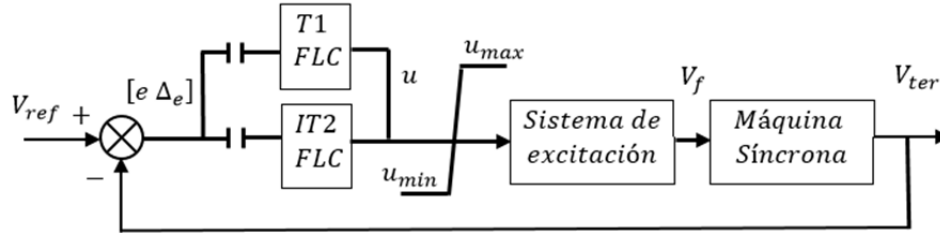


Figura 3.2 Diagrama de bloques del controlador difuso como regulador de voltaje.

Donde $[e \ \Delta e]$ es el vector de entrada para ambos controladores. Y se definen como:

$$e := error$$

$$\Delta e := Cambio del error$$

La salida del controlador es \dot{u} y se refiere al incremento requerido para la variable de salida. Y está definido como:

$$\dot{u} = u(k) - u(k - 1) \quad (3.1)$$

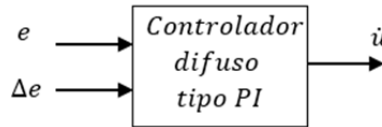


Figura 3.3 Controlador difuso tipo PI

Donde (k) al valor del error en la iteración actual y $(k - 1)$ se refiere al valor en la iteración anterior, además \dot{u} puede tomar valores positivos o negativos para incrementar la variable de salida al valor requerido.

De aquí

$$u(k) = \dot{u} + u(k - 1) \quad (3.2)$$

De esta forma es posible desarrollar un controlador difuso tipo PI [44].

Para obtener el error y cambio de error se hace simplemente:

$$e = V_{ref} - V_{ter} \quad (3.3)$$

$$\Delta e = e(k) - e(k - 1) \quad (3.4)$$

El limitador en la salida de los controladores, tiene por objeto evitar cualquier sobretensión al sistema de excitación y es una protección en el sistema los límites son:

$$u_{min} = 0 \quad (3.5)$$

$$u_{max} = 4 \quad (3.6)$$

El conjunto de ecuaciones que representan a la máquina síncrona se encuentran expresadas en por unidad (P.U). Con lo que el voltaje en terminales también lo está.

En las simulaciones, también se utiliza el regulador convencional tipo ST1 pero no es mostrado en la **Figura 3.2**.

3.3 Estructura del Controlador Tipo-1 y Tipo-2

La estructura del controlador difuso tipo-1 se muestra en la **Figura 3.4** [49]. Los elementos que lo conforman son:

1. Difusificador
2. Sistema de Inferencia
3. Base de conocimiento o reglas difusas.
4. Dedifusificador

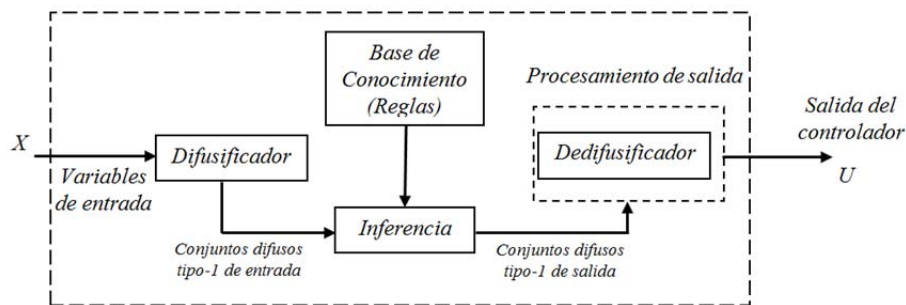


Figura 3.4 Estructura general de un controlador difuso tipo-1 convencional.

La estructura del controlador difuso tipo-2 se muestra en la **Figura 3.5**, sus elementos que lo conforman son:

1. Difusificador
2. Sistema de Inferencia
3. Base de conocimiento o reglas difusas.
4. Bloque de procesamiento de salida
 - 4.1 Algoritmo de reducción de tipo
 - 4.2 Dedifusificador

Nótese que la diferencia en la estructura entre ambos controladores radica el bloque de procesamiento de salida para el controlador tipo-2, el cual incluye un algoritmo de reducción de tipo y dedifusificador. Esto debido a los conjuntos difusos tipo-2 en intervalos utilizados por el controlador.

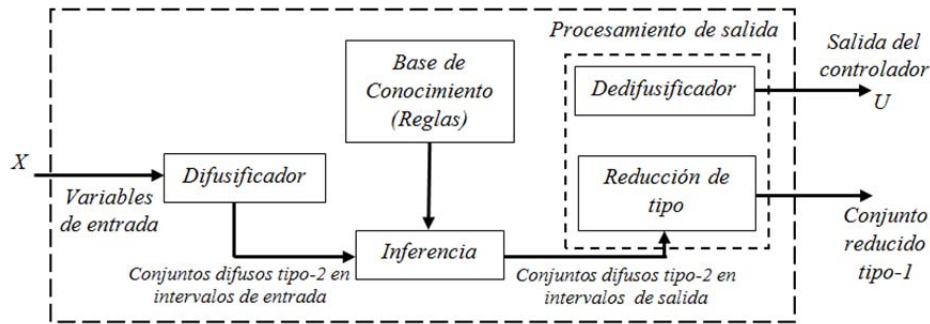


Figura 3.5 Estructura general de un controlador difuso tipo-2 en intervalos.

3.3.1 Elementos del controlador tipo-1

Difusificador

Este bloque genera los valores difusos mediante el mapeo de un valor numérico real del vector de entrada $X = (e, \Delta e)^T \rightarrow [0, 1]$ a los conjuntos difusos presentes en los correspondientes uni versos de discurso. Aquí se asignan los grados de pertenencia o membresía. Las etiquetas lingüísticas utilizadas para ambos espacios de entrada fueron:

Para un espacio de entrada con 3 conjuntos difusos:

- N = Negativo.
- CE = Cero.
- P = Positivo.

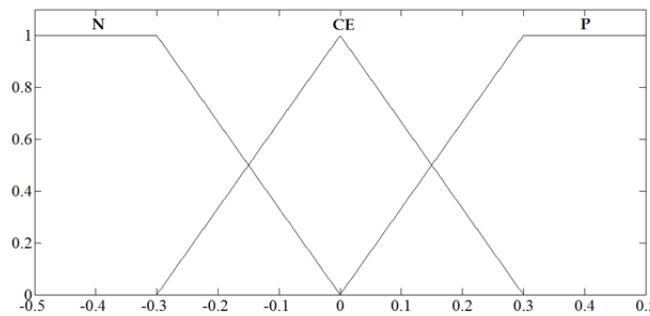


Figura 3.6 Etiquetas del universo de entrada con tres conjuntos difusos tipo-1.

Para un espacio de entrada con 5 conjuntos difusos:

- NM = Negativo mediano.
- NP = Negativo pequeño.
- CE = Cero.
- PP = Positivo pequeño.
- PM = Positivo mediano.

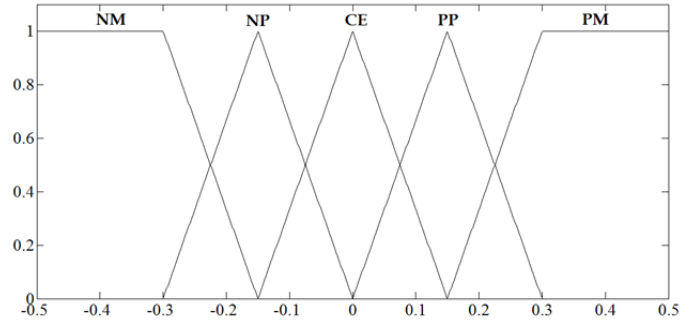


Figura 3.7 Etiquetas del universo de entrada con cinco conjuntos difusos tipo-1.

Para un espacio de entrada con 7 conjuntos difusos:

- NG = Negativo grande.
- NM = Negativo mediano.
- NP = Negativo pequeño.
- CE = Cero.
- PP = Positivo pequeño.
- PM = Positivo mediano.
- PG = Positivo grande.

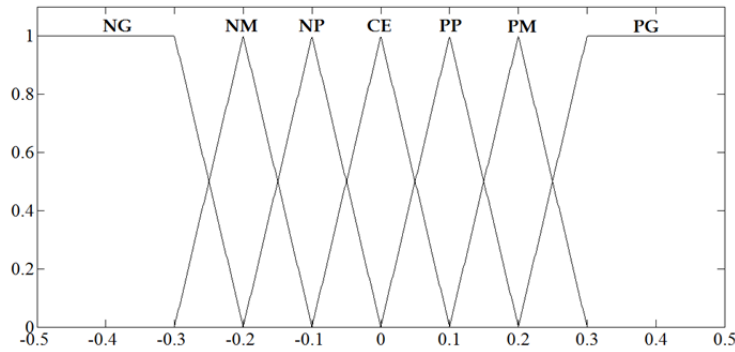


Figura 3.8 Etiquetas del universo de entrada con siete conjuntos difusos tipo-1.

Las etiquetas lingüísticas para el espacio de entrada del error y del cambio del error son idénticas.

Sistema de inferencia y Base de conocimiento

El sistema de inferencia es el encargado de combinar las reglas difusas del tipo *SI – Entonces*, que conforman la base de conocimiento.

Cada regla es interpretada como una implicación difusa, es decir una relación difusa entre el espacio de entrada $X = [e \Delta e]$ y el espacio de salida y [3]. Las matrices de conocimiento difuso para el controlador con 9, 25 y 49 reglas se muestran a continuación:

Tabla 3.1 Matriz de 9 reglas difusas

e	Δe	N	CE	P
N		NG	NM	CE
CE		NP	CE	PP
P		CE	PM	PG

Tabla 3.2 Matriz de 9 reglas difusas

e	Δe	NM	NP	CE	PP	PM
NM		NG	NG	NM	NP	CE
NP		NG	NM	NP	CE	PP
CE		NM	NP	CE	PP	PM
PP		NP	CE	PP	PM	PG
PM		CE	PP	PM	PG	PG

Tabla 3.3 Matriz de 9 reglas difusas

e	Δe	NG	NM	NP	CE	PP	PM	PG
NG		NG	NG	NG	NM	NM	NP	CE
NM		NG	NG	NM	NM	NP	CE	PP
NP		NG	NM	NM	NP	CE	PP	PM
CE		NM	NM	NP	CE	PP	PM	PM
PP		NM	NP	CE	PP	PM	PM	PG
PM		NP	CE	PP	PM	PM	PG	PG
PG		CE	PP	PM	PM	PG	PG	PG

Dedifusificador

El bloque dedifusificador se ocupa en generar el valor real para la variable de salida. Muchos métodos de dedifusificación han sido propuestos entre ellos:

- Centroide
- Centro de sumas
- Centro de barras
- Centros de conjuntos

En este trabajo para el controlador tipo-1 se utiliza el método centro de barras. En el cual cada regla difusa de salida es expresada mediante una barra con su máximo valor membresía en el centro del conjunto de salida, para luego promediar los pesos ponderados. Este método es realizado por el algoritmo Karnick & Mendel cuando se hace $\delta_e = \delta_{\Delta e} = 0$ (ver sección 3.8.1).

Las etiquetas lingüísticas para las siete barras de salida son:

- NG = Acción Negativa grande.
- NM = Acción Negativa mediana.
- NP = Acción Negativa pequeña.
- CE = Acción Cero.
- PP = Acción Positiva pequeño.
- PM = Acción Positiva mediana.
- PG = Acción Positiva grande.

3.3.2 Elementos del controlador tipo-2

El bloque dedifusificador es también utilizados por el controlador difuso tipo-2, con la diferencia de que utiliza conjuntos difusos tipo-2 en intervalos y el sistema de inferencia es como se ilustra en la **Figura 2.13** y **Figura 2.14** [49]. La base de reglas de conocimiento también es idéntica al del controlador difuso tipo-1.

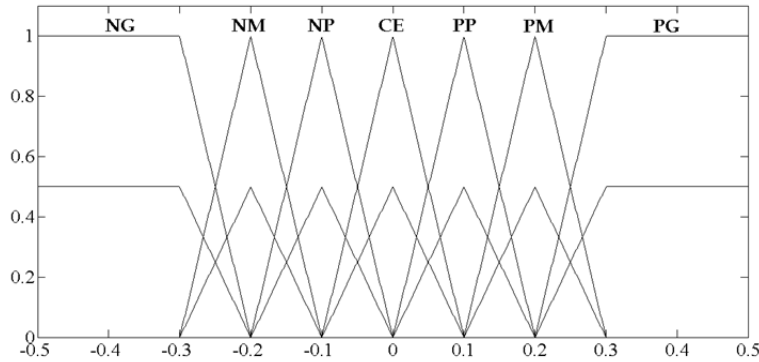


Figura 3.9 Etiquetas del universo de entrada con siete conjuntos difusos triangulares tipo-2.

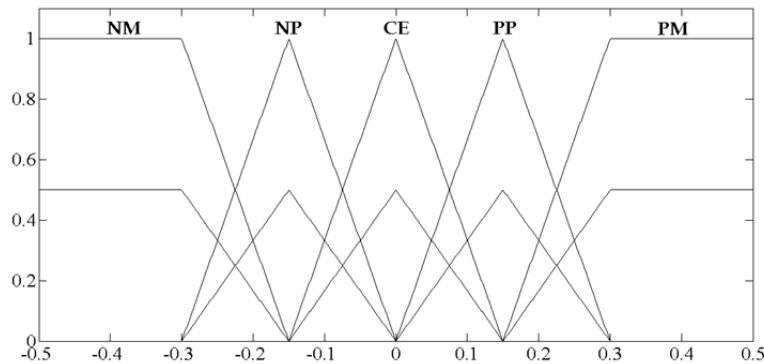


Figura 3.10 Etiquetas del universo de entrada con cinco conjuntos difusos triangulares tipo-2.

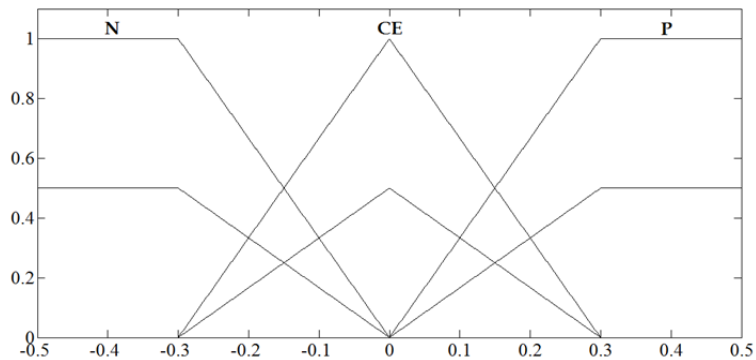


Figura 3.11 Etiquetas del universo de entrada con tres conjuntos difusos triangulares tipo-2.

Por simplicidad, la incertidumbre utilizada para los conjuntos difusos tipo-2 no cambia. Es decir, es la misma para todo conjunto presente dentro del universo de discurso en cuestión. Únicamente se aumentan o disminuye el número de conjuntos difusos en el espacio de entrada. Esto sucede también para los conjuntos difusos Gaussianos tipo-2.

Algoritmo de reducción de tipo

Este proceso en el bloque de procesamiento de salida del controlador representa un mapeo de un conjunto difuso tipo-2 a un conjunto difuso tipo-1. Como el proceso de calcular el centroide de un conjunto difuso tipo-2 es una extensión del cálculo para un tipo-1, existen entonces también varios métodos. Cuando se trata de conjuntos difusos en intervalos, el proceso de reducción de tipo se puede realizar mediante un eficiente algoritmo iterativo [3], denominado Karnik y Mendel en su versión convencional y mejorada. En el presente trabajo se utiliza en algoritmo mejorado de Karnik y Mendel en las simulaciones. La comparación en eficiencia de métodos se muestra en el apéndice A.

Dedifusificador

El proceso de dedifusificación es simplemente el promedio de los puntos \underline{y} y \bar{y} encontrados mediante el algoritmo iterativo Karnick & Mendel o Karnick & Mendel mejorado. Es decir (2.26).

3.4 Parametrización de Conjuntos Difusos Tipo-2

Las funciones de membresía utilizadas en esta tesis son de dos tipos:

- Funciones de membresía triangulares
- Funciones de membresía Gaussianas

En forma general, las primeras utilizan la formula general de la recta.

$$y - y_1 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x - x_1) \quad (3.7)$$

Donde (x_1, y_1) y (x_2, y_2) representan dos puntos conocidos; x es el valor de la variable en cuestión mapeada a los conjuntos difusos (dominio del universo de discurso) y y es el valor de pertenencia o membresía al conjunto difuso por lo que $y = \mu(x)$.

$$\mu(x) = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x - x_1) + y_1 \quad (3.8)$$

Mientras que las funciones Gaussianas utilizan:

$$\mu(x) = e^{\{-1/2[(x-m)/\sigma]^2\}} \quad (3.9)$$

3.4.1 Parametrización de los conjuntos de entrada

Con el fin de seleccionar el mínimo de parámetros para un controlador difuso tipo-2, primero se toma como punto de partida, la parametrización de los conjuntos difusos tipo-1 según:

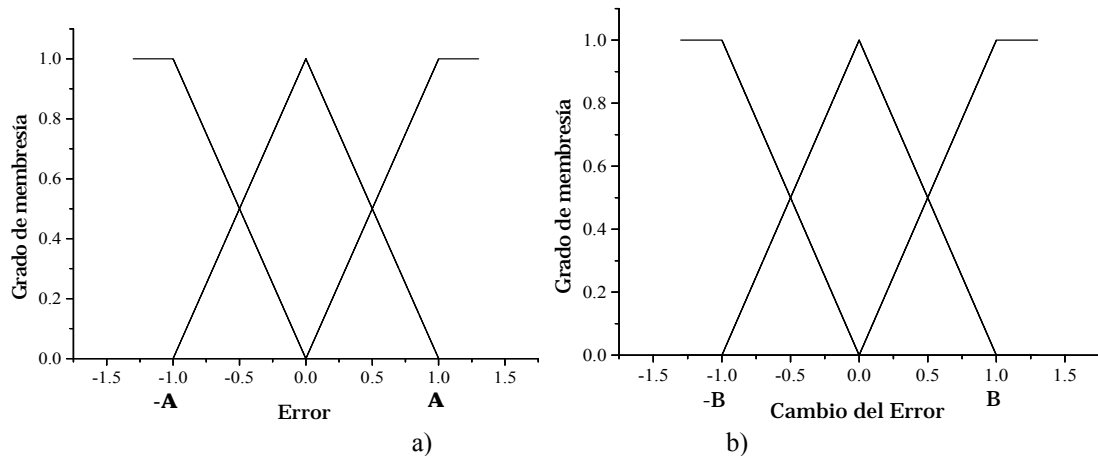


Figura 3.12 Parametrización del universo de discurso con conjuntos difusos simétricos: a) universo del error con tres funciones de membresía; b) universo del cambio del error con tres funciones de membresía, y c) Universo de salida.

En general para 3, 5, 7,...,n conjuntos difusos tipo-1 presentes en el universo de entrada:

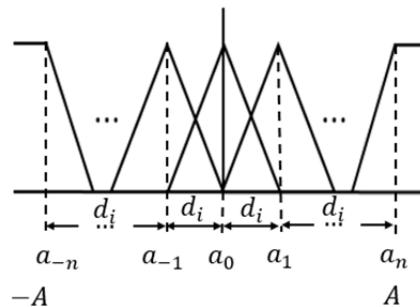


Figura 3.13 Parametrización g general de un universo de entrada.

Donde a_{-n}, \dots, a_n indica simplemente la etiqueta de los conjuntos presentes, d_i es la distancia de centros, entre dos conjuntos difusos adyacentes, el universo de discurso parametrizado en A , resultaría [44]:

$$d_i = A \left(\frac{2}{N - 1} \right) \tag{3.10}$$

Donde N es el número de conjuntos difusos presentes en el universo. Se tiene de esta forma la distancia constante entre centros de los conjuntos difusos. Con lo que ya no es necesario definir cada centro de conjunto difuso si no únicamente los centros extremos.

Supongamos que para un universo parametrizado en $A = 10$ y con $N = 5$ conjuntos difusos tendremos:

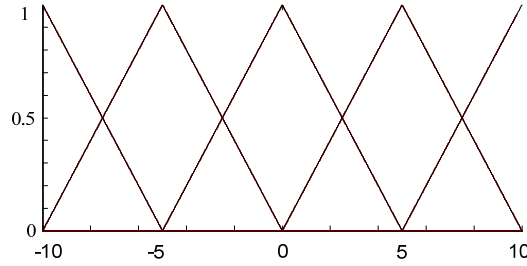


Figura 3.14 Universo de discurso con funciones de membresía triangulares parametrizado en $A = 10$, y $N = 5$.

Para las funciones de membresía Gaussianas sucede en forma similar, pero los datos requeridos en este caso son media y desviación estándar. Con lo que los centros de conjuntos equivalen a la media y la desviación estándar sería aproximadamente $1/3$ de d_i . Es decir de la distancia entre centros.

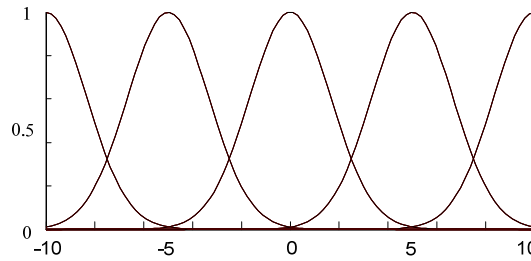


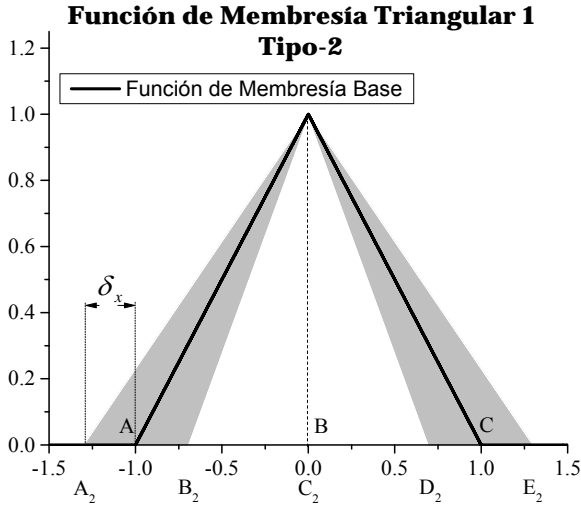
Figura 3.15 Universo de discurso con funciones de membresía Gaussianas parametrizado en $A = 10$, $d_i = 5$, $\sigma = 5/3$ y $N = 5$.

Para los conjuntos difusos tipo-2 caracterizados por su huella de incertidumbre (ver sección 2.3), es necesario expresar dicha incertidumbre mediante dos funciones de membresía denominadas: superior (UMF) e inferior (LMF). Con lo que se utiliza (3. 8) y (3. 9), para el modelado de las funciones de membresía más un parámetro δ_x que define la incertidumbre en los conjuntos difusos utilizados en este trabajo, **Figura 3.16**.

La incertidumbre en los conjuntos difusos de entrada se define como:

$$\begin{aligned} \delta_e &:= \text{Incertidumbre en los conjuntos difusos del universo del error.} \\ \delta_{\Delta e} &:= \text{Incertidumbre en los conjuntos difusos del cambio del error} \end{aligned}$$

La figura dsiguiente describe las funciones de mebresía utilizadas en este trabajo. Estos fueron tomados de [3][36][37][47][48][59]. La incertidumbre añadida en los conjuntos difusos es de tipo simétrica. Además, por motivos de simplificación, es igual en todos los conjuntos difusos presentes en el universo de discurso. Aunque en aplicaciones prácticas puede no ser así.



$$A_2 = A - \delta_x, B_2 = A + \delta_x;$$

$$C_2 = B,$$

$$D_2 = C - \delta_x, E_2 = C + \delta_x;$$

con: $A < B < C$

$$LMF \begin{cases} 0 & D_2 \leq x \text{ or } x \leq B_2 \\ \frac{x - B_2}{C_2 - B_2} & B_2 < x \leq C_2 \\ \frac{x - D_2}{C_2 - D_2} & C_2 < x < D_2 \end{cases}$$

$$UMF \begin{cases} 0 & E_2 \leq x \text{ or } x \leq A_2 \\ \frac{x - A_2}{C_2 - A_2} & A_2 < x \leq C_2 \\ \frac{x - E_2}{C_2 - E_2} & C_2 < x < E_2 \end{cases}$$

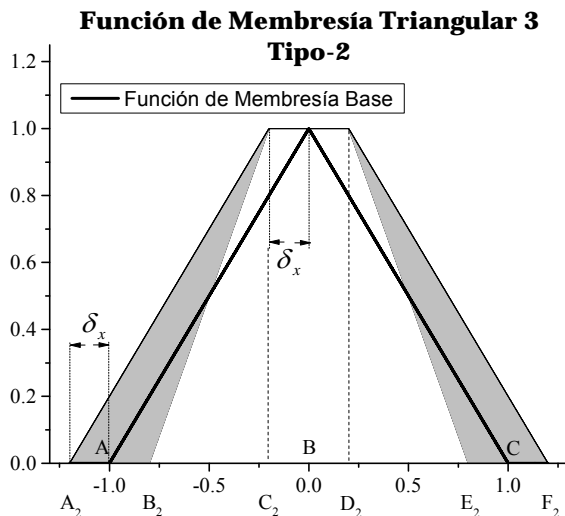
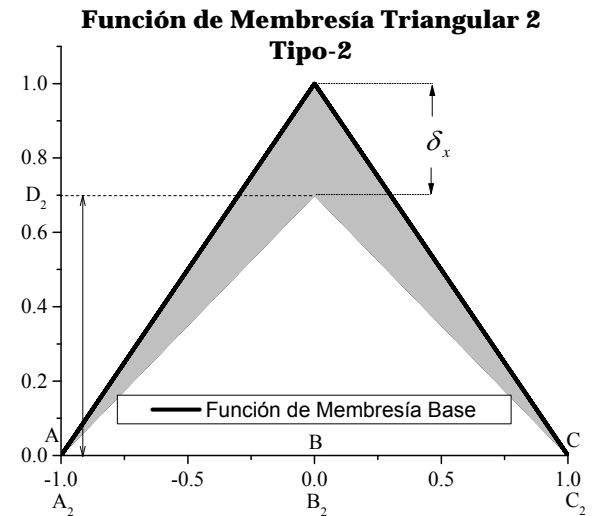
$$A_2 = A, B_2 = B;$$

$$C_2 = C, D_2 = 1 - \delta_x$$

$$LMF \begin{cases} 0 & C_2 \leq x \text{ or } x \leq A_2 \\ \left(\frac{x - A_2}{B_2 - A_2}\right) D_2 & A_2 < x < B_2 \\ \left(\frac{C_2 - x}{C_2 - B_2}\right) D_2 & B_2 < x < C_2 \end{cases}$$

$$UMF \begin{cases} 0 & C_2 \leq x \text{ or } x \leq A_2 \\ \frac{x - A_2}{B_2 - A_2} & A_2 < x < B_2 \\ \frac{C_2 - x}{C_2 - B_2} & B_2 < x < C_2 \end{cases}$$

con: $A < B < C$



$$A_2 = A - \delta_x, B_2 = A + \delta_x;$$

$$C_2 = B - \delta_x, D_2 = B + \delta_x$$

$$E_2 = C - \delta_x, G_2 = C + \delta_x;$$

con: $A < B < C$

$$LMF \begin{cases} 0 & E_2 \leq x \text{ or } x \leq B_2 \\ \frac{x - B_2}{C_2 - B_2} & B_2 < x < C_2 \\ 1 & C_2 \leq x < D_2 \\ \frac{E_2 - x}{E_2 - D_2} & D_2 \leq x < E_2 \end{cases}$$

$$UMF \begin{cases} 0 & F_2 \leq x \text{ or } x \leq A_2 \\ \frac{x - A_2}{C_2 - A_2} & A_2 < x < C_2 \\ 1 & C_2 \leq x < D_2 \\ \frac{F_2 - x}{F_2 - D_2} & D_2 \leq x < F_2 \end{cases}$$

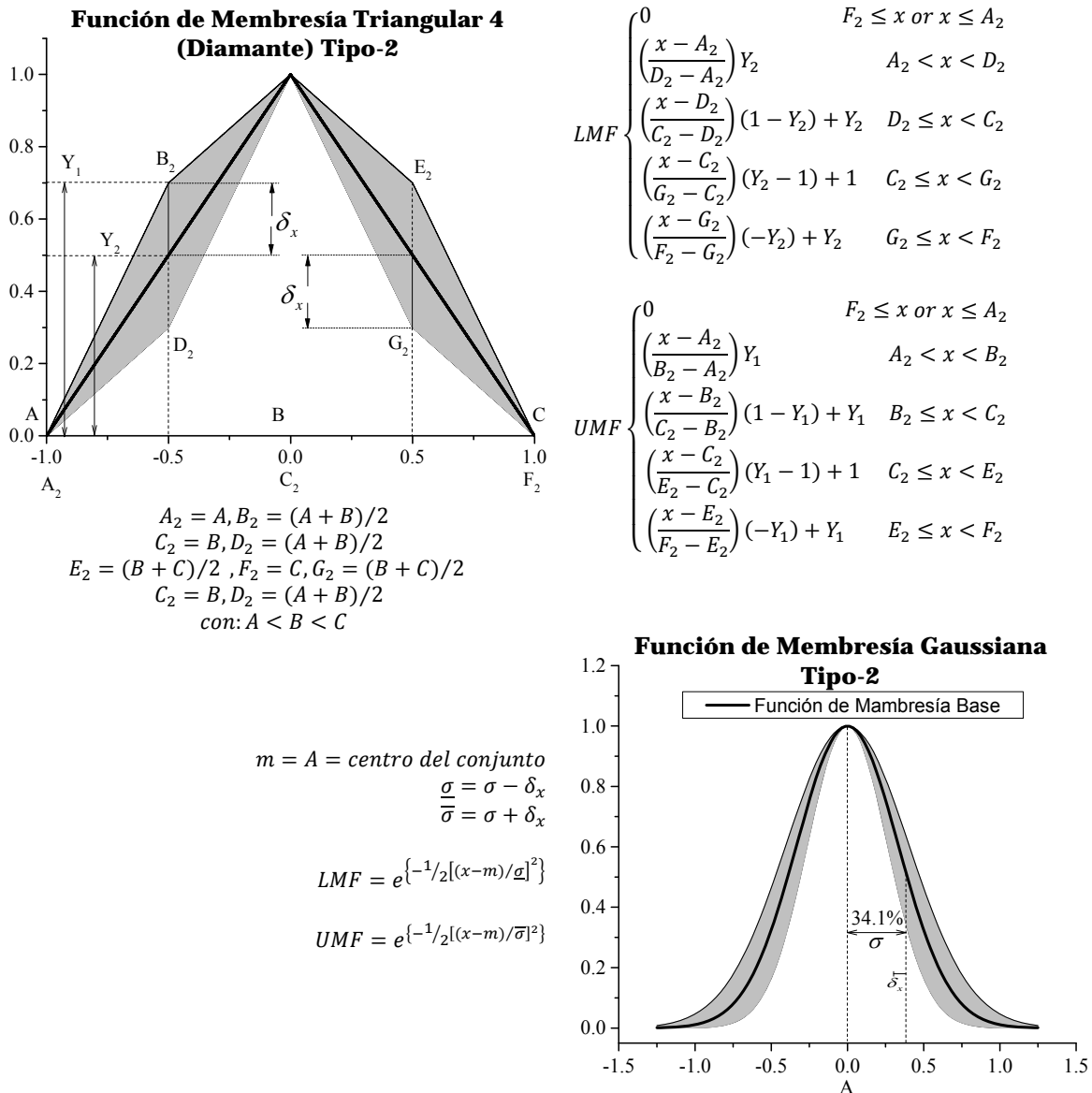


Figura 3.16 Funciones de membresía para diferentes conjuntos tipo-2 en intervalos.

3.4.2 Parametrización de las barras de salida

Para las barras de salida únicamente se define el centro de barra, en donde se considera el máximo valor de membresía según el sistema de inferencia min-max. Para el caso del controlador difuso tipo-2, es posible añadir cierta incertidumbre en esta barra de salida desplazando hacia la izquierda y derecha del centro de barra, pudiendo ser la incertidumbre simétrica o no simétrica.

Incertidumbre en las barras del universo de discurso de salida se define mediante:

- $c :=$ centro de barra en el universo de salida
- $SL =$ Incertidumbre a la izquierda de c
- $SR =$ Incertidumbre a la derecha de c

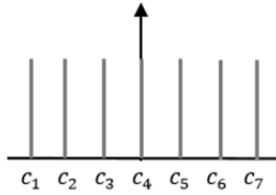


Figura 3.17 Universo de salida con siete funciones tipo barra.



Figura 3.18 Incertidumbre en barra de salida.

3.5 Superficie de control

La superficie de control de un controlador lógico difuso muestra el mapeo no lineal de las variables de entrada a la variable de salida. En otras palabras, permite ver en forma gráfica el comportamiento de la variable de salida del controlador ante la variación de las variables de entrada del mismo. Es una forma de representar en forma compacta el comportamiento no lineal del controlador difuso [45]. Nótese también que esta representación se encuentra limitada a dos variables de entrada al controlador como máximo [45][46]. Y su forma no lineal depende del valor de los parámetros del controlador [45][46], y de las reglas difusas utilizadas por el controlador.

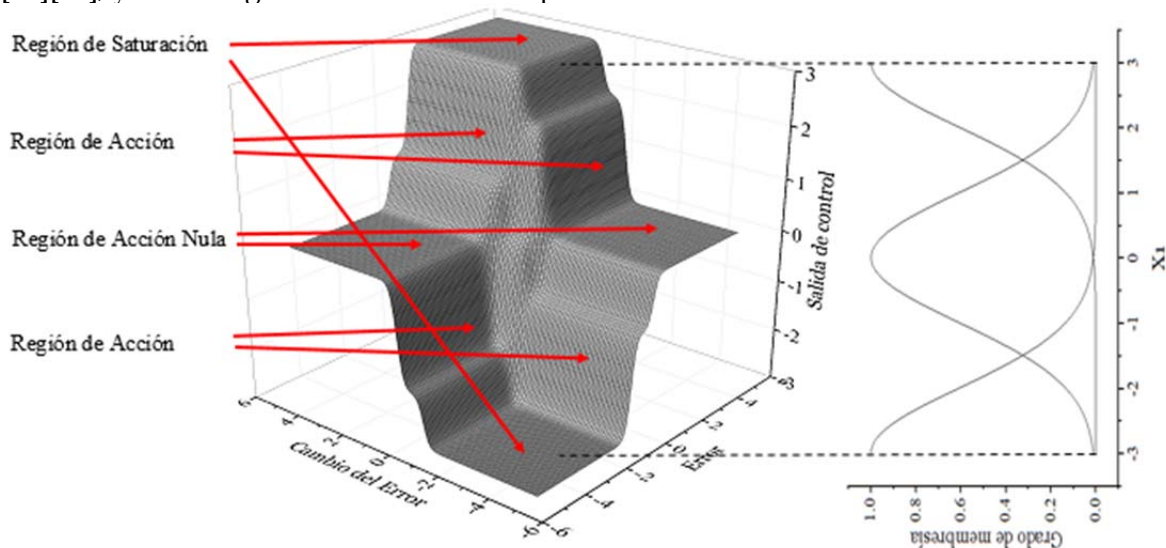


Figura 3.19 Superficie de control del T1 FLC con conjuntos difusos de entrada simétricos y matriz de conocimiento difuso simétrica..

La figura anterior muestra es el resultado de utilizar conjunto difusos tipo-1 con conjuntos difusos difusos de entrada uniformemente distribuidos. Es decir con traslape del 50%. Y además matriz de conocimiento difuso con consecuentes distribuidos simétricamente (ver Figura 3.6 y **Tabla 3.1**).

Dada su forma se identifican tres regiones:

Región de saturación: donde el controlador proporciona el máximo incremento positivo o negativo para su variable de salida \dot{u} .

Región de acción: donde el controlador actúa ante la variación sus cariables de entrada e y Δe .

Región de saturación: donde el controlador proporciona un incremento nulo para su variable de salida \dot{u} .

Nótese que los centros de las funciones consecuentes se encuentran distribuidas uniformemente al igual que la matriz de conocimiento difuso (Tabla 3.1). esta superficie de control puede ser distorciónada por ejemplo:

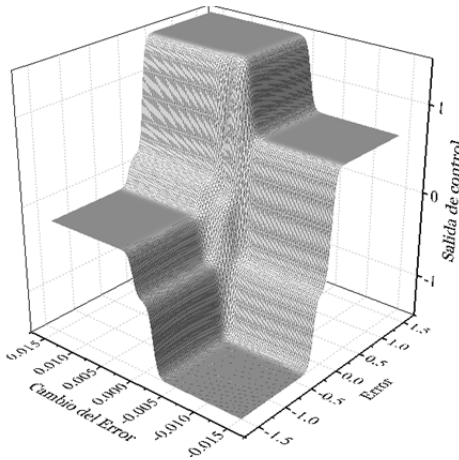


Figura 3.20 Superficie de control para los parámetros en Figura 3.6, Figura 3.21 y Tabla 3.4.

Tabla 3.4 Matriz de conocimiento difuso para la superficie de control de la Figura 3.20

e	Δe	N	CE	P
N	NG	NG	NM	
CE	NP	CE	PP	
P	PM	PG	PG	

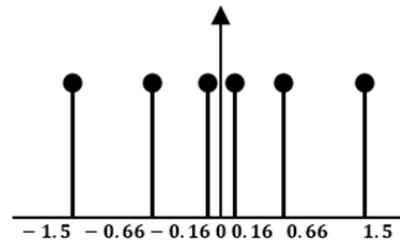


Figura 3.21 Universo de salida no uniforme para la Figura 3.20.

3.6 Fuentes de Incertidumbre a Simular en el Controlador

Dado que la principal justificante del uso de conjuntos difusos tipo-2 dentro de los controladores difusos, es la presencia de incertidumbre. En el presente trabajo se realizan pruebas con:

- Incertidumbre (ruido) en el lazo de retroalimentación del voltaje en las terminales del generador V_{ter} .
- Incertidumbre en la señal de entrada del voltaje de referencia V_{ref} .

De esta forma las incertidumbres consideradas afectan directamente al sistema de control. No se consideran incertidumbres propias en el sistema de potencia. La **Figura 3.22** muestra la forma de realizar la implementación en las simulaciones realizadas, para evaluar el desempeño los controladores.

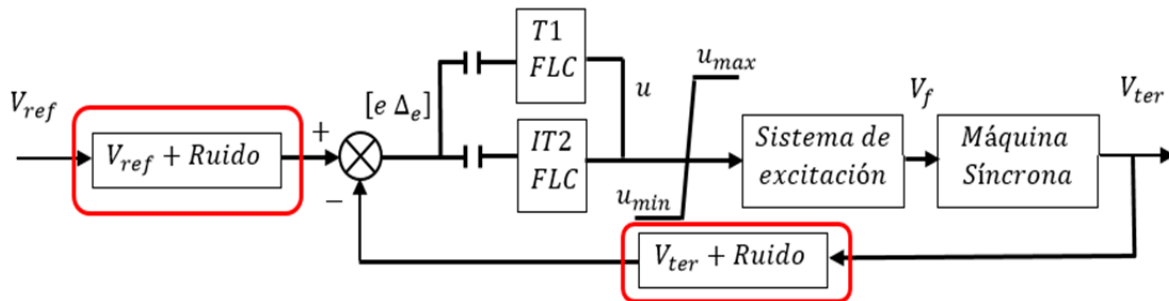


Figura 3.22 Diagrama de bloques del sistema de control con fuentes de incertidumbre.

Note que la diferencia con la **Figura 3.2** es la llamada a la función randn de la forma: $a + b * randn$; la cual genera un número aleatorio normalmente distribuido con media en a y desviación estándar b . Entonces, para un voltaje constante en terminales del generador con valor de 1 P.U como valor medio en dicho valor de tensión y desviación estándar del 1% del valor del voltaje terminal, la llamada a la función sería siguiente:

$$V_{ter_ruido}(k) = V_{ter} + V_{ter} * 0.01 * rand; \quad (3.11)$$

Donde $v_{ter_ruido}(k)$ es un vector para almacenar el voltaje con incertidumbre en la iteración k .

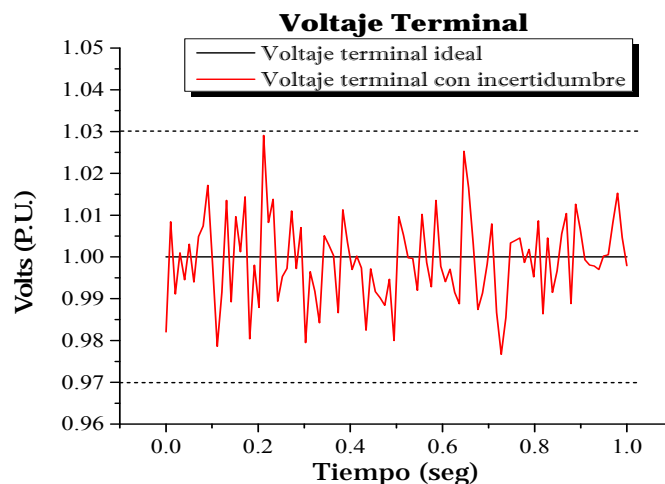


Figura 3.23 Voltaje en terminales ideal y con incertidumbre.

Con un valor para la desviación estándar igual 1% del valor actual en el voltaje en terminales, y como la simulación de la incertidumbre es normalmente distribuida. Se establece una banda de variación de $\pm 3\%$ cuando el valor del voltaje en terminales se mantiene constante. Esto debido a que el 99.7% de los valores de una variable aleatoria continua con distribución normal residen en ± 3 veces la desviación estándar.

Si denotamos Ruido a:

$$Ruido = V_{ter} * 0.01 * rand; \quad (3.12)$$

El error del sistema se redefine como:

$$\text{Error} = V_{\text{ref}} - (V_{\text{ter}} + \text{Ruido}); \quad (3.13)$$

Note que el término entre paréntesis en (3.13) es idéntico al lado derecho de (3.11). Y cuando se hace $\text{Ruido} = 0$; todo se reduce al caso ideal; es decir medición sin incertidumbre.

La incertidumbre del voltaje en terminales y el voltaje de referencia, se analizan por separado en 4.5 y 4.8.

3.7 Caso Ejemplo de Operaciones en el IT2 FLC

En esta sección se ilustra para un mejor entendimiento, las operaciones realizadas por el controlador difuso tipo-2 en cada paso de integración de la simulación del sistema. Los parámetros de simulación son los siguientes.

Tabla 3.5 Parámetros del Controlador tipo-2.	
$A = 0.3, B = 0.003$	
$\delta_e = 0.5; \delta_{\Delta e} = 0.5$	
$C = 3; SL = 0.3; SR = 0.1$	
Num. de barras de salida = 7	
Num. de reglas = 9	
Función de membresía triangular 2 tipo-2	

Supongamos que para el instante de tiempo $t = 1.136\text{s}$ tenemos un vector de entrada al controlador difuso tipo-2 $X = [0.0397 \ -0.0025]$

Los valores de membresía para el universo de entrada del error son:

$$N_e = [0 \ 0] \quad CE_e = [0.4339 \ 0.8677] \quad P_e = [0.0661 \ 0.1323]$$

$$N_{\Delta e} = [0.4236 \ 0.8473] \quad CE_{\Delta e} = [0.0764 \ 0.1527] \quad P_{\Delta e} = [0 \ 0]$$

Note que el subíndice indica el universo de entrada.

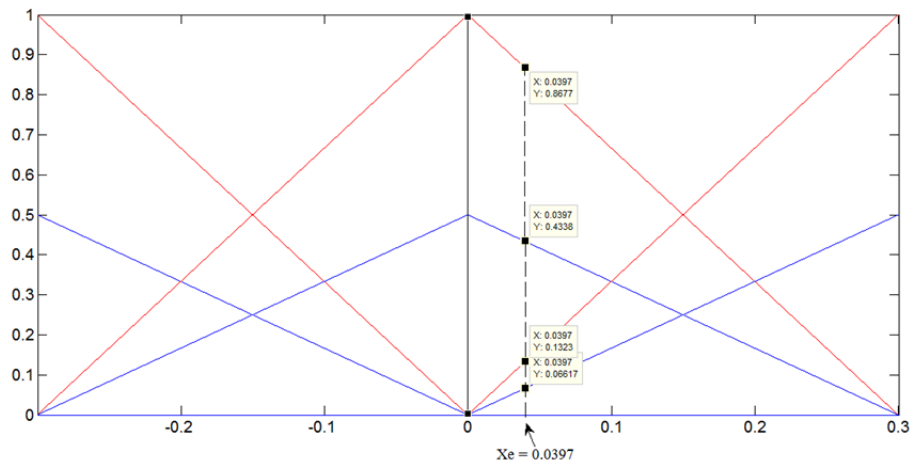


Figura 3.24 Universo de entrada del error.

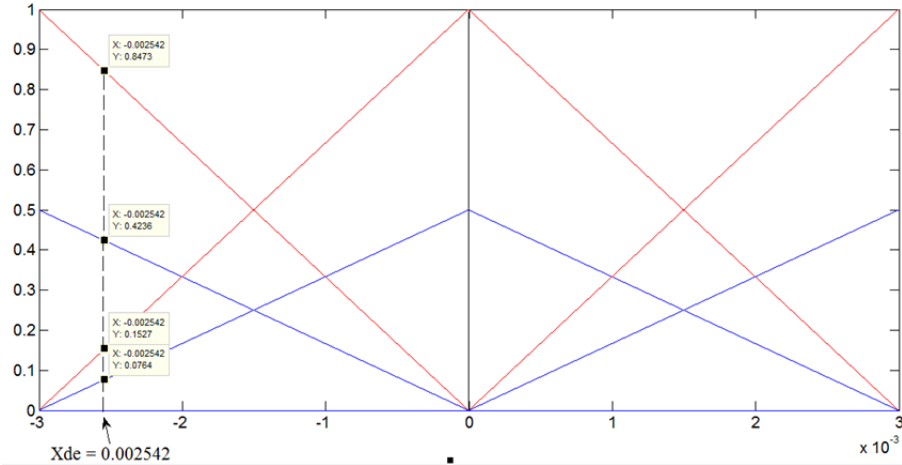


Figura 3.25 Universo de entrada del cambio de error.

Para las nueve reglas difusas se tiene:

Tabla 3.6 Inferencia del controlador tipo-2, operador mínimo.		
Membresía	Centro	Incertidumbre en barra
$[0 \wedge 0.4236, 0 \wedge 0.8473] = [0 \ 0]$	$R_{1,1} = -3$	$[-3.3 \ -2.9]$
$[0 \wedge 0.0764, 0 \wedge 0.1527] = [0 \ 0]$	$R_{1,2} = -2$	$[-2.3 \ -1.9]$
$[0 \wedge 0, 0 \wedge 0] = [0 \ 0]$	$R_{1,3} = 0$	$[0 \ 0]$
$[0.4339 \wedge 0.4236, 0.8677 \wedge 0.8473] = [0.4236 \ 0.8473]$	$R_{2,1} = -1$	$[-1.3 \ -0.9]$
$[0.4339 \wedge 0.0764, 0.8677 \wedge 0.1527] = [0.0764 \ 0.1527]$	$R_{2,2} = 0$	$[0 \ 0]$
$[0.4339 \wedge 0, 0.8677 \wedge 0] = [0 \ 0]$	$R_{2,3} = 1$	$[0.7 \ 1.1]$
$[0.0661 \wedge 0.4236, 0.1323 \wedge 0.8473] = [0.0661 \ 0.1323]$	$R_{3,1} = 0$	$[0 \ 0]$
$[0.0661 \wedge 0.0764, 0.1323 \wedge 0.1527] = [0.0661 \ 0.1323]$	$R_{3,2} = 2$	$[1.7 \ 2.1]$
$[0.0661 \wedge 0, 0.1323 \wedge 0] = [0 \ 0]$	$R_{3,3} = 3$	$[2.7 \ 3.1]$

Aplicando el operador máximo

Tabla 3.7 Inferencia del controlador tipo-2, operador máximo.		
Membresía	Centro	Incertidumbre en barra
$[0 \ 0]$	$R_{1,1} = -3$	$[-3.3 \ -2.9]$
$[0 \ 0]$	$R_{1,2} = -2$	$[-2.3 \ -1.9]$
$[0.4236 \ 0.8473]$	$R_{2,1} = -1$	$[-1.3 \ -0.9]$
$[0.0764 \ 0.1527]$	$R_{2,2} = 0$	$[0 \ 0]$
$[0 \ 0]$	$R_{2,3} = 1$	$[0.7 \ 1.1]$
$[0.0661 \ 0.1323]$	$R_{3,2} = 2$	$[1.7 \ 2.1]$
$[0 \ 0]$	$R_{3,3} = 3$	$[2.7 \ 3.1]$

De aquí es posible utilizar el algoritmo KM y el EKM por lo que se muestra el desarrollo de ambos:

Con el algoritmo de Karnik & Mendel

Para \underline{y}

Centros de barra

$$[-3.3 \ -2.3 \ -1.3 \ 0 \ 0.7 \ 1.7 \ 2.7]$$

SEPI-ESIME-ZACATENCO CAPÍTULO 3. SISTEMA DE EXCITACIÓN UTILIZANDO CONTROL DIFUSO TIPO-2

$$\theta_i = [0 \ 0 \ 0.6355 \ 0.1146 \ 0 \ 0.0992 \ 0]$$

$$\underline{y}'_l = \frac{-3.3 \times 0 - 2.3 \times 0 - 1.3 \times 0.6355 + 0 \times 0.1146 + 0.7 \times 0 + 1.7 \times 0.0992 + 2.7 \times 0}{0 + 0 + 0.6355 + 0.1146 + 0 + 0.0992 + 0}$$

$$\underline{y}'_l = -0.7742$$

$$k = 3$$

De aquí

$$\theta_{i_correg} = [0 \ 0 \ 0.8473 \ 0.0764 \ 0 \ 0.0661 \ 0]$$

$$\underline{y}_l = \frac{-3.3 \times 0 - 2.3 \times 0 - 1.3 \times 0.8473 + 0 \times 0.0764 + 0.7 \times 0 + 1.7 \times 0.0661 + 2.7 \times 0}{0 + 0 + 0.8473 + 0.0764 + 0 + 0.0661 + 0}$$

$$\underline{y}_l = -0.9993$$

Como $\underline{y}'_l \neq \underline{y}_l$ entonces $\underline{y}'_l = -0.9993$

$$k = 3$$

$$\theta_{i_correg} = [0 \ 0 \ 0.8473 \ 0.0764 \ 0 \ 0.0661 \ 0]$$

$$\underline{y}_l = \frac{-3.3 \times 0 - 2.3 \times 0 - 1.3 \times 0.8473 + 0 \times 0.0764 + 0.7 \times 0 + 1.7 \times 0.0661 + 2.7 \times 0}{0 + 0 + 0.8473 + 0.0764 + 0 + 0.0661 + 0}$$

$$\underline{y}_l = -0.9993$$

Como $\underline{y}'_l = \underline{y}_l$ entonces $\underline{y} = -0.9993, L = 3$ e $iter = 2$

Para \bar{y}

Centros de barra

$$[-2.9 \ -1.9 \ -0.9 \ 0 \ 1.1 \ 2.1 \ 3.1]$$

$$\theta_i = [0 \ 0 \ 0.6355 \ 0.1146 \ 0 \ 0.0992 \ 0]$$

$$\bar{y}'_r = \frac{-2.9 \times 0 - 1.9 \times 0 - 0.9 \times 0.6355 + 0 \times 0.1146 + 1.1 \times 0 + 2.1 \times 0.0992 + 3.1 \times 0}{0 + 0 + 0.6355 + 0.1146 + 0 + 0.0992 + 0}$$

$$\bar{y}'_r = -0.4282$$

$$k = 3$$

De aquí

$$\theta_{i_correg} = [0 \ 0 \ 0.4236 \ 0.1527 \ 0 \ 0.1323 \ 0]$$

$$\bar{y}_r = \frac{-2.9 \times 0 - 1.9 \times 0 - 0.9 \times 0.4236 + 0 \times 0.1527 + 1.1 \times 0 + 2.1 \times 0.1323 + 3.1 \times 0}{0 + 0 + 0.4236 + 0.1527 + 0 + 0.1323 + 0}$$

$$\bar{y}_r = -0.1459$$

Como $\bar{y}'_r \neq \bar{y}_r$ entonces $\bar{y}'_r = -0.1459$

$$k = 3$$

De aquí

$$\theta_{i_correg} = [0 \ 0 \ 0.4236 \ 0.1527 \ 0 \ 0.1323 \ 0]$$

$$\bar{y}_r = \frac{-2.9 \times 0 - 1.9 \times 0 - 0.9 \times 0.4236 + 0 \times 0.1527 + 1.1 \times 0 + 2.1 \times 0.1323 + 3.1 \times 0}{0 + 0 + 0.4236 + 0.1527 + 0 + 0.1323 + 0}$$

$$\bar{y}_r = -0.1459$$

Como $\bar{y}'_r = \bar{y}_r$ entonces $\bar{y} = -0.1459, R = 3$ e $iter = 2$

Finalmente

$$u = \frac{-0.9993 - 0.1459}{2} = -0.5726$$

Con el algoritmo Mejorado de Karnik & Mendel

Para \underline{y}

Centros de barra

$$[-3.3 \ -2.3 \ -1.3 \ 0 \ 0.7 \ 1.7 \ 2.7]$$

$$k = 7/2.4 \approx 3$$

$$\frac{a}{b} = \frac{-3.3 \times 0 - 2.3 \times 0 - 1.3 \times 0.8473 + 0 \times 0.0764 + 0.7 \times 0 + 1.7 \times 0.0661 + 2.7 \times 0}{0 + 0 + 0.8473 + 0.0764 + 0 + 0.0661 + 0}$$

$$\frac{a}{b} = -0.9993$$

$$k' = 3$$

Como $k' = k$ entonces $\underline{y} = -0.9993, L = 3$ e $iter = 1$

Para \bar{y}

$$[-2.9 \quad -1.9 \quad -0.9 \quad 0 \quad 1.1 \quad 2.1 \quad 3.1]$$

$$k = 7/1.7 \approx 4$$

$$\frac{a}{b} = \frac{-2.9 \times 0 - 1.9 \times 0 - 0.9 \times 0.4236 + 0 \times 0.0764 + 1.1 \times 0 + 2.1 \times 0.1323 + 3.1 \times 0}{0 + 0 + 0.4236 + 0.0764 + 0 + 0.1323 + 0}$$

$$\frac{a}{b} = -0.1635$$

$$k' = 3$$

Como $k' \neq k$

$$s = \text{signo}(3 - 4) = -1$$

$$a = -0.10341 + [0(0.1527 - 0.0764)] = -0.10341$$

$$b = 0.6323 + [(0.1527 - 0.0764)] = 0.7086$$

$$\frac{a}{b} = -0.1459$$

$$k = 3$$

$$k' = 3$$

Como $k' = k$ entonces $\bar{y} = -0.1459, R = 3$ e $iter = 2$

Finalmente

$$u = \frac{-0.9993 - 0.1459}{2} = -0.5726$$

Como puede observarse ambos controladores convergen al mismo resultado. Para este caso el controlador infiere que la salida debe decrementar en $\dot{u} = -0.5726$.

3.7.1 Controlador difuso Tipo-2 como controlador difuso Tipo-1

Como se ha dicho, todo conjunto difuso tipo-2 en intervalos, es un conjunto difuso tipo-1 cuando su incertidumbre $\delta_i \rightarrow 0$. Por lo que un controlador difuso tipo-2 deberá comportarse como un controlador difuso tipo-1 cuando sus todas las incertidumbres de sus conjuntos difusos en intervalos se hacen cero. En lo siguiente se muestra, que así sucede en la simulación del controlador tipo-2.

La **Figura 3.26** muestra los datos de entrada necesarios para los controladores difusos tipo-1 y tipo-2.

SEPI-ESIME-ZACATENCO CAPÍTULO 3. SISTEMA DE EXCITACIÓN UTILIZANDO CONTROL DIFUSO TIPO-2

```

%*****
%Datos del Controlador difuso tipo 1:

% Universo de discurso del error
1

% Universo de discurso del cambio del error
0.005

% Número de reglas difusas: 9 reglas [9]    25 reglas [25]    49 reglas [49]
25

% Numero de barras de salida: 5 [5]    7 [7]
7

% Universo de discurso de Salida
3

a)
%*****
%Datos del Controlador difuso tipo 2:

% Universo de discurso del error
1

%Incertidumbre para conjuntos difusos del error
0

% Universo de discurso del cambio del error
0.005

%Incertidumbre para conjuntos difusos del error
0

% Número de reglas difusas: 9 reglas [9]    25 reglas [25]    49 reglas [49]
25

% Numero de barras de salida: 5 [5]    7 [7]
7

```

Figura 3.26 Datos de simulación requeridos por los controladores: (a) difuso tipo-1, (b) difuso tipo-2.

Como puede apreciarse, se requieren para ambos controladores los parámetros de los universos de entrada. Para el controlador difuso tipo-2 se requiere además la incertidumbre asociada a sus conjuntos difusos. Es posible seleccionar una incertidumbre para el universo del error $\delta_e = 0$ y para el cambio del error $\delta_{\Delta e} = 0$, y mantener el universo de salida con funciones tipo barra, con lo que el desempeño del controlador será igual a un controlador difuso tipo-1.

Con el objeto de comprobar este hecho, se desarrolló el algoritmo del control difuso tipo-1 y se compararon los resultados entre este y el control difuso tipo-2 en intervalos utilizando el algoritmo convencional Karnik y Mendel haciendo:

$$\begin{aligned}
 \delta_e &= \delta_{\Delta e} = 0 \\
 SL &= SR = 0
 \end{aligned}
 \tag{3.14}$$

Las simulaciones fueron realizadas con el generador síncrono sin carga, línea de transmisión corta y sujeto a una falla trifásica que ocurre en $t=1$ seg con duración de 6 ciclos. Los parámetros de ambos controladores se resumen en la **Tabla 3.8**.

Tabla 3.8 Parámetros de Controladores
$A_1 = A_2 = 1$
$B_1 = B_2 = 0.005$
$C_1 = C_2 = 3$
<i>Num. de barras (singletons) de salida = 25</i>
<i>Num. de reglas = 25</i>

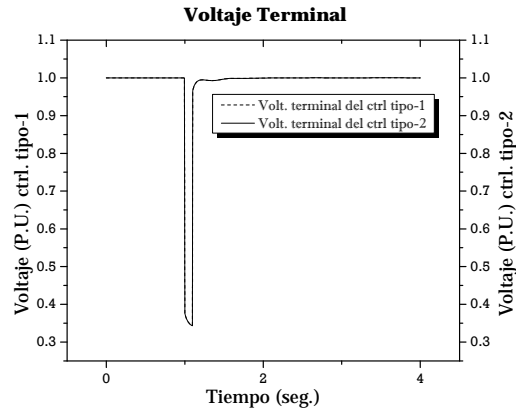


Figura 3.27 Comportamiento del voltaje terminal de la maquina síncrona con controladores difusos con parámetros idénticos y $\delta_e = \delta_{\Delta e} = 0$.

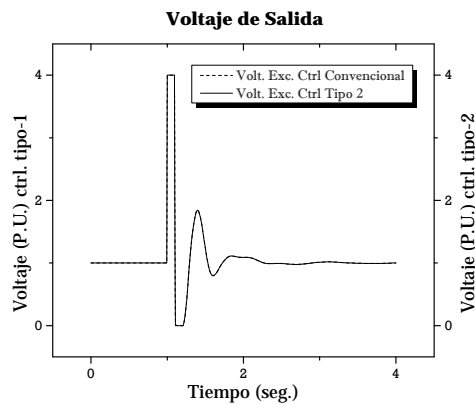


Figura 3.28 Comportamiento del voltaje de excitación de la maquina síncrona con controladores difusos con parámetros idénticos y $\delta_e = \delta_{\Delta e} = 0$.

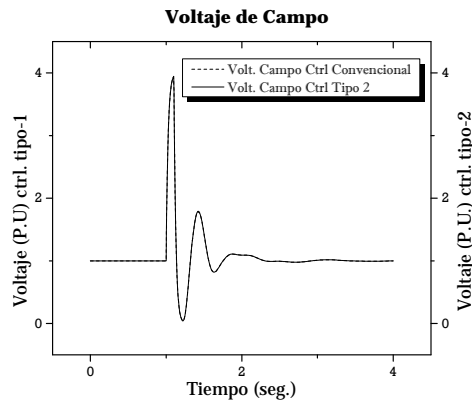


Figura 3.29 Comportamiento del voltaje de campo de la maquina síncrona con controladores difusos con parámetros idénticos y $\delta_e = \delta_{\Delta e} = 0$.

La superposición de las respuestas de ambos controladores, confirman el comportamiento idéntico de ambos algoritmos. No obstante se muestra numéricamente el valor de salida de ambos controladores para diferentes iteraciones de tiempo.

Tabla 3.9 Salida de Controladores

Tiempo	iter	Ctrl Convencional	Ctrl Tipo 2
0.000000	1	1.000000	1.000000
0.000000	2	1.000000	1.000000
0.004000	3	1.000000	1.000000
0.008000	4	1.000000	1.000000
0.012000	5	1.000000	1.000000
.	.	.	.
.	.	.	.
.	.	.	.
1.332000	335	1.485883	1.485883
1.336000	336	1.523125	1.523125
1.340000	337	1.557352	1.557352
1.344000	338	1.588484	1.588484
1.348000	339	1.617531	1.617531
.	.	.	.
.	.	.	.
.	.	.	.
1.992000	500	1.090034	1.090034
1.996000	501	1.090121	1.090121
2.000000	502	1.090232	1.090232
2.004000	503	1.090355	1.090355
2.008000	504	1.090486	1.090486

Los resultados anteriores dejan entrever el desempeño idéntico de ambos algoritmos. Por lo que en las simulaciones posteriores simplemente se hace (3. 14) en el algoritmo del IT2FLC, para cuando se requiere la simulación del controlador difuso convencional tipo-1.

3.8 Metodología de Sintonización del IT2 FLC

Varias técnicas se han desarrollado para la sintonización del IT2 FLC [19][46][47][48],[49]. Por simplicidad en este trabajo para el proceso de sintonización del IT2 FLC se toma al T1 FLC como base; es decir, primero se busca un ajuste adecuado de un T1 FLC y de ahí se añade incertidumbre en los conjuntos difusos de los antecedentes y consecuentes. De esta forma se tienen ciertas ventajas [19]:

- Una inteligente inicialización de los parámetros de un IT2 FLC.
- Un sistema base T1 FLC el cual puede ser comparado con el IT2 FLC, donde la única diferencia entre ellos es la incertidumbre añadida en los conjuntos difusos.
- Si se emplea un algoritmo de búsqueda inteligente como en [49] sólo requerirá buscar la cantidad de incertidumbre requerida para cada conjunto en particular.
- Cuando todas las incertidumbres presentes en los conjuntos difusos se hacen cero, se tiene un T1 FLC sintonizado.

El proceso de sintonización es el siguiente:

Para el T1 FLC [45]

- 1) Ajuste manual de los universos de discurso de entrada.
- 2) Ajuste manual del universo de discurso de salida

Tomando como base al T1 FLC, para el IT2 FLC

- 3) Ajuste manual de la incertidumbre a los universos de discurso de entrada.
- 4) Ajuste manual de la incertidumbre en el universo de discurso de salida.

Enseguida se describen brevemente:

1) Ajuste manual de los universos de discurso de entrada.

El ajuste manual requiere definir el número y forma de las funciones de membresía, además de inicializar los parámetros A y B para los universos de entrada del error y del cambio de error respectivamente y variarlos en forma conservadora. Esto es, variar un parámetro a la vez considerando la evolución del diagrama de fases. Además inicialmente se eligen A y B con un valor en C sugerido según el rango de variación deseable en la variable de salida del controlador [45]. Luego, la elección de C mejora según los valores elegidos para A y B .

2) Ajuste manual del universo de discurso de salida.

Requiere definir el número de barras de salida así como el parámetro C del controlador (ver sección 3.4.2), buscando mejorar la respuesta del controlador en base a la respuesta dinámica de la variable a controlar, el diagrama de fases y la superficie de control.

3) Ajuste manual de la incertidumbre a los universos de discurso de entrada.

Este paso requiere añadir incertidumbre en alguna forma a los conjuntos difusos triangulares y Gaussianos tratados en el presente trabajo (ver sección 3.4.1).

4) Ajuste manual de la incertidumbre en el universo de discurso de salida.

Se añade incertidumbre en las barras de salida según sección 3.4.2.

La elección primero de δ_e y $\delta_{\Delta e}$ y luego la de Sl y SR sugieren la misma idea de la elección ordenada de A , B y C en el T1 FLC.

Como se ha dicho, partiendo del controlador difuso tipo-1 sintonizado manualmente como base, se agrega incertidumbre a los conjuntos difusos en el espacio de entrada y posteriormente en los consecuentes del universo de discurso de salida. Con el fin de llevar un orden en el análisis se muestra el diagrama de flujo.

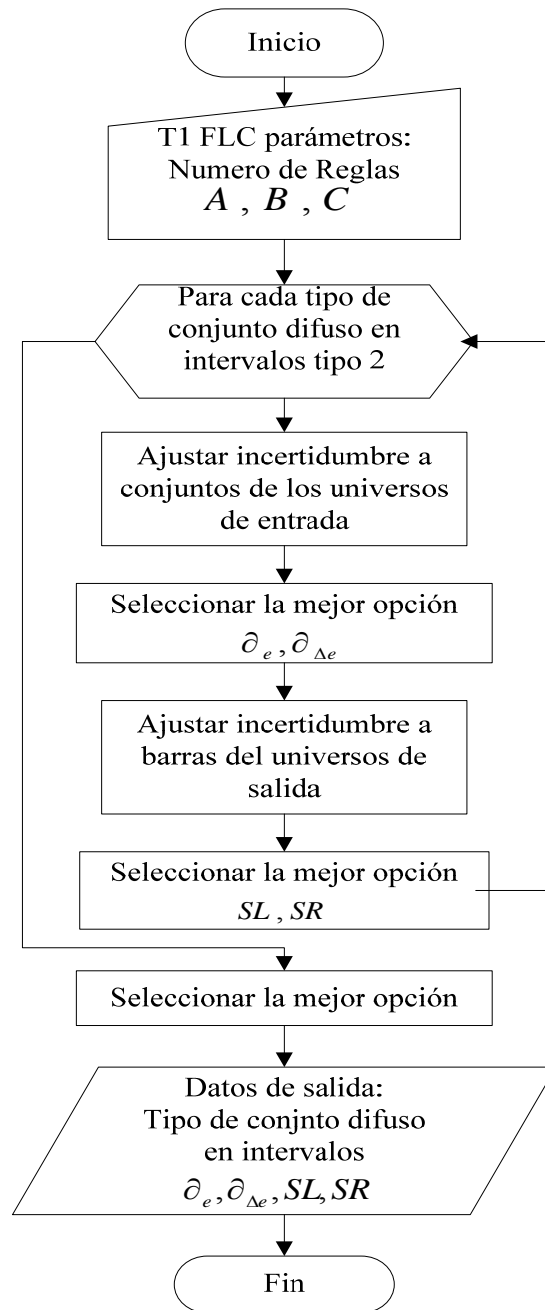


Figura 3.30 Diagrama de flujo del proceso de ajuste de incertidumbre en conjuntos difusos tipo-2.

Para la elección de los parámetros $A, B, C, \delta_e, \delta_{\Delta e}, SL$ y SR ; se consideran los índices de comportamiento ISE, IAE e ITAE, el digrama de fase del controlador y la respuesta dinámica de voltaje en terminales del generador. La interpretación de los índices de desempeño ISE, IAE e ITAE se muestra en el Apéndice 0.

Por su parte el diagrama de fases permite visualizar la dinámica de las variables de entrada $[e \Delta e]$ del controlador y su convergencia alrededor del origen. Excesivas vueltas alrededor del origen indican demasiadas oscilaciones en el voltaje terminal. Pero una

forma en espiral sugiere un recorrido adecuado alrededor de la matriz de reglas difusas del controlador.

El objetivo principal del controlador difuso tipo-1 y tipo-2 presentado en este trabajo es corregir la salida del controlador en forma no lineal en base al error y cambio de error presente en sus entradas, hasta obtener $[e \Delta e] \approx 0$. En este caso se utiliza únicamente el plano $(e, \Delta e)$ por lo que la variable tiempo se encuentra implícito en el diagrama.

Finalmente es posible para ambos controladores, la optimización de la matriz de reglas difusas por el método de búsqueda por Tabú. El método de búsqueda por Tabú no es el principal objeto de este trabajo por lo que se muestran únicamente algunas simulaciones del controlador utilizando el método. Siendo no necesario para la sintonización requerida en este trabajo.

Note que, para un sistema de dos entradas una salida:

- El número de reglas difusas afecta en forma directa al número de conjuntos difusos en los universos de entrada.
- El número de consecuentes afecta en forma directa al número de barras para la variable de salida del controlador.

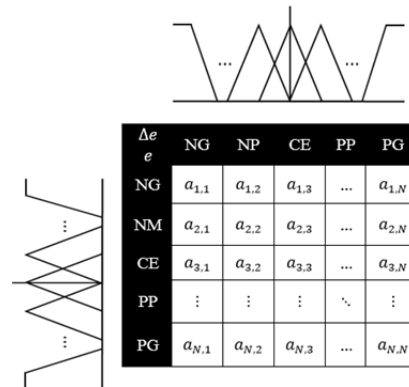


Figura 3.31 Matriz de asociación difusa.

Si NR es el número de reglas difusas y N es el número de conjuntos difusos en ambos universos de entrada, entonces:

$$N = \sqrt{NR} \tag{3.15}$$

El número de reglas disponibles en este trabajo son: 9, 25 y 49. Lo que permite una matriz de asociación difusa de 3×3 , 5×5 y 7×7 respectivamente.

CAPÍTULO 4

*Enseñarás a volar, pero no volarán tu vuelo. Enseñarás a soñar, pero no soñarán tu sueño.
Enseñarás a vivir, pero no vivirán tu vida. Sin embargo... en cada vuelo, en cada vida,
en cada sueño, perdurará siempre la huella del camino enseñado.
Madre Teresa de Calcuta. Religiosa.*

4. PRUEBAS Y RESULTADOS

4.1 Introducción

En el presente capítulo se presentan los resultados de la sintonización de los controladores T1 FLC e IT2 FLC y pruebas de estos en diversas situaciones con el generador sujeto a carga eléctrica en sus terminales de salida [9][12].

Las pruebas y simulaciones consideran un generador síncrono de 645 MVA[8], conectado a un bus infinito mediante una línea de transmisión corta y larga, términos que hacen referencia al valor de la impedancia en la línea, según [8][9][12][17][18]:

Tabla 4.1 Datos de líneas de transmisión.	
Línea de transmisión	Impedancia Ω
Corta	$0.024 + j0.115$
Larga	$0.12 + j1.1$

La sintonización se realizó con el generador síncrono en vacío. El proceso se detalla en la sección 4.2 y 4.3 .

Una vez determinados los parámetros del IT2 FLC, el controlador es explorado ante pequeños y grandes disturbios para casos con 9, 25 y 49 reglas de conocimiento difuso. Las simulaciones consideran al IT2 FLC con el algoritmo EKM. La comparación entre los algoritmos KM y EKM se detallan en el apéndice C.

Los disturbios a los que el sistema fue sometido se clasifican en:

- Pequeños disturbios: pequeños cambios en el voltaje de referencia y toma de carga iniciando con el generador síncrono en vacío.
- Gran disturbio: corto circuito severo en el bus infinito.

En la sección 4.4 se considera el sistema con línea de transmisión corta sujeto a un disturbio grande, el cual permite analizar al controlador en situaciones de corto circuito con duración de hasta diez ciclos de la frecuencia fundamental de generador. El desempeño del controlador en este caso se cuantifica mediante los índices de error ISE (considera sobretiros), ITAE (considera error en estado estable) e IAE (considera tanto sobretiros como error en estado estable). En la sección 4.5 se considera el mismo sistema sujeto al disturbio grande, pero con incertidumbre en la lectura del voltaje terminal. Aquí se estudia el manejo de incertidumbre en el T1 FLC y el IT2 FLC.

La sección 4.6 considera el caso de línea corta sujeta a un disturbio grande, sin incertidumbre.

En la sección 4.7 considera el sistema con línea de transmisión conrta sujeto a un pequeño disturbio, esto es un pequeño cambio en la referencia de voltaje. Esto permiten analizar el desempeño del controlador ante una pequeña entrada escalón al sistema y medir comportamientos como sobretiro, tiempo de respuesta, tiempo en alcanzar el valor pico y tiempo de estabilización [11]. La sección 4.8 las mismas simulaciones que 4.7, pero cn incertidumbre en la señal de voltaje de referencia, esto con el fin de tener una segunda fuente de incertidumbre dentro del sistema de control en el cual analizar el contolador difuso ipo-2.

Por último las secciones 4.9 y 4.10 muestran los resultados de toma de carga del generador síncrono para el caso línea corta y larga respectivamente. Esto para analizar el controlador durante los arranques con el generador en vacío y hasta alcanzar una condición de plena carga.

4.2 Sintonización manual para una línea de transmisión corta y generador síncrono en vacío

La sintonización manual de los universos de entrada, implica la selección de los adecuados parámetros A y B para los universos de discurso [45] (ve Figura 3.12).

Se realizan extensas simulaciones con el fin de buscar los mejores parámetros adecuados para el T1 FLC. En estas simulaciones se expanden y contraen los universos de entrada buscando los mejores resultados b)

Figura 4.1.

Las simulaciones siguientes consideran una falla trifásica en bus infinito, a 1 segundo de simulación y con duración de 10 ciclos (≈ 0.16 seg), y el generador síncrono en vacío. Se]muestran también los resultados utilizando el regulador de voltaje tipo ST1.

Como se menciona en la sección 3.8, se procede a realizar la sintonización del controlador IT2 FLC a partir del T1 FLC. Se considera el mínimo conocimiento del posible para este caso en la matriz de reglas difusas. Esto es nueve reglas difusas lo que implica tres funciones de membresía por cada universo de entrada. El proceso para 25 y 49 reglas de conocimiento es idéntico, pero en este trabajo únicamente se realiza el análisis con 9 reglas de conocimiento. Los resultados de las simulaciones con 25 y 49 reglas de conocimiento consideran los mismos parámetros obtenidos de la sintonización con 9 reglas de conocimiento. Teniendo cuidado de que la incertidumbre δ_e y δ_{Δ_e} obtenida para 9 reglas no produzca un traslape excesivo entre conjuntos tipo-2 en intervalos para los casos con 25 y 49 reglas de conocimiento. Para la elección de parámetros se cosideraran los índices de desempeño ISE, IAE e ITAE, el diagrama de fase y la respuesta dinámica del voltaje terminal.

Una vez sintonizado el controlador en vacío se procede a realizar simulaciones con el generador síncrono bajo carga y bajo ciertas condiciones iniciales.

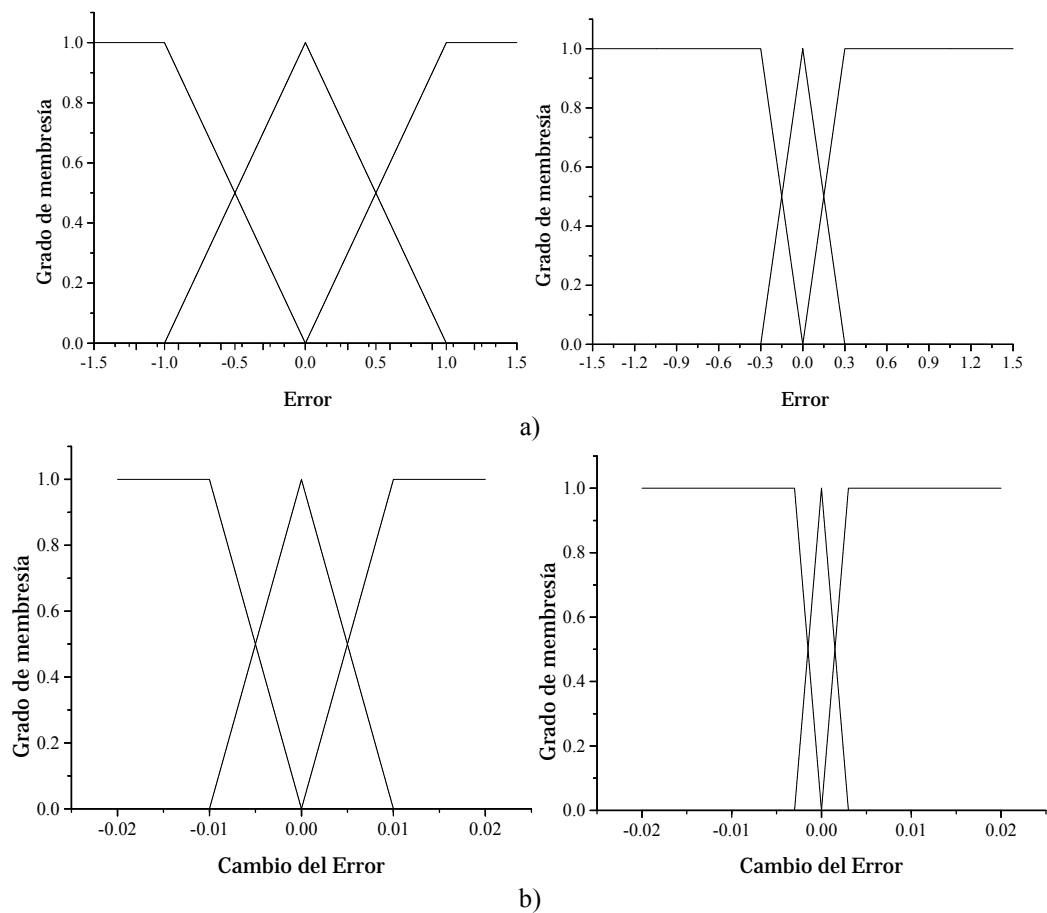


Figura 4.1 Expansión y contracción de los universos de entrada: a) universo del error; b) universo del cambio del error.

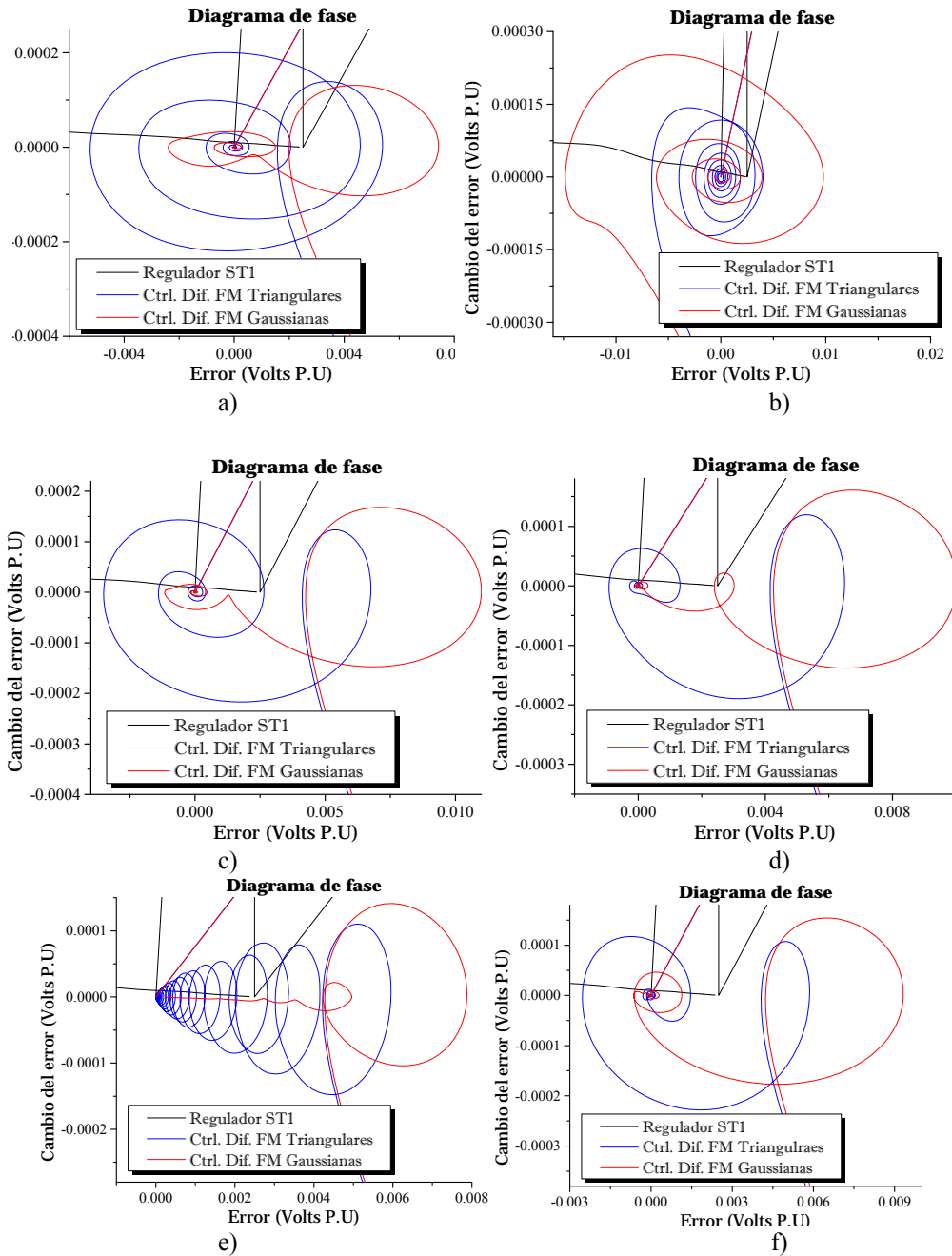


Figura 4.2 Evolución del diagrama de fases para línea corta: a) $A = 0.5, B = 0.01$; b) $A = 0.5, B = 0.02$; c) $A = 0.5, B = 0.005$; d) $A = 0.5, B = 0.003$; e) $A = 0.7, B = 0.001$; f) $A = 0.3, B = 0.003$.

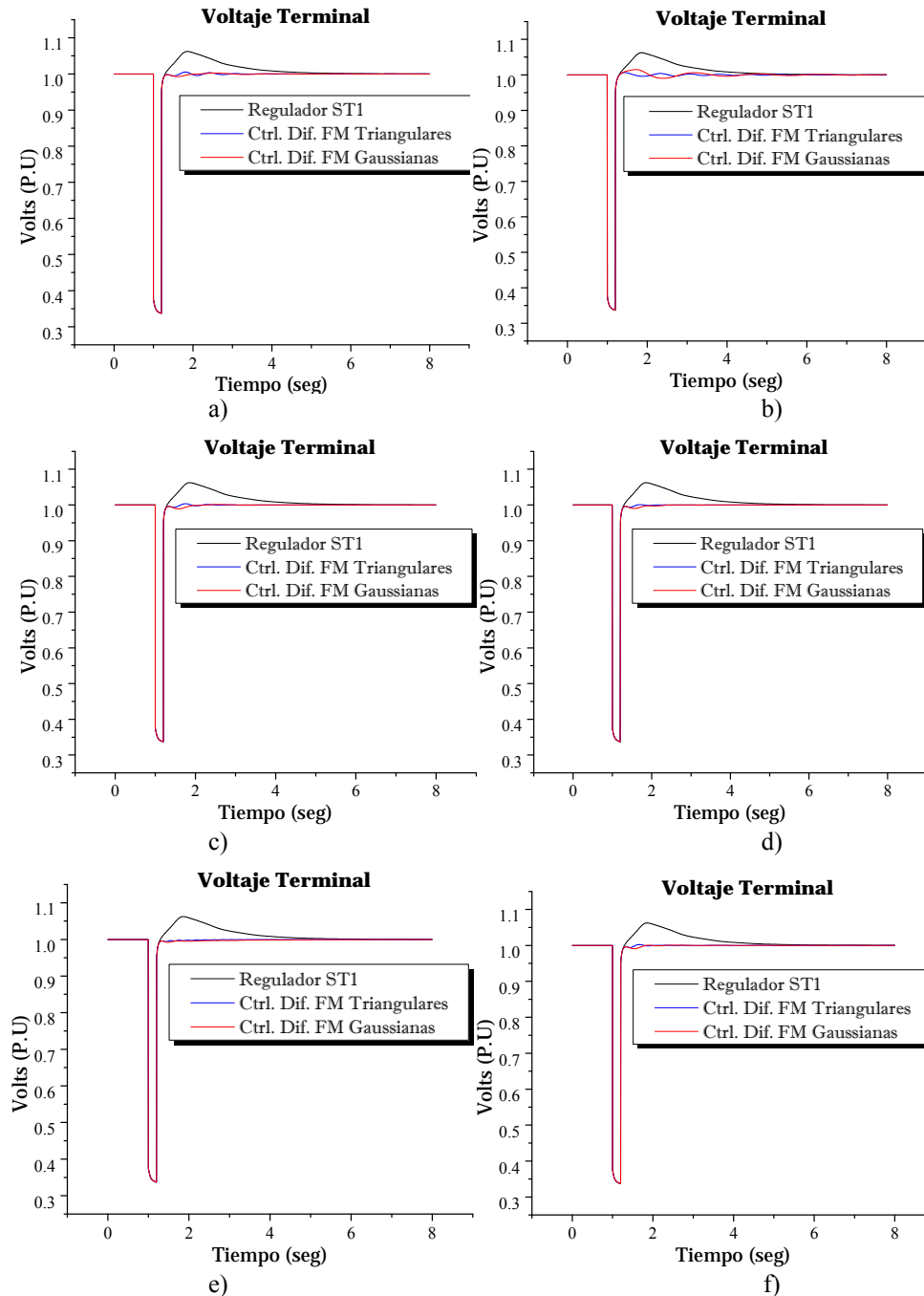


Figura 4.3 Evolución del voltaje terminal para línea corta: a) $A = 0.5, B = 0.01$; b) $A = 0.5, B = 0.02$; c) $A = 0.5, B = 0.005$; d) $A = 0.5, B = 0.003$; e) $A = 0.7, B = 0.001$; f) $A = 0.3, B = 0.003$.

La siguiente tabla muestra un compendio de los índices del error, de las simulaciones anteriores, para los casos: función de membresía triangular y gaussiana. Los valores sombreados indican los valores mínimos para cada índice.

Tabla 4.2 Índices para sintonización del T1 FLC, línea corta.							
Parámetro		Función de Membresía Triangular			Función de Membresía Gaussiana		
A	B	ISE	IAE	ITAE	ISE	IAE	ITAE
0.5	0.01	0.07129946	0.11405082	0.12702978	0.07129465	0.11263078	0.12475198
0.5	0.02	0.07131586	0.11871176	0.14314069	0.07146117	0.13255288	0.18354779
0.5	0.008	0.07130526	0.11471295	0.12756963	0.07133569	0.11806201	0.13460040
0.5	0.005	0.07130386	0.11364170	0.12486278	0.07133210	0.11665496	0.13042537
0.5	0.003	0.07130223	0.11288707	0.12333603	0.07132719	0.11647351	0.12990170
0.5	0.001	0.07130483	0.11460086	0.12752647	0.07132950	0.12084234	0.14626787
0.3	0.001	0.07130152	0.11471034	0.13154483	0.07131557	0.11685607	0.13263373
0.7	0.001	0.07131462	0.11587394	0.13055320	0.07134501	0.12453009	0.16144792
1.0	0.001	0.07133512	0.11809294	0.13765704	0.07137821	0.12936085	0.18343464
0.3	0.003	0.07130038	0.11269782	0.12305319	0.07131643	0.11469445	0.12656438
0.7	0.003	0.07131029	0.11355763	0.12451718	0.07134003	0.11863616	0.13508755

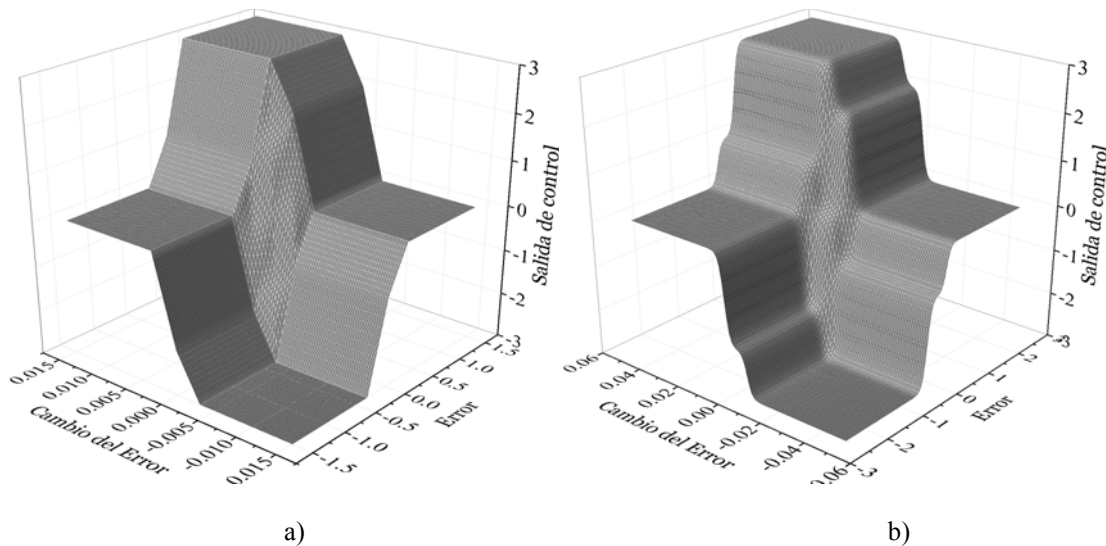


Figura 4.4 Superficies de control de para un control tipo-1 con: a) funciones de membresía triangulares $A = 0.3$, $B = 0.003$, $C = 3$; b) funciones de membresía Gaussianas $A = 0.5$, $B = 0.01$, $C = 3$.

La diferencia entre las superficies de control para cada caso particular (funciones de membresía triangulares y gaussianas), cuando los parámetros varían dentro del rango analizado $A | A = [0.3 \ 1]$ y $B | B = [0.001 \ 0.01]$ son casi idénticas. Por lo que el comportamiento del controlador también lo será, de ahí que la diferencia entre los índices de error también sea mínima.

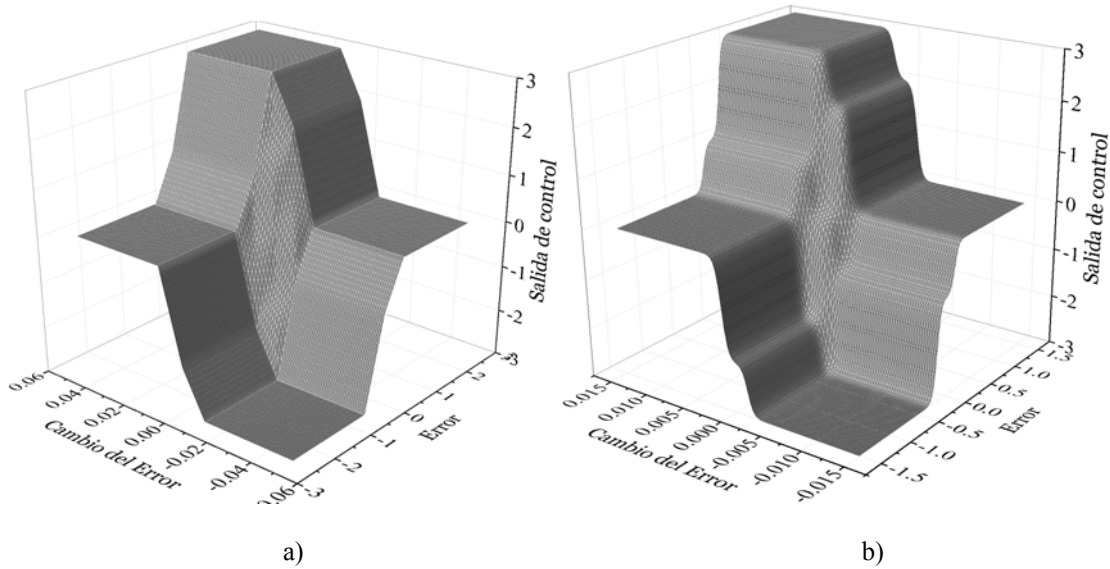


Figura 4.5 Superficies de control de para un control tipo-1 con: a) funciones de membresía triangulares $A = 0.5, B = 0.01, C = 3$; b) funciones de membresía Gaussianas $A = 0.3, B = 0.003, C = 3$.

Tabla 4.3 Índices del error ante diversos valores para C .						
C	Función de Membresía Triangular $A = 0.3, B = 0.003$			Función de Membresía Gaussiana $A = 0.5, B = 0.01$		
	ISE	IAE	ITAE	ISE	IAE	ITAE
1	0.07137505	0.11309754	0.12438815	0.07207332	0.15210362	0.23232824
3	0.07130038	0.11269782	0.12305319	0.07129465	0.11263078	0.12475198
5	0.07130068	0.11381090	0.12754102	0.07132470	0.11617684	0.12980468

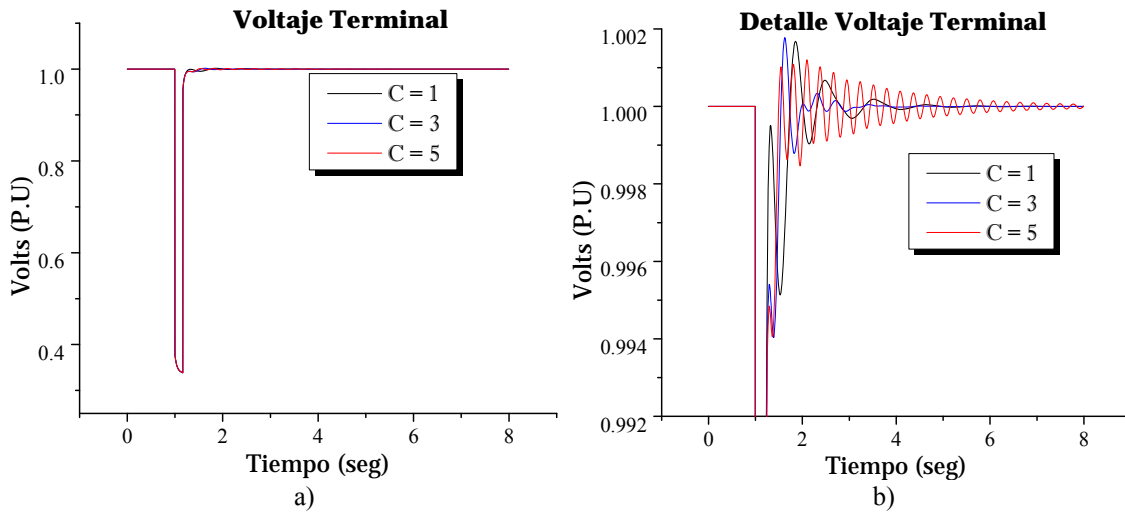
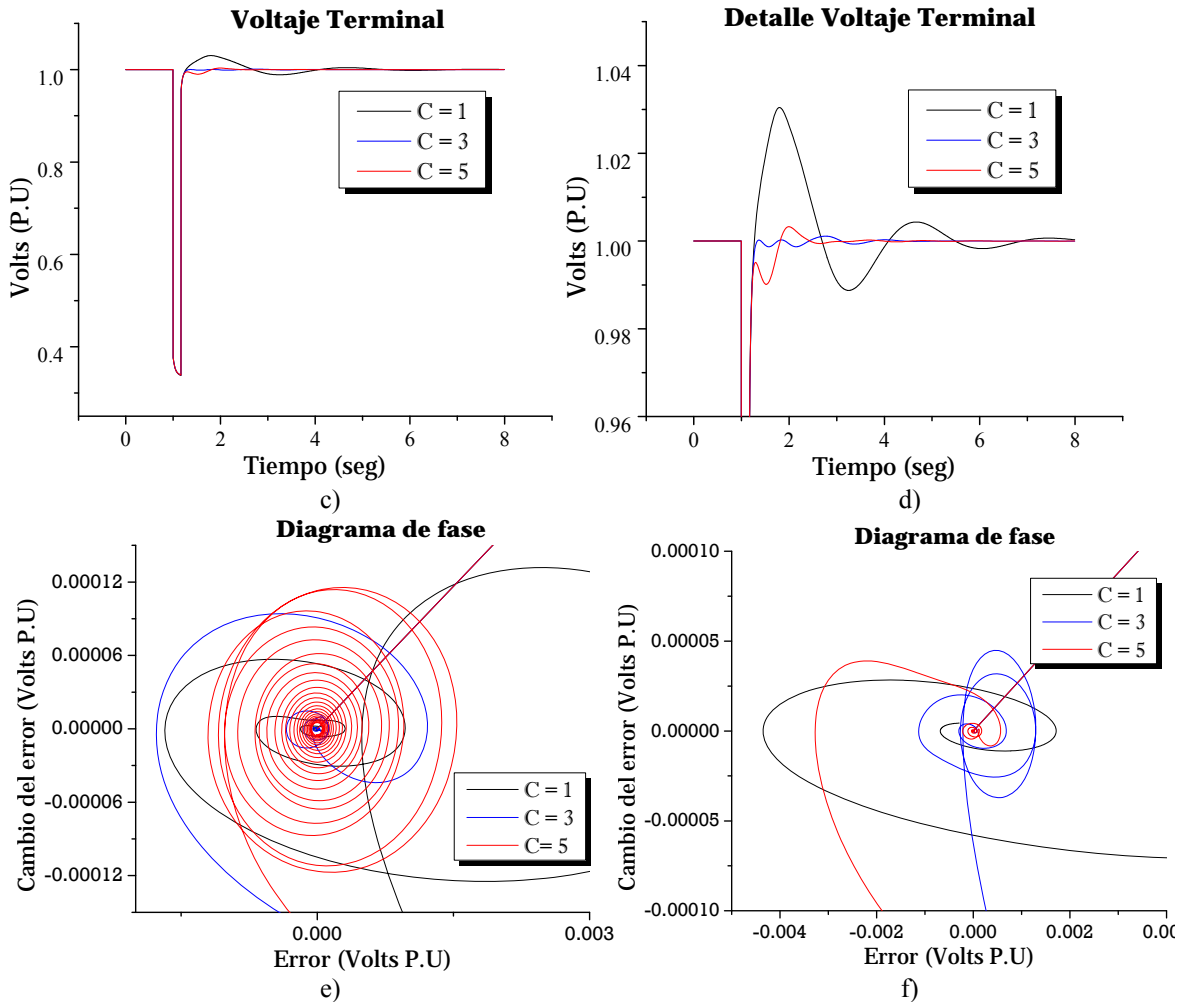


Figura 4.6 Respuesta del voltaje terminal ante una falla de corto circuito de 10 ciclos de duración: a) y b) Universos de entrada con funciones de membresía triangulares; c) y d) universos de entrada con funciones de membresía Gaussianas; e) diagrama de fases, funciones de membresía triangulares; f) diagrama de fases, funciones de membresía Gaussianas.



Continuación de Figura 4.6 Respuesta del voltaje terminal ante una falla de corto circuito de 10 ciclos de duración: a) y b) Universos de entrada con funciones de membresía triangulares; c) y d) universos de entrada con funciones de membresía Gaussianas; e) diagrama de fases, funciones de membresía triangulares; f) diagrama de fases, funciones de membresía Gaussianas.

Modificando el número de barras de salida para el controlador en 5 y 7 barras, el resultado es casi idéntico; para el caso funciones de membresía Gaussianas, la respuesta del voltaje terminal con el controlador con 7 barras de salida presenta menores oscilaciones.

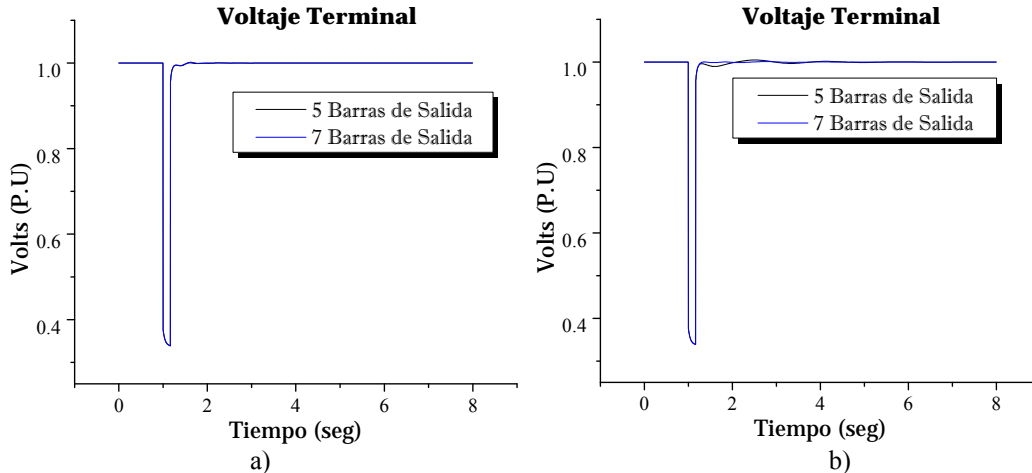


Figura 4.7 Respuesta del voltaje terminal al modificar el número de barras de salida del controlador tipo-1 con: a) universos de entrada con funciones de membresía triangulares; b) universos de entrada con funciones de membresía Gaussianas.

Tabla 4.4 Índices del error ante diferentes números de barras de salida.						
No. Barras de salida	Función de Membresía Triangular $A = 0.3, B = 0.003, C = 3$			Función de Membresía Gaussiana $A = 0.3, B = 0.003, C = 5$		
	ISE	IAE	ITAE	ISE	IAE	ITAE
5	0.07130182	0.11272544	0.12302361	0.07134768	0.12212127	0.14987064
7	0.07130038	0.11269782	0.12305319	0.07129465	0.11263078	0.12475198

4.2.1 Análisis de resultados de la sintonización del T1 FLC caso línea corta

Los índices de error así como la respuesta dinámica del voltaje terminal permiten inferir los valores adecuados para los universos de entrada y salida. EL objetivo fué localizar a que valores para A, B y C se tienen los menores índices de error y una respuesta aceptable del voltaje terminal. Como puede observarse el comportamiento del diagrama de fase influye directamente en la elección de A y B . De esta forma los valores elegidos mostraron una forma espiral con pocas oscilaciones. Esto asegura que la gran parte de las reglas difusas fueron recorridas. Por su parte, la superficie de control mostro ser diferente entre funciones de membresía triangulares y Gaussianas, a pesar de tener pocas variaciones entre funciones de membresía del mismo tipo. La superficie de control resultante resultó simétrica con pendiente pronunciada debido a los consecuentes propuestos para la matriz de reglas difusas. Se observó que con las funciones de mebresía tipo Gaussianas se tiene una forma mas suave en las pendientes a diferencia de las funciones de membresía tipo triangular. Esto debido a la forma misma de las funciones de membresía.

El número de barras de salida en base a la **Tabla 4.4**, es de 7 barras, a pesar de la mínima diferencia en los índices.

En suma los parámetros elegidos fueron:

Tabla 4.5 Parámetros para el controlador Tipo-1.		
Parámetros	Función de Membresía Triangular	Función de Membresía Gaussiana
A	0.3	0.5
B	0.003	0.01
C	3	3

4.2.2 Ajuste de la Incertidumbre en las funciones de membresía tipo-2 caso línea corta

Enseguida se muestran los resultados al añadir incertidumbre en los conjuntos difusos de entrada.

Triangular 1 Tipo-2

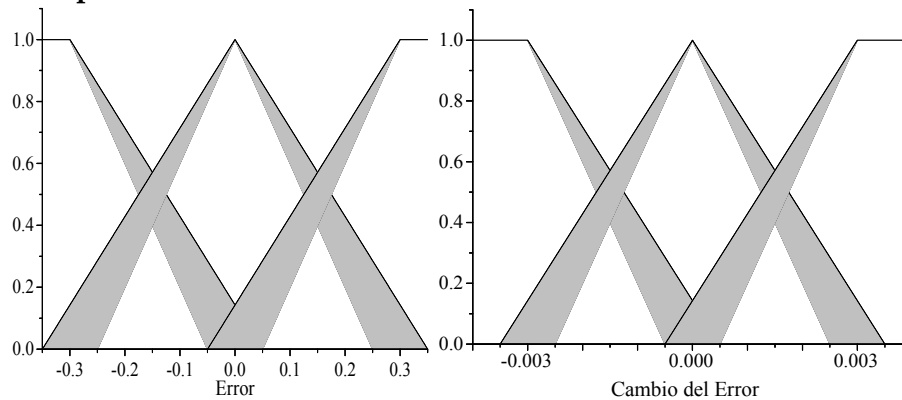


Figura 4.8 Funciones de membresía triangulares 1 tipo-2, para una línea de transmisión corta.

Tabla 4.6 Índices al añadir incertidumbre en los conjuntos triangulares 1 tipo-2, caso línea corta.			
Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.07130038	0.11269782	0.12305319
$\delta_e = 0.1, \delta_{\Delta e} = 0.001$	0.07131129	0.11428080	0.12588540
$\delta_e = 0.15, \delta_{\Delta e} = 0.001$	0.07131884	0.11849532	0.13841971
$\delta_e = 0.15, \delta_{\Delta e} = 0.0015$	0.07133521	0.12312306	0.16625012
$\delta_e = 0.1, \delta_{\Delta e} = 0.0005$	0.07131698	0.11738340	0.13423924
$\delta_e = 0.1, \delta_{\Delta e} = 0.0003$	0.07131575	0.11740103	0.13462308
$\delta_e = 0.05, \delta_{\Delta e} = 0.0005$	0.07130445	0.11325141	0.12395901

Tabla 4.7 Índices al añadir incertidumbre en las barras de salida, caso triangular 1 y línea corta.			
Incertidumbre de entrada $\delta_e = 0.05, \delta_{\Delta e} = 0.0005$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.07130445	0.11325141	0.12395901
$c = 3, IL = 0.1, IR = 0.1$	0.07130414	0.11321353	0.12388424
$c = 3, SL = 0.1, SR = 0.15$	0.07101878	0.12131152	0.16364774
$c = 3, SL = 0.1, SR = 0.2$	0.07076819	0.13007458	0.20407069
$c = 3, SL = 0.1, SR = 0.3$	0.07036149	0.14708841	0.28181936
$c = 3, SL = 0.25, SR = 0.3$	0.07103349	0.12071756	0.16122063
$c = 3, SL = 0.2, SR = 0.15$	0.07161222	0.12332030	0.16611634
$c = 3, SL = 0.3, SR = 0.1$	0.07266055	0.15306578	0.28986243

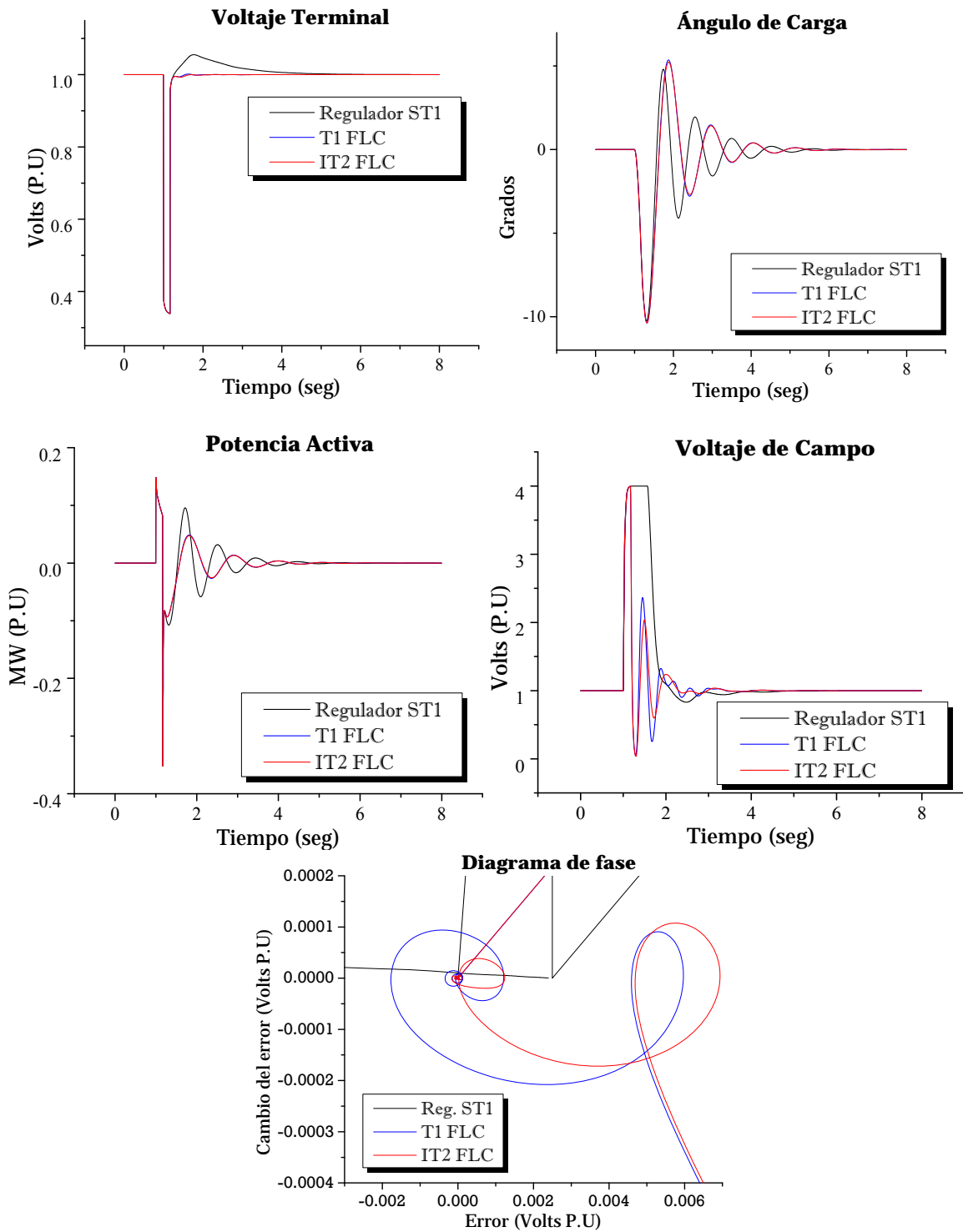
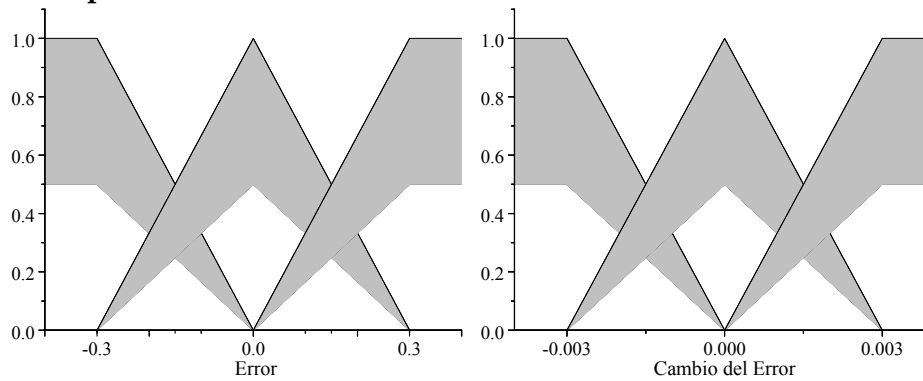


Figura 4.9 Respuesta del sistema utilizando las funciones de membresía triangulares 1 tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.

Triangular 2 Tipo-2**Figura 4.10** Funciones de membresía triangulares 2 tipo-2, para una línea de transmisión corta.**Tabla 4.8** Índices al añadir incertidumbre en los conjuntos triangulares 2 tipo-2, caso línea corta.

Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.07130038	0.11269782	0.12305319
$\delta_e = 0.05, \delta_{\Delta e} = 0.05$	0.07133985	0.11278736	0.12316377
$\delta_e = 0.1, \delta_{\Delta e} = 0.1$	0.07130035	0.11269304	0.12304687
$\delta_e = 0.2, \delta_{\Delta e} = 0.2$	0.07130025	0.11267668	0.12302347
$\delta_e = 0.3, \delta_{\Delta e} = 0.3$	0.07133954	0.11273451	0.12308450
$\delta_e = 0.5, \delta_{\Delta e} = 0.5$	0.07129945	0.11256761	0.12289265
$\delta_e = 0.7, \delta_{\Delta e} = 0.7$	0.07129911	0.11366613	0.12761962
$\delta_e = 0.7, \delta_{\Delta e} = 0.1$	0.07129911	0.11363620	0.12762877
$\delta_e = 0.1, \delta_{\Delta e} = 0.7$	0.07130503	0.11797971	0.15074218
$\delta_e = 0.1, \delta_{\Delta e} = 0.85$	0.07130333	0.11727186	0.14756586

Tabla 4.9 Índices al añadir incertidumbre en las barras de salida, caso triangular 2 y línea corta.

Incertidumbre de entrada $\delta_e = 0.5, \delta_{\Delta e} = 0.5$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.07129945	0.11256761	0.12289265
$c = 3, SL = 0.1, SR = 0.1$	0.07129936	0.11255379	0.12288844
$c = 3, SL = 0.1, SR = 0.15$	0.07129926	0.11256502	0.12292445
$c = 3, SL = 0.1, SR = 0.2$	0.07129916	0.11257730	0.12296126
$c = 3, SL = 0.1, SR = 0.3$	0.07129900	0.11260787	0.12304448
$c = 3, SL = 0.25, SR = 0.3$	0.07129917	0.11258066	0.12300612
$c = 3, SL = 0.2, SR = 0.15$	0.07130006	0.11254619	0.12288405
$c = 3, SL = 0.3, SR = 0.1$	0.07130240	0.11255475	0.12286877

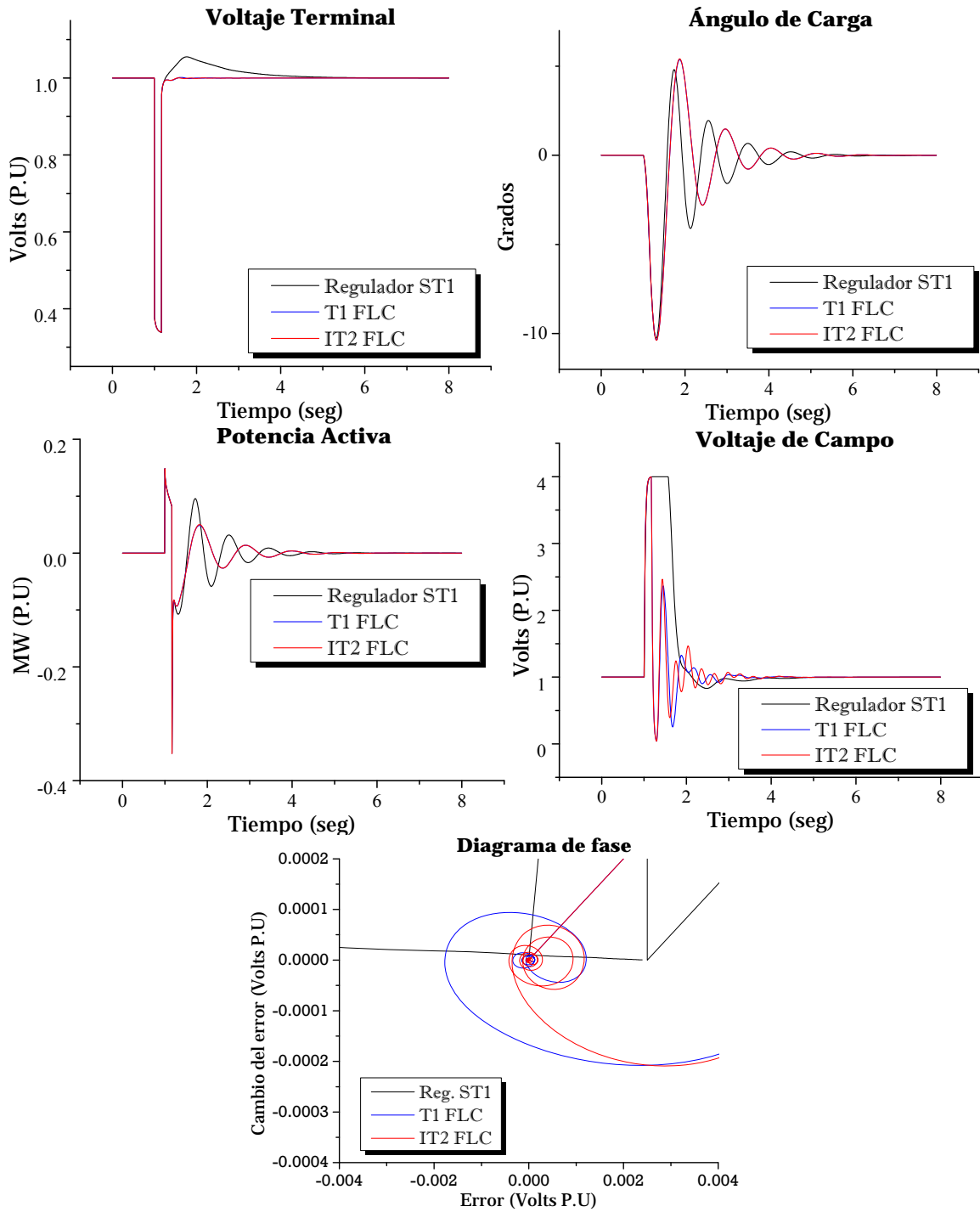


Figura 4.11 Respuesta del sistema utilizando las funciones de membresía triangulares 2 tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.

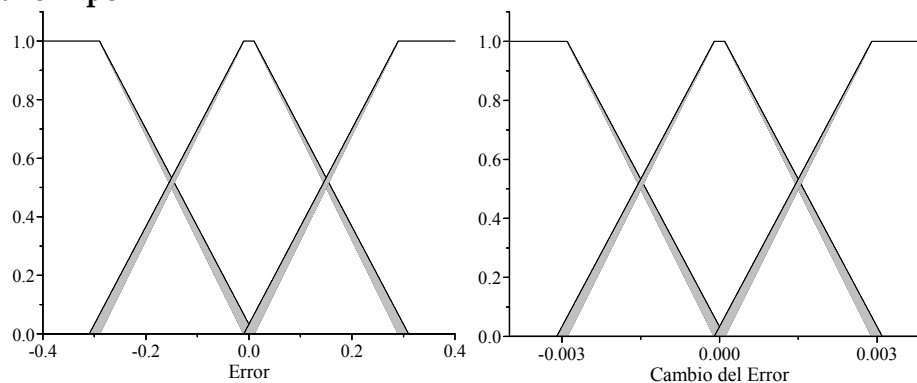
Triangular 3 Tipo-2**Figura 4.12** Funciones de membresía triangulares 3 tipo-2, para una línea de transmisión corta.

Tabla 4.10 Índices al añadir incertidumbre en los conjuntos triangulares 3 tipo-2. Caso línea corta			
Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.07130038	0.11269782	0.12305319
$\delta_e = 0.1, \delta_{\Delta e} = 0.001$	0.07131024	0.11426697	0.12590511
$\delta_e = 0.1, \delta_{\Delta e} = 0.0005$	0.07131466	0.11663935	0.13197064
$\delta_e = 0.1, \delta_{\Delta e} = 0.0003$	0.07131284	0.11638376	0.13144385
$\delta_e = 0.12, \delta_{\Delta e} = 0.001$	0.07131910	0.11729593	0.13344008
$\delta_e = 0.01, \delta_{\Delta e} = 0.0001$	0.07130033	0.11263305	0.12291649
$\delta_e = 0.01, \delta_{\Delta e} = 0.00005$	0.07130126	0.11419664	0.13008845

Tabla 4.11 Índices al añadir incertidumbre en las barras de salida, caso triangular 3 y línea corta.			
Incertidumbre de entrada $\delta_e = 0.01, \delta_{\Delta e} = 0.0001$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.07130033	0.11263305	0.12291649
$c = 3, SL = 0.1, SR = 0.1$	0.07148781	0.11895941	0.14944503
$c = 3, SL = 0.1, SR = 0.15$	0.07125303	0.11375904	0.12892195
$c = 3, SL = 0.1, SR = 0.2$	0.07120825	0.11500034	0.13511596
$c = 3, SL = 0.1, SR = 0.3$	0.07112559	0.11751554	0.14726089
$c = 3, SL = 0.25, SR = 0.3$	0.07125717	0.11370905	0.12861967
$c = 3, SL = 0.2, SR = 0.15$	0.07134715	0.11417177	0.12949890
$c = 3, SL = 0.3, SR = 0.1$	0.07148781	0.11895941	0.14944503

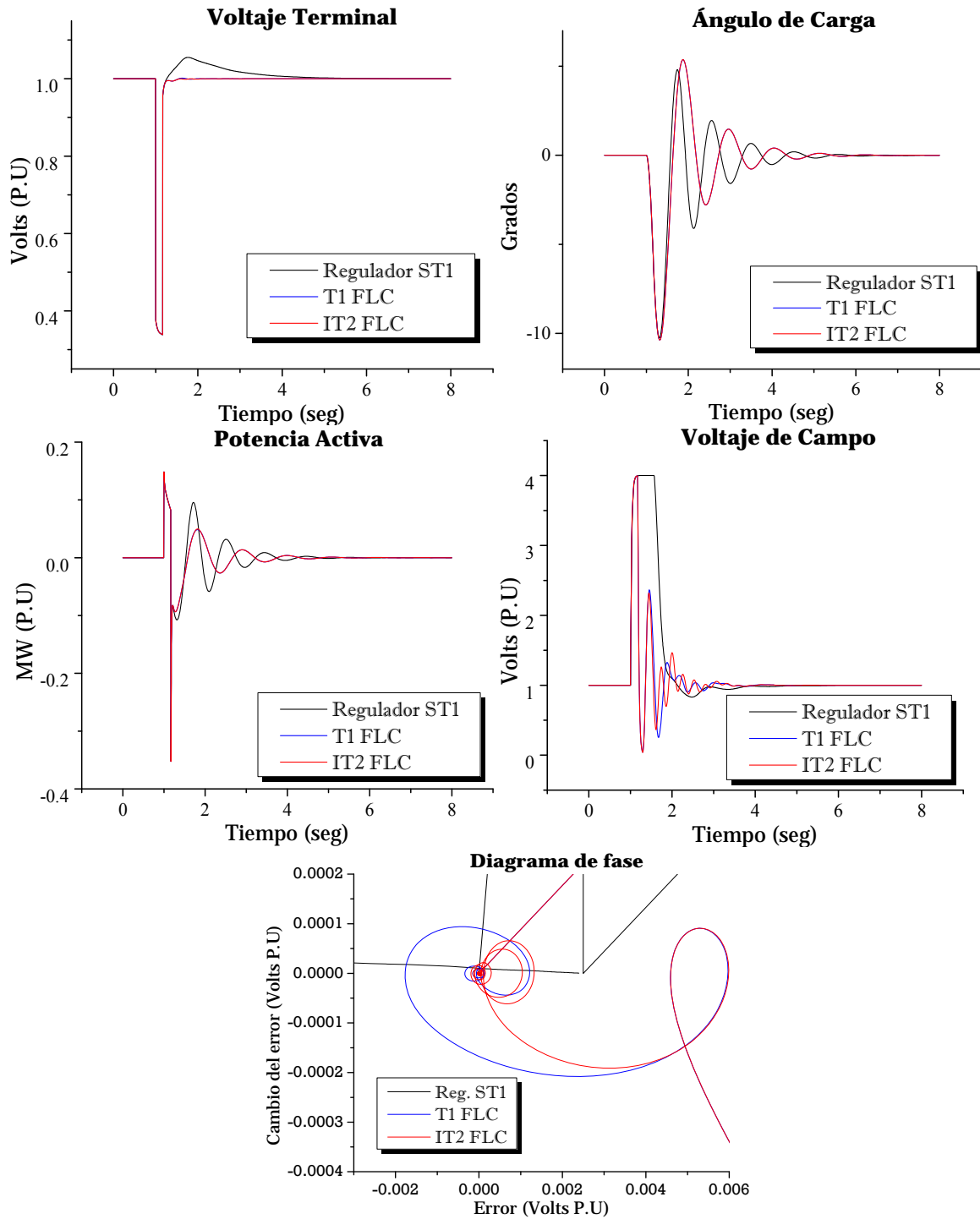
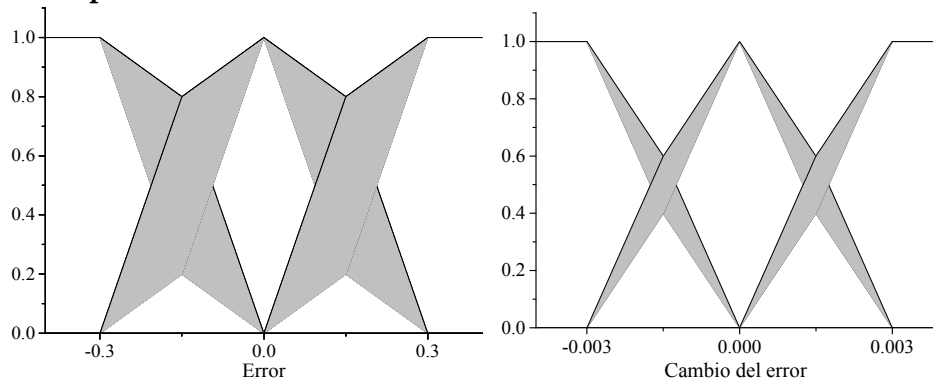


Figura 4.13 Respuesta del sistema utilizando las funciones de membresía triangulares 3 tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.

Triangular 4 Tipo-2**Figura 4.14** Funciones de membresía triangulares 4 tipo-2, para una línea de transmisión corta.**Tabla 4.12** Índices al añadir incertidumbre en los conjuntos triangulares 4 tipo-2. Caso línea corta.

Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.07130038	0.11269782	0.12305319
$\delta_e = 0.1, \delta_{\Delta e} = 0.1$	0.07130038	0.11269776	0.12305285
$\delta_e = 0.1, \delta_{\Delta e} = 0.2$	0.07130040	0.11269930	0.12305419
$\delta_e = 0.1, \delta_{\Delta e} = 0.3$	0.07130052	0.11271126	0.12306841
$\delta_e = 0.1, \delta_{\Delta e} = 0.4$	0.07130075	0.11273934	0.12310519
$\delta_e = 0.2, \delta_{\Delta e} = 0.1$	0.07130040	0.11269848	0.12305252
$\delta_e = 0.3, \delta_{\Delta e} = 0.1$	0.07130045	0.11269983	0.12305077
$\delta_e = 0.4, \delta_{\Delta e} = 0.1$	0.07130054	0.11270474	0.12305349
$\delta_e = 0.3, \delta_{\Delta e} = 0.3$	0.07130041	0.11270267	0.12305974

Tabla 4.13 índices al añadir incertidumbre en las barras de salida, caso triangular 4 y línea corta.

Incertidumbre de entrada $\delta_e = 0.3, \delta_{\Delta e} = 0.1$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.07130045	0.11269983	0.12305077
$c = 3, IL = 0.1, IR = 0.1$	0.07130038	0.11269159	0.12304390
$c = 3, SL = 0.1, SR = 0.15$	0.07130028	0.11269347	0.12305106
$c = 3, SL = 0.1, SR = 0.2$	0.07130019	0.11269880	0.12306504
$c = 3, SL = 0.1, SR = 0.3$	0.07130005	0.11271527	0.12310366
$c = 3, SL = 0.25, SR = 0.3$	0.07130018	0.11268528	0.12304735
$c = 3, SL = 0.2, SR = 0.15$	0.07130108	0.11268848	0.12304044
$c = 3, SL = 0.3, SR = 0.1$	0.07130339	0.11271384	0.12307758

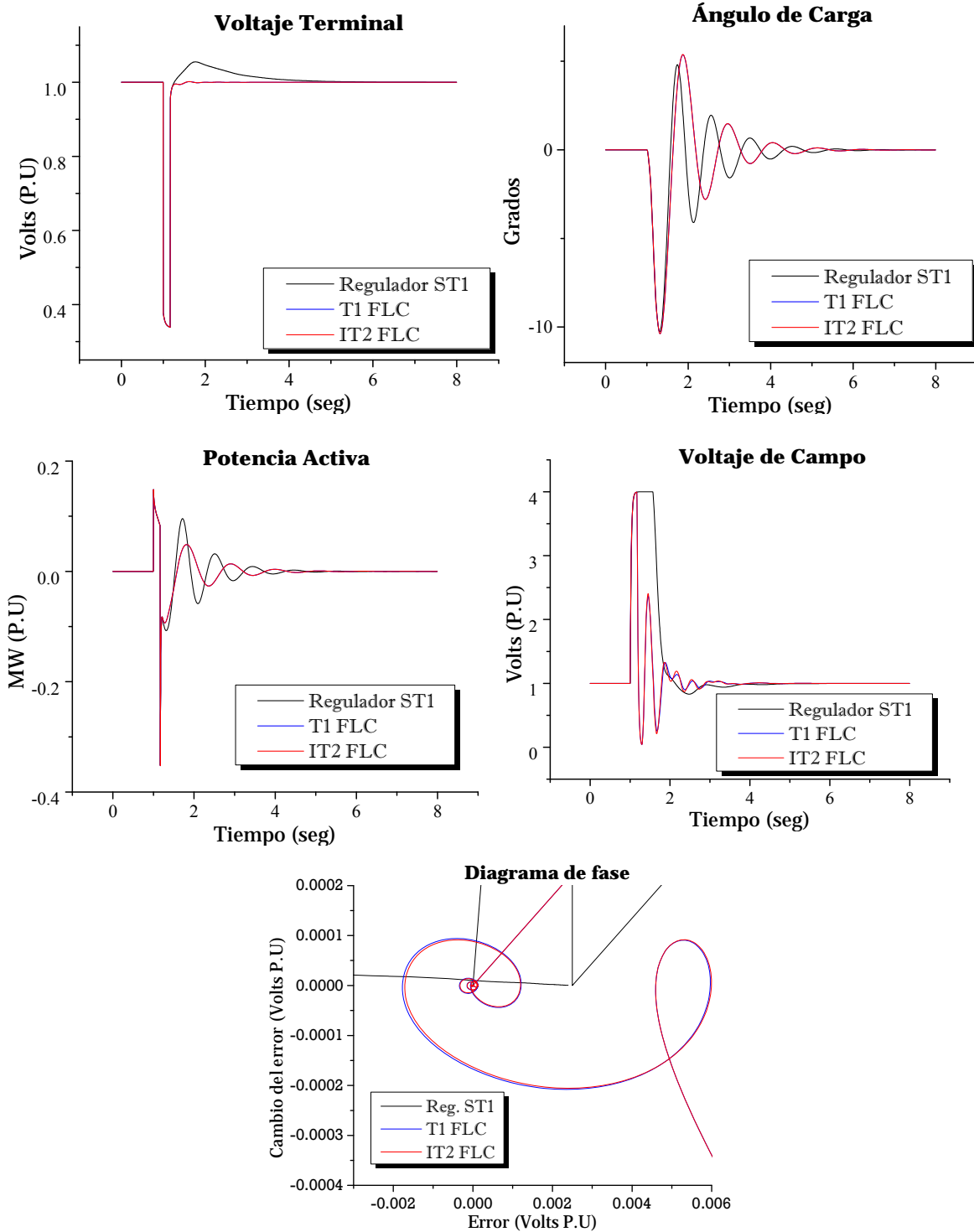


Figura 4.15 Respuesta del sistema utilizando las funciones de membresía triangulares 4 tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.

Gaussiana Tipo-2 con incertidumbre

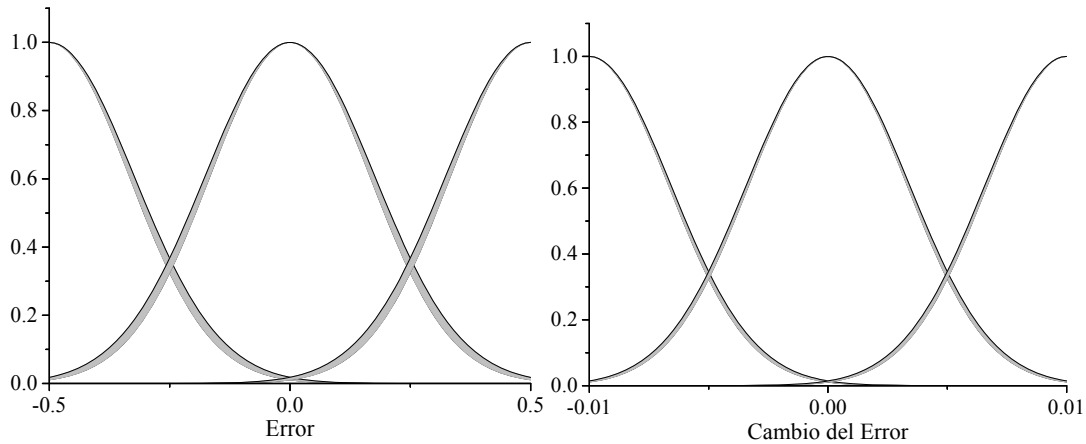


Figura 4.16 Funciones de membresía Gaussianas tipo-2, para una línea de transmisión corta.

Tabla 4.14 Índices al añadir incertidumbre en los conjuntos Gaussianos tipo-2. Caso línea corta.			
Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.07129465	0.11263078	0.12475198
$\delta_e = 0.01, \delta_{\Delta e} = 0.0001$	0.07129509	0.11275524	0.12465838
$\delta_e = 0.01, \delta_{\Delta e} = 0.0003$	0.07129504	0.11325711	0.12770653
$\delta_e = 0.01, \delta_{\Delta e} = 0.0005$	0.07129566	0.11363522	0.12909431
$\delta_e = 0.03, \delta_{\Delta e} = 0.0001$	0.07130016	0.11426254	0.12757758
$\delta_e = 0.05, \delta_{\Delta e} = 0.0001$	0.07130360	0.11492758	0.12924798
$\delta_e = 0.1, \delta_{\Delta e} = 0.0001$	0.07130309	0.11485569	0.13032930

Tabla 4.15 Índices al añadir incertidumbre en las barras de salida. Caso Gaussianas y línea corta.			
Incertidumbre de entrada $\delta_e = 0.01, \delta_{\Delta e} = 0.0001$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.07129509	0.11275524	0.12465838
$c = 3, IL = 0.1, IR = 0.1$	0.07129537	0.11288353	0.12487394
$c = 3, SL = 0.1, SR = 0.15$	0.07093095	0.12032182	0.16319423
$c = 3, SL = 0.1, SR = 0.2$	0.07059917	0.13010678	0.20659301
$c = 3, SL = 0.1, SR = 0.3$	0.07003232	0.15011730	0.29399699
$c = 3, SL = 0.25, SR = 0.3$	0.07093693	0.12028300	0.16301513
$c = 3, SL = 0.2, SR = 0.15$	0.07168997	0.12264712	0.16622218
$c = 3, SL = 0.3, SR = 0.1$	0.07305443	0.15524305	0.30033073

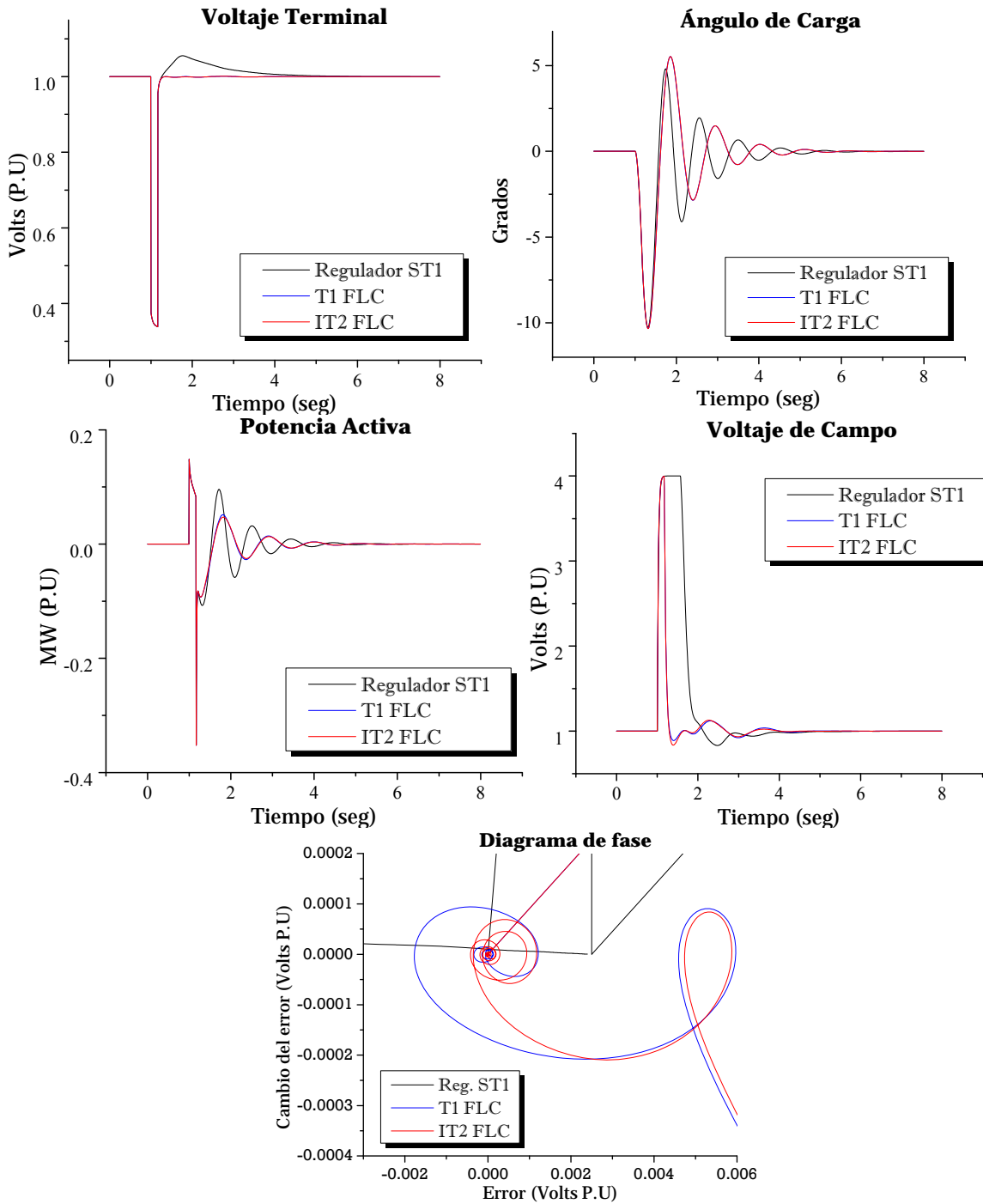


Figura 4.17 Respuesta del sistema utilizando las funciones de membresía Gaussiana tipo-2 en una línea corta ante un disturbio de 10 ciclos y 9 reglas difusas.

4.2.3 Análisis de resultados de la sintonización del IT2 FLC caso línea corta

En suma, los parámetros adecuados para el controlador tipo-2 según el tipo de función de membresía utilizada resulta:

Tabla 4.16 Resumen mejores parámetros, caso línea corta.				
Tipos de funciones de membresía tipo 2	Incertidumbre en los universos de entrada		Incertidumbre en el universo de salida	
	δ_e	$\delta_{\Delta e}$	SL	SR
Triangular 1 tipo-2 $A = 0.3; B = 0.003$ y $C = 3$	0.05	0.0005	0.1	0.1
Triangular 2 tipo-2 $A = 0.3; B = 0.003$ y $C = 3$	0.5	0.5	0.3	0.1
Triangular 3 tipo-2 $A = 0.3; B = 0.003$ y $C = 3$	0.01	0.0001	0.0	0.0
Triangular 4 tipo-2 $A = 0.3; B = 0.003$ y $C = 3$	0.3	0.1	0.25	0.3
Gaussianas tipo-2 $A = 0.5; B = 0.01$ y $C = 3$	0.01	0.0001	0.0	0.0

Tabla 4.17 Resumen de mejores índices según Tabla 4.16, caso línea corta.			
Tipo de función de Membresía	Índices del Error		
	ISE	IAE	ITAE
Triangular 1 tipo-2	0.07130414	0.11321353	0.12388424
Triangular 2 tipo-2	0.07130240	0.11255475	0.12286877
Triangular 3 tipo-2	0.07130033	0.11263305	0.12291649
Triangular 4 tipo-2	0.07130018	0.11268258	0.12304735
Gaussiana tipo-2	0.07129509	0.11275524	0.12465838

De la **Tabla 4.17**, se infiere que los mejores parámetros para el controlador tipo-2 con conjuntos en intervalos con 9 reglas difusas y para una línea corta resulta:

Tabla 4.18 Parámetros finales para el IT2 FLC, caso línea corta.				
Tipos de funciones de membresía tipo 2	Incertidumbre en los universos de entrada		Incertidumbre en el universo de salida	
	δ_e	$\delta_{\Delta e}$	SL	SR
Triangular 2 tipo-2 $A = 0.3; B = 0.003$ y $C = 3$	0.5	0.5	0.3	0.1
Gaussianas tipo-2 $A = 0.5; B = 0.01$ y $C = 3$	0.01	0.0001	0.0	0.0

Estos parámetros más los presentados en la **Tabla 4.5**, complementan el conjunto de parámetros requeridos en este trabajo para el IT2 FLC. Estos parámetros serán utilizados en las simulaciones del generador síncrono con una línea de transmisión corta.

La función de membresía triangular 2 tipo-2 mostro ser la mejor entre su tipo, considerando el resultado en los índices de desempeño. Aunque debido a la estrecha diferencia en los índices cualquiera pudo ser elegida. Como por ejemplo, la función de membresía triangular 1 tipo-2 mostró una respuesta mas suave en el voltaje de campo. Los resultados finaes en la **Tabla 4.18** muestran que la incertidumbre depende de la forma de la función de membresía, para este caso las funciones triangulares requirieron una incertidumbre alta del orden de 0.5 mientras que para las Gaussianas fue una mínima. Además la incertidumbre en los consecuentes para el caso funciones de membresía Gaussianas tipo-2 fue nula. Aunque la comparación directa es difícil debido a que la interpretación de la incertidumbre en cada función de membresía es diferente en cada caso. La forma de la superficie de control para el IT2 FLC mostro muy poca diferencia con respecto al T1 FLC con los mismos parámetros. Esto debido a que la incetidumbre fue

simétrica en las funciones de membresía de los conjuntos difusos tipo-2, con lo que la afectación a estos conjuntos no fue abrupta.

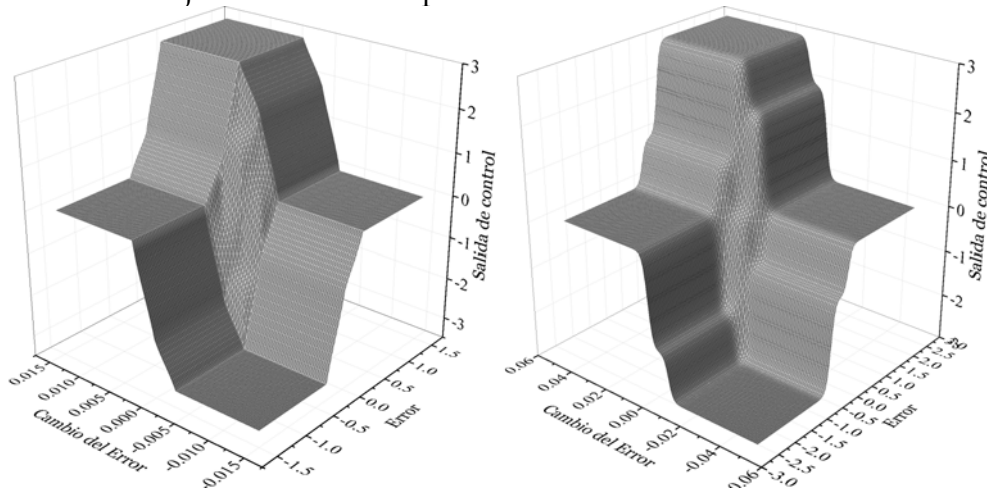


Figura 4.18 Superficie de control resultante del IT2 FLC con los parámetros según **Tabla 4.18** y 9 reglas de conocimiento difuso.

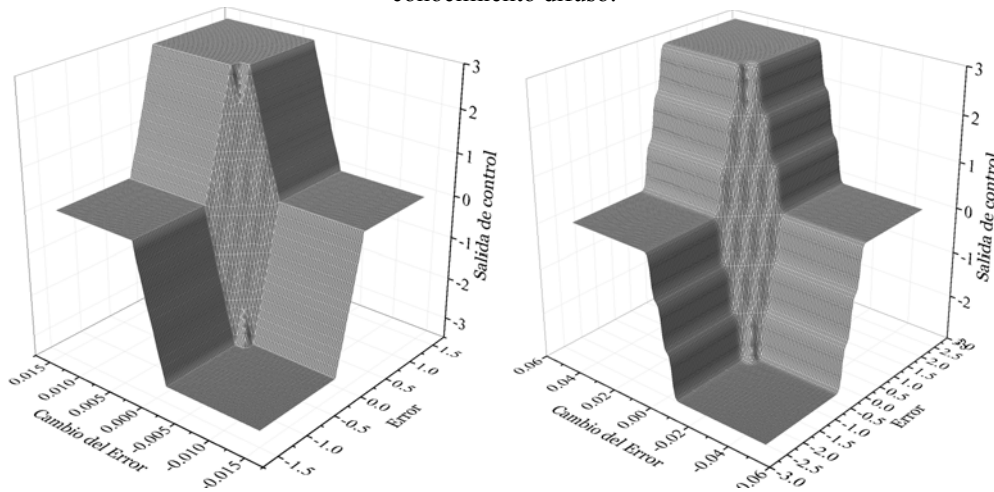


Figura 4.19 Superficie de control resultante del IT2 FLC con los parámetros según **Tabla 4.18** y 25 reglas de conocimiento difuso.

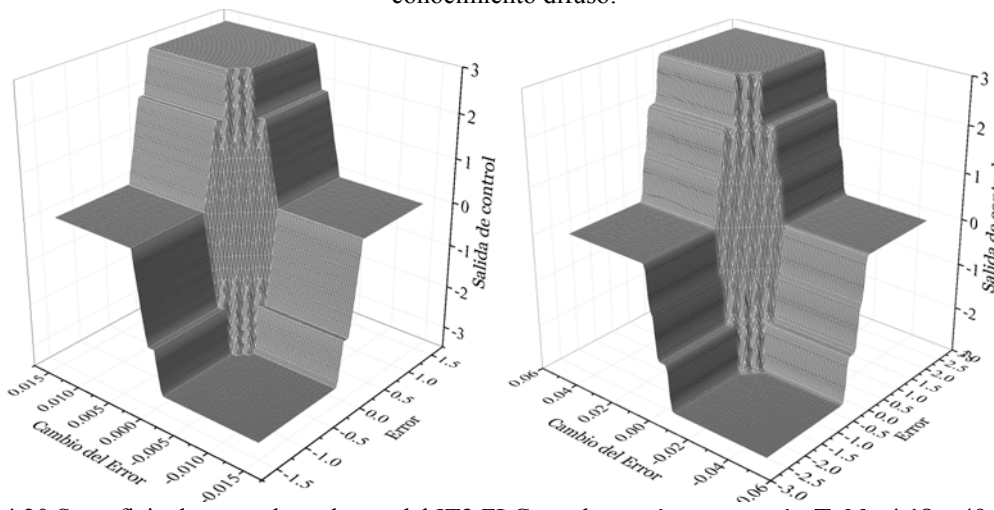


Figura 4.20 Superficie de control resultante del IT2 FLC con los parámetros según **Tabla 4.18** y 49 reglas de conocimiento difuso.

4.3 Sintonización manual para una línea de transmisión larga con el generador síncrono en vacío

De igual forma que para la configuración línea corta con el generador síncrono en vacío, es decir sin carga eléctrica en el bus, se tiene para una línea de transmisión larga lo siguiente:

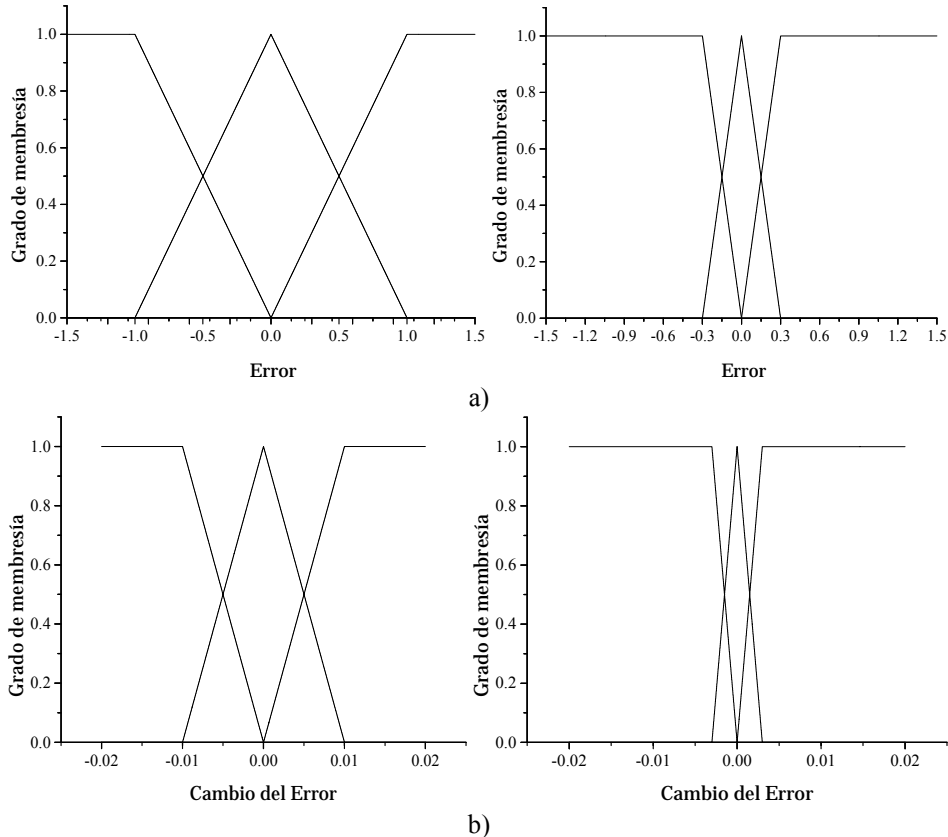


Figura 4.21 Expansión y contracción de los universos de entrada: a) Universo del error; b) Universo del cambio del error.

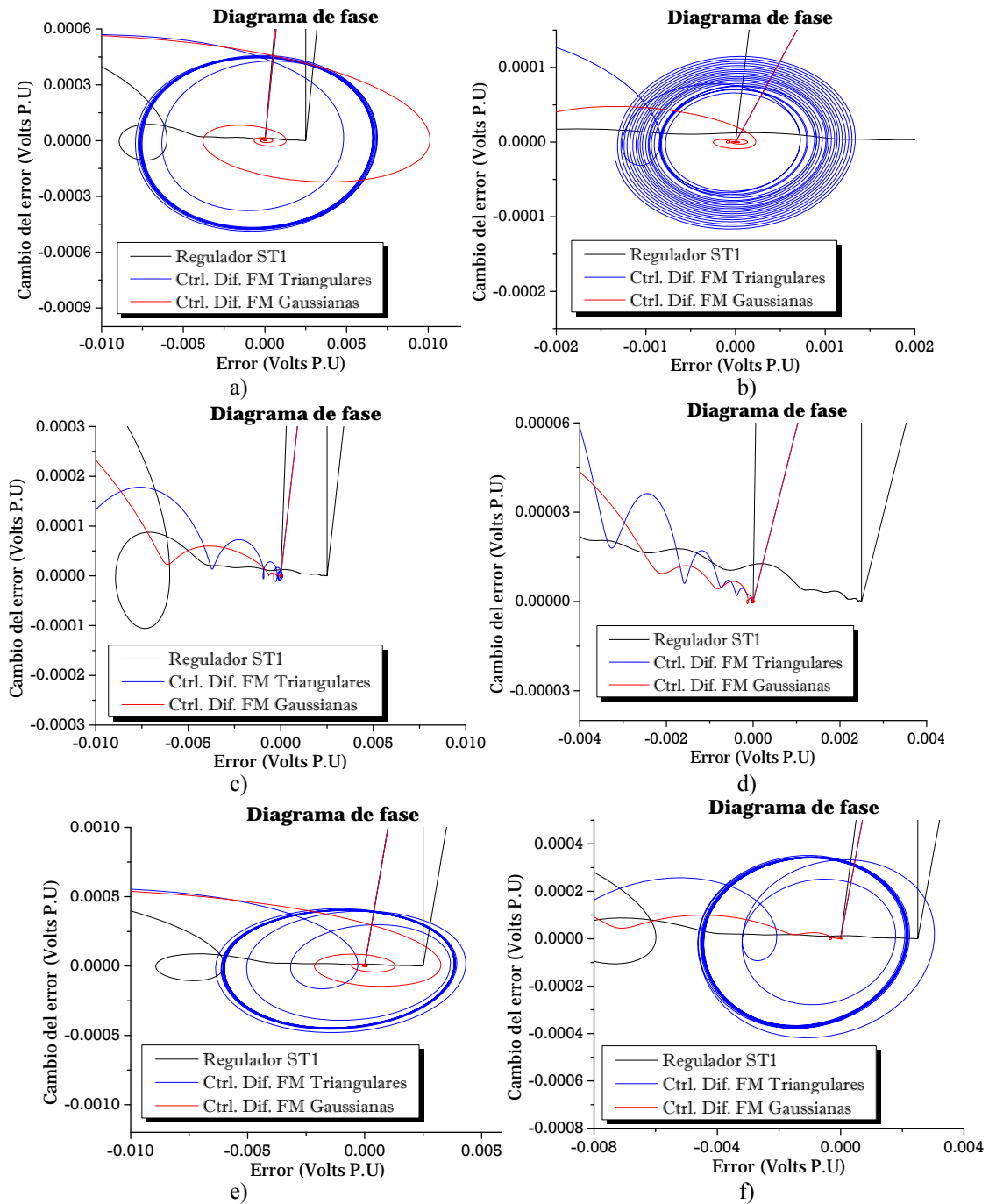


Figura 4.22 Evolución del diagrama de fases para línea corta: a) $A = 0.5, B = 0.01$; b) $A = 0.5, B = 0.005$; c) $A = 0.7, B = 0.005$; d) $A = 1.0, B = 0.005$; e) $A = 0.3, B = 0.005$; f) $A = 0.3, B = 0.003$.

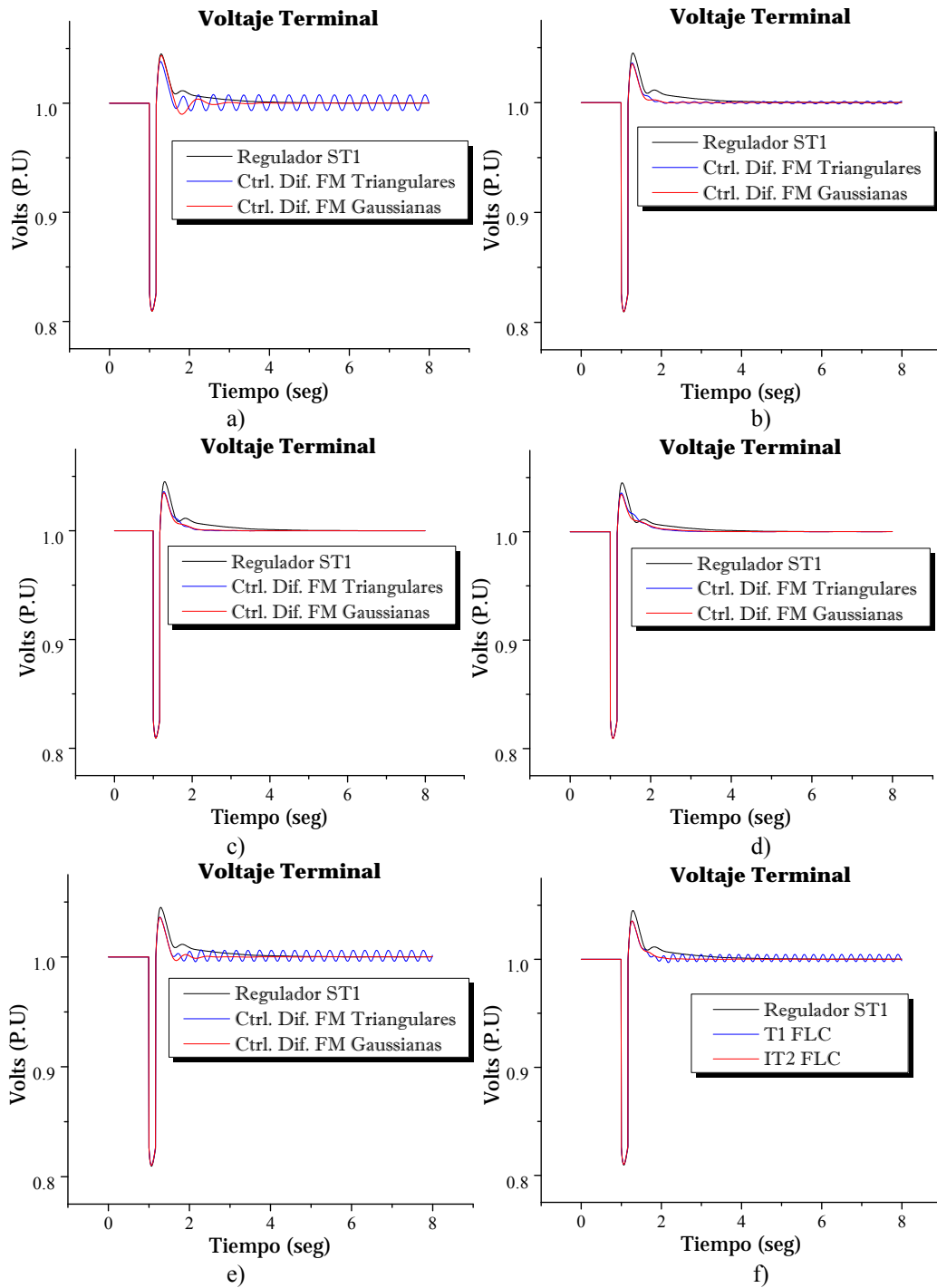


Figura 4.23 Evolución del voltaje terminal para línea corta: a) $A = 0.5, B = 0.01$; b) $A = 0.5, B = 0.005$; c) $A = 0.7, B = 0.005$; d) $A = 1.0, B = 0.005$; e) $A = 0.3, B = 0.005$; f) $A = 0.3, B = 0.003$.

La siguiente tabla muestra un resumen de los índices obtenidos en varias simulaciones:

Tabla 4.19 Resumen de índices para la sintonización del T1 FLC, caso línea larga.

Parámetro		Función de Membresía Triangular			Función de Membresía Gaussiana		
A	B	ISE	IAE	ITAE	ISE	IAE	ITAE
0.5	0.01	0.00611688	0.06888339	0.18419141	0.00612833	0.04724562	0.05920497
0.5	0.02	0.00643803	0.08766279	0.26932876	0.00668686	0.08306072	0.19838846
0.5	0.008	0.00604231	0.06243738	0.15758507	0.00601789	0.04326137	0.05169985
0.5	0.005	0.00593000	0.04499072	0.06822877	0.00592505	0.04076529	0.04755526
0.5	0.003	0.00598765	0.05509211	0.11013197	0.00597629	0.04635472	0.05840966
0.7	0.005	0.00595626	0.04308419	0.05128703	0.00595268	0.04276367	0.05136888
1.0	0.005	0.00600682	0.04641324	0.05785104	0.00599447	0.04609199	0.05856980
1.5	0.005	0.00609520	0.05215668	0.07098721	0.00606095	0.05187846	0.07387177
0.3	0.005	0.00599034	0.05949452	0.14186088	0.00590266	0.04057543	0.04727650
0.3	0.003	0.00594958	0.05451252	0.11486800	0.00591002	0.04196457	0.04951554

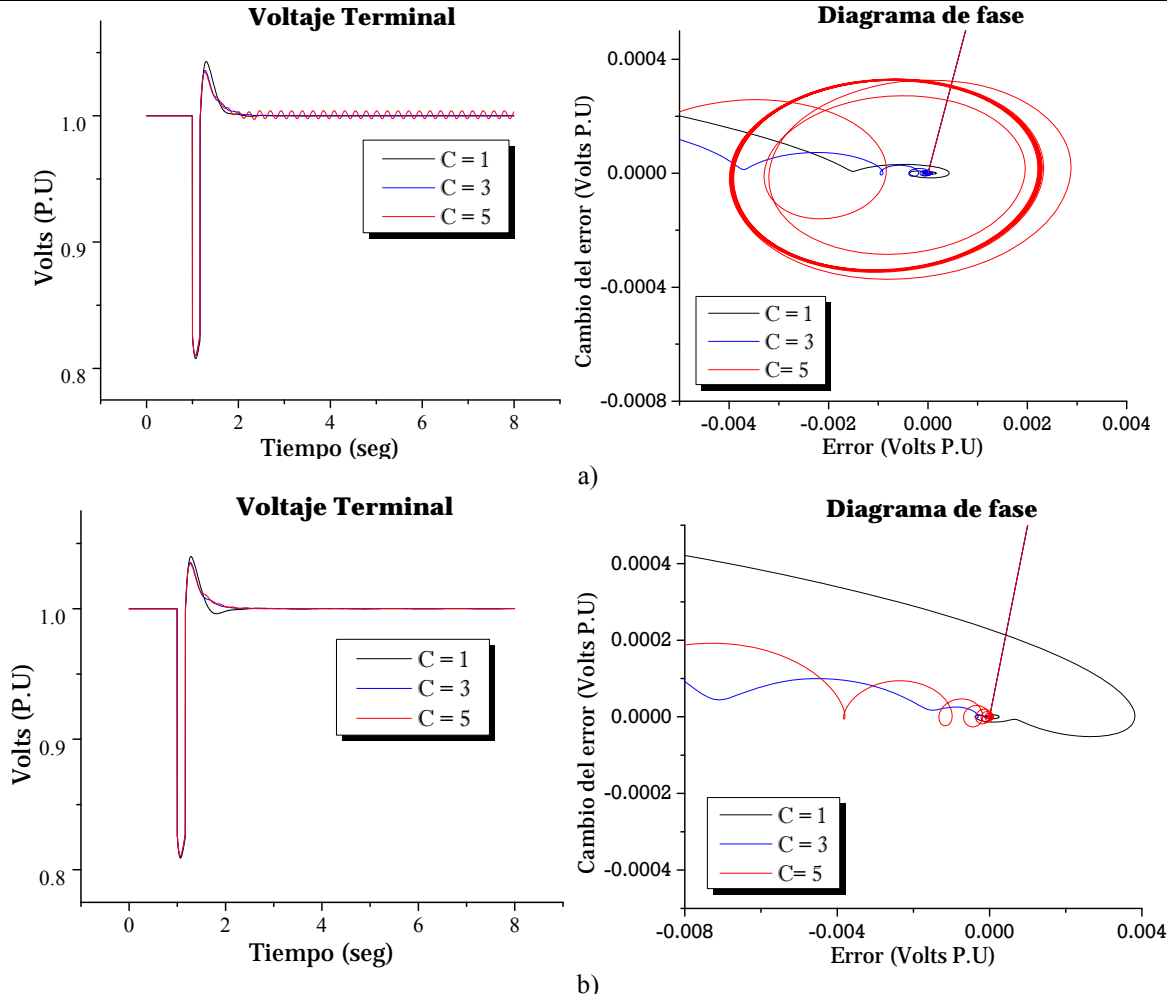


Figura 4.24 Respuesta del voltaje terminal al variar el parámetro C en el universo de salida del T1 FLC con : a) universos de entrada con funciones de membresía triangulares; b) universos de entrada con funciones de membresía Gaussianas.

Tabla 4.20 Índices del error ante distintos valores del parámetro C , caso línea larga.

C	Función de Membresía Triangular $A = 0.7, B = 0.005$			Función de Membresía Gaussiana $A = 0.3, B = 0.003$		
	ISE	IAE	ITAE	ISE	IAE	ITAE
1	0.00622575	0.04457213	0.05299910	0.00606228	0.04325339	0.05181634
3	0.00595626	0.04308419	0.05128703	0.00591002	0.04196457	0.04951554
5	0.00594586	0.05429915	0.11025556	0.00589182	0.04227825	0.05017648

Modificando el número de barras de salida para el controlador en 5 y 7 barras, los resultados son:

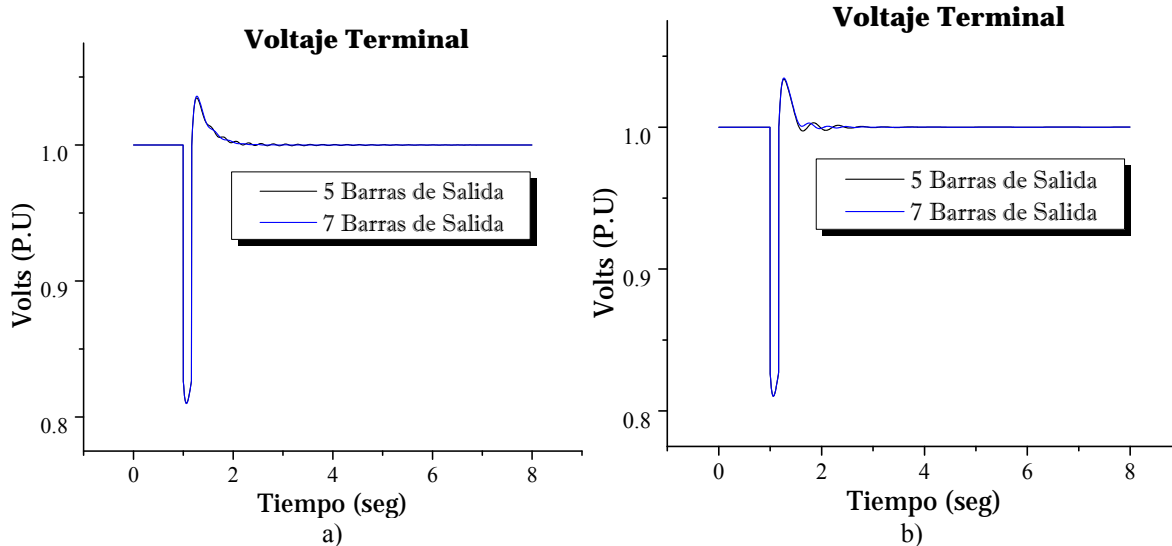


Figura 4.25 Respuesta del voltaje terminal al modificar el número de barras de salida del controlador tipo-1 con: a) universos de entrada con funciones de membresía triangulares; b) universos de entrada con funciones de membresía Gaussianas.

Tabla 4.21 Índices ante diferentes números de barras de salida, caso línea larga.						
No. Barras de salida	Función de Membresía Triangular $A = 0.7, B = 0.005, C = 3$			Función de Membresía Gaussiana $A = 0.3, B = 0.005, C = 5$		
	ISE	IAE	ITAE	ISE	IAE	ITAE
5	0.00592790	0.04385500	0.05494238	0.00585565	0.04080423	0.04750754
7	0.00595626	0.04308419	0.05128703	0.00589182	0.04227825	0.05017648

4.3.1 Análisis de la sintonización del T1FLC caso línea larga

Para el caso línea larga, la variación en los parámetros A y B mostraron marcada diferencia en la respuesta del voltaje terminal con el generador síncrono en vacío. De igual forma que en la sección 4.2 se siguió el criterio obtener una respuesta dinámica adecuada del voltaje terminal, índices del error con los mínimos valores y una respuesta en espiral y poco oscilante en el diagrama de fases. Nótese que para este caso, en la respuesta dinámica del voltaje terminal se tiene sobretiro en todos los casos y las funciones de membresía tipo Gaussianas mostraron mejor comportamiento en el voltaje terminal. Además de que para el caso de funciones de membresía Gaussianas aunque el controlador presenta menores índices en $A = 0.3$ y $B = 0.005$ el comportamiento del voltaje y diagrama de fase sugieren buena opción $A = 0.3$ y $B = 0.003$.

Se utilizan eligen 7 barras de salida para ambas funciones de membresía a pesar de que para las funciones de membresía Gaussianas se tienen menores índices con 5 barras. Esto para tener a todos los controladores trabajando con 7 barras de salida y pueda existir uniformidad en este parámetro.

Parámetro	Función de Membresía Triangular	Función de Membresía Gaussiana
<i>A</i>	0.7	0.3
<i>B</i>	0.005	0.003
<i>C</i>	3	5

4.3.2 Ajuste de la Incertidumbre en las funciones de membresía tipo-2 caso línea larga

Enseguida se muestran los resultados al añadir incertidumbre en los conjuntos difusos de entrada.

Triangular 1 Tipo-2 con incertidumbre

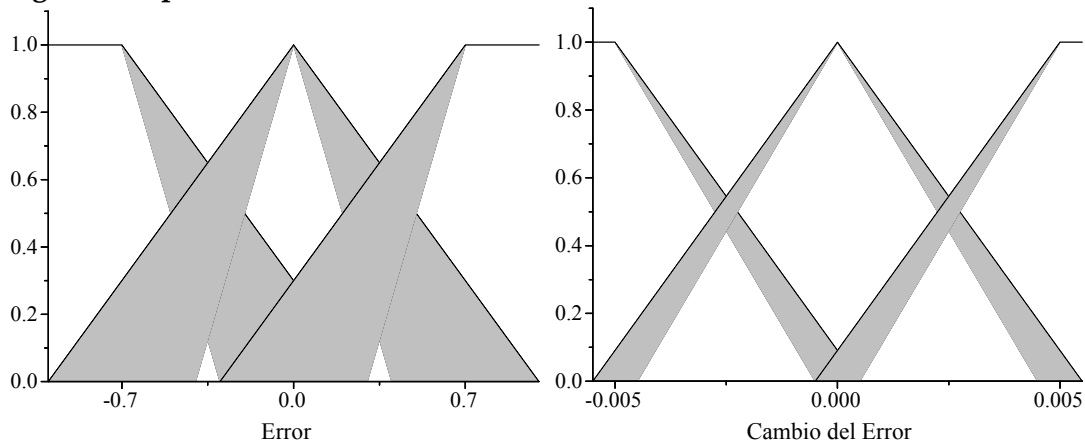


Figura 4.26 Funciones de membresía triangulares 1 tipo-2, para una línea de transmisión corta.

Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.00595626	0.04308419	0.05128703
$\delta_e = 0.1, \delta_{\Delta e} = 0.001$	0.00603494	0.06360899	0.16265937
$\delta_e = 0.2, \delta_{\Delta e} = 0.001$	0.00629964	0.06818817	0.12077903
$\delta_e = 0.05, \delta_{\Delta e} = 0.001$	0.00617435	0.07570615	0.22020772
$\delta_e = 0.1, \delta_{\Delta e} = 0.002$	0.00673343	0.10355620	0.35207102
$\delta_e = 0.3, \delta_{\Delta e} = 0.001$	0.00644139	0.07496602	0.14238064
$\delta_e = 0.3, \delta_{\Delta e} = 0.0005$	0.00594647	0.04804508	0.08074424

Incertidumbre de entrada $\delta_e = 0.3, \delta_{\Delta e} = 0.0005$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.00594647	0.04804508	0.08074424
$c = 3, IL = 0.1, IR = 0.1$	0.00651387	0.07912982	0.15778785
$c = 3, SL = 0.1, SR = 0.2$	0.00879797	0.18474983	0.65506063
$c = 3, SL = 0.1, SR = 0.3$	0.01412965	0.28772812	1.14069292
$c = 3, SL = 0.2, SR = 0.1$	0.00744184	0.11900742	0.45163534
$c = 3, SL = 0.3, SR = 0.1$	0.01138865	0.20447997	0.90273464
$c = 3, SL = 0.2, SR = 0.3$	0.00864057	0.18031762	0.63328474

Triangular 2 Tipo-2 con incertidumbre

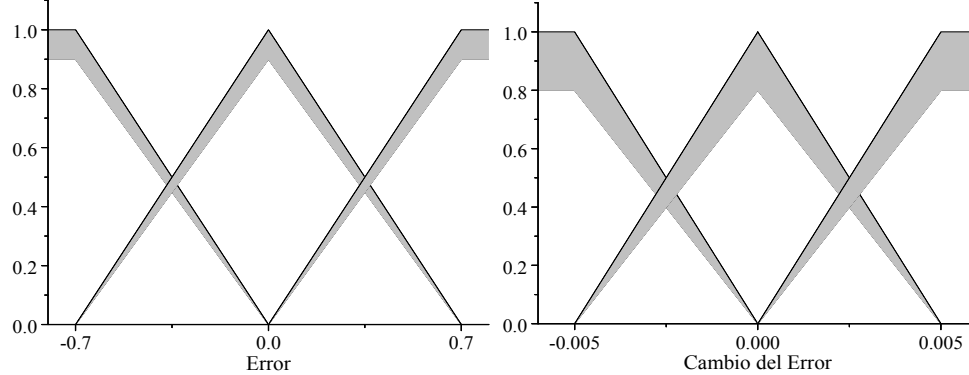


Figura 4.27 Funciones de membresía triangulares 2 tipo-2, para una línea de transmisión corta.

Tabla 4.25 Índices al añadir incertidumbre en los conjuntos triangulares 2 tipo-2, laso línea larga.

Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.00595626	0.04308419	0.05128703
$\delta_e = 0.1, \delta_{\Delta e} = 0.1$	0.00595661	0.04310937	0.05134022
$\delta_e = 0.1, \delta_{\Delta e} = 0.2$	0.00595223	0.04285395	0.05102407
$\delta_e = 0.1, \delta_{\Delta e} = 0.3$	0.00594844	0.04280271	0.05154082
$\delta_e = 0.1, \delta_{\Delta e} = 0.4$	0.00594500	0.04390309	0.05816400
$\delta_e = 0.2, \delta_{\Delta e} = 0.2$	0.00595768	0.04319470	0.05153659
$\delta_e = 0.3, \delta_{\Delta e} = 0.2$	0.00596503	0.04376347	0.05279475
$\delta_e = 0.5, \delta_{\Delta e} = 0.1$	0.00600399	0.05118204	0.08786643

Tabla 4.26 Índices al añadir incertidumbre en las barras de salida, caso triangular 3 y línea larga.

Incertidumbre de entrada $\delta_e = 0.1, \delta_{\Delta e} = 0.2$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.00595223	0.04285395	0.05102407
$c = 3, SL = 0.1, SR = 0.1$	0.00595341	0.04298555	0.05133883
$c = 3, SL = 0.1, SR = 0.2$	0.00597911	0.04461589	0.05433068
$c = 3, SL = 0.1, SR = 0.3$	0.00597911	0.04461589	0.05433068
$c = 3, SL = 0.2, SR = 0.1$	0.00594405	0.04238656	0.05034670
$c = 3, SL = 0.3, SR = 0.1$	0.00593568	0.04183321	0.04949749
$c = 3, SL = 0.2, SR = 0.3$	0.00596550	0.04386186	0.05306106

Triangular 3 Tipo-2 con incertidumbre

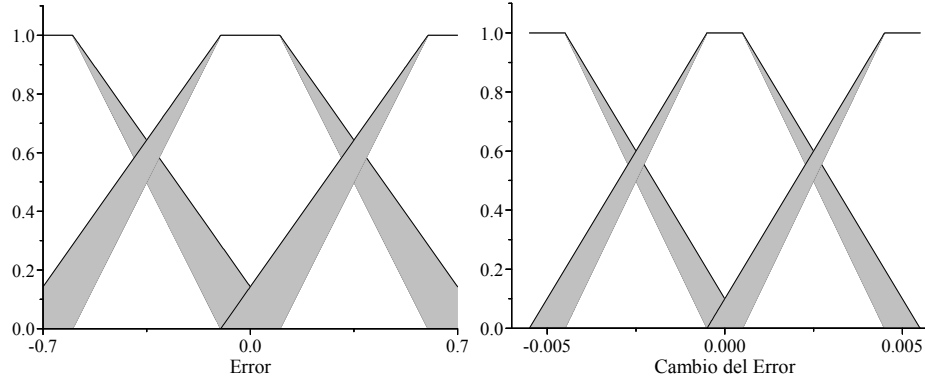


Figura 4.28 Funciones de membresía triangulares 3 tipo-2, para una línea de transmisión corta.

Tabla 4.27 Índices al añadir incertidumbre en los conjuntos triangulares 3 tipo-2, caso línea larga.

Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.00595626	0.04308419	0.05128703
$\delta_e = 0.1, \delta_{\Delta e} = 0.001$	0.00606637	0.06672305	0.17909647
$\delta_e = 0.1, \delta_{\Delta e} = 0.002$	0.00657089	0.09728089	0.32295397
$\delta_e = 0.1, \delta_{\Delta e} = 0.0005$	0.00620865	0.06584798	0.12853405
$\delta_e = 0.2, \delta_{\Delta e} = 0.001$	0.00630256	0.06723096	0.11617984
$\delta_e = 0.05, \delta_{\Delta e} = 0.001$	0.00608768	0.06876779	0.18346081

Tabla 4.28 Índices al añadir incertidumbre en las barras de salida, caso triangular 3 y línea larga.

Incertidumbre de entrada $\delta_e = 0.1, \delta_{\Delta e} = 0.0005$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.00620865	0.06584798	0.12853405
$c = 3, SL = 0.1, SR = 0.1$	0.00621241	0.06648575	0.13348377
$c = 3, SL = 0.1, SR = 0.2$	0.00648867	0.10306497	0.29803513
$c = 3, SL = 0.1, SR = 0.3$	0.00711856	0.13807519	0.45765220
$c = 3, SL = 0.2, SR = 0.1$	0.00632342	0.06829448	0.15927404
$c = 3, SL = 0.3, SR = 0.1$	0.00675359	0.09381050	0.29755559
$c = 3, SL = 0.2, SR = 0.3$	0.00652385	0.10418552	0.30625809

Triangular 4 Tipo-2 con incertidumbre

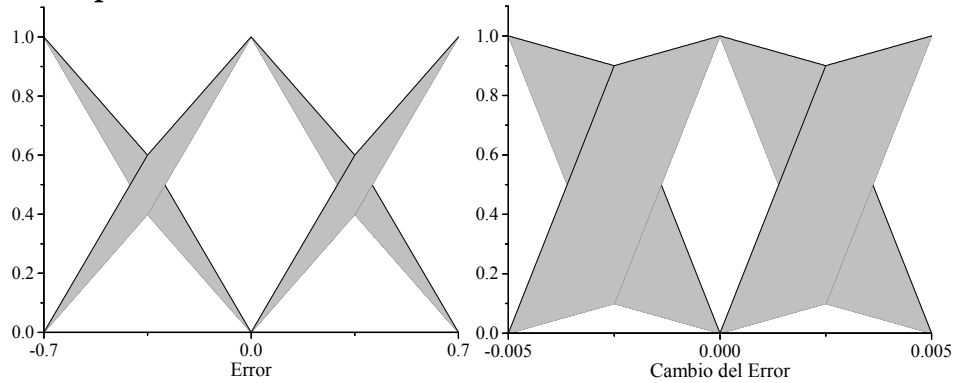


Figura 4.29 Funciones de membresía triangulares 4 tipo-2, para una línea de transmisión corta.

Tabla 4.29 Índices al añadir incertidumbre en los conjuntos triangulares 4 tipo-2, caso línea larga.			
Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.00595626	0.04308419	0.05128703
$\delta_e = 0.1, \delta_{\Delta e} = 0.1$	0.00595624	0.04308849	0.05129521
$\delta_e = 0.1, \delta_{\Delta e} = 0.2$	0.00595524	0.04305283	0.05123653
$\delta_e = 0.1, \delta_{\Delta e} = 0.3$	0.00595326	0.04296905	0.05109025
$\delta_e = 0.1, \delta_{\Delta e} = 0.4$	0.00594397	0.04253551	0.05034519
$\delta_e = 0.2, \delta_{\Delta e} = 0.1$	0.00595668	0.04309773	0.05131008
$\delta_e = 0.3, \delta_{\Delta e} = 0.1$	0.00595745	0.04311486	0.05133640
$\delta_e = 0.4, \delta_{\Delta e} = 0.1$	0.00595862	0.04314300	0.05138169
$\delta_e = 0.5, \delta_{\Delta e} = 0.1$	0.00596027	0.04318545	0.05145273

Tabla 4.30 Índices al añadir incertidumbre en las barras de salida, caso Gaussianas y línea larga.			
Incertidumbre de entrada $\delta_e = 0.1, \delta_{\Delta e} = 0.4$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 3, IL = 0.0, IR = 0.0$	0.00594397	0.04253551	0.05034519
$c = 3, IL = 0.1, IR = 0.1$	0.00595581	0.04335576	0.05186621
$c = 3, SL = 0.1, SR = 0.2$	0.00597079	0.04442441	0.05387394
$c = 3, SL = 0.1, SR = 0.3$	0.00598781	0.04556252	0.05610693
$c = 3, SL = 0.2, SR = 0.1$	0.00594758	0.04290943	0.05114994
$c = 3, SL = 0.3, SR = 0.1$	0.00594093	0.04251956	0.05054526
$c = 3, SL = 0.2, SR = 0.3$	0.00597698	0.04500389	0.05519046

Gaussiana Tipo-2 con incertidumbre

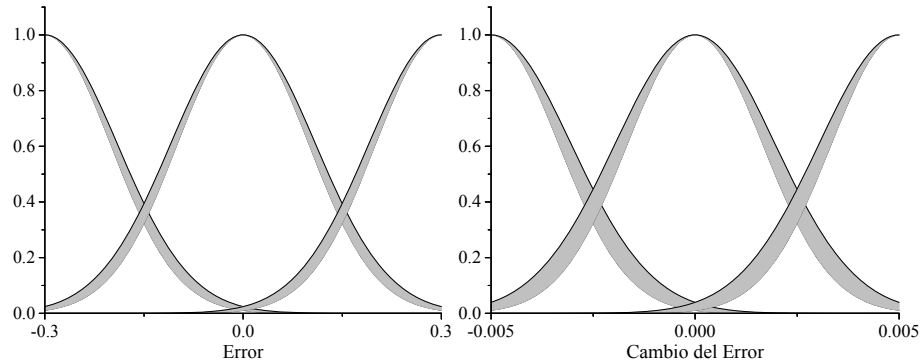


Figura 4.30 Funciones de membresía Gaussianas tipo-2, para una línea de transmisión corta.

Tabla 4.31 Índices al añadir incertidumbre en los conjuntos Gaussianos tipo-2, caso línea larga.			
Incertidumbre	Índices del Error		
	ISE	IAE	ITAE
$\delta_e = 0.0, \delta_{\Delta e} = 0.0$	0.00589182	0.04227825	0.05017648
$\delta_e = 0.01, \delta_{\Delta e} = 0.0001$	0.00589218	0.04236452	0.05048834
$\delta_e = 0.01, \delta_{\Delta e} = 0.0003$	0.00586957	0.03988342	0.04587661
$\delta_e = 0.01, \delta_{\Delta e} = 0.0005$	0.00587005	0.04060075	0.04748973
$\delta_e = 0.01, \delta_{\Delta e} = 0.00005$	0.00590223	0.04389518	0.05384190
$\delta_e = 0.03, \delta_{\Delta e} = 0.0001$	0.00594618	0.05039367	0.07946928

Tabla 4.32 Índices al añadir incertidumbre en las barras de salida, caso Gaussianas y línea larga.			
Incertidumbre de entrada $\delta_e = 0.01, \delta_{\Delta e} = 0.0003$	Índices del Error		
	ISE	IAE	ITAE
Incertidumbre en los consecuentes			
$c = 5, IL = 0.0, IR = 0.0$	0.00586957	0.03988342	0.04587661
$c = 5, IL = 0.1, IR = 0.1$	0.00587101	0.04005924	0.04620128
$c = 5, SL = 0.1, SR = 0.2$	0.00583857	0.04853276	0.08194619
$c = 5, SL = 0.1, SR = 0.3$	0.00582815	0.05707004	0.11766481
$c = 5, SL = 0.2, SR = 0.1$	0.00592539	0.04728672	0.07984737
$c = 5, SL = 0.3, SR = 0.1$	0.00599719	0.05492036	0.11410515
$c = 5, SL = 0.2, SR = 0.3$	0.00587251	0.04023884	0.04655158

4.3.3 Análisis de la sintonización del IT2 FLC caso línea larga

De igual forma que en la sección 4.2.3, se sigue el criterio de una adecuada respuesta dinámica de la variable a controlar, un correcto recorrido del diagrama de fases, aunque en este caso solo se muestran los valores de los índices de error. La superficie de control no varía según línea largo o corta, puesto que no depende del sistema a controlar, si no mas bien de los parámetros del controlador.

En suma, los parámetros adecuados para el controlador tipo 2 según el tipo de función de membresía utilizada, tenemos:

Tabla 4.33 Resumen de mejores parámetros del IT2 FLC, caso línea larga.

Tipos de funciones de membresía tipo 2	Incertidumbre en los universos de entrada		Incertidumbre en el universo de salida	
	δ_e	$\delta_{\Delta e}$	SL	SR
Triangular 1 tipo-2 $A = 0.7; B = 0.005$ y $C = 3$	0.3	0.0005	0.0	0.0
Triangular 2 tipo-2 $A = 0.7; B = 0.005$ y $C = 3$	0.1	0.2	0.3	0.1
Triangular 3 tipo-2 $A = 0.7; B = 0.005$ y $C = 3$	0.1	0.0005	0.0	0.0
Triangular 4 tipo-2 $A = 0.7; B = 0.005$ y $C = 3$	0.1	0.4	0.3	0.1
Gaussianas tipo-2 $A = 0.3; B = 0.005$ y $C = 5$	0.01	0.0003	0.0	0.0

Tabla 4.34 Mejores índices según Tabla 4.33.

Tipo de función de Membresía	Índices del Error		
	ISE	IAE	ITAE
Triangular 1 tipo-2	0.00594647	0.04804508	0.08074424
Triangular 2 tipo-2	0.00593568	0.04183321	0.04949749
Triangular 3 tipo-2	0.00620865	0.06584798	0.12853405
Triangular 4 tipo-2	0.00594093	0.04251956	0.05054526
Gaussiana tipo-2	0.00586957	0.03988342	0.04587661

De la tabla anterior, se infiere que el mejor caso en la funciones de membresía triangulares resulto ser triangular 2 tipo-2. Pero a diferencia que en línea corta, esta vez se requirió incertidumbre del orden de 0.1, 0.2. Mientras que la incertidumbre en las funciones de membresía Gaussianas sigue siendo pequeña para los conjuntos difusos de entrada, y nula en las barras consecuentes de salida.

Tabla 4.35 Parámetros para el controlador tipo-2, caso línea larga.

Tipos de funciones de membresía tipo 2	Incertidumbre en los universos de entrada		Incertidumbre en el universo de salida	
	δ_e	$\delta_{\Delta e}$	SL	SR
Triangular 2 tipo-2 $A = 0.7; B = 0.005$ y $C = 3$	0.1	0.2	0.3	0.1
Gaussianas tipo-2 $A = 0.3; B = 0.003$ y $C = 5$	0.01	0.0003	0.0	0.0

4.4 Pruebas para una línea de transmisión corta, sujeta a un disturbio grande

Se presentan los resultados con el generador con carga y sujeto a un corto circuito en el bus que ocurre a 1s de tiempo de simulación, con duración de 6 y 10 ciclos. Las condiciones iniciales fueron:

$$V_{ter} = 1 P.U. \quad \theta = 6.8077^\circ \quad P_A = 1 P.U. \quad P_R = -0.1474 P.U.$$

Los parámetros de ambos controladores son tomados de la **Tabla 4.5** y **Tabla 4.18**. Para el caso del controlador tipo-2 con funciones de membresía triangulares se obtuvieron

mejores resultados sin añadir incertidumbre en los consecuentes por lo que $SL = 0$ y $SR = 0$.

4.4.1 Controladores con funciones de membresía triangulares y 9 reglas de conocimiento

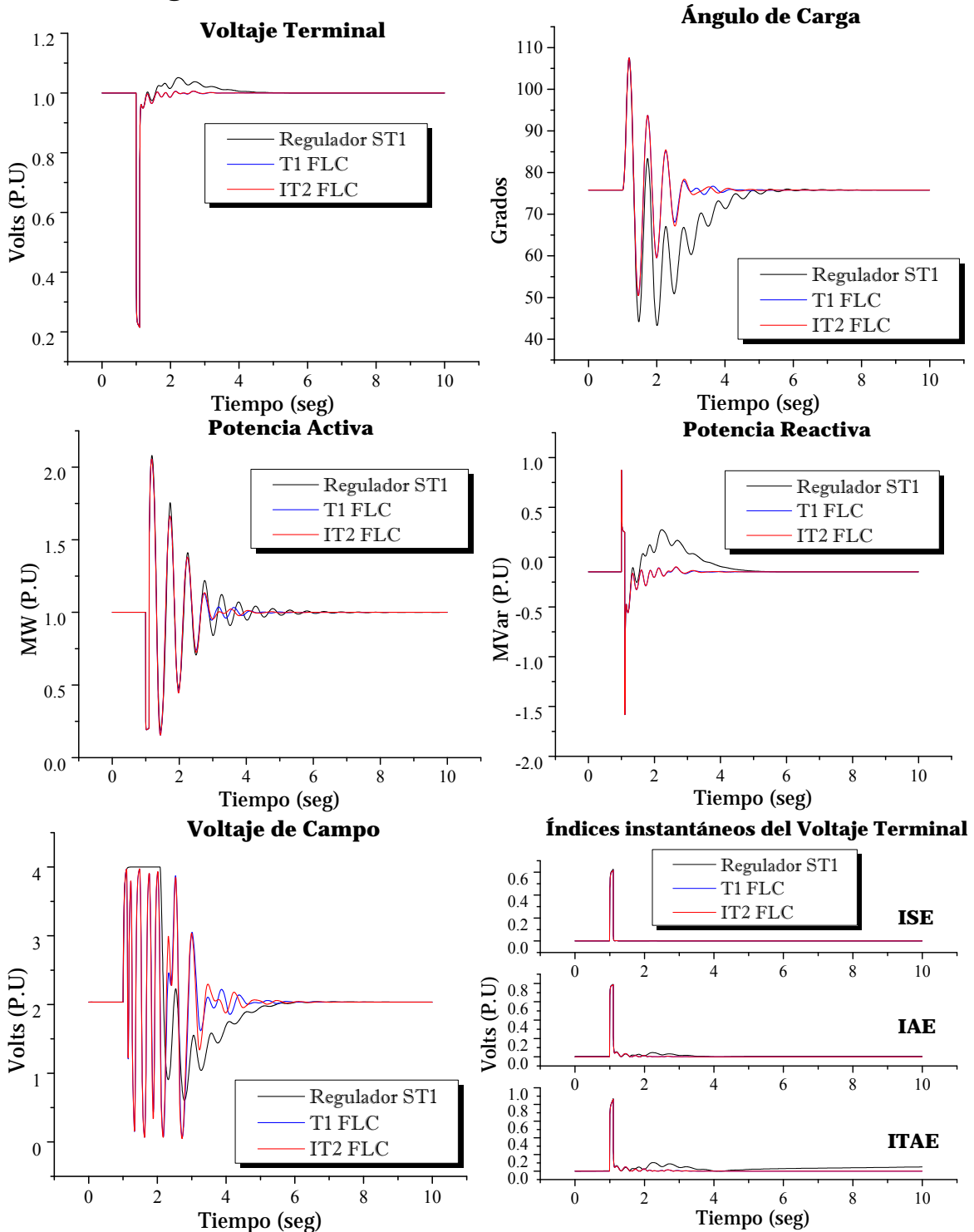
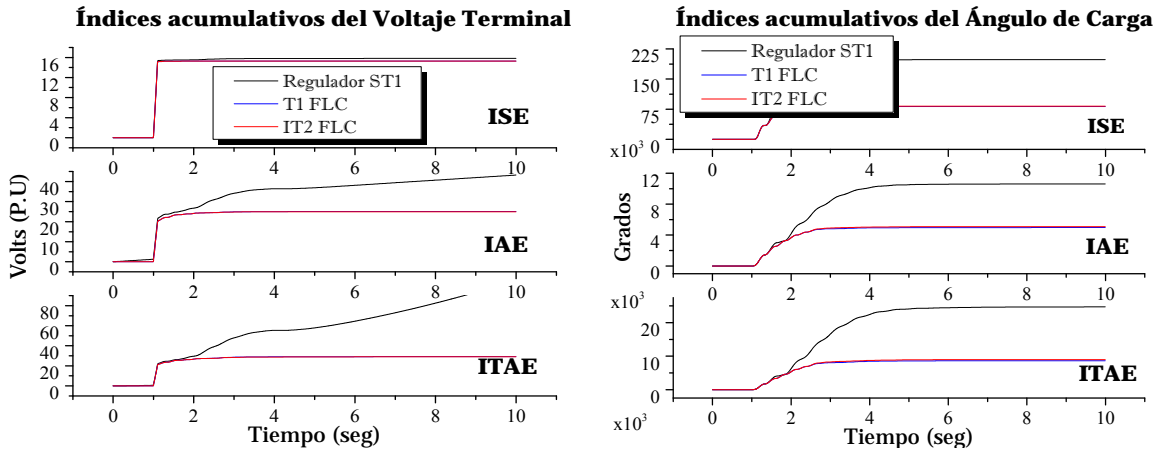


Figura 4.31 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.



Continuación de Figura 4.31 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.

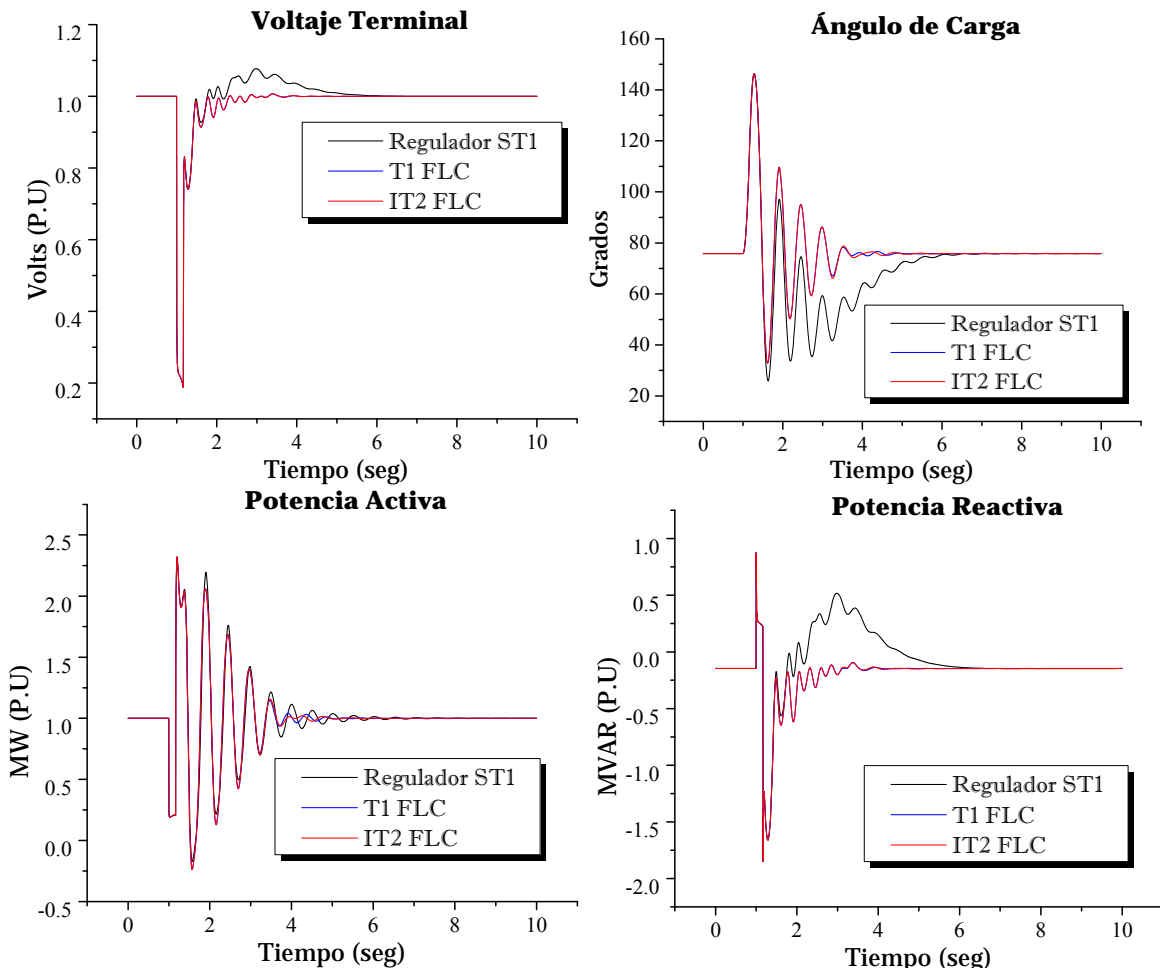
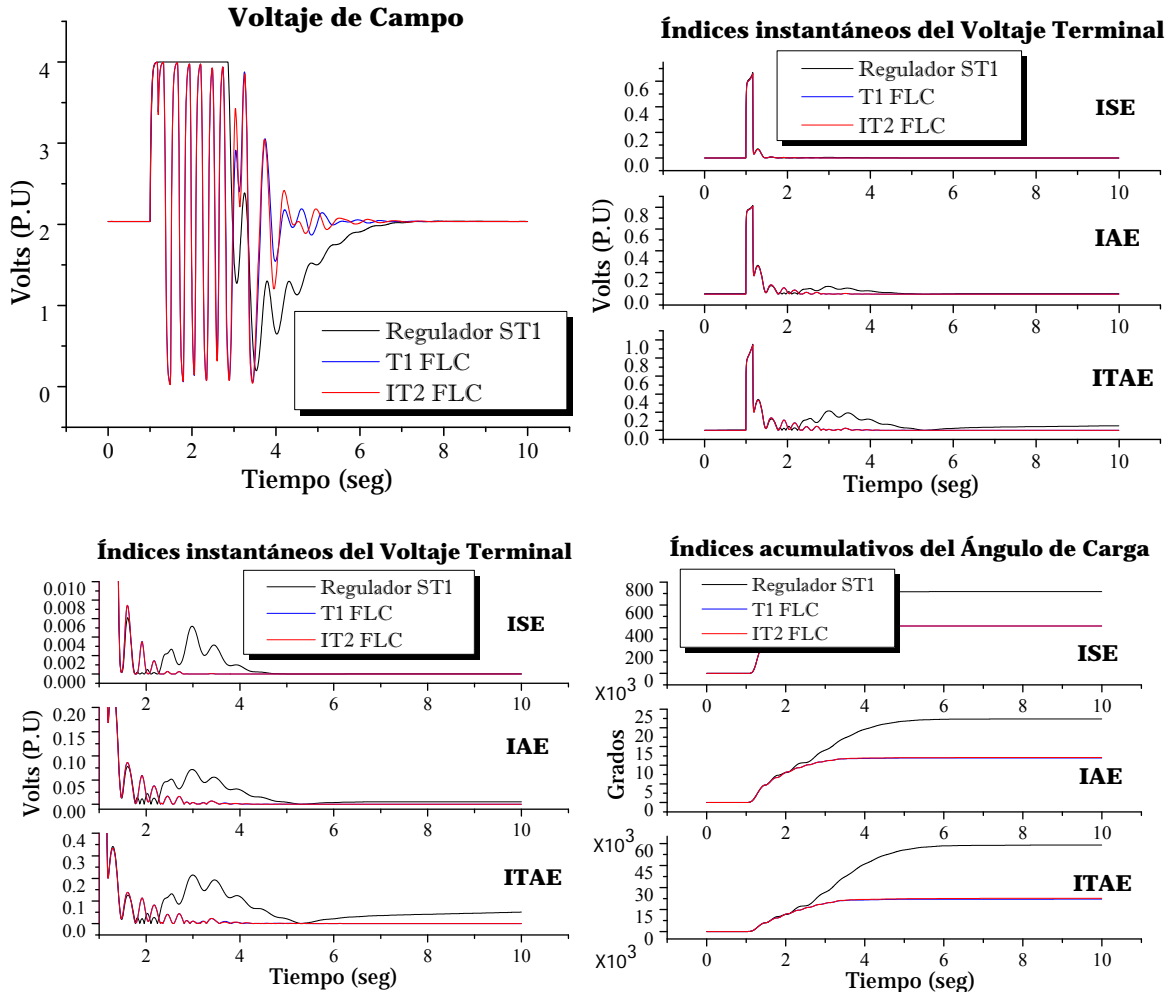


Figura 4.32 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.



Continuación de Figura 4.32 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.

4.4.2 Controladores con funciones de membresía triangulares y 25 reglas de conocimiento

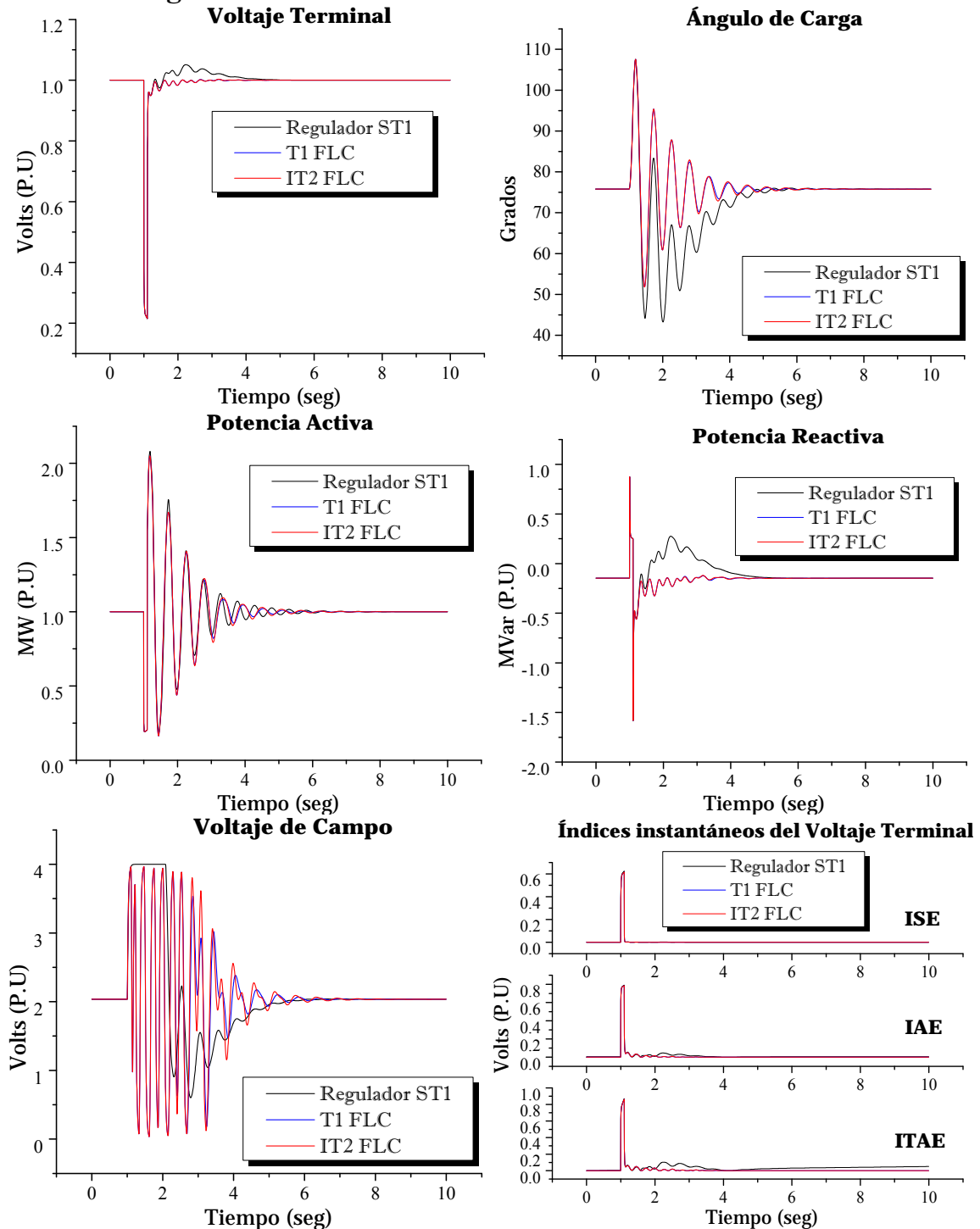
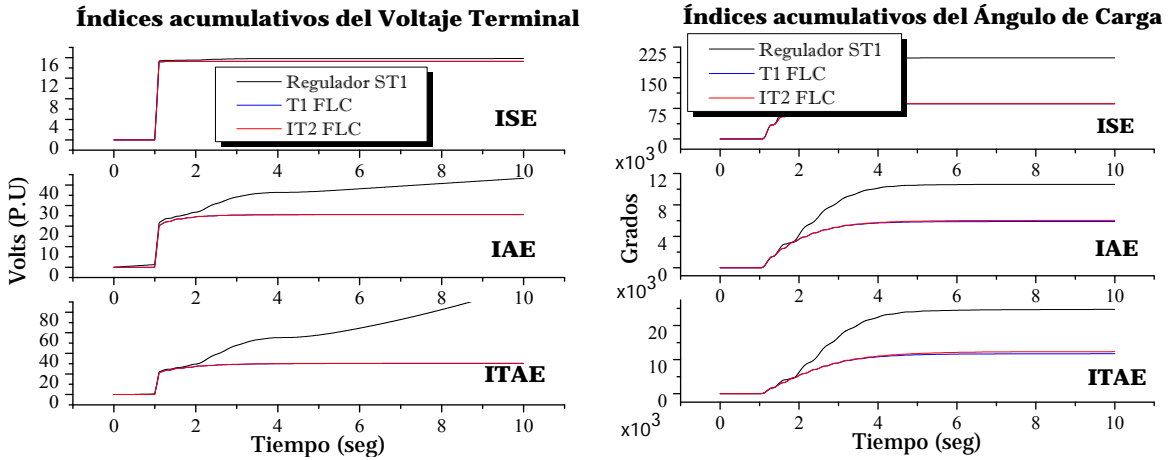


Figura 4.33 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.



Continuación de Figura 4.33 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.

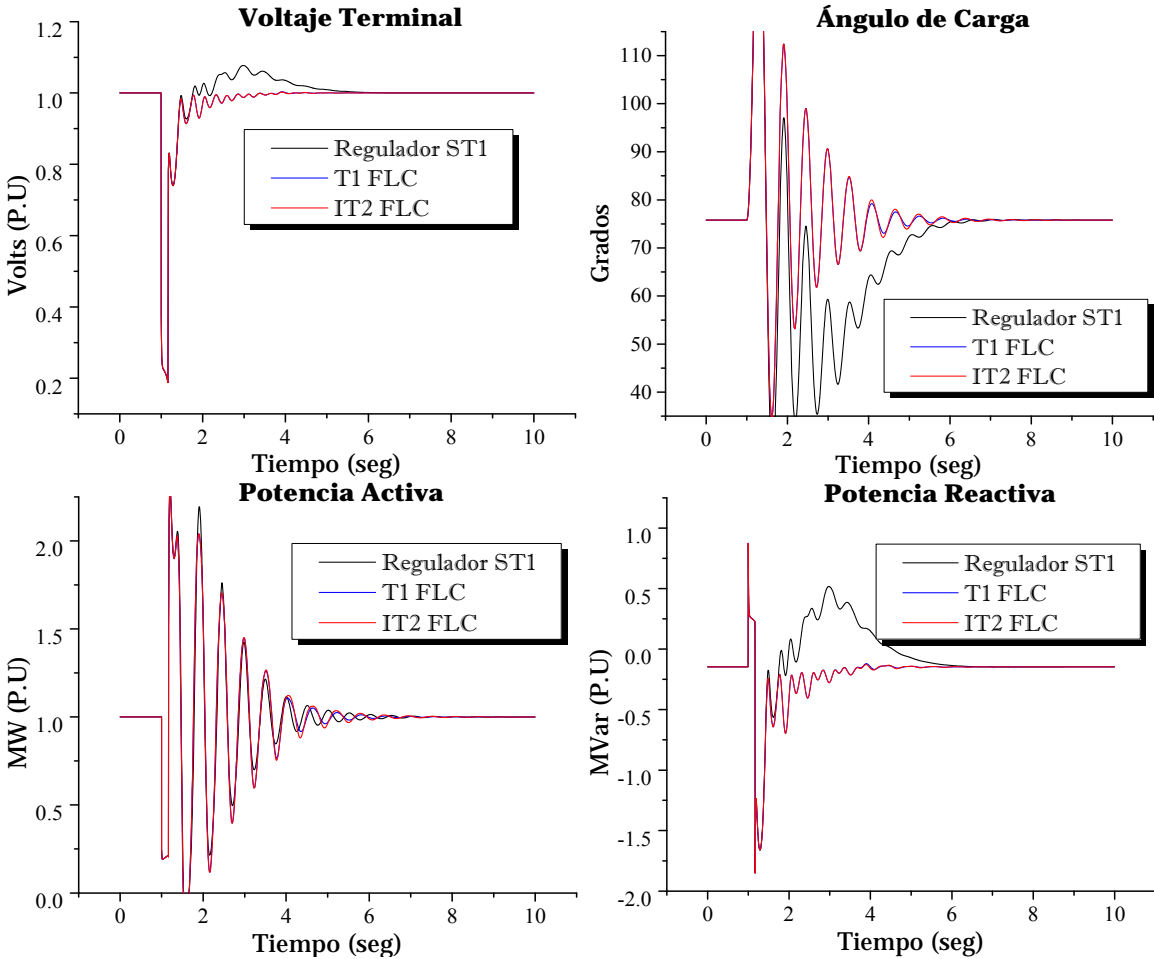
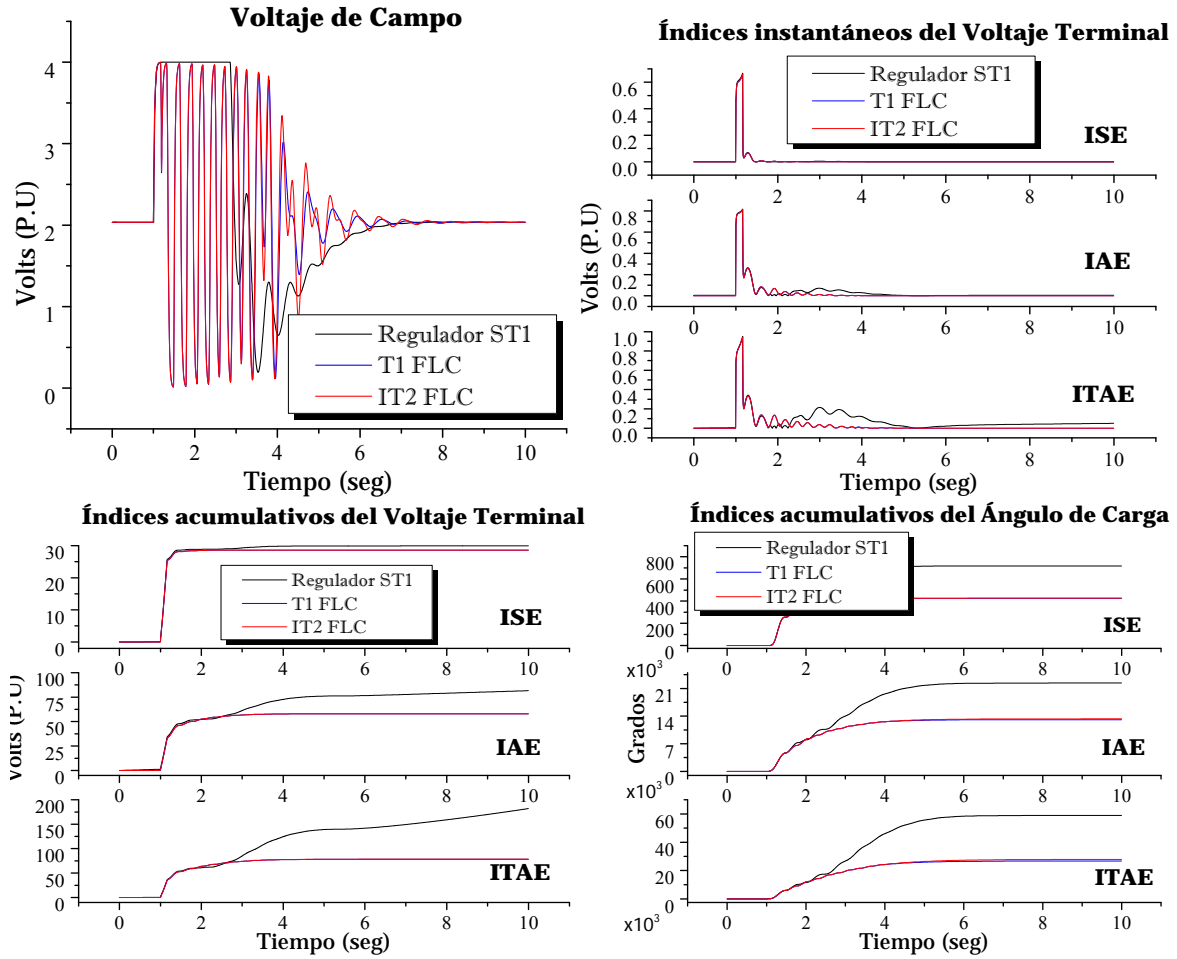


Figura 4.34 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.



Continuación de Figura 4.34 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.

4.4.3 Controladores con funciones de membresía triangulares y 49 reglas de conocimiento

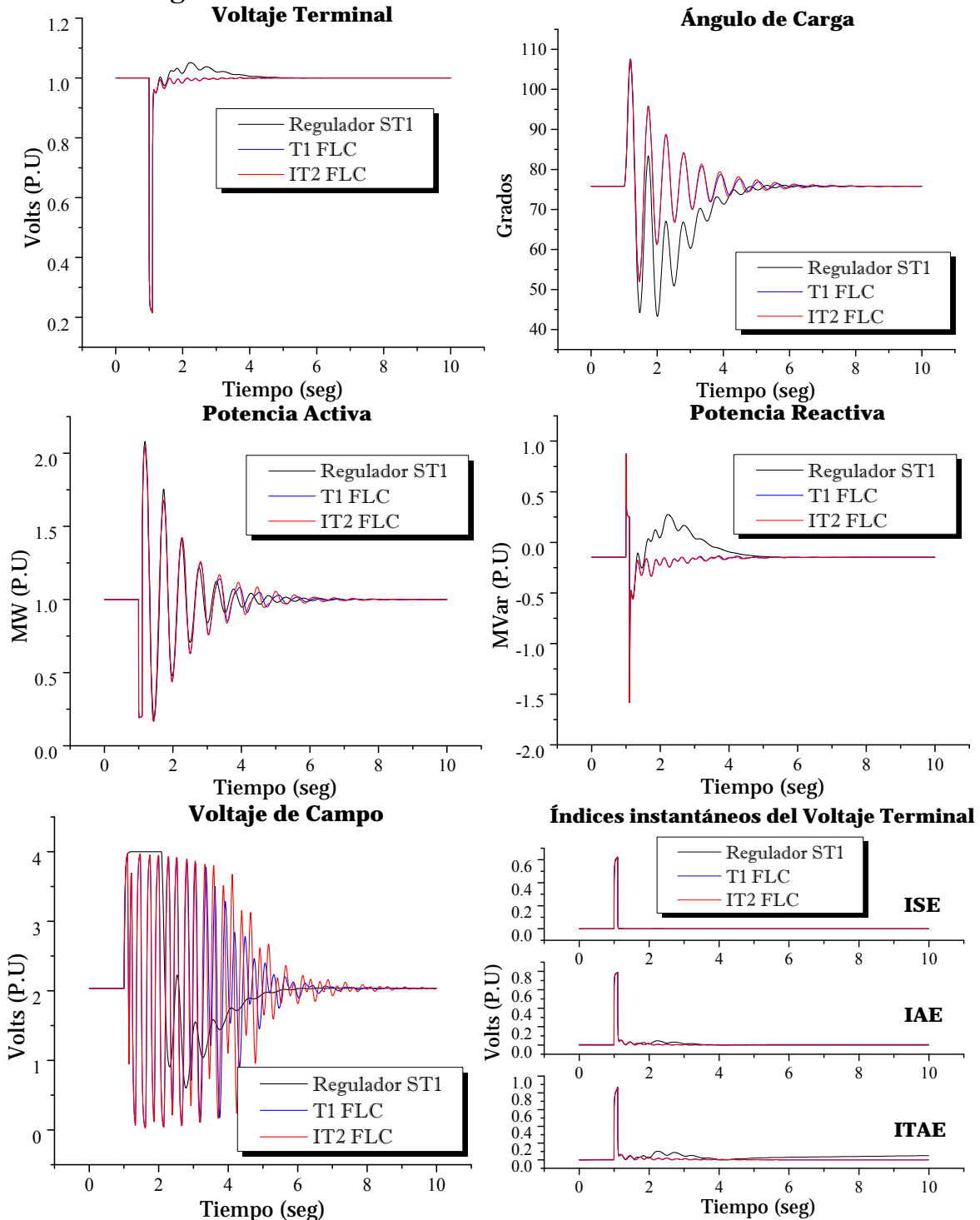
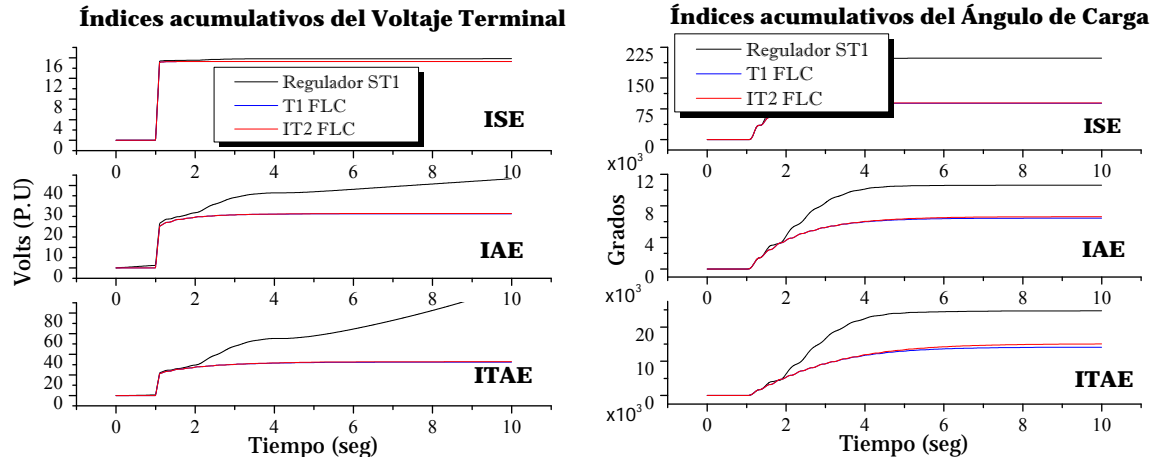


Figura 4.35 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.



Continuación de Figura 4.35 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.

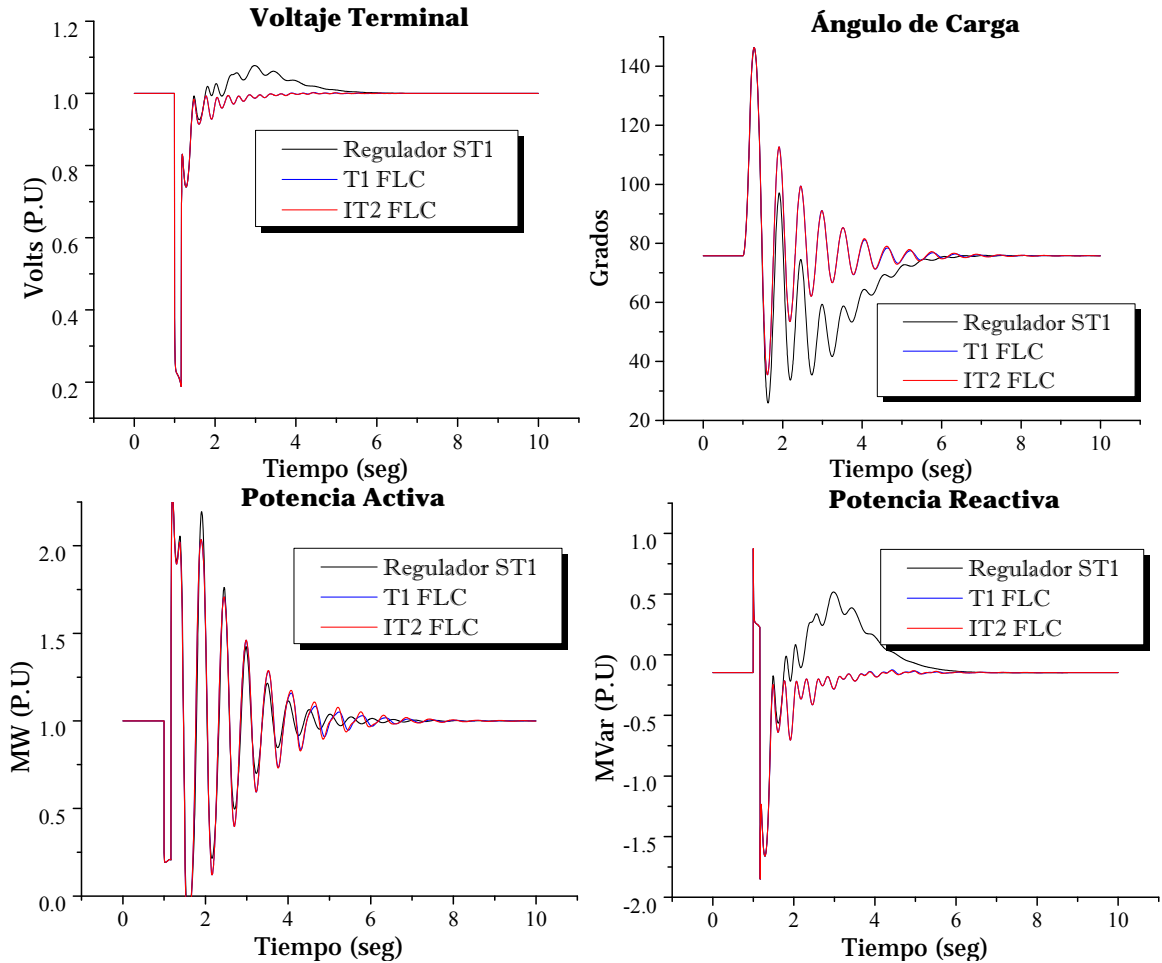
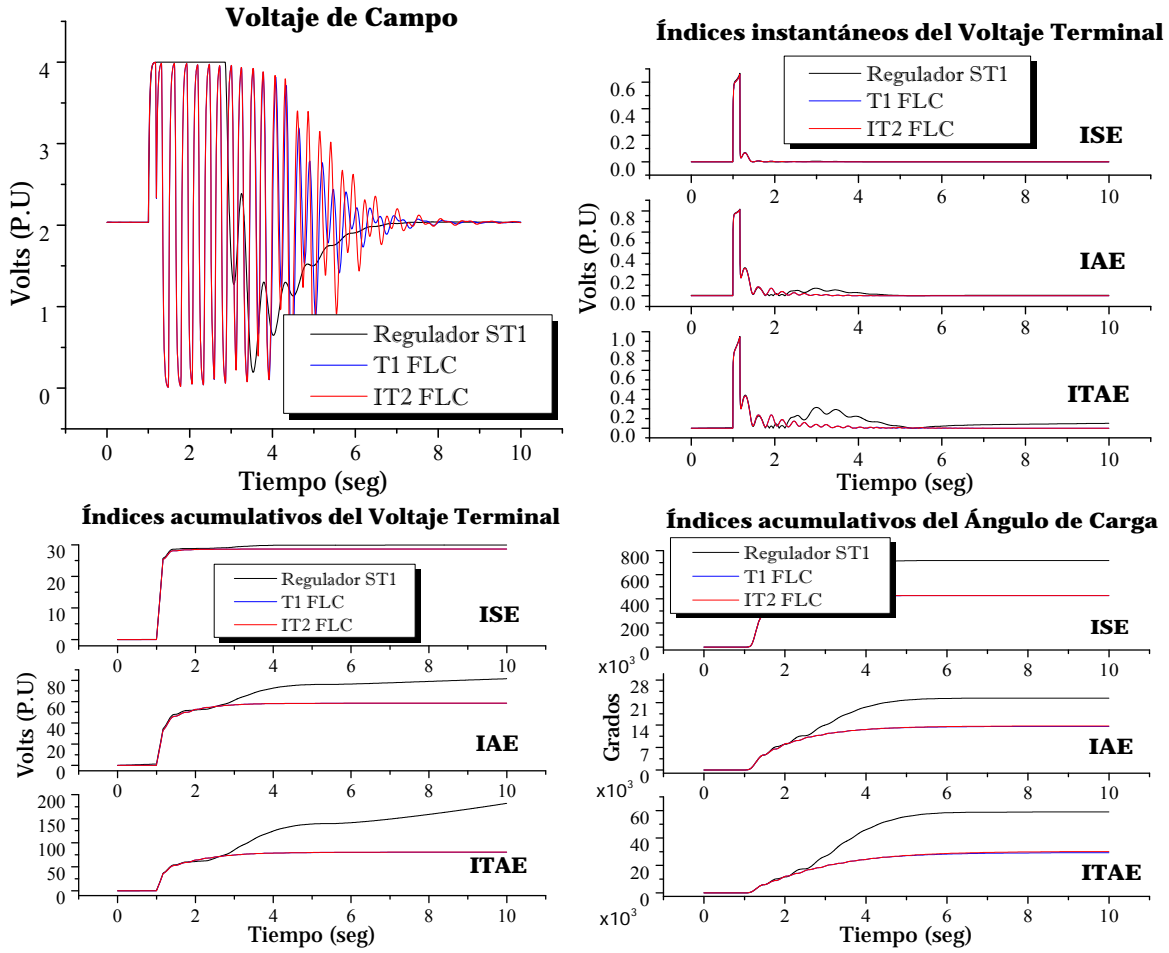


Figura 4.36 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.



Continuación de Figura 4.36 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.

4.4.4 Controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento

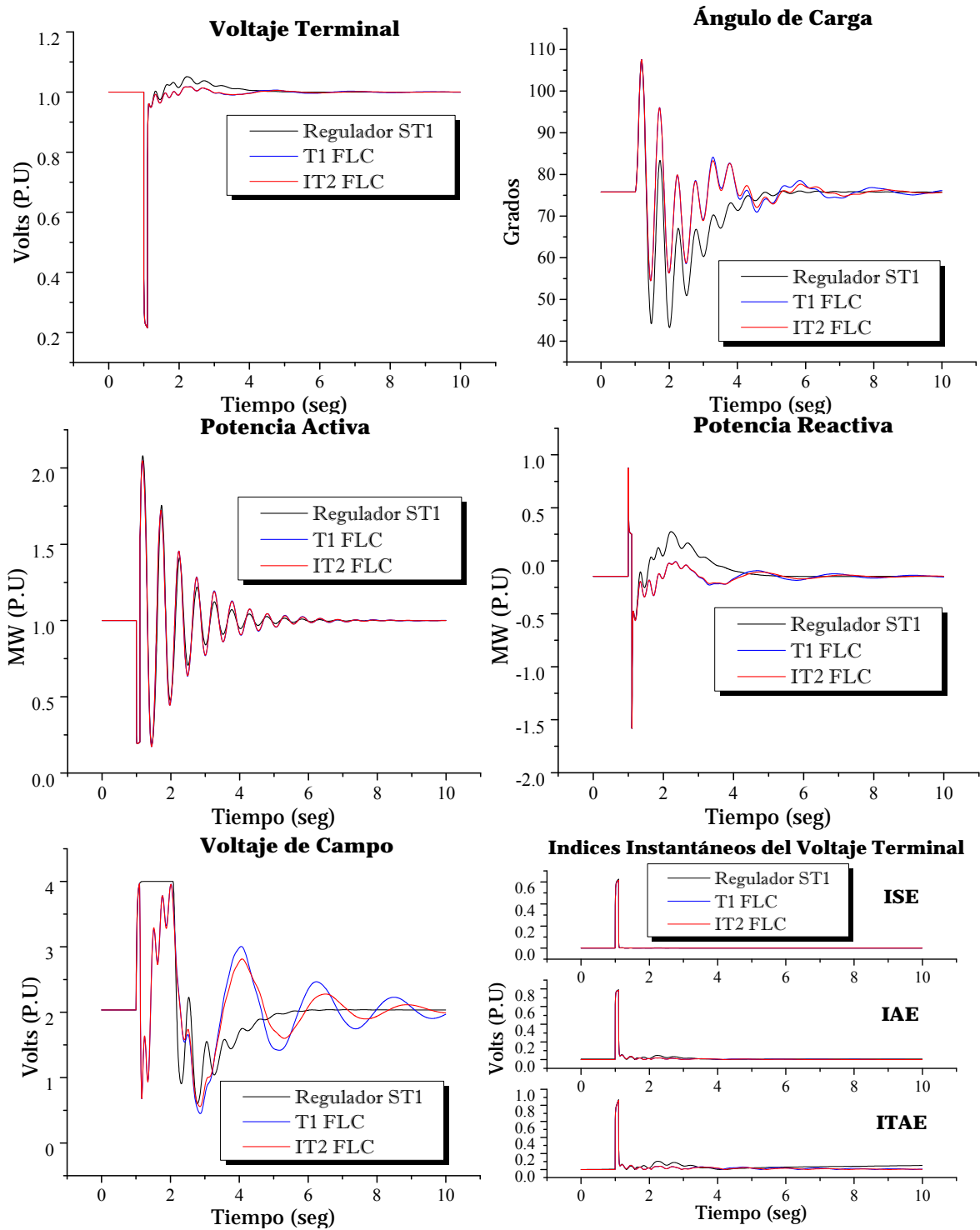
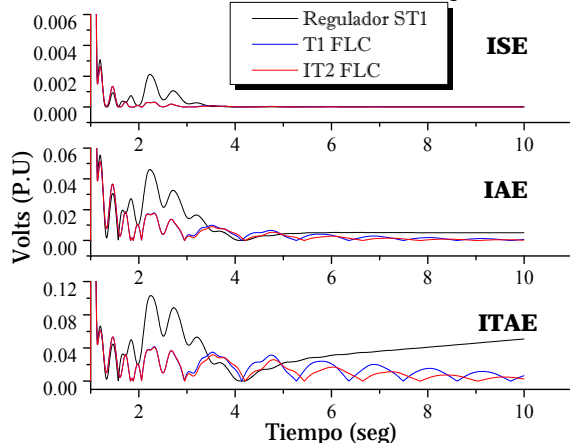
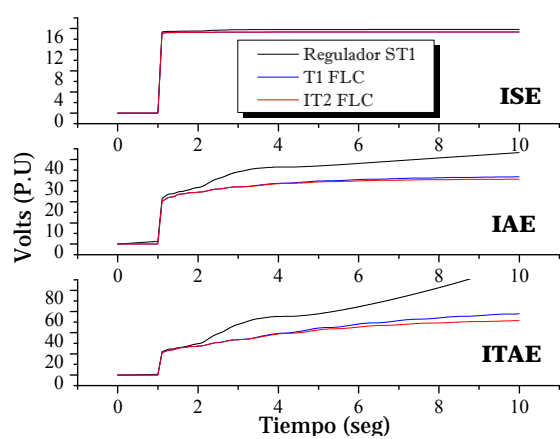
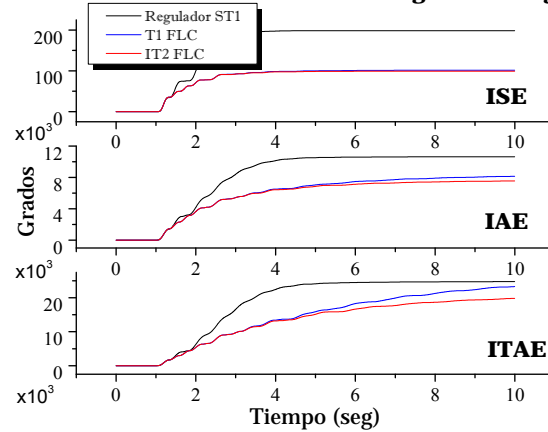


Figura 4.37 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.

Detalle Índices Instantáneos del Voltaje Terminal**Índices acumulativos del Voltaje Terminal****Índices acumulativoS del Ángulo de Carga**

Continuación de Figura 4.37 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.

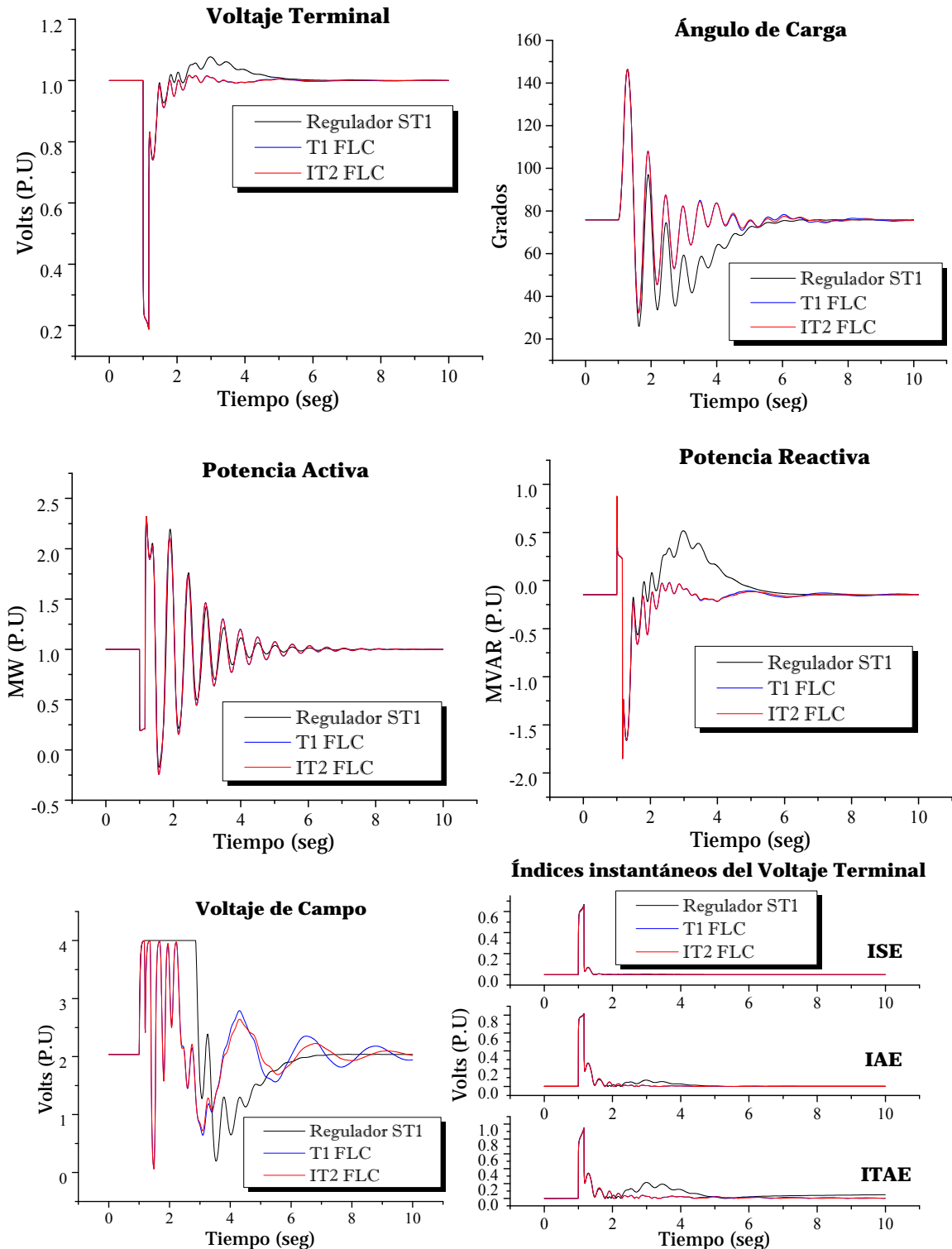
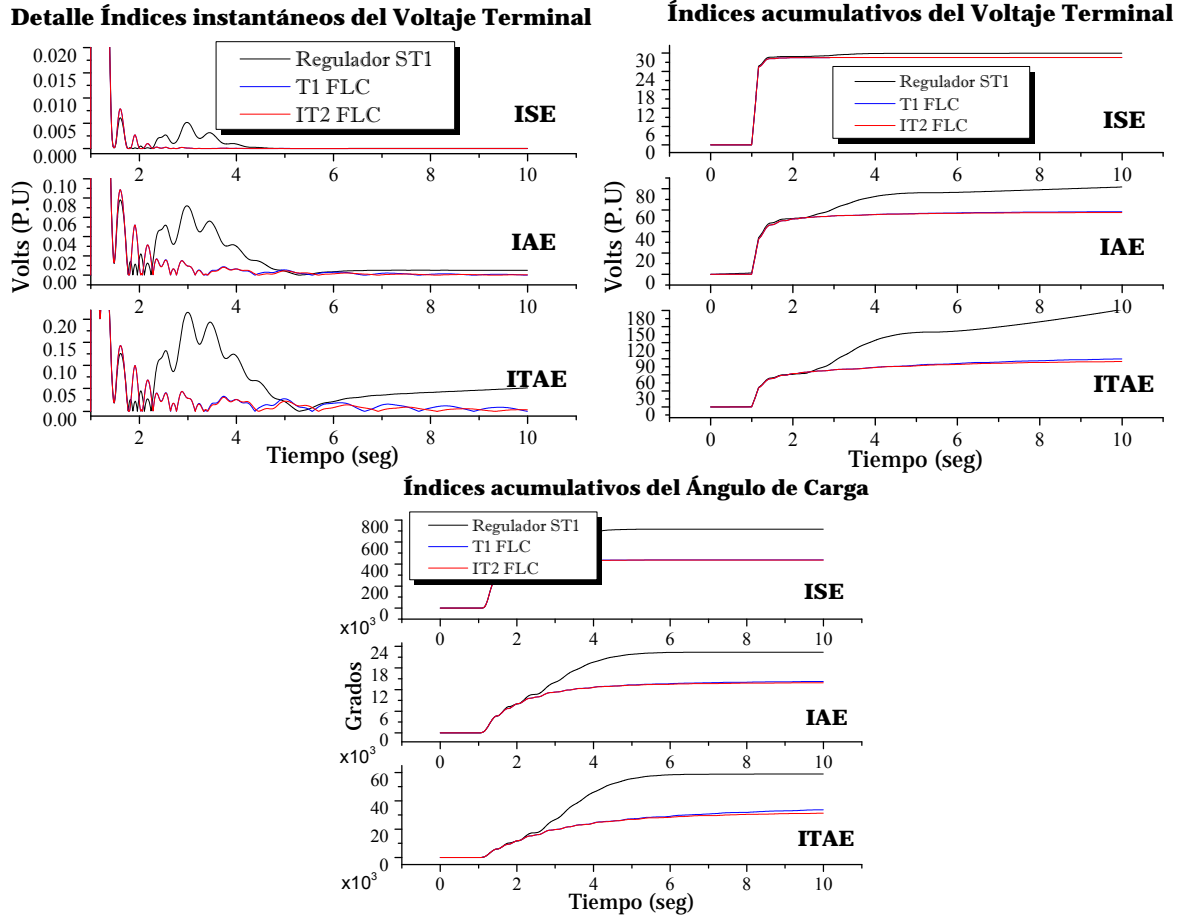


Figura 4.38 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.



Continuación de Figura 4.38 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.

4.4.5 Controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento

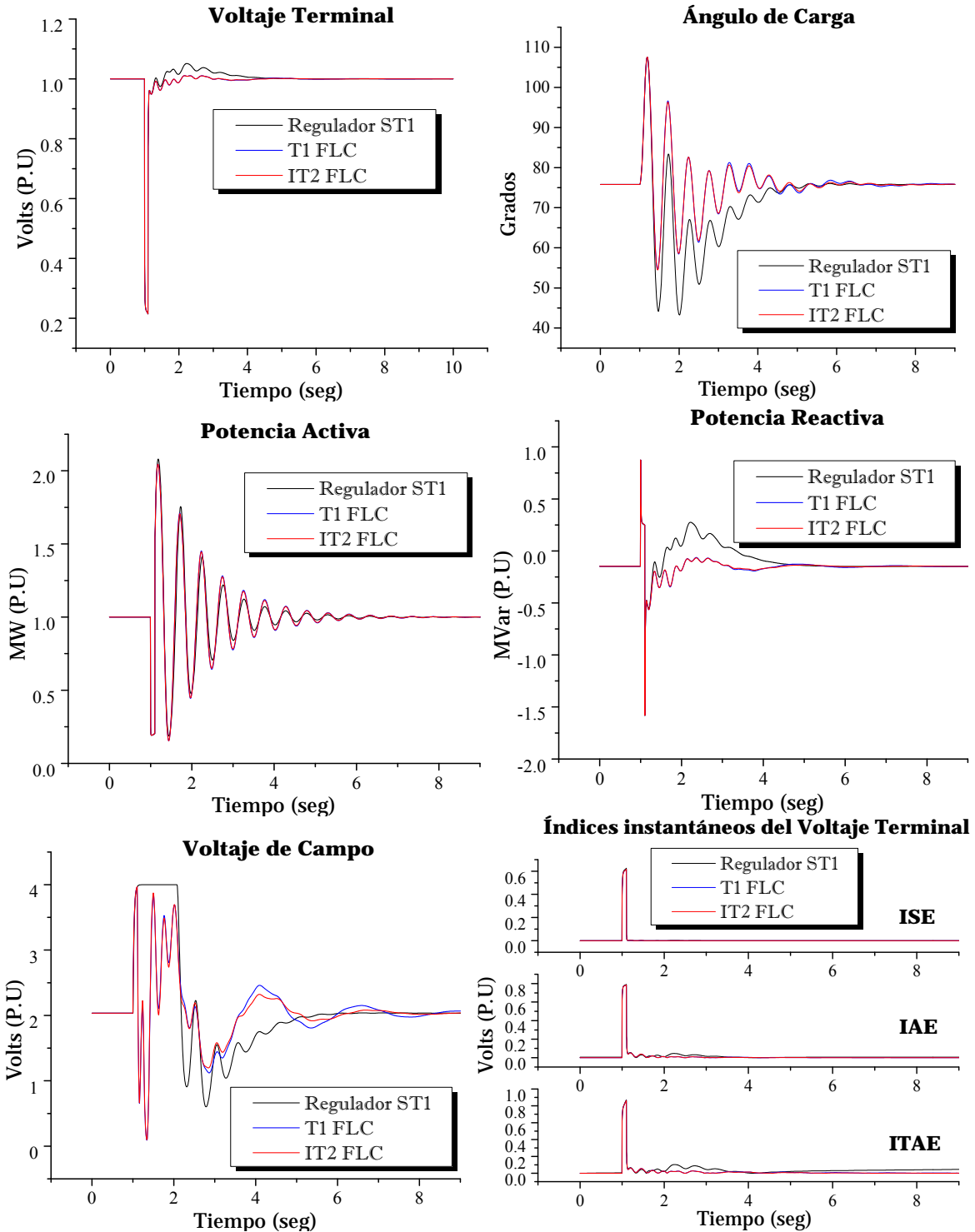
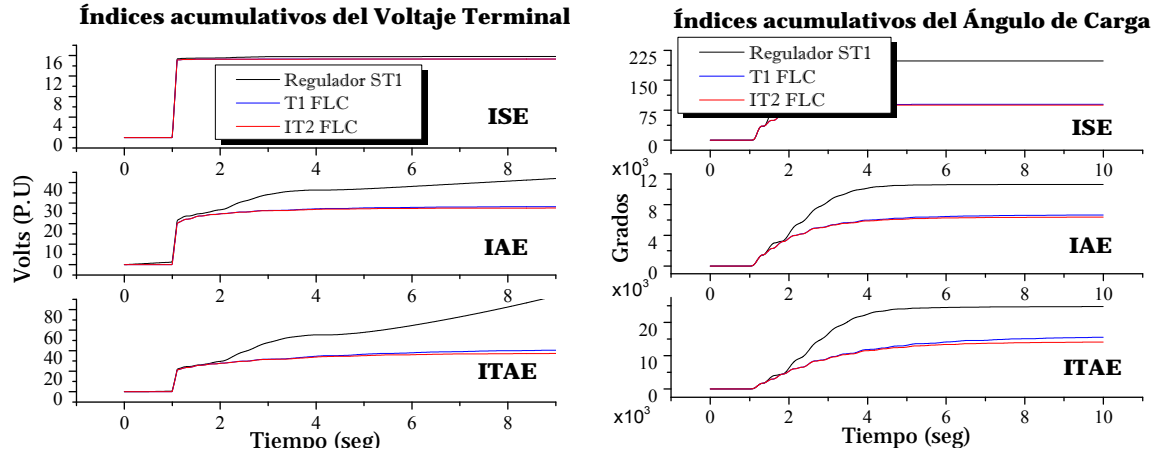


Figura 4.39 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.



Continuación de Figura 4.39 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.

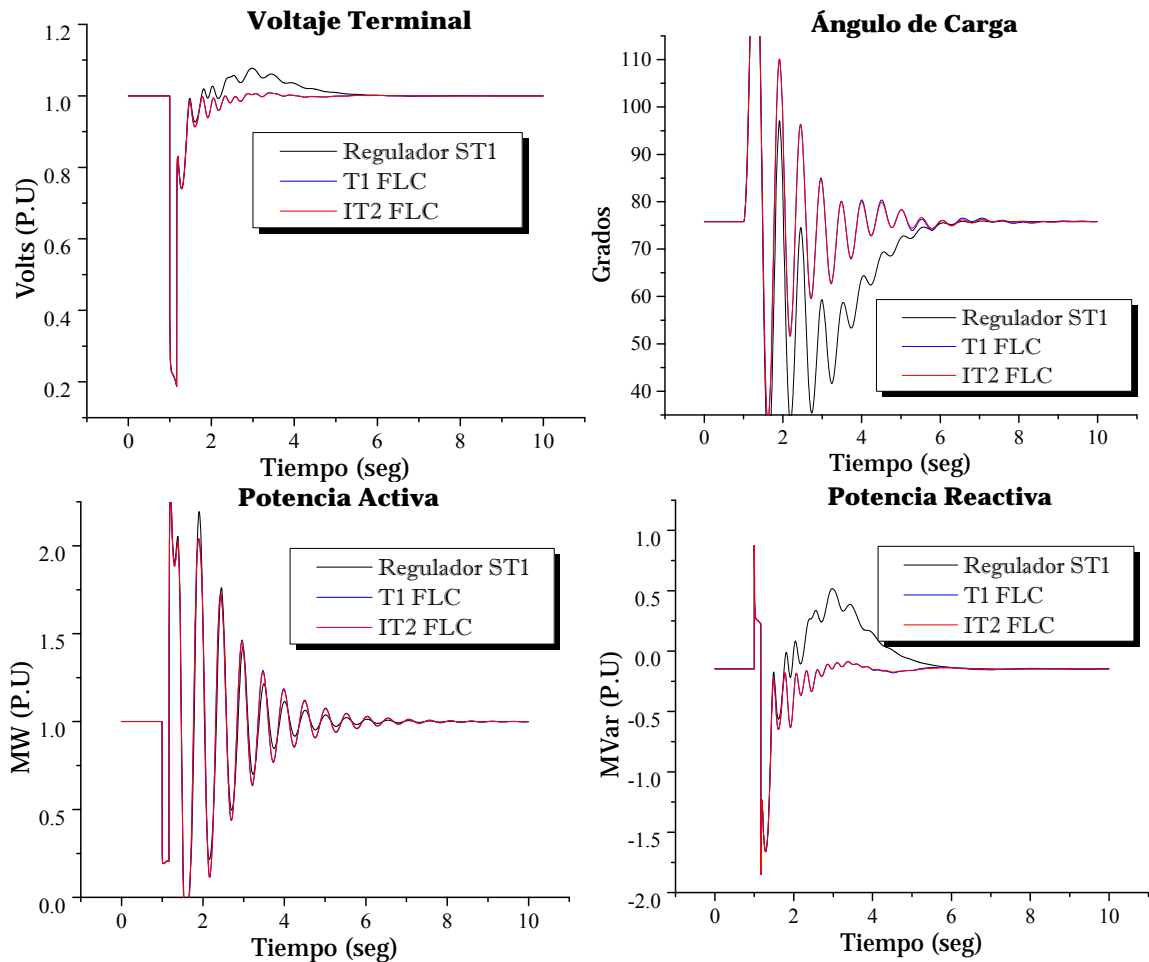
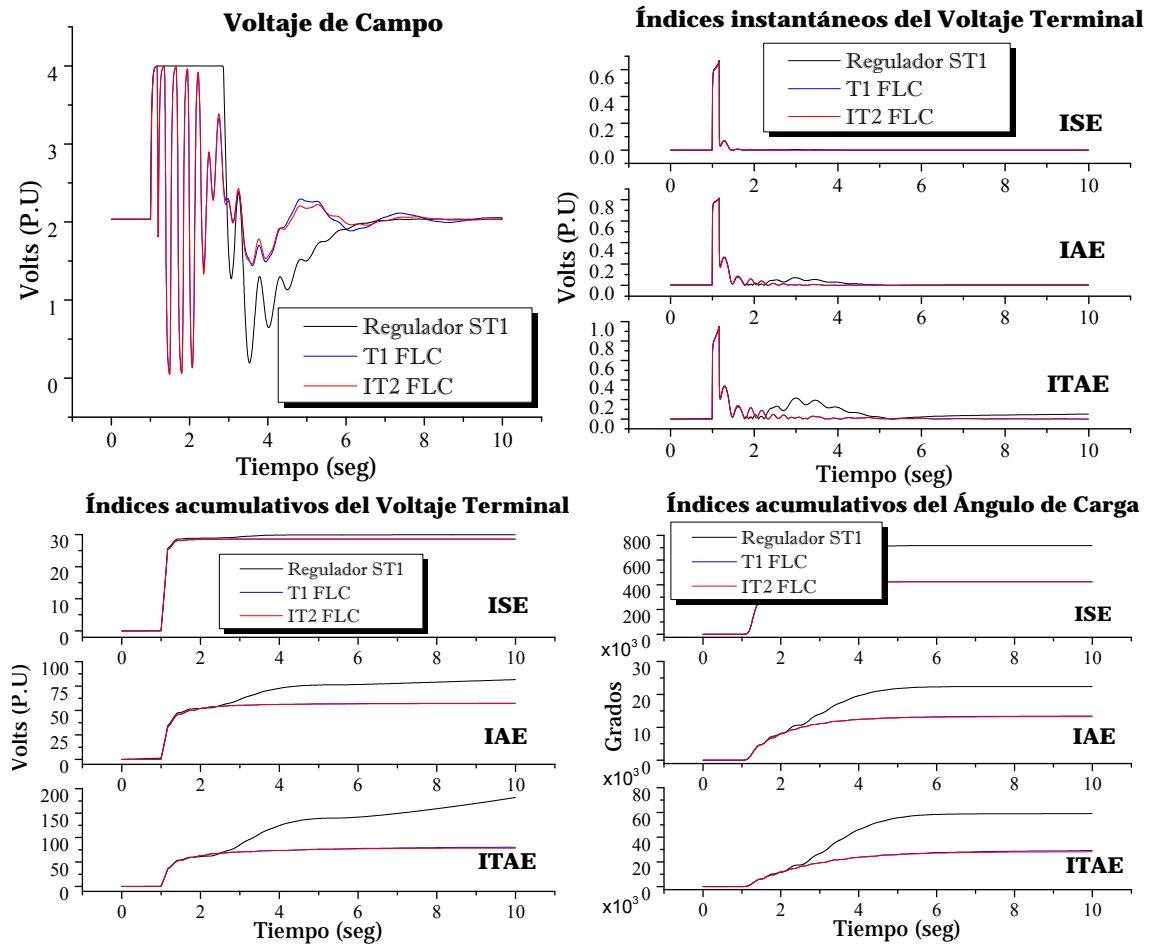


Figura 4.40 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.



Continuación de Figura 4.40 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.

4.4.6 Controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento

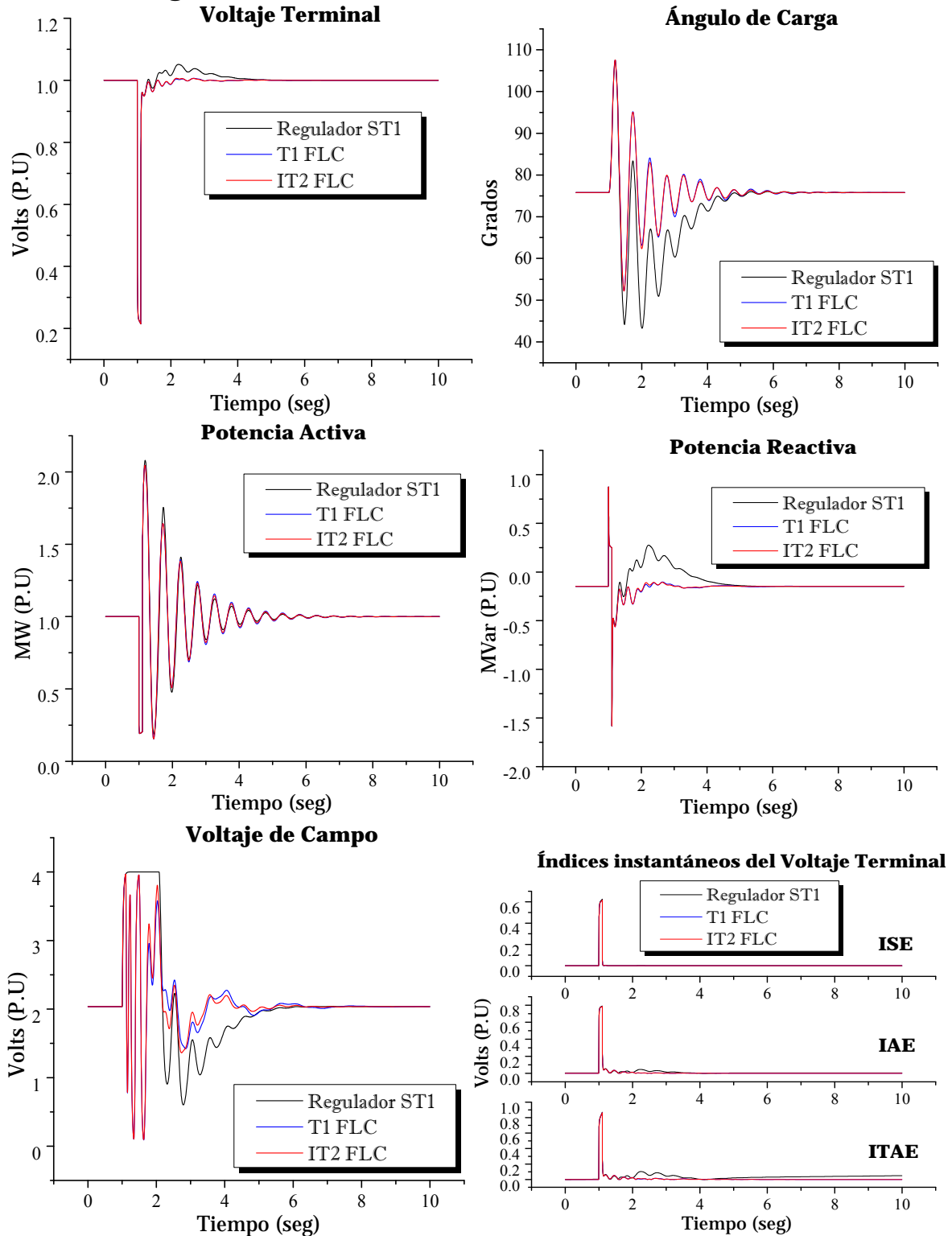
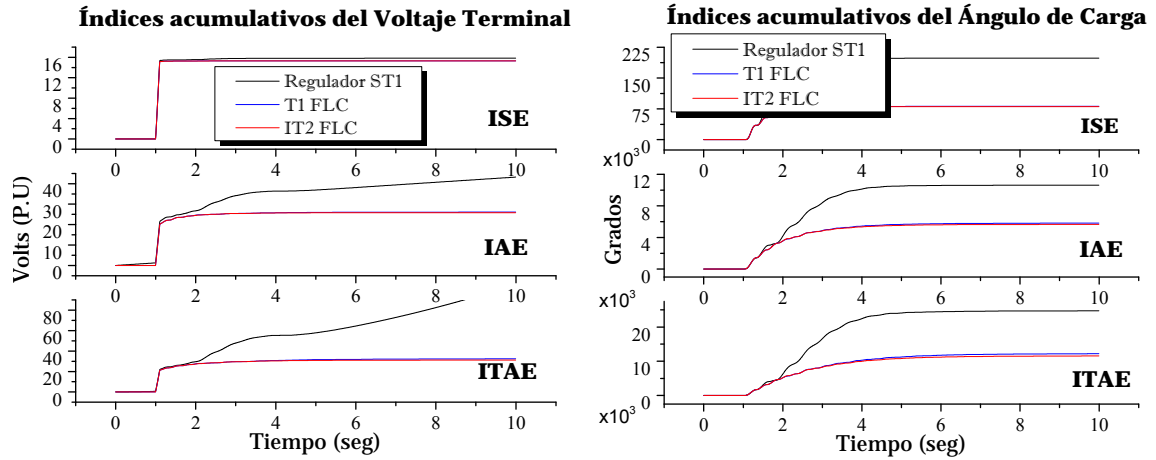


Figura 4.41 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.



Continuación de Figura 4.41 Respuesta del sistema sujeto a corto circuito con 6 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.

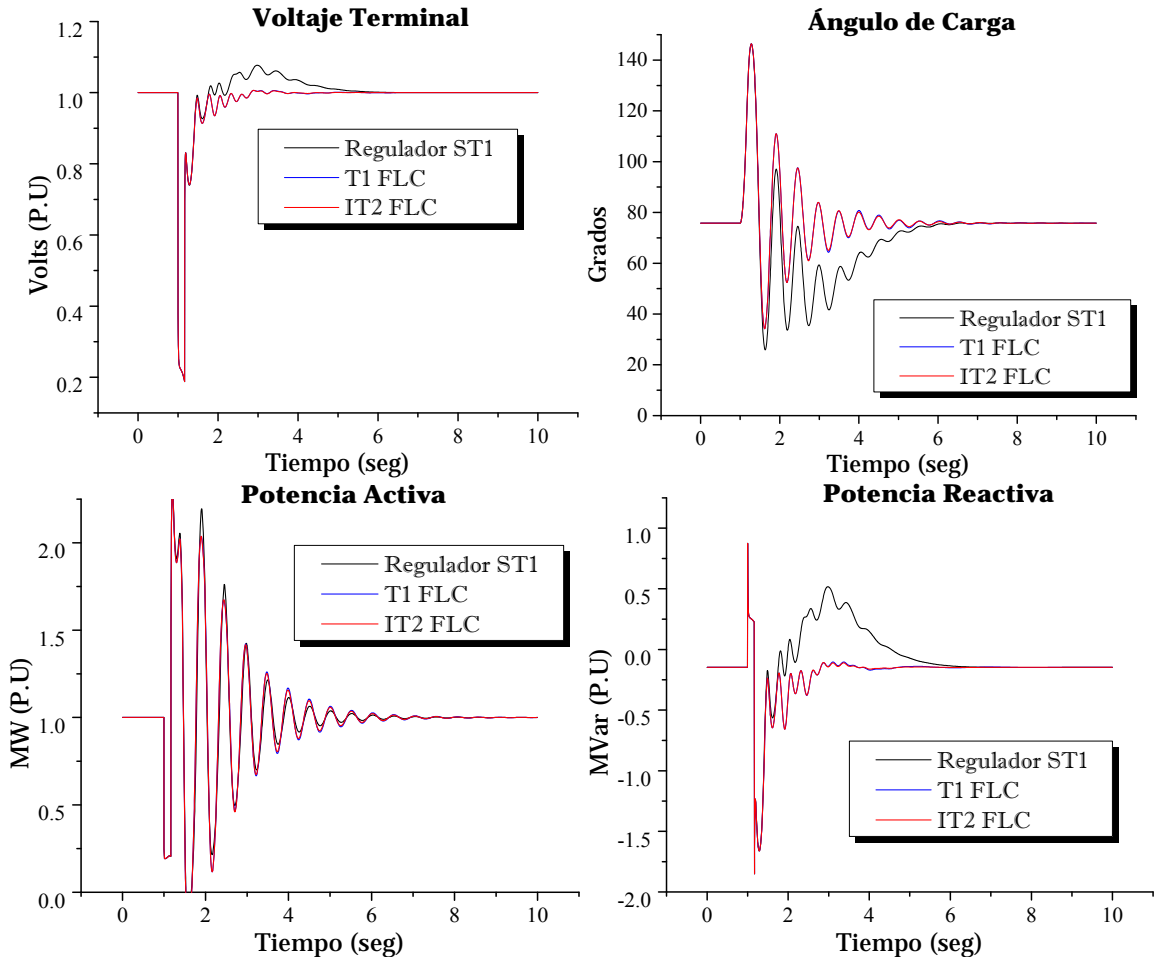
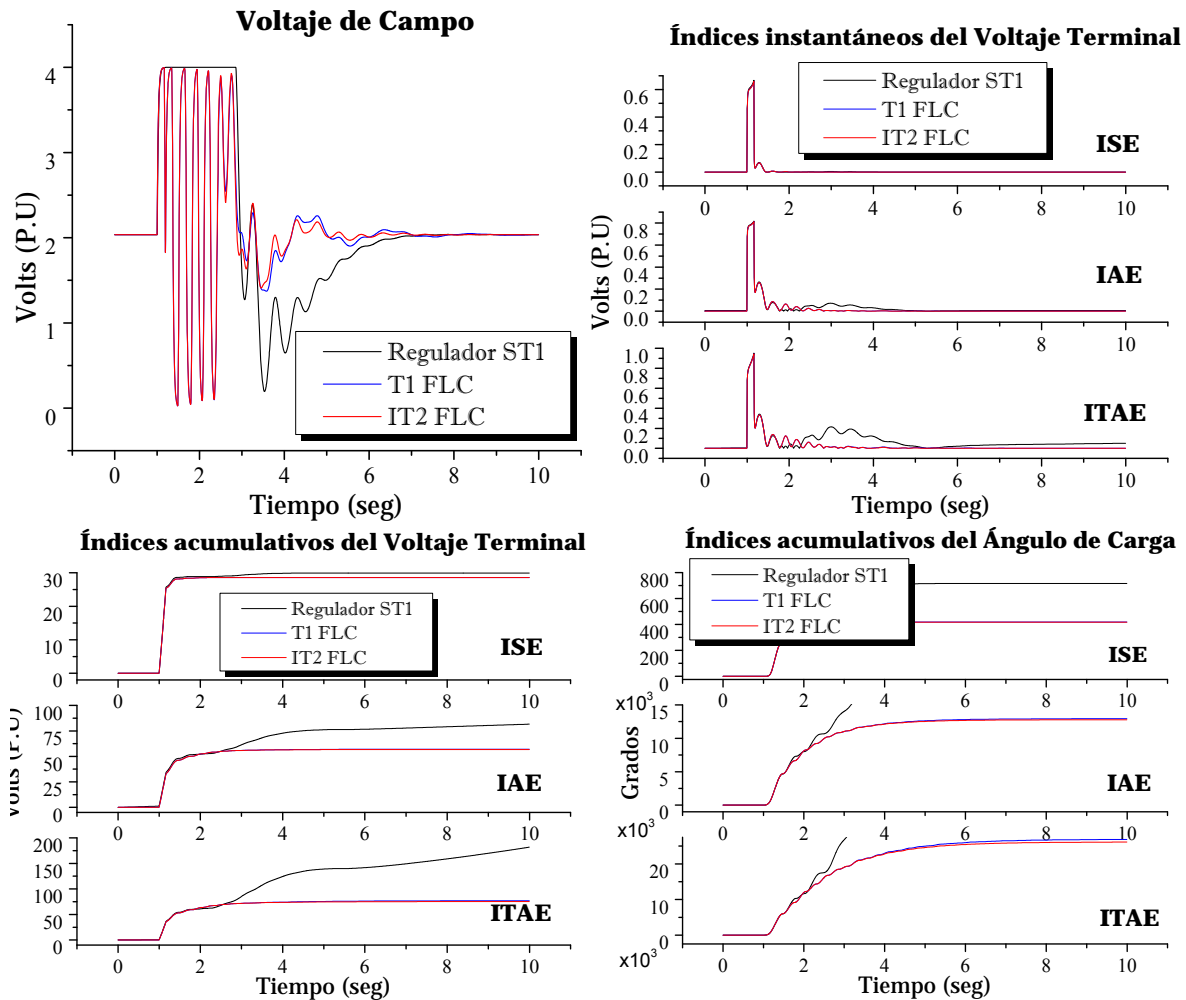


Figura 4.42 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.



Continuación de Figura 4.42 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.

4.4.7 Tiempo crítico de liberación de falla

El tiempo crítico de liberación de la falla para el generador síncrono con una línea de transmisión corta, y controlador con funciones de membresía Gaussianas y 9 reglas de conocimiento se muestra la **Figura 4.43**. Este tiempo crítico fue el mismo para las funciones de membresía Gaussianas 25 y 49 reglas de conocimiento y también para funciones de membresía triangulares con 9, 25 y 49 reglas de conocimiento.

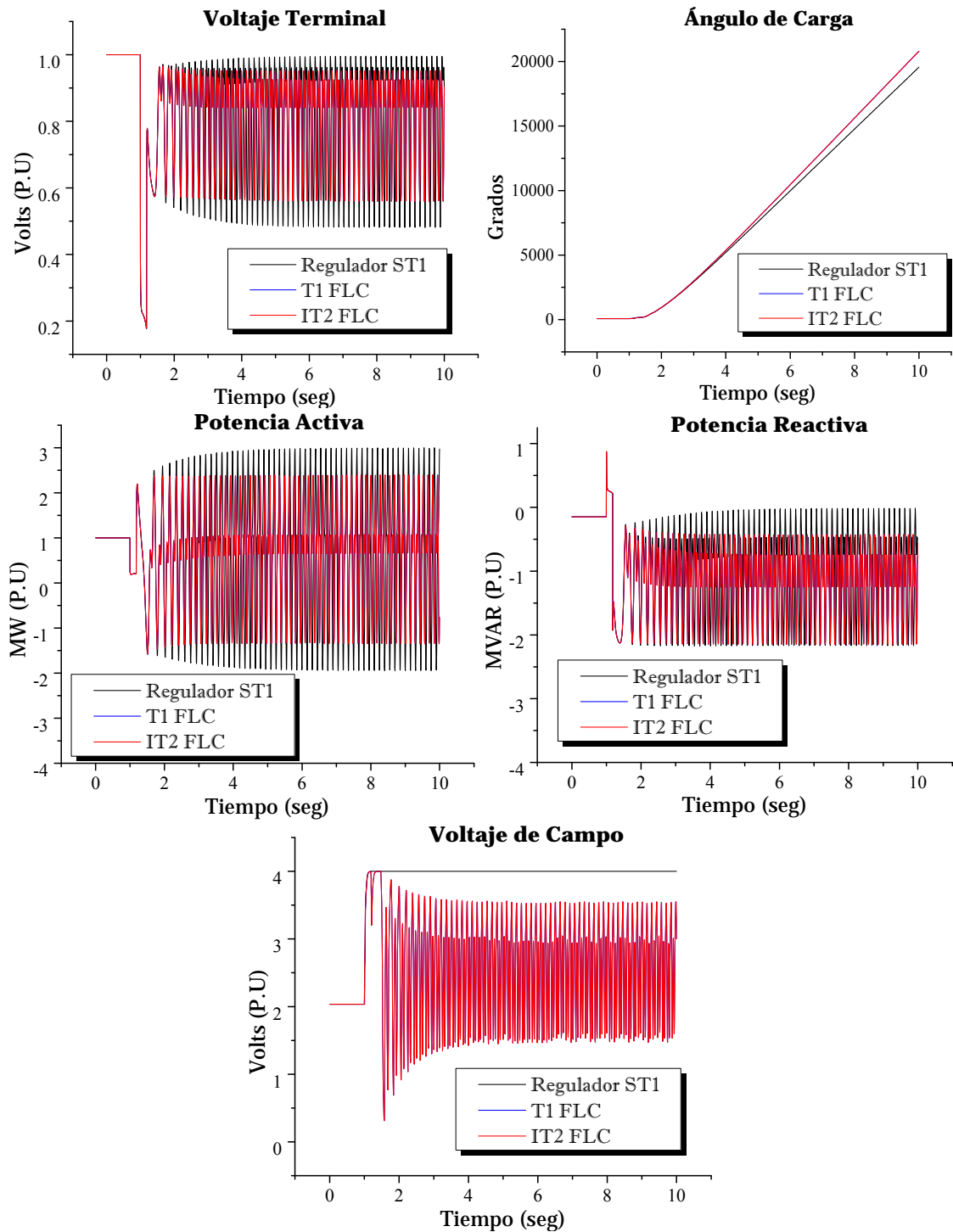


Figura 4.43 Respuesta del sistema sujeto a corto circuito con 11 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.

4.4.8 Análisis de resultados

Para el caso de funciones de membresía triangulares, la respuesta ante un gran disturbio, es casi idénticas entre el controlador tipo-1 y tipo-2. Siendo ligeramente superiores en algunos casos el controlador tipo-2, para fallas con duración de 6 ciclos. Los índices acumulativos sea del voltaje terminal o del ángulo de carga se encuentran en la mayoría de los casos sobrepuestos. Así como las respuestas de la potencia reactiva y a ángulo de carga.

Para el caso de funciones de membresía Gaussianas, la respuesta del controlador tipo-2 es superior en la mayoría de los casos. El voltaje de campo de presenta siempre mejor respuesta en el controlador tipo-2 sobre el convencional tipo-1. Además ambos controladores estabilizan el voltaje de campo en la máquina síncrona cuando se aumenta el conocimiento del controlador (número de reglas difusas). Los índices instantáneos y acumulativos del error en el voltaje terminal, muestran mejor comportamiento para el controlador IT2 FLC sobre el T1 FLC.

Ambos controladores difusos superan al regulador convencional ST1.

El tiempo de liberación crítica de falla tanto para el regulador ST1, el T1 FLC y el IT2 FLC es de 11 ciclos para la condición inicial presentada.

Nótese que el multiplicador de $\times 10^3$ en la escala de los índices acumulativos del ángulo de carga se deben a que el error fue cuantificado directamente en grados y no en P.U.

4.5 Para una línea de transmisión corta, sujeto a un disturbio grande e incertidumbre en el voltaje terminal

Las siguientes pruebas siguientes muestran los resultados al añadir incertidumbre en la lectura del voltaje terminal según **Figura 3.22**. La condición inicial considerada es:

$$V_{ter} = 1 P.U. \quad \theta = 6.8077^\circ \quad P_A = 1 P.U. \quad P_R = -0.1474 P.U.$$

Se inicia añadiendo baja incertidumbre del orden del $\pm 3\%$ en la variación del voltaje terminal y posteriormente aumenta el margen de variación del orden de $\pm 9\%$. Esta incertidumbre sugiere un error o daño en el transductor para el voltaje terminal.

Los parámetros de ambos controladores son tomados de la **Tabla 4.5** y **Tabla 4.18**. Para el caso del controlador tipo-2 con funciones de membresía triangulares se obtuvieron mejores resultados sin añadir incertidumbre en los consecuentes por lo que $SL = 0$ y $SR = 0$. Estos resultados son los que se muestran.

4.5.1 Controladores con funciones de membresía Gaussianas e incertidumbre en el voltaje terminal

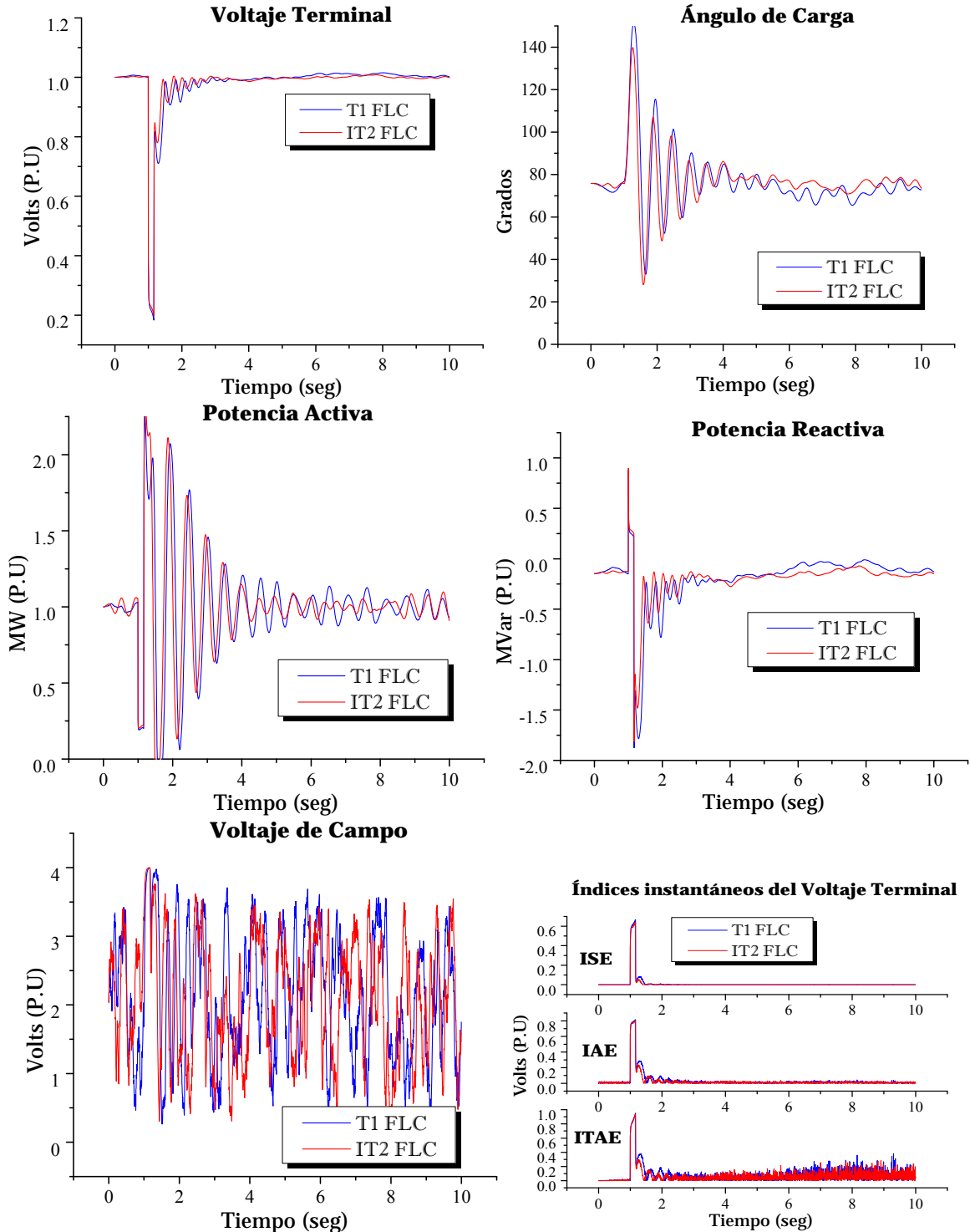
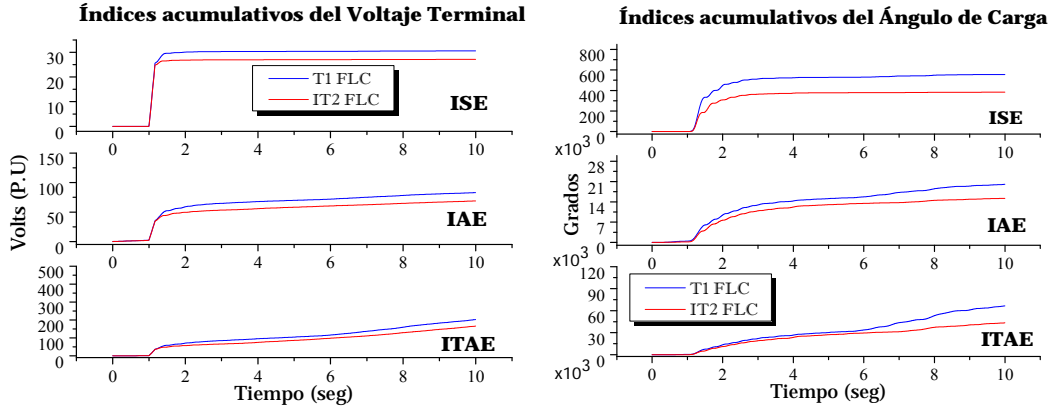


Figura 4.44 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 9 reglas de conocimiento y $\pm 3\%$ de incertidumbre en la lectura del voltaje terminal.



Continuación de Figura 4.44 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 9 reglas de conocimiento y $\pm 3\%$ de incertidumbre en la lectura del voltaje terminal.

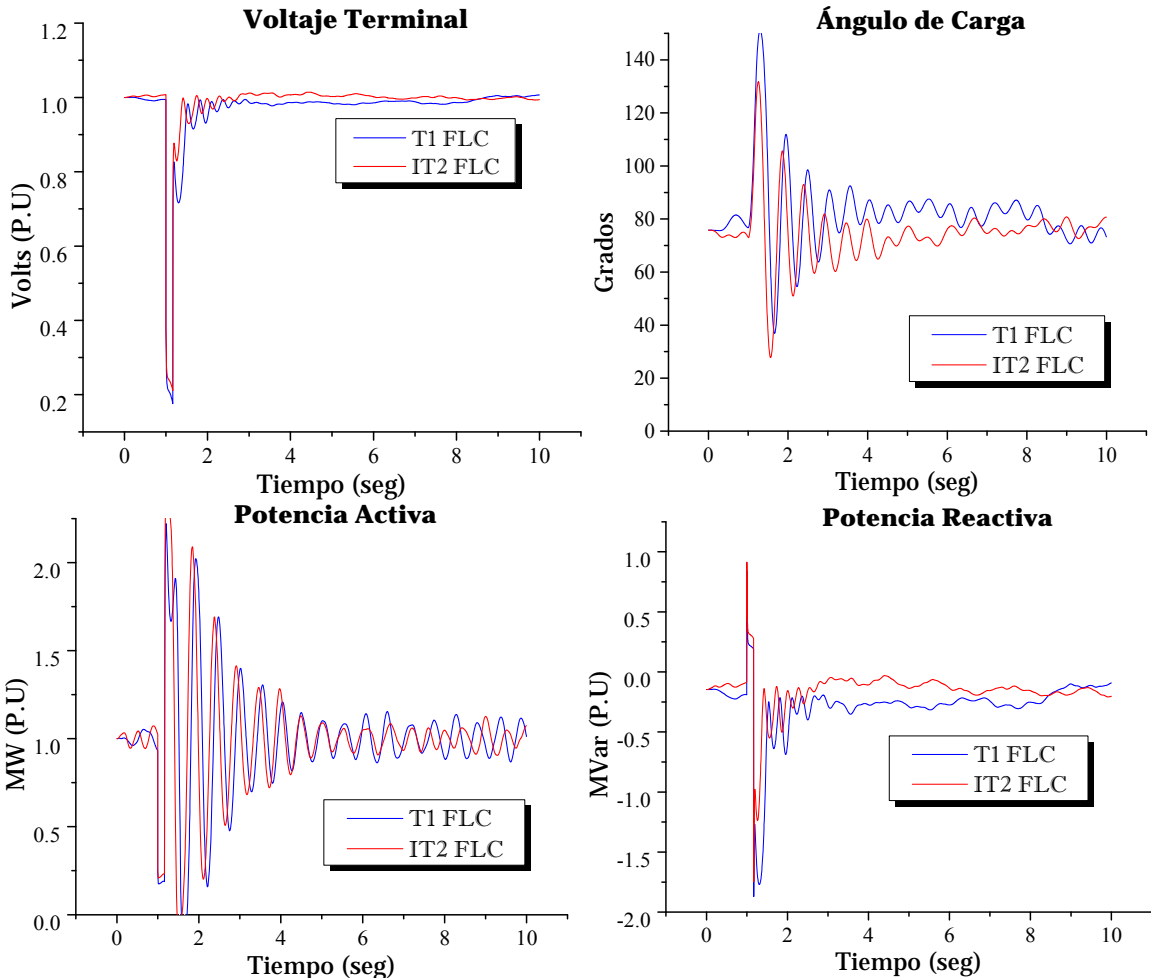
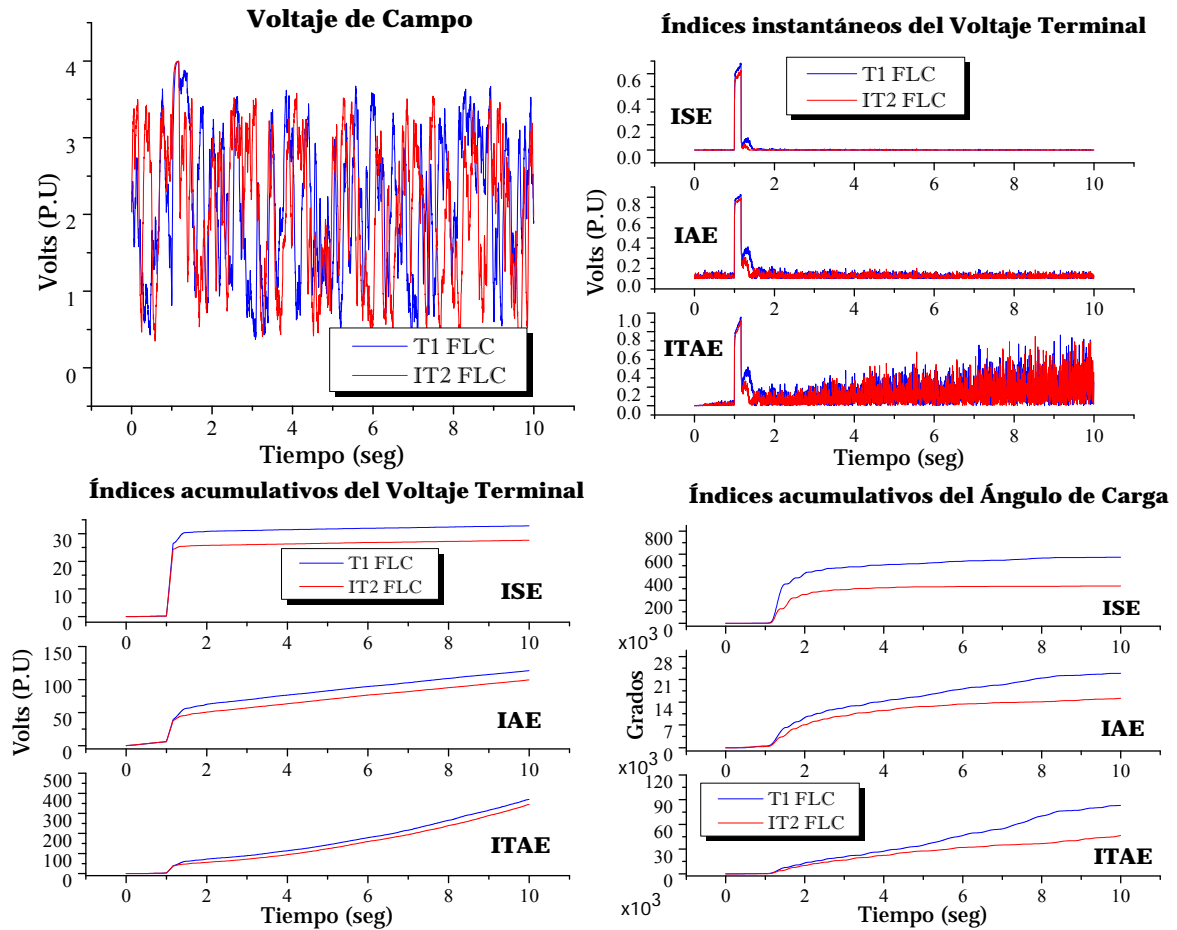


Figura 4.45 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 9 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.



Continuación de Figura 4.45 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 9 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.

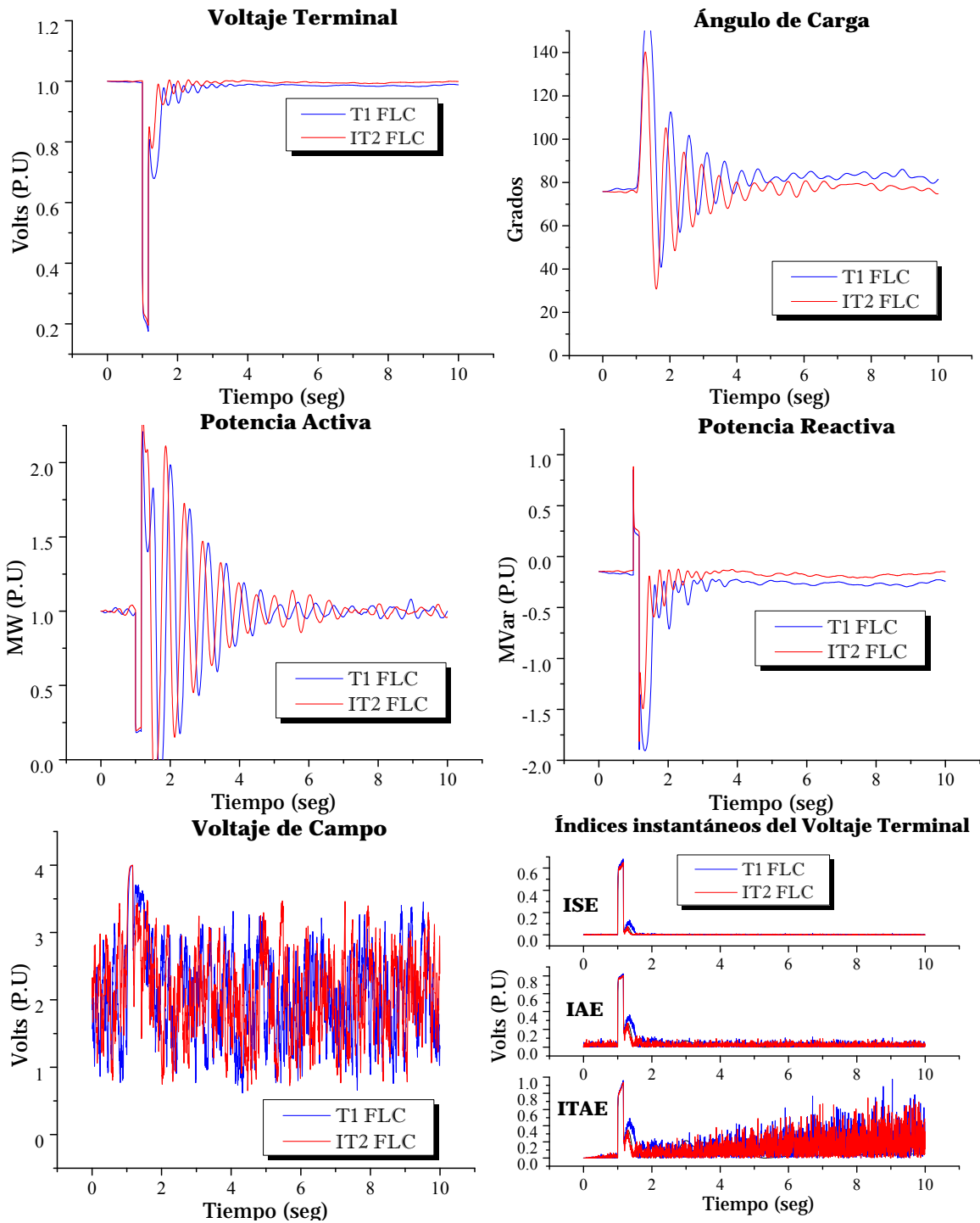
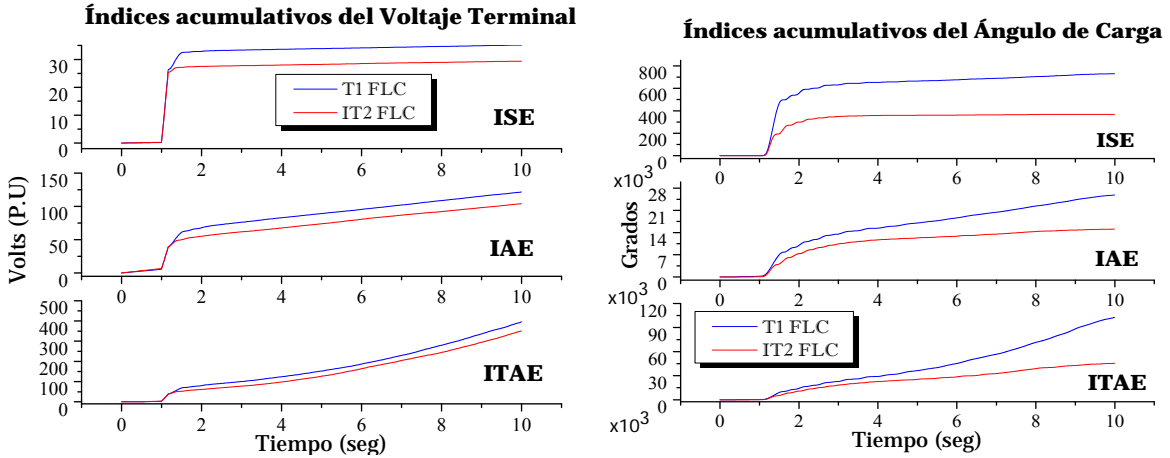


Figura 4.46 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussinas, 25 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.



Continuación de Figura 4.46 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.

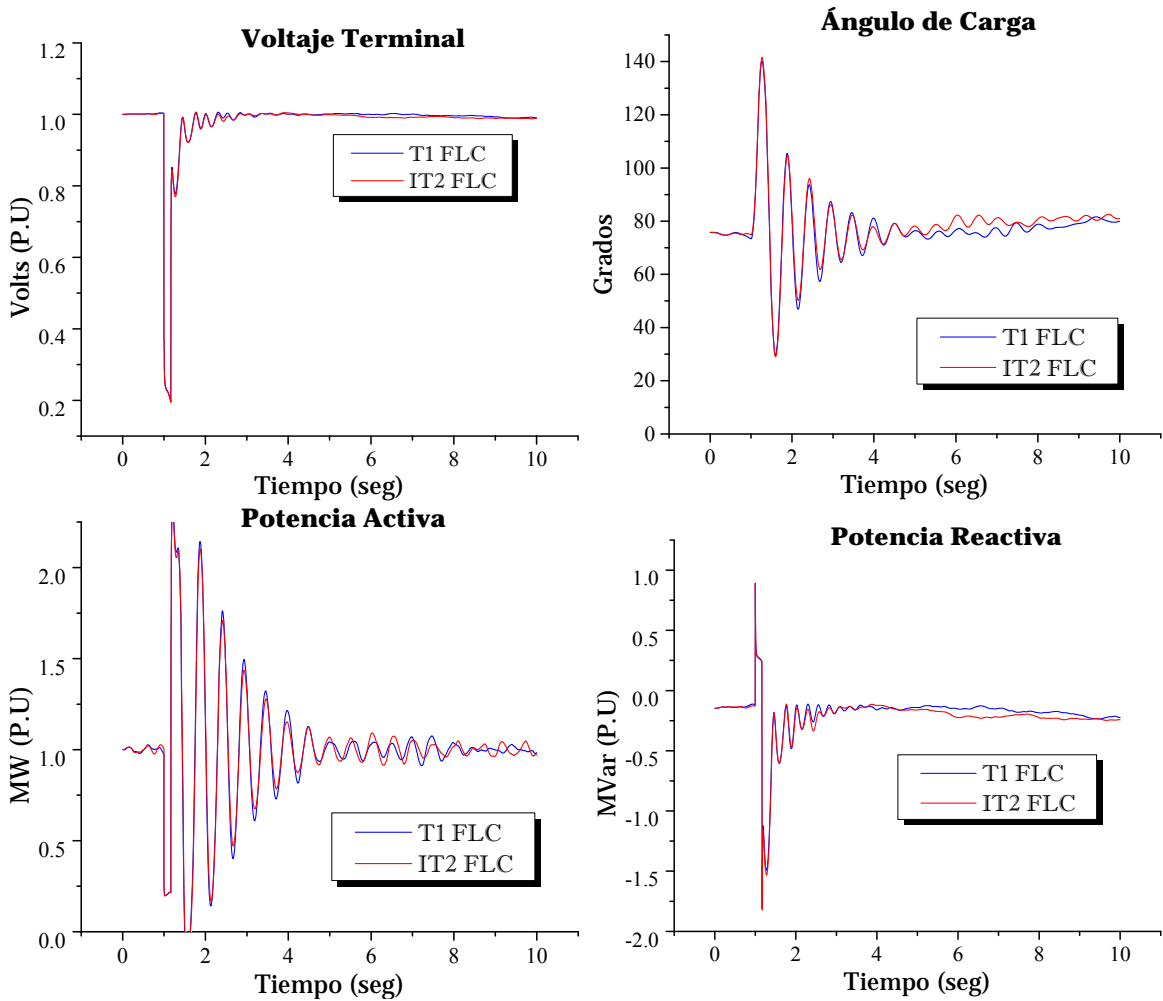
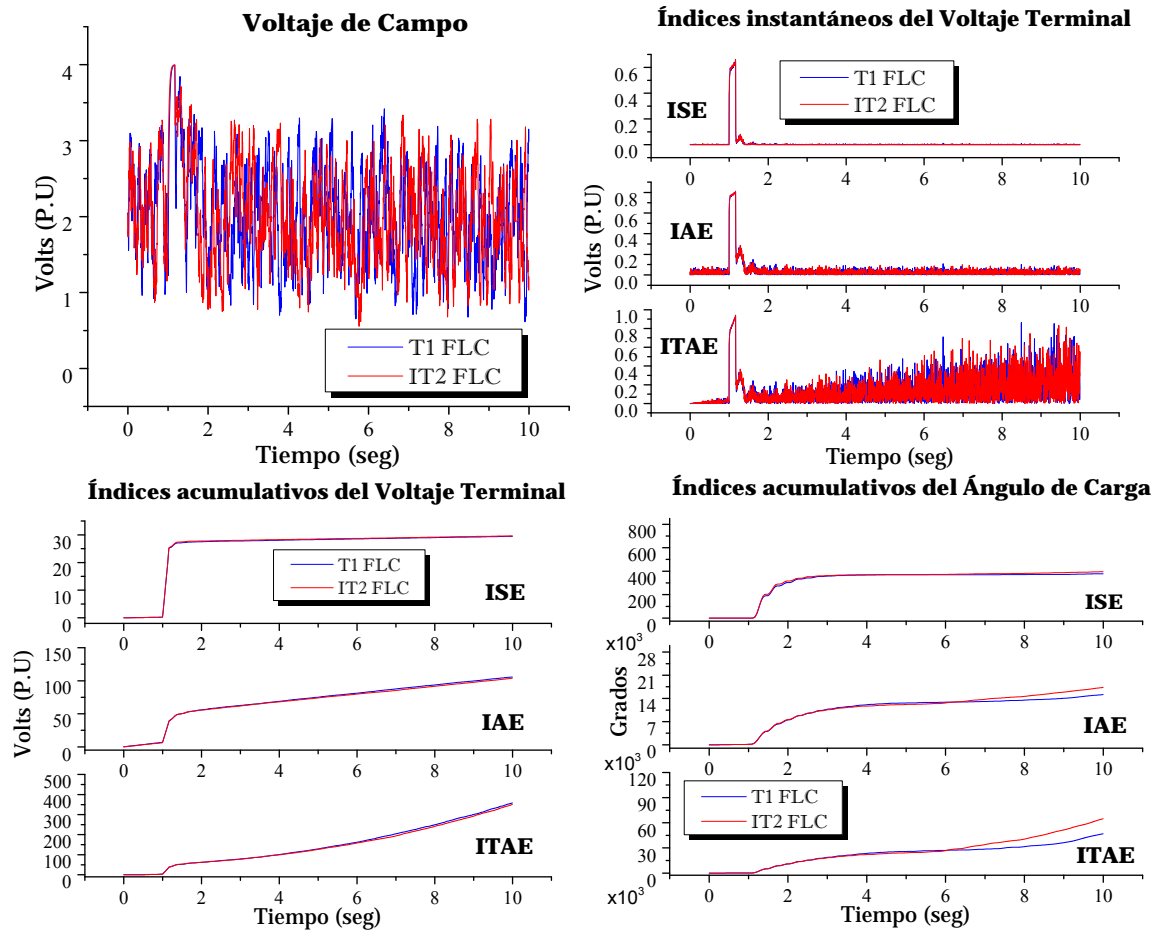


Figura 4.47 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 49 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.



Continuación de Figura 4.47 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas, 49 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.

4.5.2 Controladores con funciones de membresía triangulares e incertidumbre en el voltaje terminal

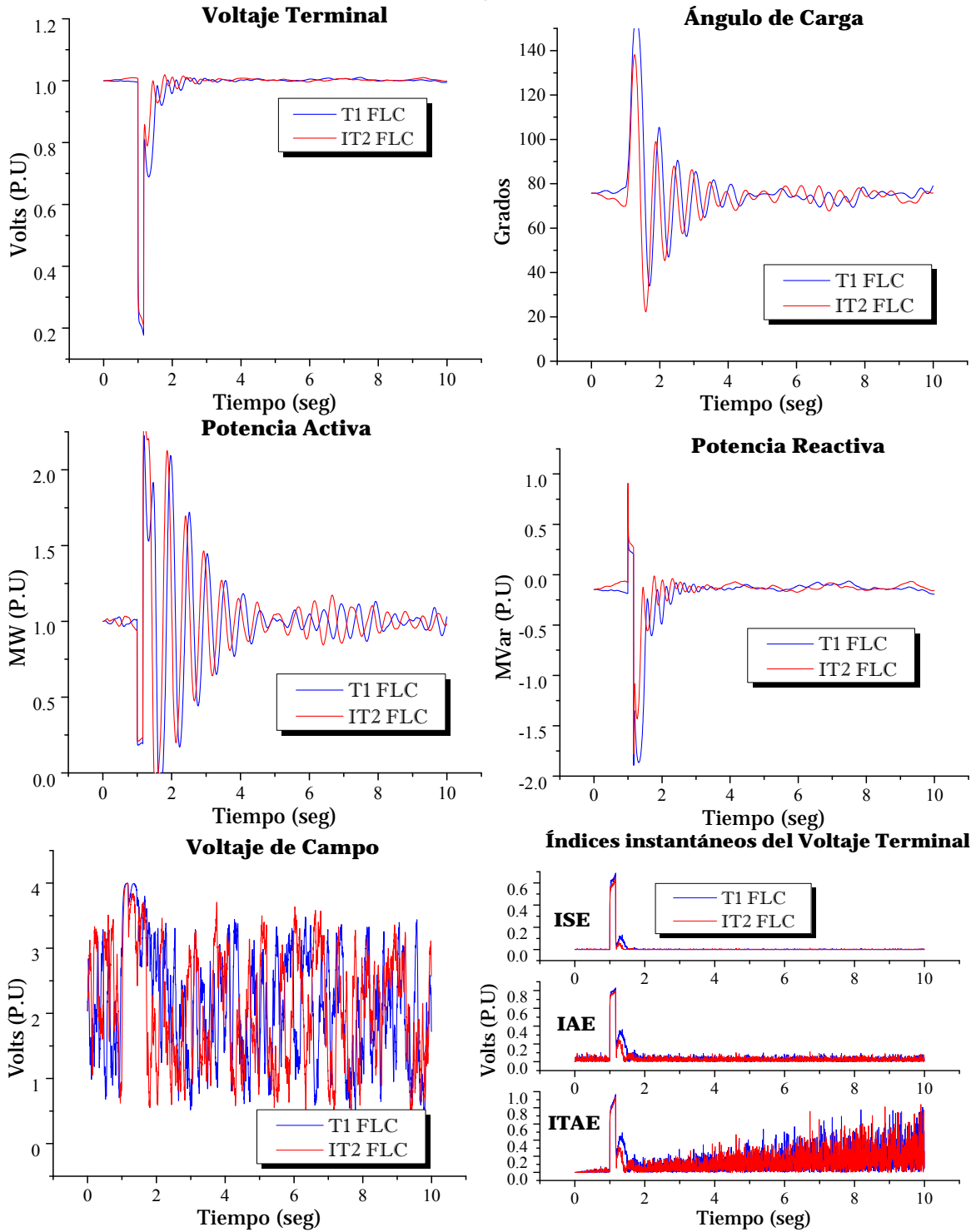
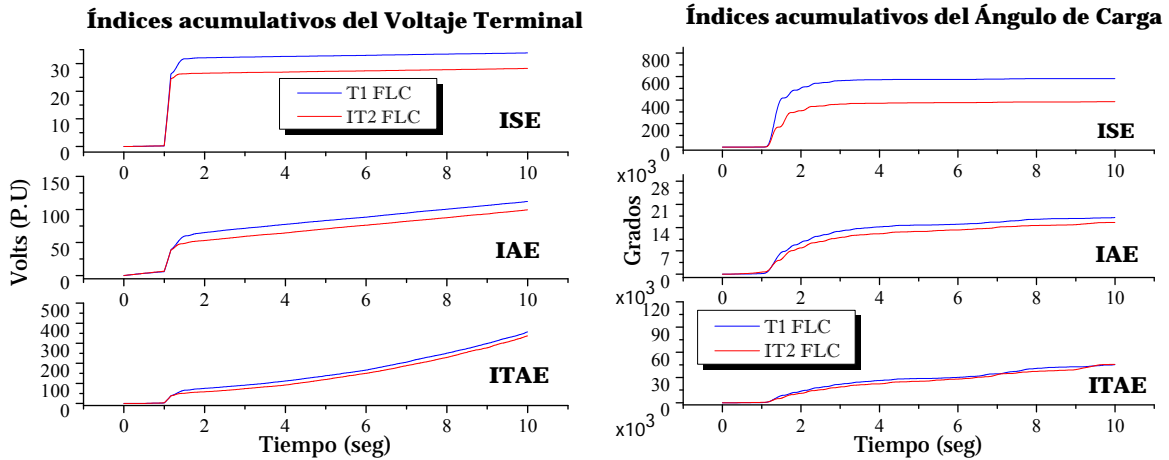


Figura 4.48 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 9 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.



Continuación de Figura 4.48 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 9 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.

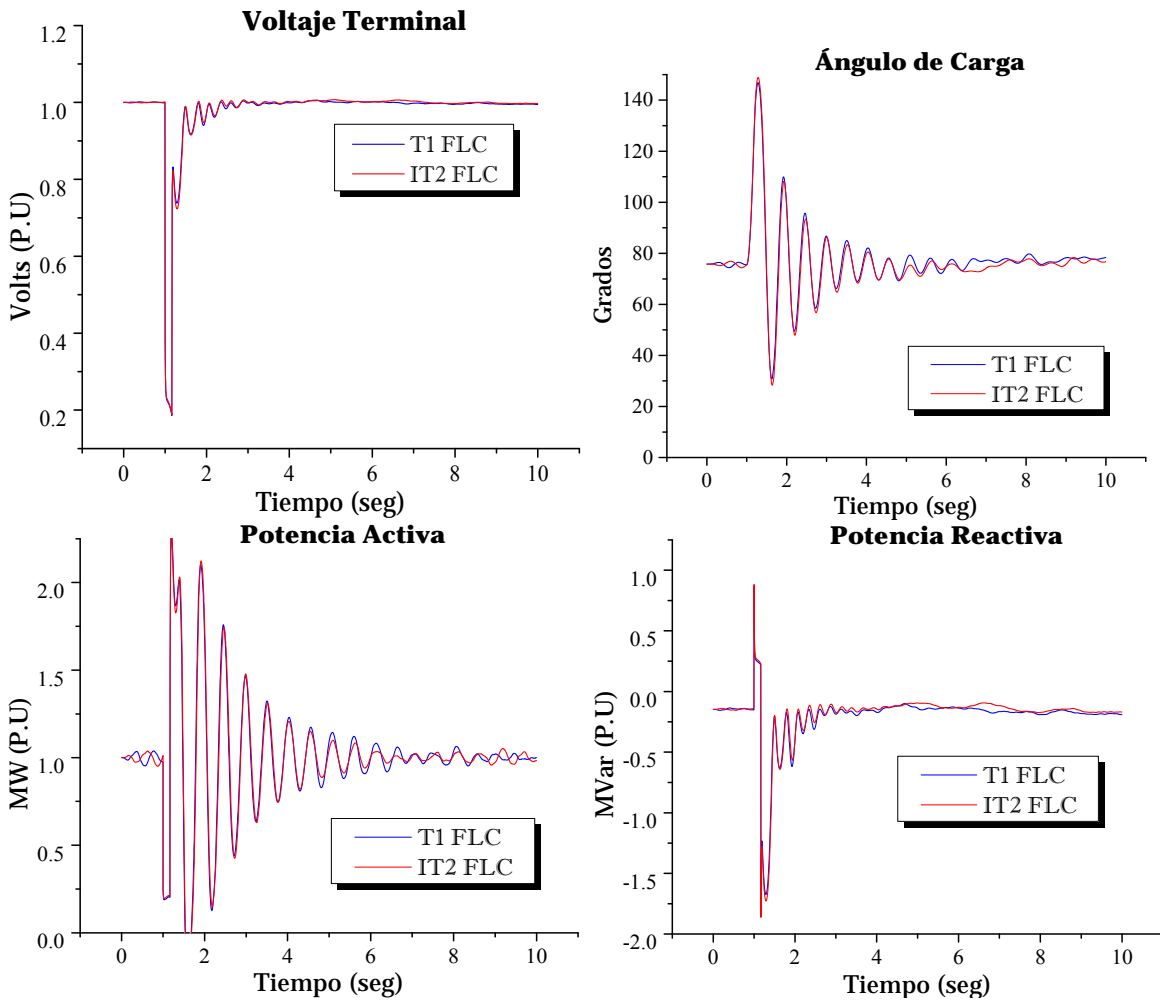
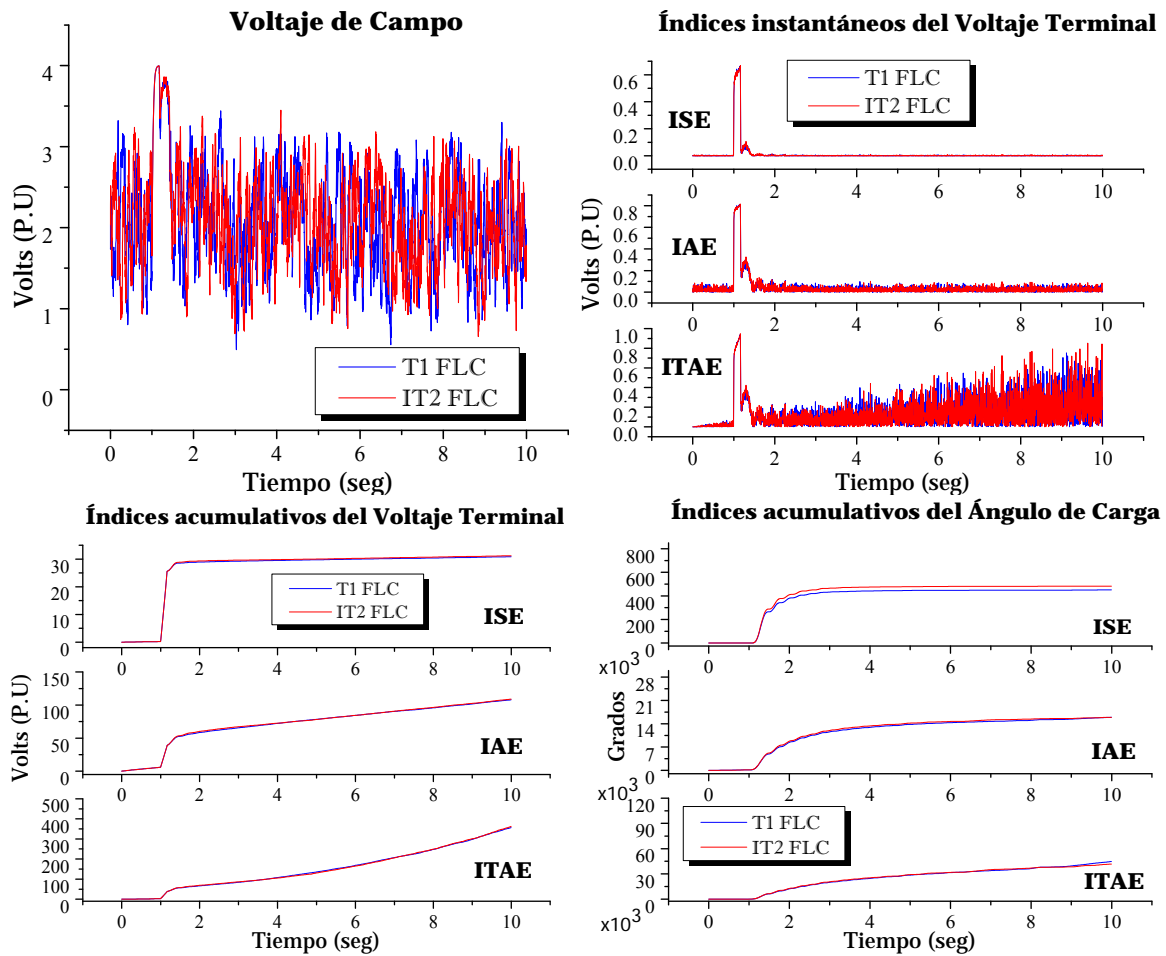


Figura 4.49 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 25 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.



Continuación de Figura 4.49 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 25 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.

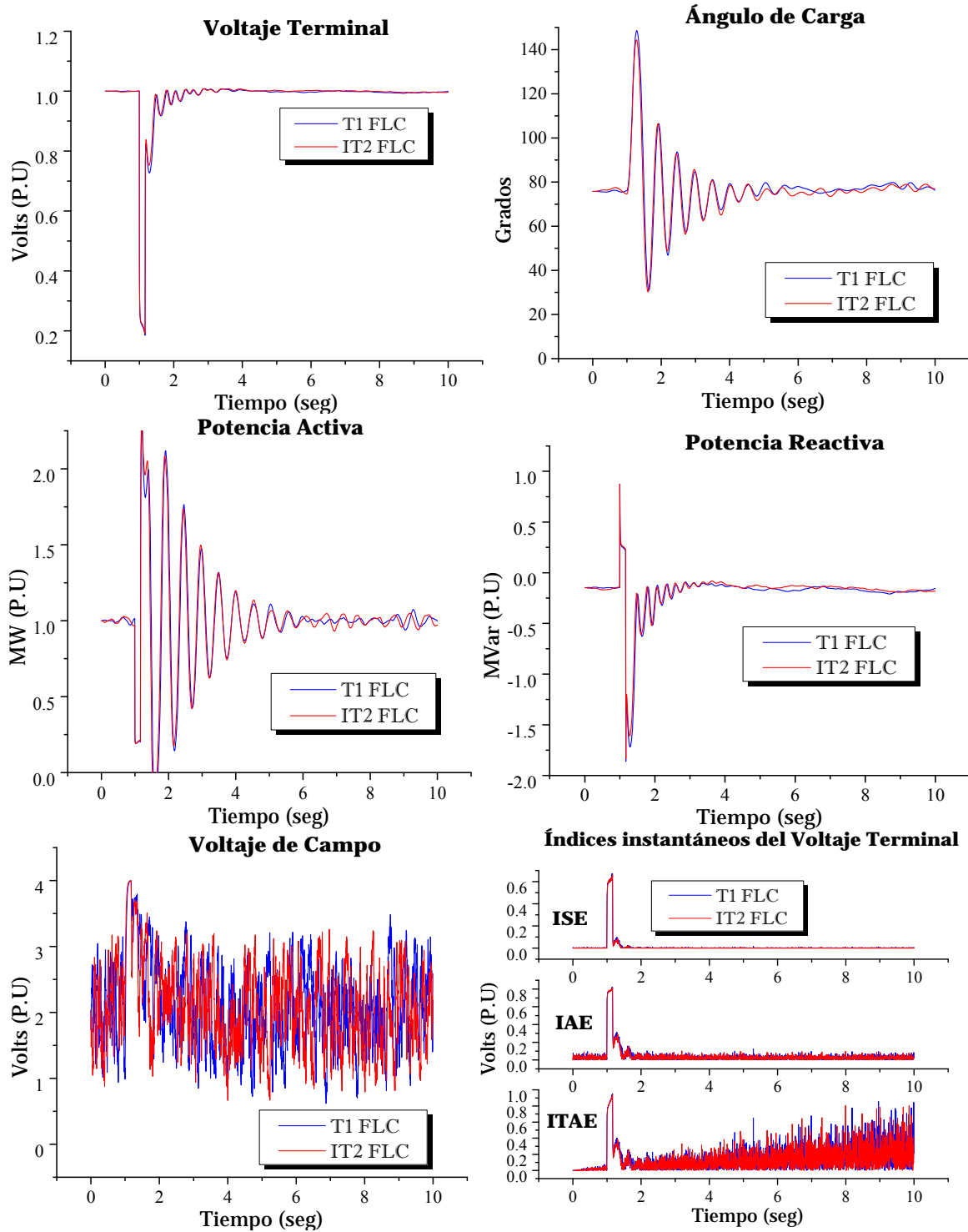
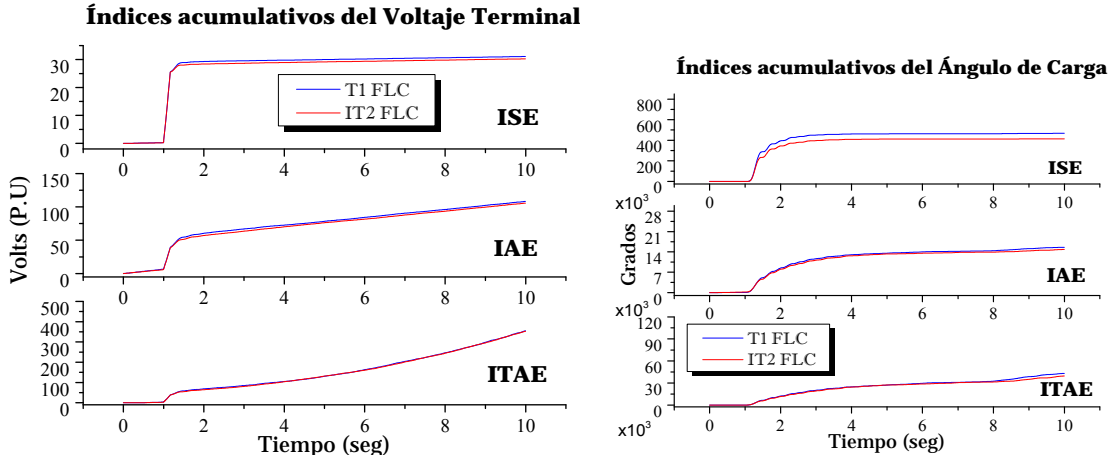


Figura 4.50 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 49 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.



Continuación de Figura 4.50 Respuesta del sistema sujeto a corto circuito con 10 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares, 49 reglas de conocimiento y $\pm 9\%$ de incertidumbre en la lectura del voltaje terminal.

4.5.3 Análisis de resultados

En este caso, el desempeño del controlador tipo-2 con funciones de membresía Gaussianas, es mejor que el controlador tipo-1. Esto se confirma mediante los índices de desempeño del voltaje terminal y ángulo de carga. Pero es más notorio cuando se utiliza una matriz de conocimiento de tan solo 9 reglas difusas, ya que conforme el número de reglas aumenta, el desempeño de ambos controladores es similar. Esto se debe a que el controlador fue sintonizado a 9 reglas de conocimiento y para los casos de 25 y 49 reglas, si bien no se tienen casos inestables se entiende que los parámetros de incertidumbre no son los más adecuados.

En las gráficas de los índices de desempeño instantáneo y acumulativo: la inferioridad en el índice ISE para el control tipo-2, indica que presentará menos sobretiro; mientras que el ITAE indica que se tendrán menores errores en el voltaje terminal a lo largo de la simulación.

Para el caso funciones de membresía triangulares, el desempeño del controlador tipo-2 con 25 y 49 reglas de conocimiento es muy similar al controlador difuso tipo-1.

Ambos casos de funciones de membresía triangulares y Gaussianas tipo-2 con sólo 9 reglas de conocimiento mostraron una aceptable respuesta dinámica del voltaje terminal e índices instantáneos del error en el voltaje terminal.

La constante variación del voltaje de campo para todos los casos en ambos controladores, indica que los controladores difusos siempre se encuentran variando dicho voltaje buscando ajustar al voltaje de referencia y que el error lo más aproximado a cero.

4.6 Pruebas para una línea de transmisión larga, sujeta a un disturbio grande

En esta prueba se conecta el generador a una línea de transmisión larga. El generador se enciende con carga y sujeto a un corto circuito en el bus que ocurre a 1s de tiempo de simulación, con duración de 2 y 4 ciclos. Las condiciones iniciales para este caso fueron:

$$V_{ter} = 1 P.U. \quad \theta = 52.3997^\circ \quad P_A = 0.75 P.U. \quad P_R = -0.2726 P.U.$$

Los parámetros de ambos controladores son tomados de la **Tabla 4.22** y **Tabla 4.35**.

4.6.1 Controladores con funciones de membresía triangulares y 9 reglas de conocimiento

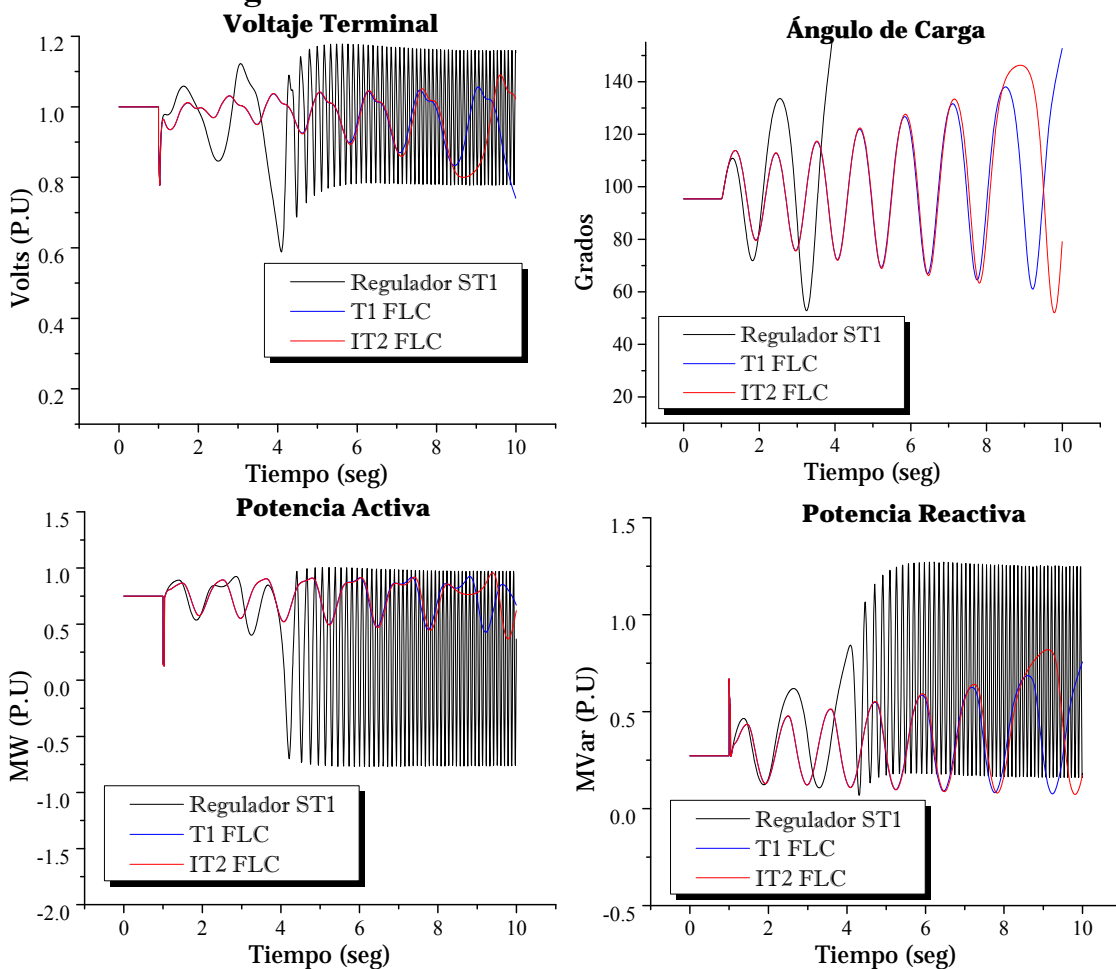
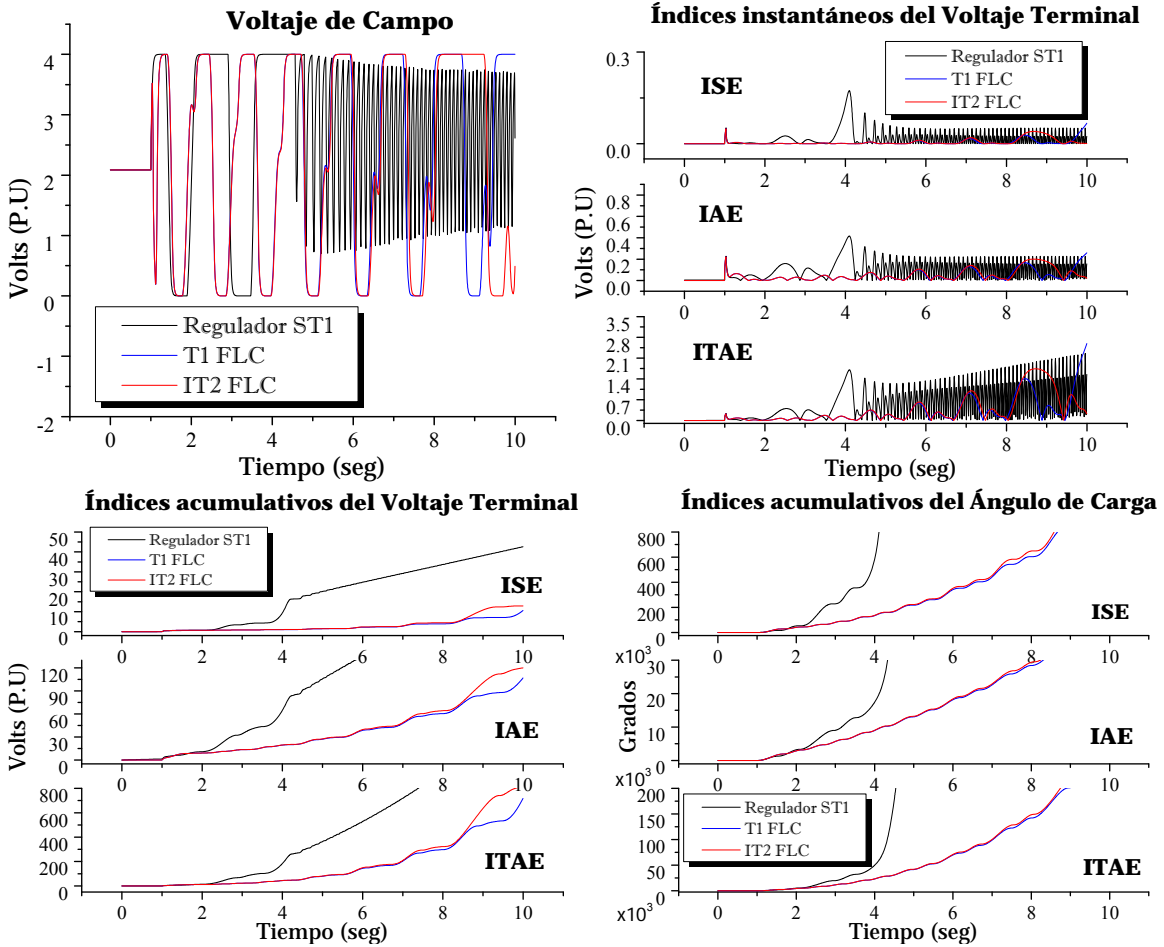


Figura 4.51 Respuesta del sistema sujeta a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.



Continuación de Figura 4.51 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.

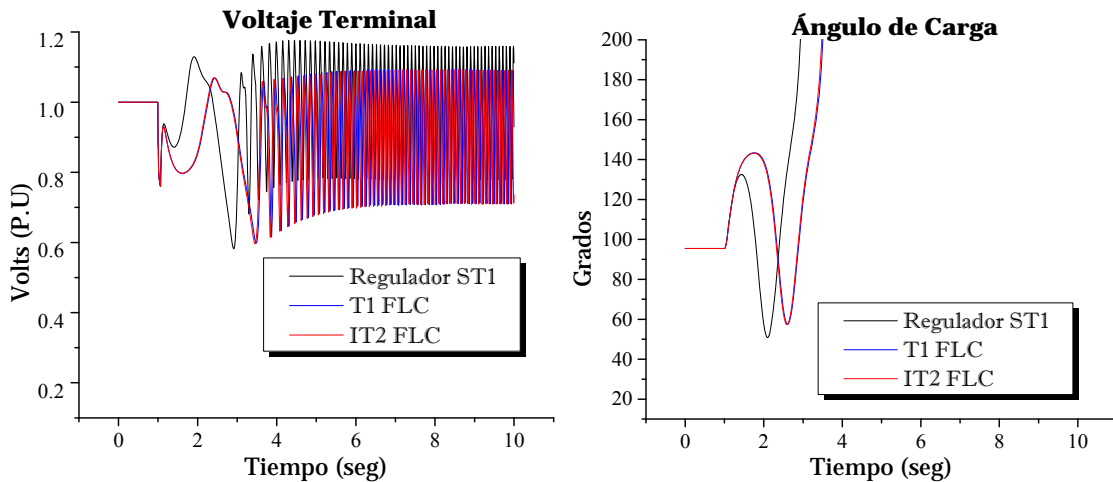
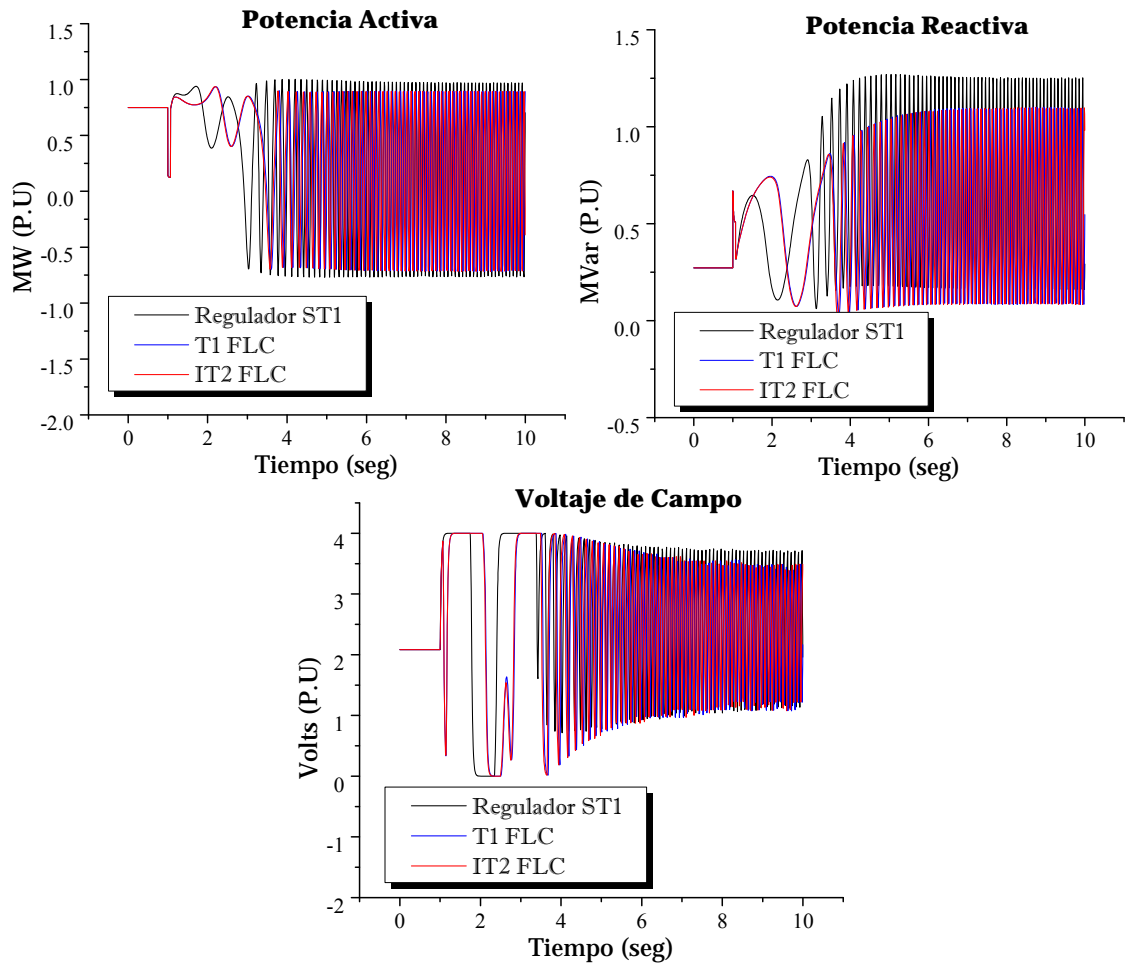


Figura 4.52 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.



Continuación de Figura 4.52 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.

4.6.2 Controladores con funciones de mmebresía triangulares y 25 reglas de conocimiento

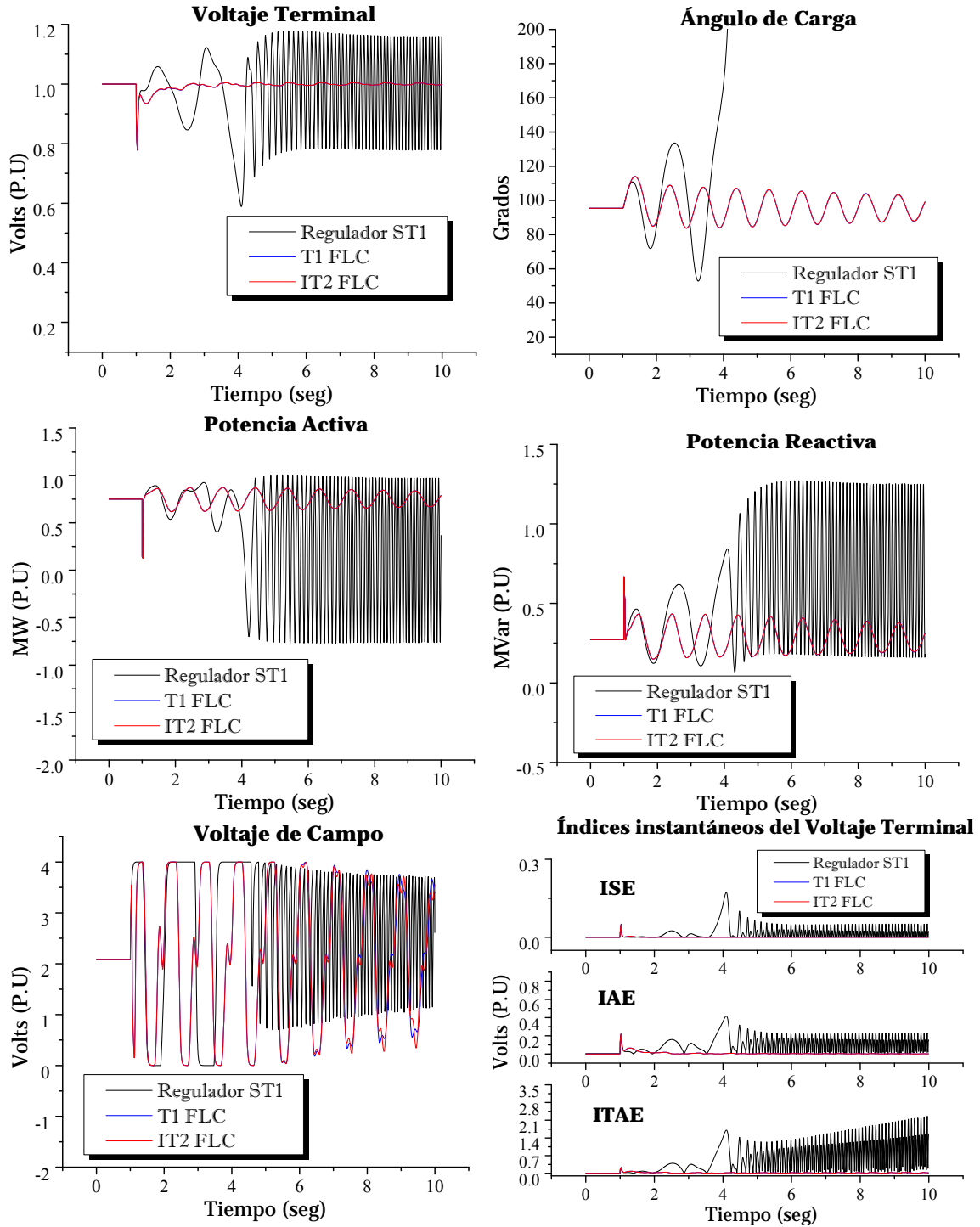
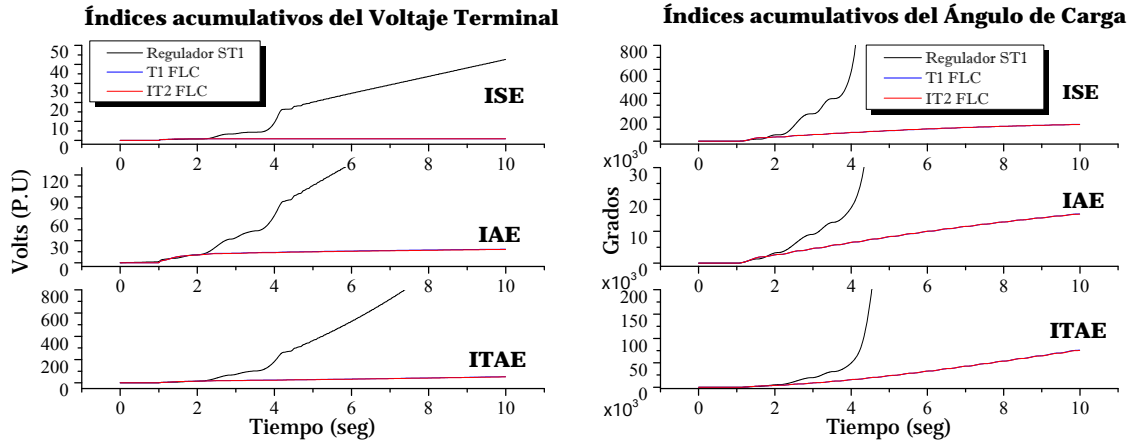


Figura 4.53 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.



Continuación de Figura 4.53 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.

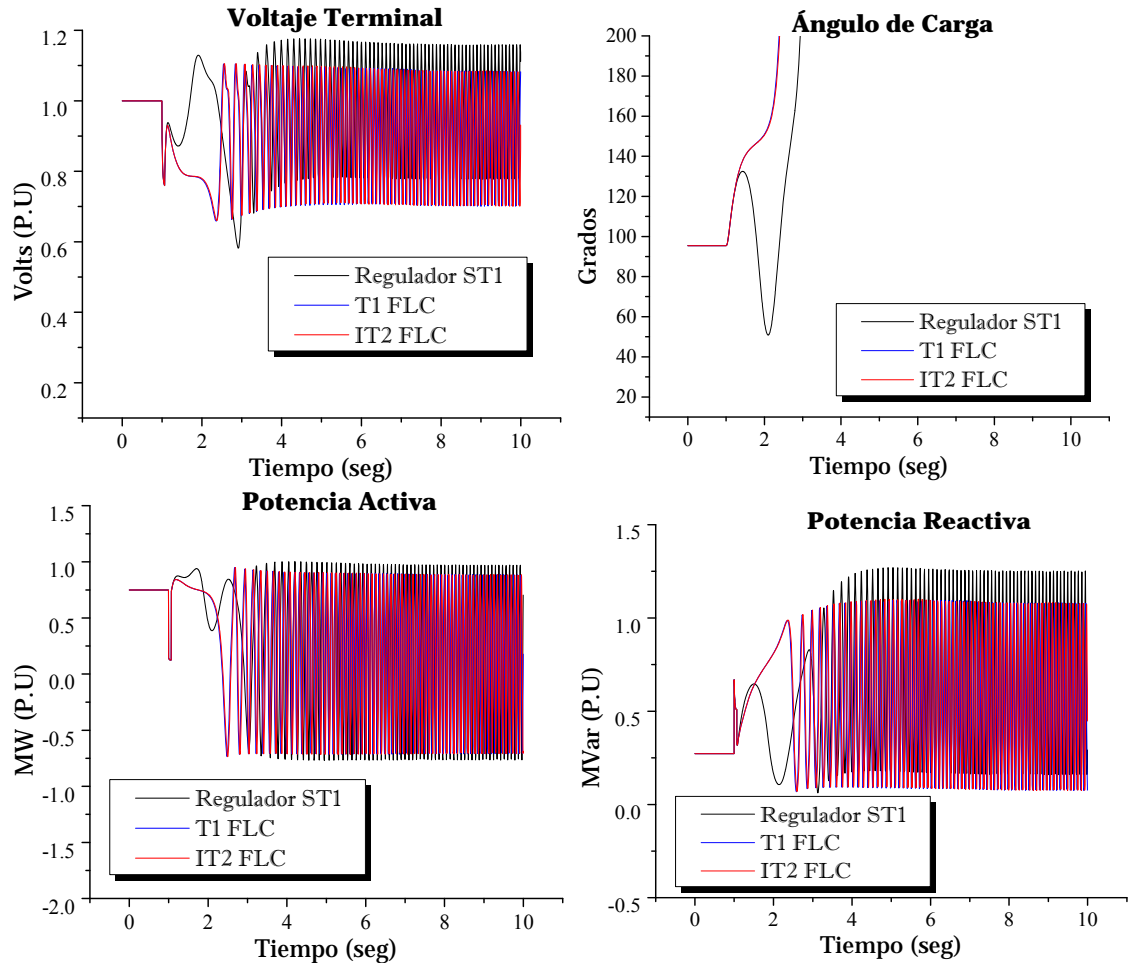


Figura 4.54 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 25 reglas de conocimiento.

4.6.3 Controladores con funciones de membresía triangulares y 49 reglas de conocimiento

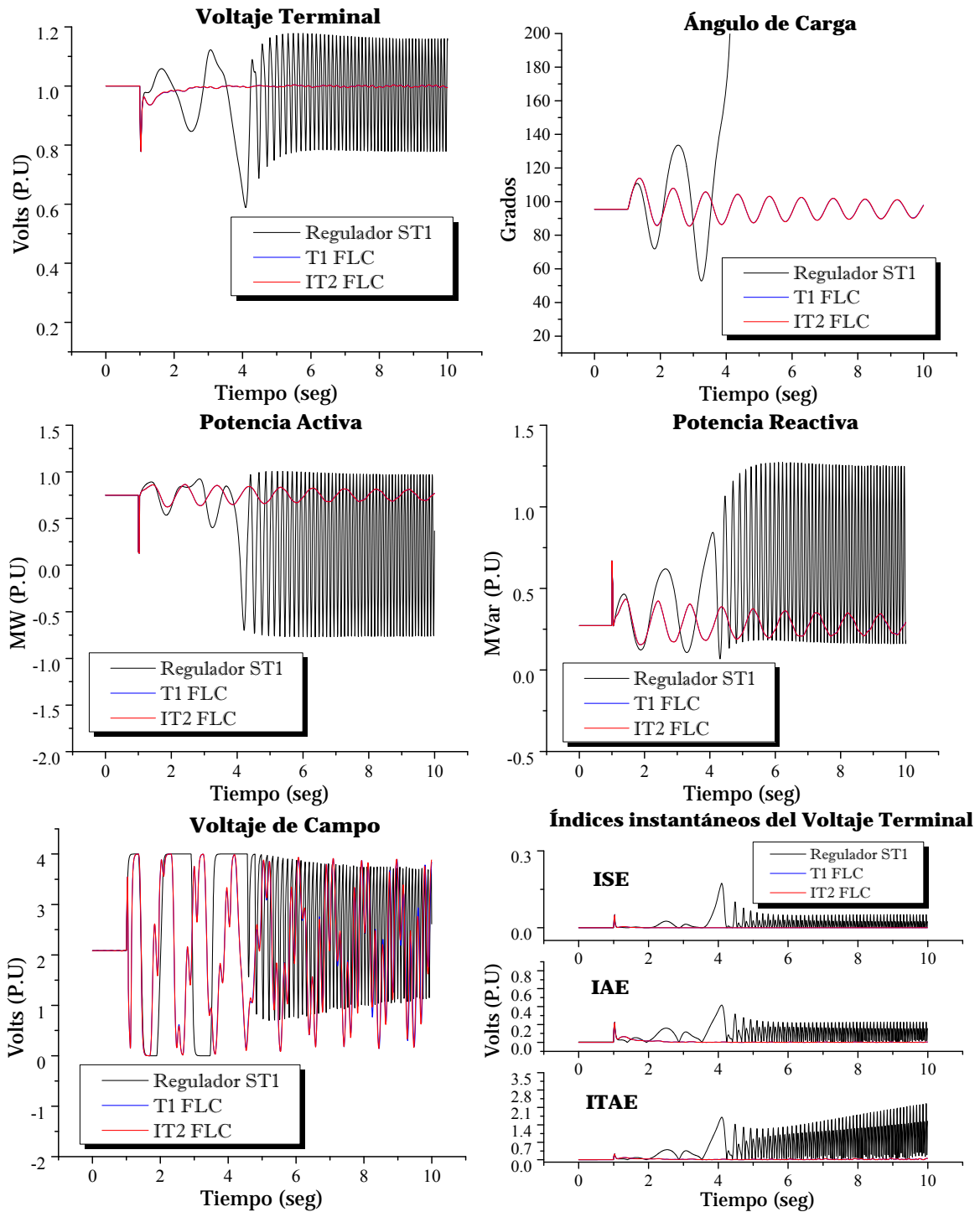
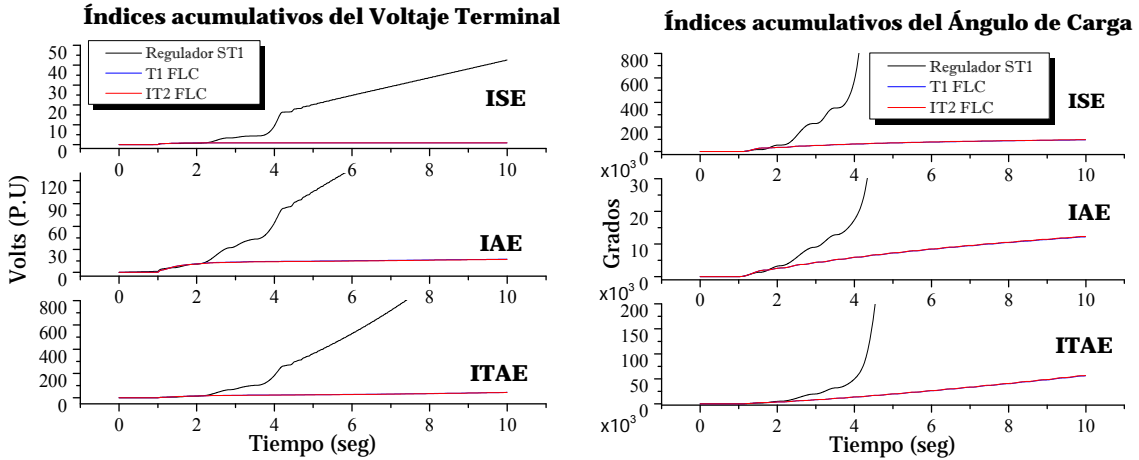


Figura 4.55 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.



Continuación de Figura 4.55 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.

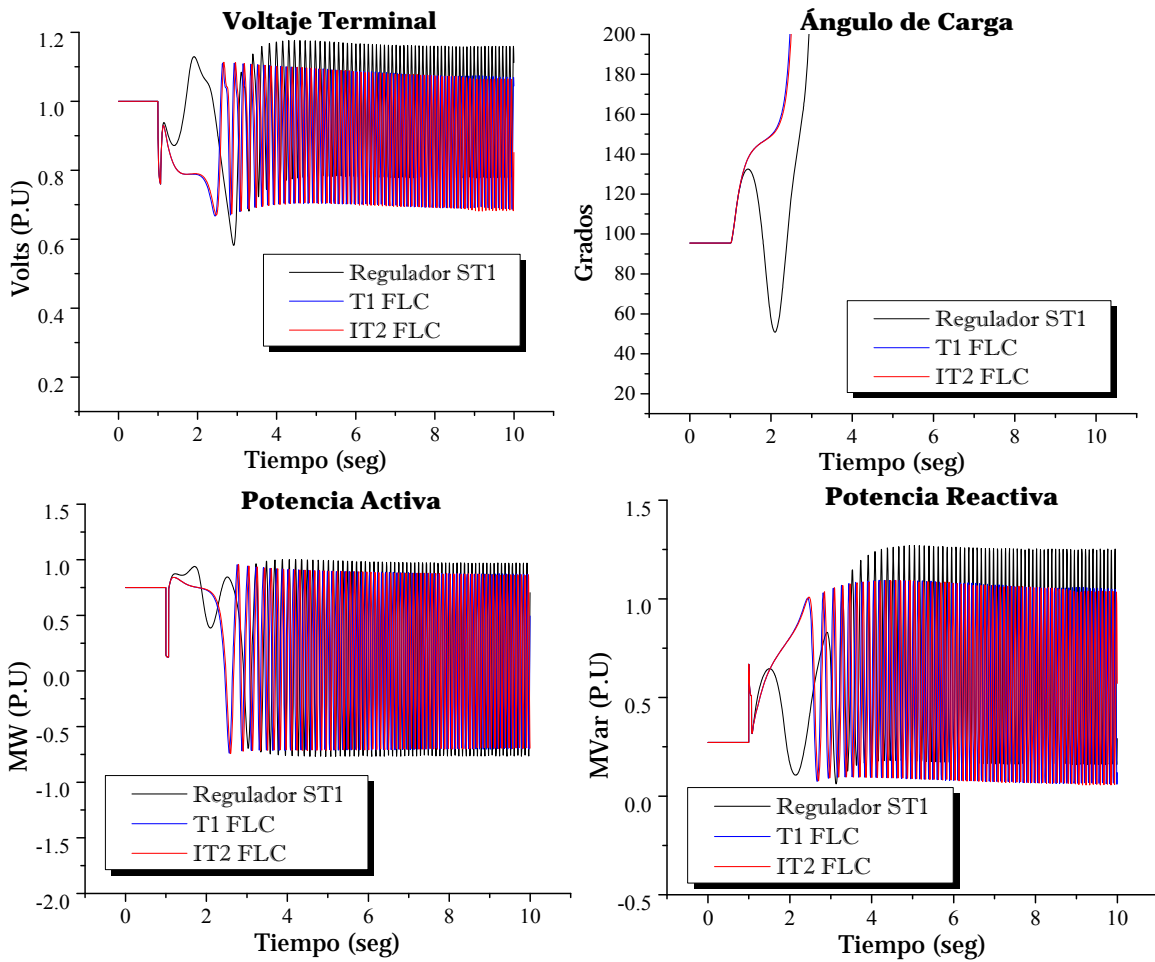


Figura 4.56 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía triangulares y 49 reglas de conocimiento.

4.6.4 Controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento

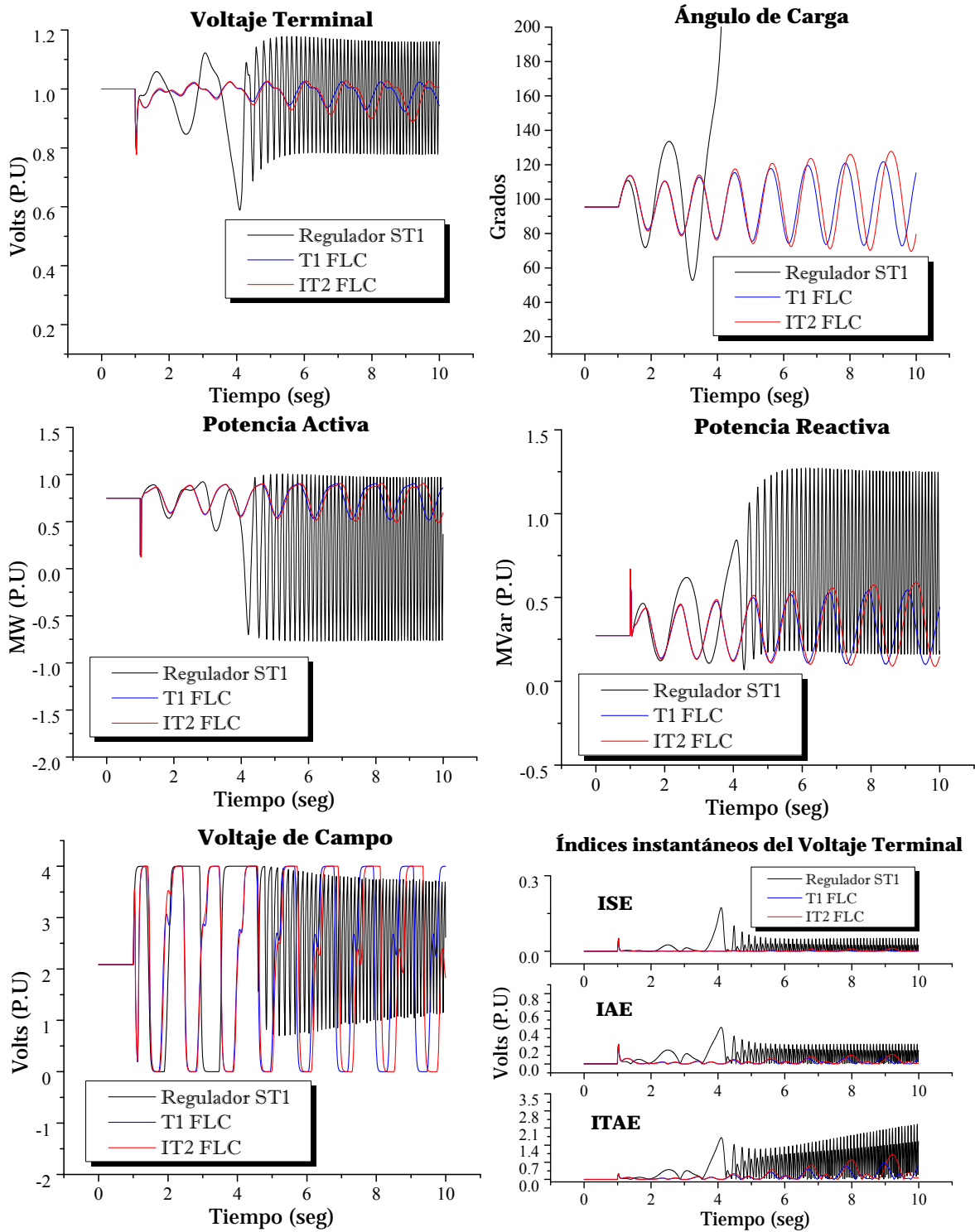
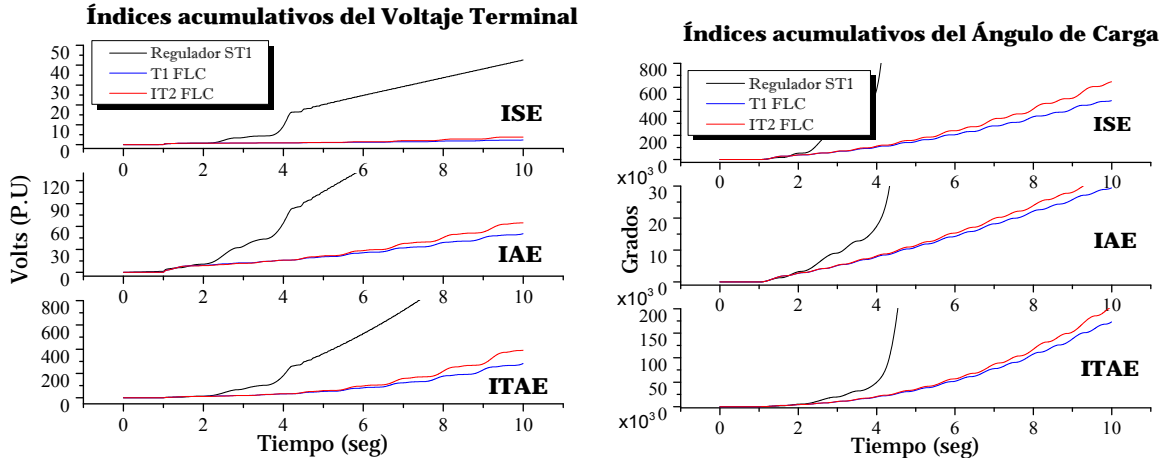


Figura 4.57 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.



Continuación de Figura 4.57 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.

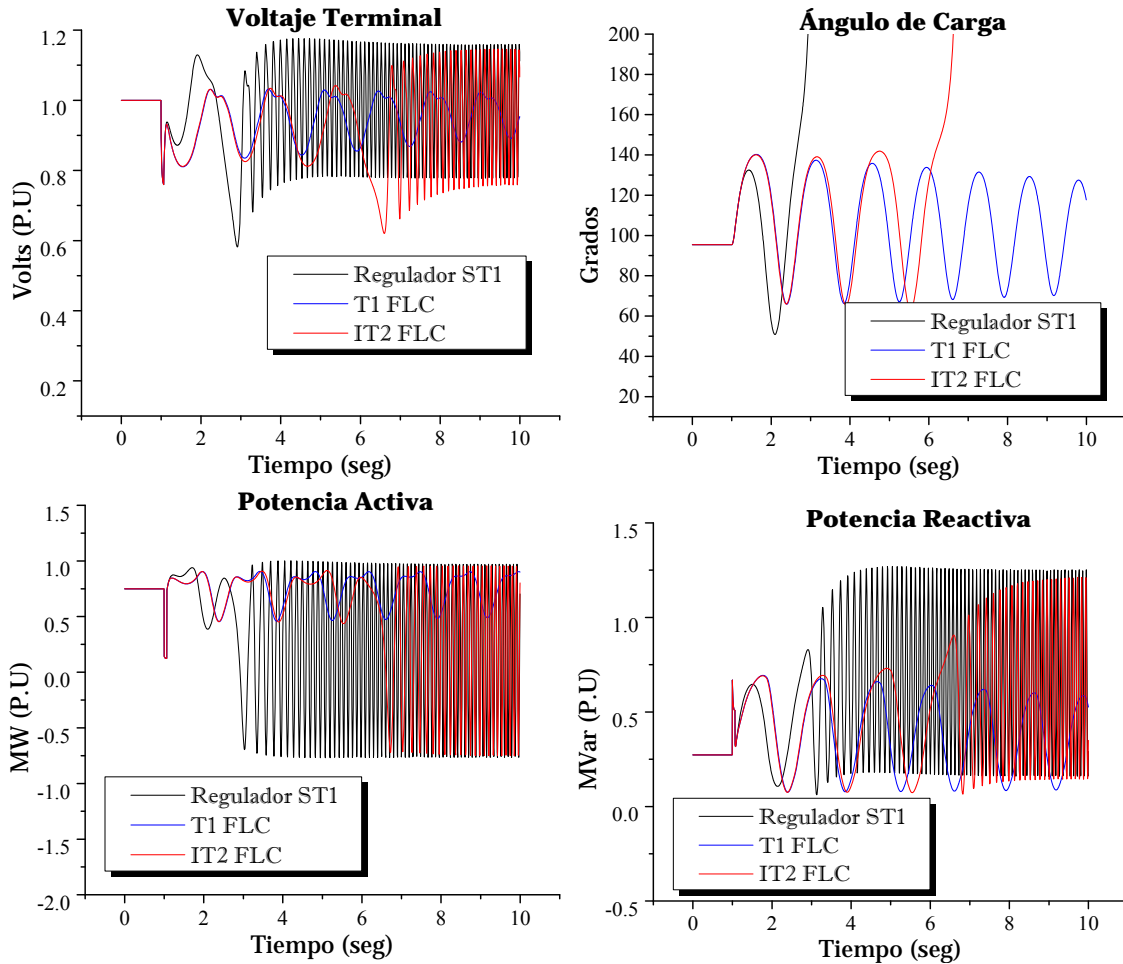


Figura 4.58 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.

4.6.5 Controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento

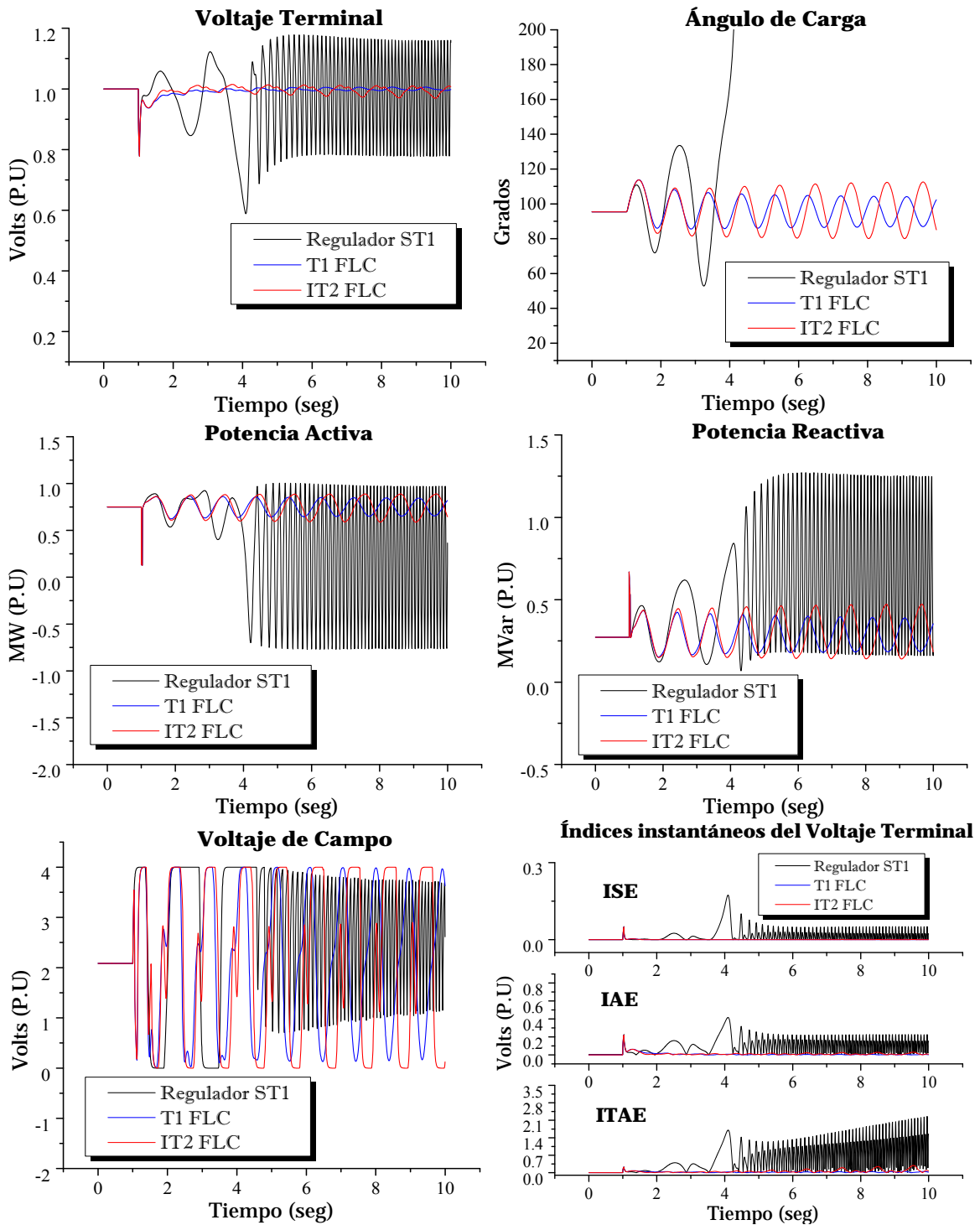
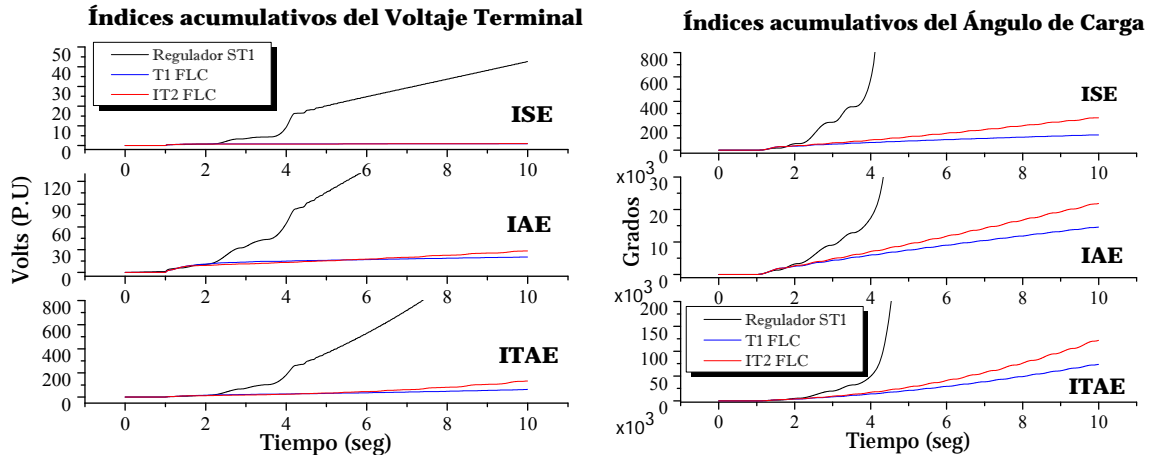


Figura 4.59 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.



Continuación de Figura 4.59 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.

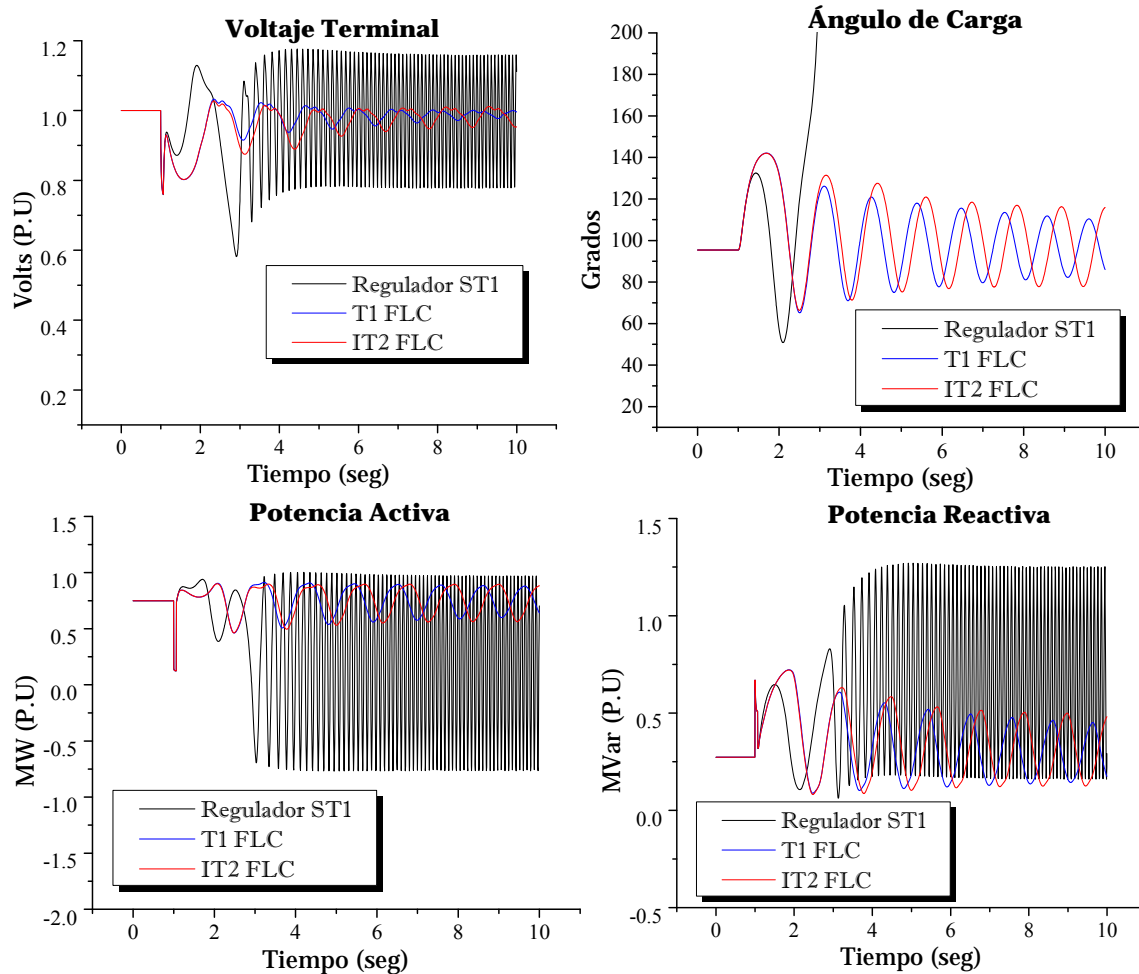


Figura 4.60 Respuesta del sistema sujeto a corto circuito con 4 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 25 reglas de conocimiento.

4.6.6 Controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento

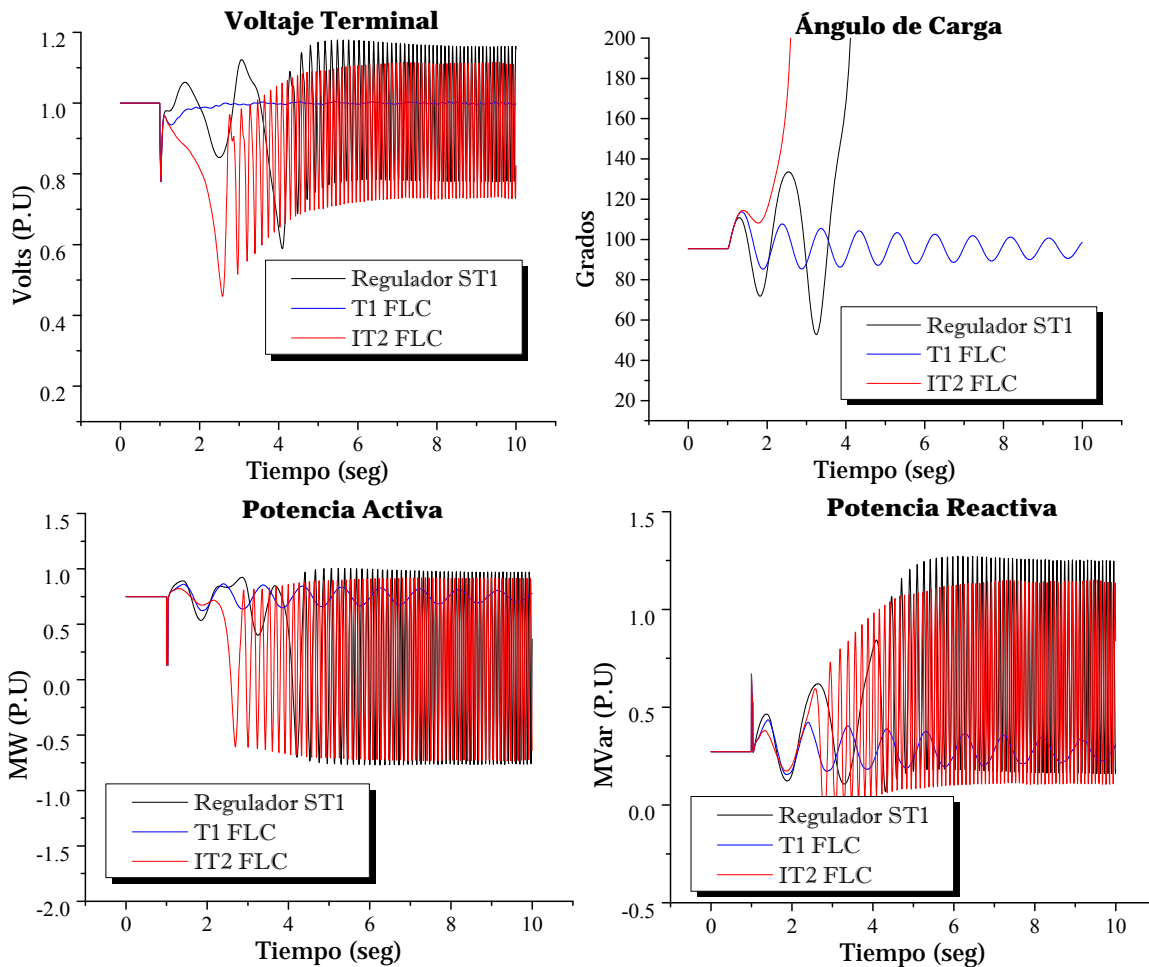


Figura 4.61 Respuesta del sistema sujeto a corto circuito con 2 ciclos de duración; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.

4.6.7 Análisis de resultados

En este caso utilizando una línea de transmisión larga, el tiempo crítico de liberación de falla disminuye considerablemente al orden de 2 ciclos. Esto se debe a que dada la condición inicial, el generador síncrono se encuentra operando en sus límites de estabilidad, con el ángulo de carga muy cercano a su límite teórico (90°).

Para el caso funciones de membresía triangulares, el desempeño de ambos controladores es muy similar. Cuando se utilizan funciones de membresía Gaussianas, el control tipo-2 en ciertas pruebas se ve superado, como en el caso de falla del orden de 4 ciclos con 9 reglas de conocimiento en donde el control se vuelve inestable.

En la mayoría de los casos no se muestra consistencia, ya que un IT2 FLC con funciones de membresía triangulares con 25 reglas muestra mantener estable el voltaje terminal, pero un IT2 FLC con funciones de membresía Gaussianas y 49 reglas de conocimiento muestra ser

inestable. Sin embargo el controlador tipo-2 muestra superioridad sobre el St1 en la gran mayoría de los casos.

4.7 Para una línea de transmisión corta sujeta un pequeño disturbio en el voltaje de referencia de 1 a 1.05 P.U.

Para este caso de pequeño disturbio, se realiza un pequeño cambio en el voltaje de referencia del 5%, y se comparan las respuestas del controlador tipo-1 y tipo-2. La condición inicial del generador síncrono es:

$$V_{ter} = 1 P.U. \quad \theta = 6.8077^\circ \quad P_A = 1 P.U. \quad P_R = -0.1474 P.U.$$

En seguida se presentan dos casos de simulación. Los parámetros de ambos controladores son tomados de la **Tabla 4.5** y **Tabla 4.18**. Para el caso del controlador tipo-2 con funciones de membresía triangulares se obtuvieron mejores resultados sin añadir incertidumbre en los consecuentes por lo que $SL = 0$ y $SR = 0$.

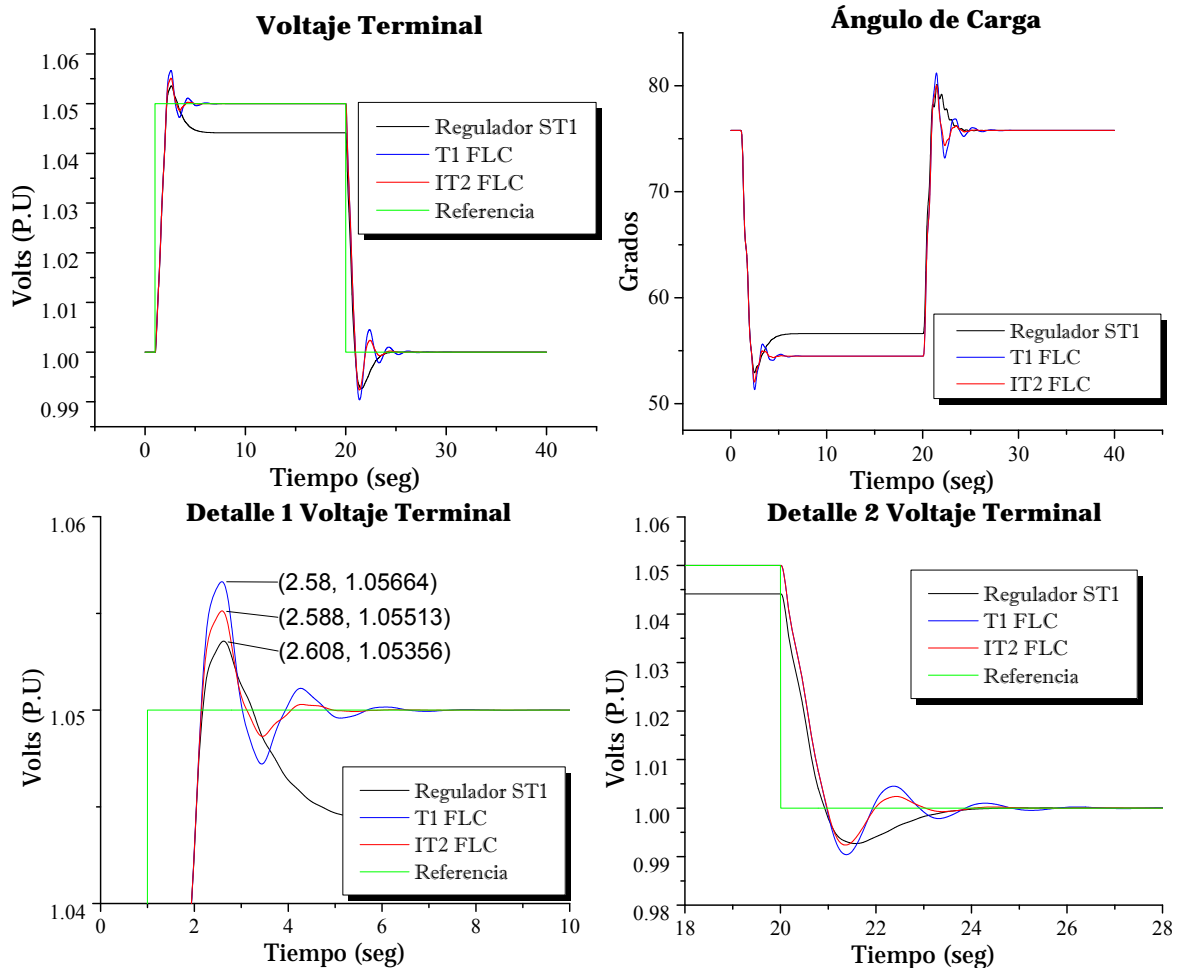
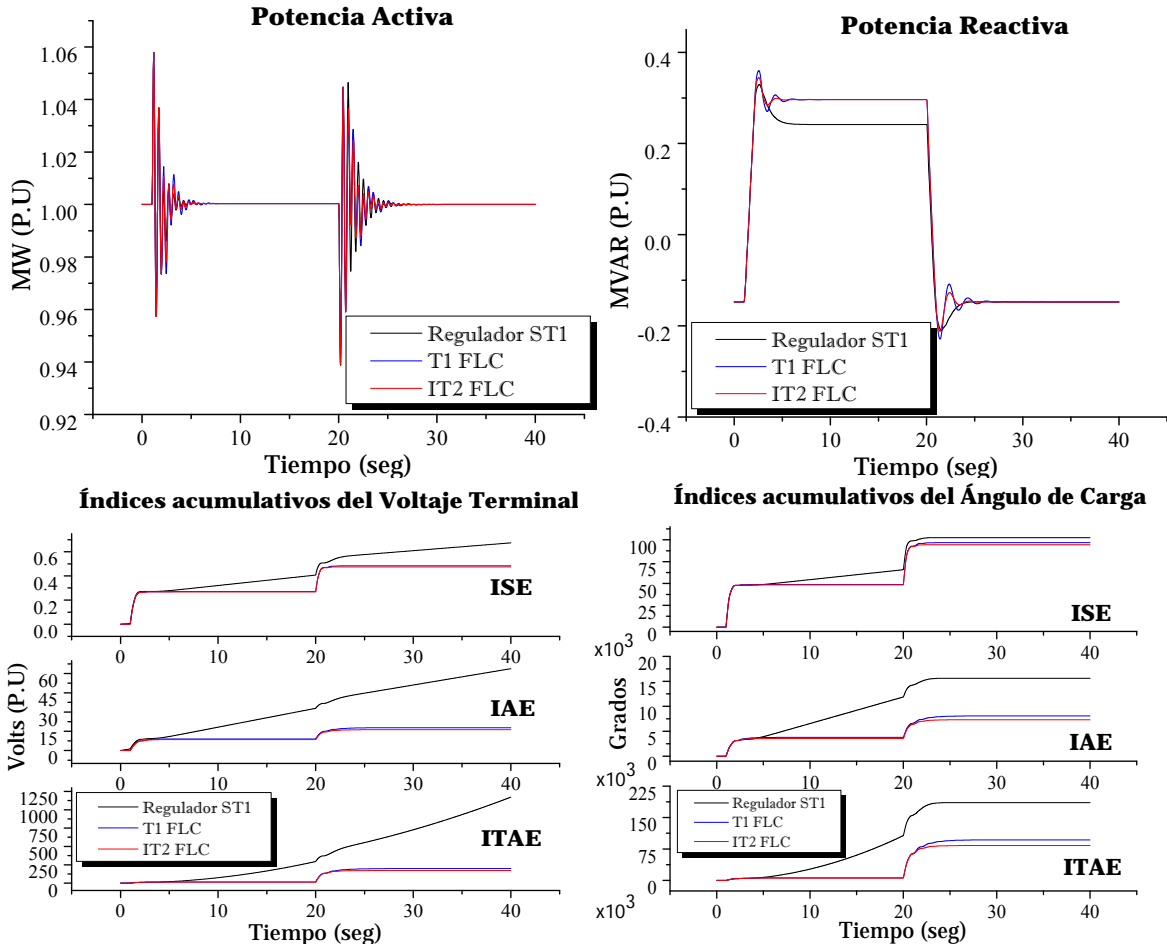


Figura 4.62 Respuesta del sistema sujeta a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.



Coninuación de Figura 4.62 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.

Tabla 4.36 Índices de comportamiento dinámico de detalle 1 voltaje terminal Figura 4.62.		
Índice	Controlador Tipo-1	Controlador Tipo-2
Sobretiro	13.28%	10.26%
Tiempo de respuesta	0.884 s	0.884 s
Tiempo de estabilización	3.604 s	2.836 s
Tiempo en alcanzar el valor pico	1.58 s	1.588 s

4.7.1 Controladores con funciones de Membresía triangulares y 9 reglas de conocimiento

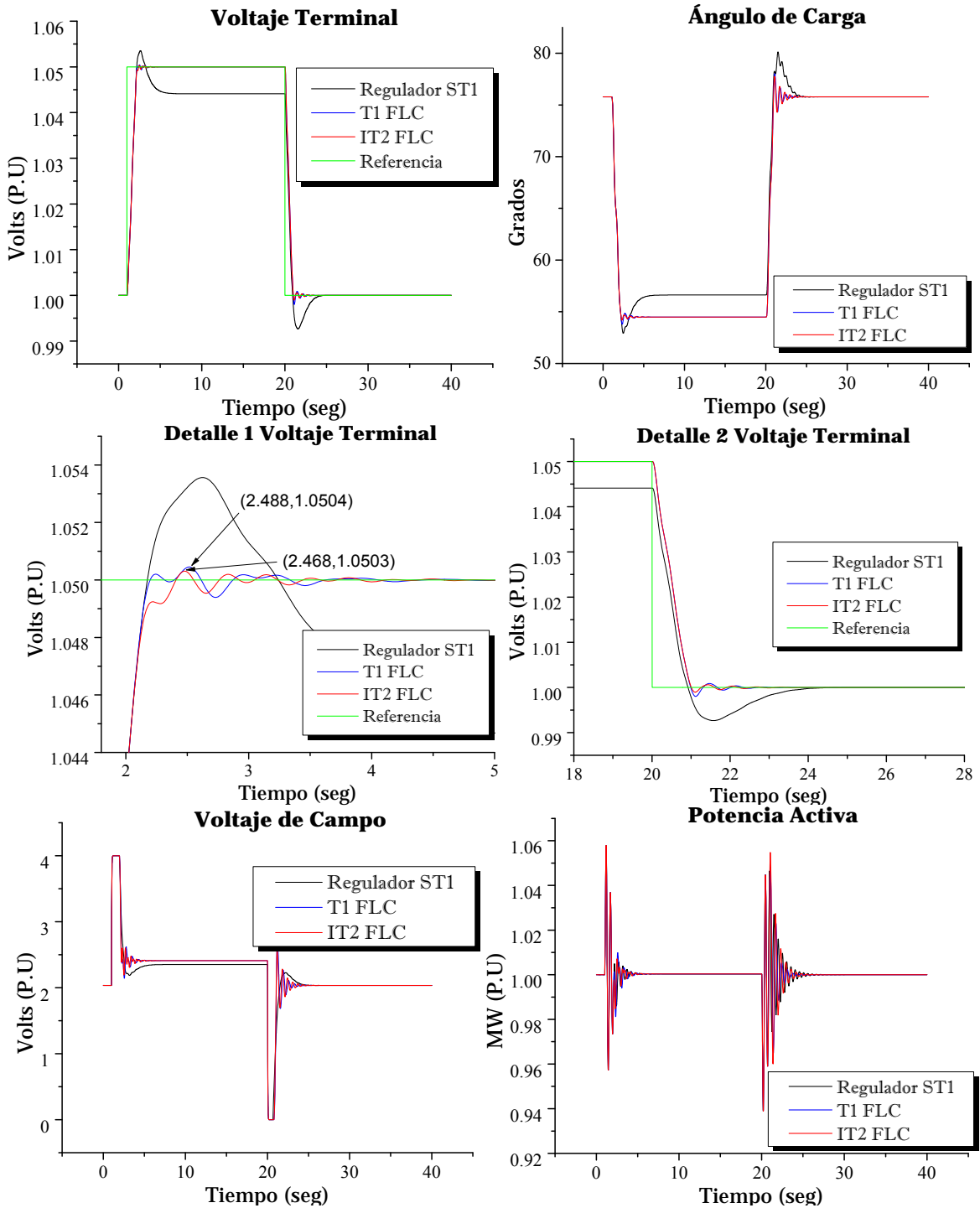
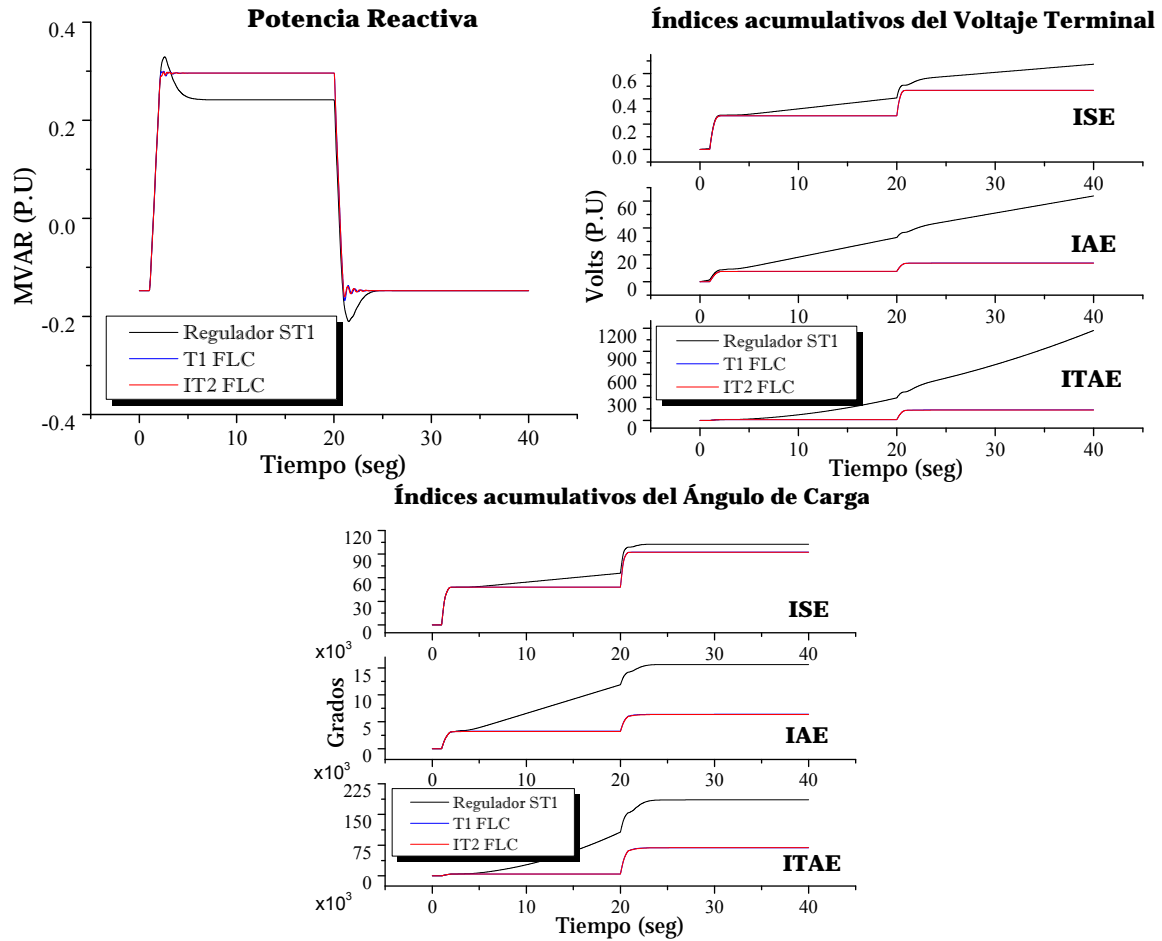


Figura 4.63 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.



Continuación de Figura 4.63 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.

Tabla 4.37 Índices de comportamiento dinámico de detalle 1 voltaje terminal Figura 4.62.

Índice	Controlador Tipo-1	Controlador Tipo-2
Sobretiro	0.8%	0.6%
Tiempo de respuesta	0.884 s	0.884 s
Tiempo de estabilización	1.160 s	1.356 s
Tiempo en alcanzar el valor pico	1.488 s	1.468 s

4.7.2 Análisis de resultados

El controlador difuso tipo-2, mostró mejor desempeño y respuesta dinámica. Para el caso funciones de membresía Gaussianas y 49 reglas de conocimiento, el controlador difuso tipo-2 mostró menor sobretiro y mejor tiempo de estabilización sobre el T1 FLC. El tiempo de respuesta de ambos controladores es idéntico, lo que indica que al inicio del disturbio el comportamiento de ambos controladores es casi idéntico al momento de seguir la referencia. Se observa además, que el regulador convencional ST1 presenta un menor sobretiro incluso que ambos controladores difusos, pero al tratarse de un control proporcional el error en estado estable es siempre presente, es decir no alcanza el valor deseado por la señal de referencia.

Para el caso de funciones de membresía triangulares, el controlador tipo-2 también presenta menor sobretiro y menor tiempo en alcanzar el valor pico en la respuesta, pero el controlador tipo-1 es superior en el tiempo de estabilización. Al igual que al utilizar funciones de membresía Gaussianas, el tiempo de respuesta es idéntico en ambos controladores.

4.8 Para una línea de transmisión corta sujeta a un pequeño disturbio en el voltaje de referencia de 1 a 1.05 P.U. e incertidumbre de $\pm 1\%$

En este apartado se añade incertidumbre en el valor de referencia en el orden de $\pm 1\%$ en el valor de referencia, y se aplica además un incremento del mismo de orden del 5%. Se compara las respuestas del controlador tipo-1 y tipo-2. La condición inicial del generador síncrono es:

$$V_{ter} = 1 P.U. \quad \theta = 6.8077^\circ \quad P_A = 1 P.U. \quad P_R = -0.1474 P.U.$$

Los parámetros de ambos controladores son tomados de la **Tabla 4.5** y **Tabla 4.18**. Para el caso del controlador tipo-2 con funciones de membresía triangulares se obtuvieron mejores resultados sin añadir incertidumbre en los consecuentes por lo que $SL = 0$ y $SR = 0$. Estos resultados son los que se muestran.

4.8.1 Controladores con funciones de Membresía Gaussianas y 49 reglas de conocimiento

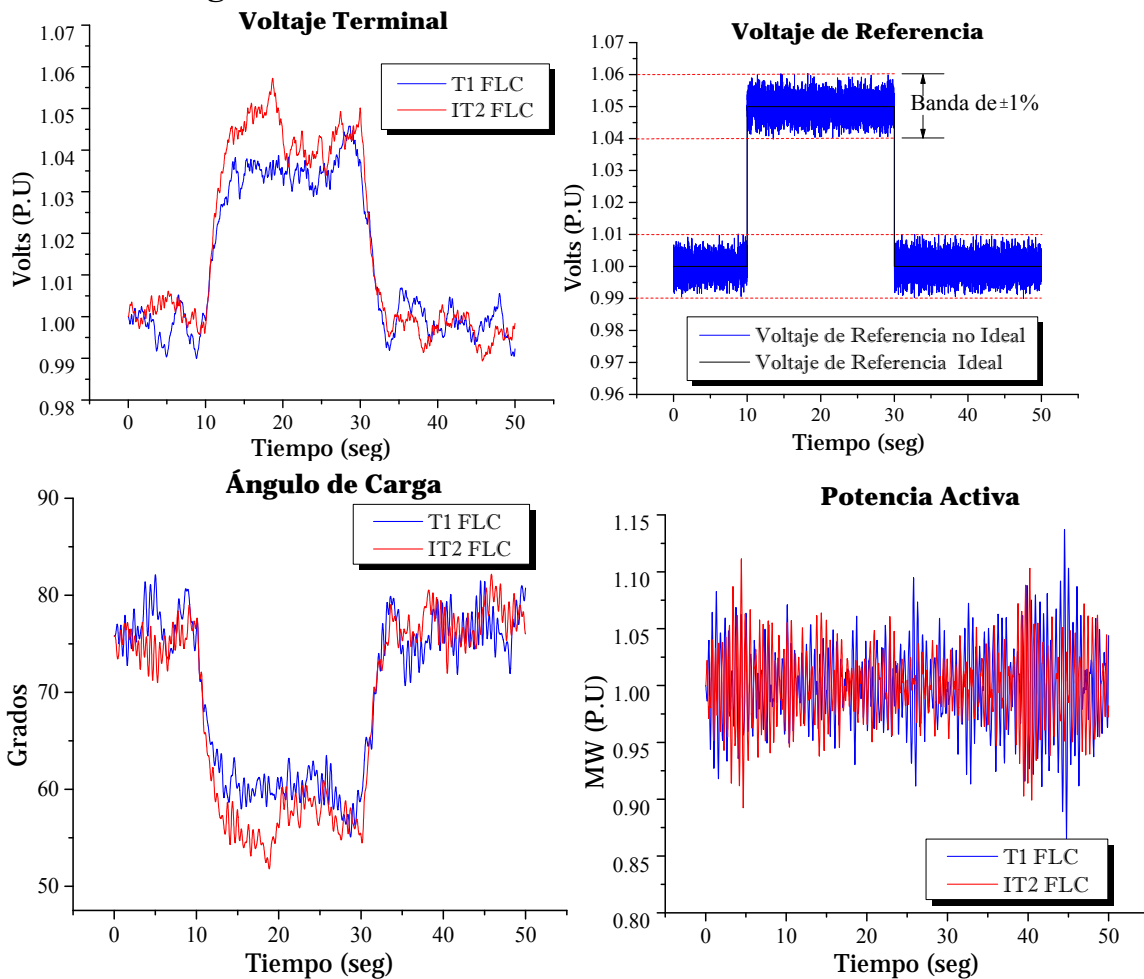
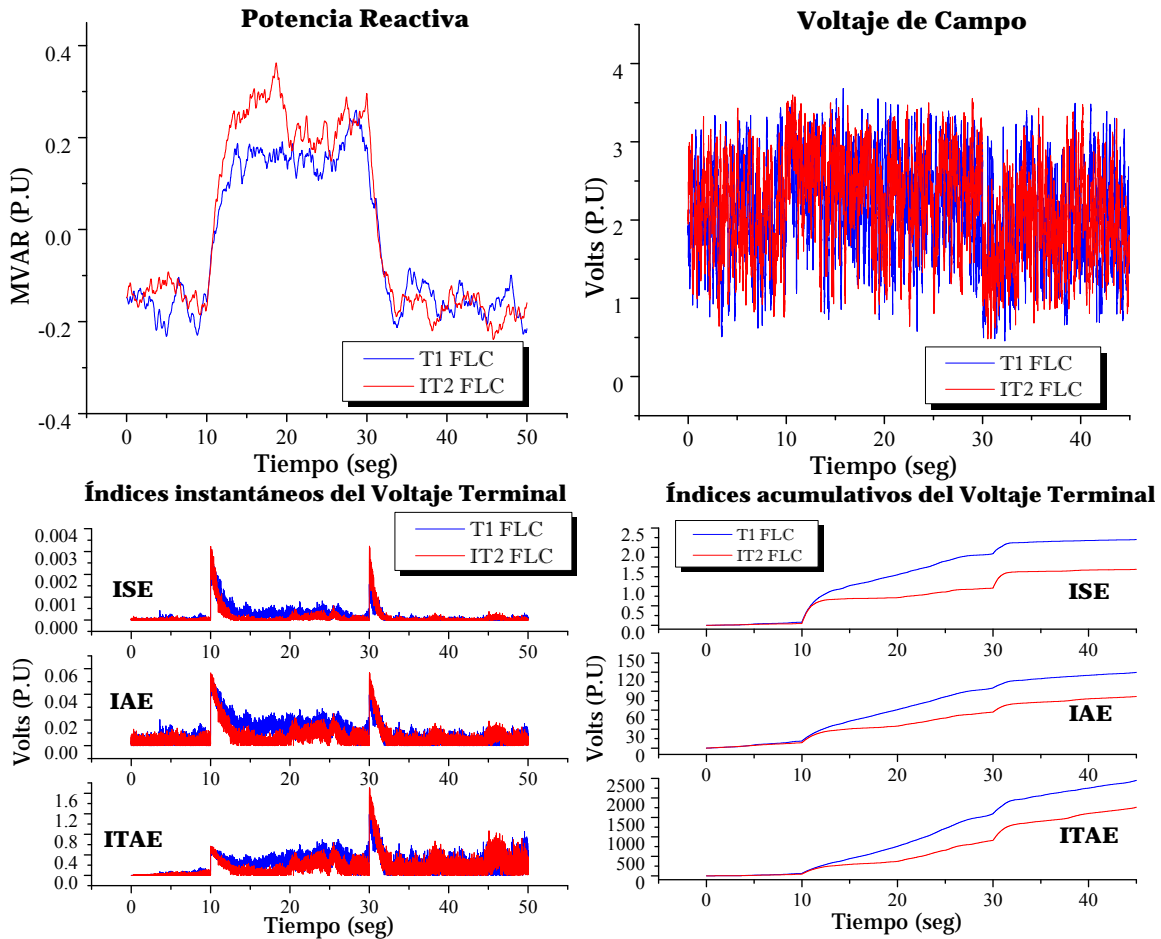


Figura 4.64 Respuesta del sistema sujeta a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento e incertidumbre de $\pm 2\%$ en el valor del voltaje de referencia.



Continuación de Figura 4.64 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento e incertidumbre de $\pm 2\%$ en el valor del voltaje de referencia.

4.8.2 Controladores con funciones de Membresía triangulares y 9 reglas de conocimiento

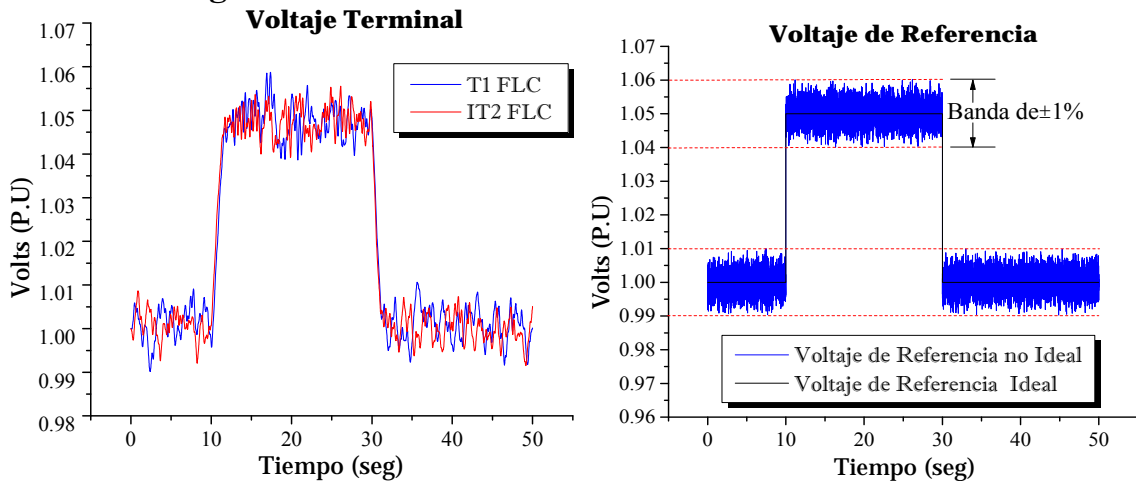
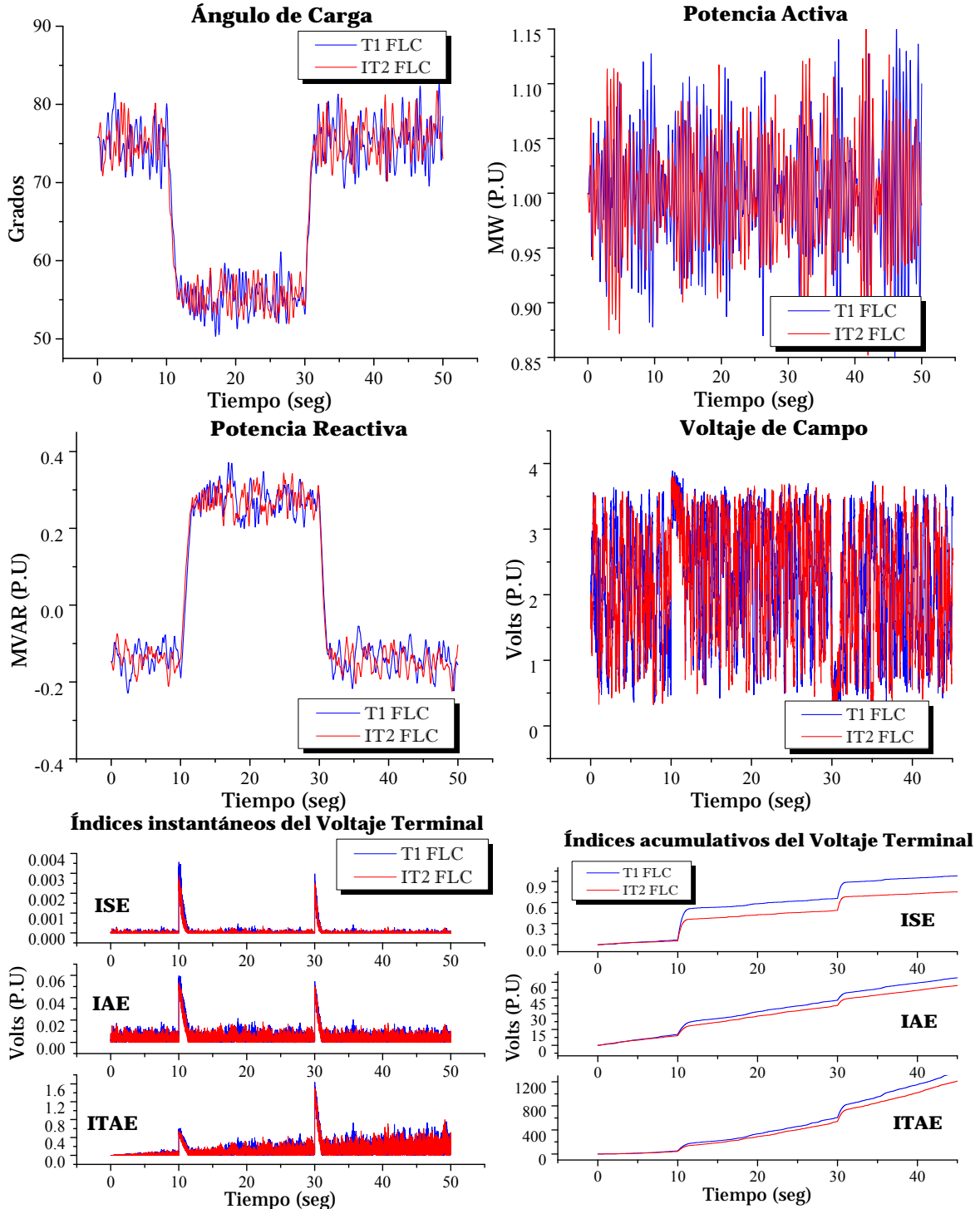


Figura 4.65 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía triangulares, 9 reglas de conocimiento e incertidumbre de $\pm 2\%$ en el valor del voltaje de referencia.



Continuación de Figura 4.65 Respuesta del sistema sujeto a un cambio en el voltaje de referencia; caso línea corta; controladores con funciones de membresía triangulares, 9 reglas de conocimiento e incertidumbre de $\pm 2\%$ en el valor del voltaje de referencia.

4.8.3 Análisis de resultados

Al añadir incertidumbre en la señal de voltaje de referencia, la señal de respuesta dinámica del voltaje terminal, ángulo de carga, potencia activa, reactiva y voltaje de campo se distorsionan debido a la incertidumbre presente en el sistema. El diagrama de fase también se distorsiona pero este no es mostrado en las figuras.

El desempeño en este caso es medido a partir de los índices del error, ya que medir sobretiros y tiempos de estabilización es difícil.

Los índices del error instantáneos y acumulativos del voltaje terminal, permiten visualizar un mejor manejo en la incertidumbre por parte del controlador tipo-2. El controlador tipo-2 presenta un menores índices acumulativos. Esta ventaja del controlador tipo-2 sobre el tipo-1 es mas notorio -para este caso- cuando se utilizan funciones de membresía Gaussianas tipo-2

En este caso de simulación se deja entrever la mayor robustez del controlador tipo-2 cuando existe incertidumbre en el sistema, para este caso en la señal de referencia de voltaje.

4.9 Para una línea de transmisión corta, toma de carga de 0 – 1 P.U.

En este apartado se realizan pequeños incrementos en el mando de la turbina desde 0 hasta 1 P.U. Los cambios no son súbitos, sino mediante rampas en tiempos preestablecidos. Al inicio el generador se encuentra sin carga y conectado al bus infinito a mediante de una línea de transmisión corta. Se compara el desempeño de los controladores tipo-1 y tipo-2 y el regulador convencional tipo St1.

Los parámetros de ambos controladores son tomados de la **Tabla 4.5** y **Tabla 4.18**.

4.9.1 Controladores con funciones de membresía triangulares

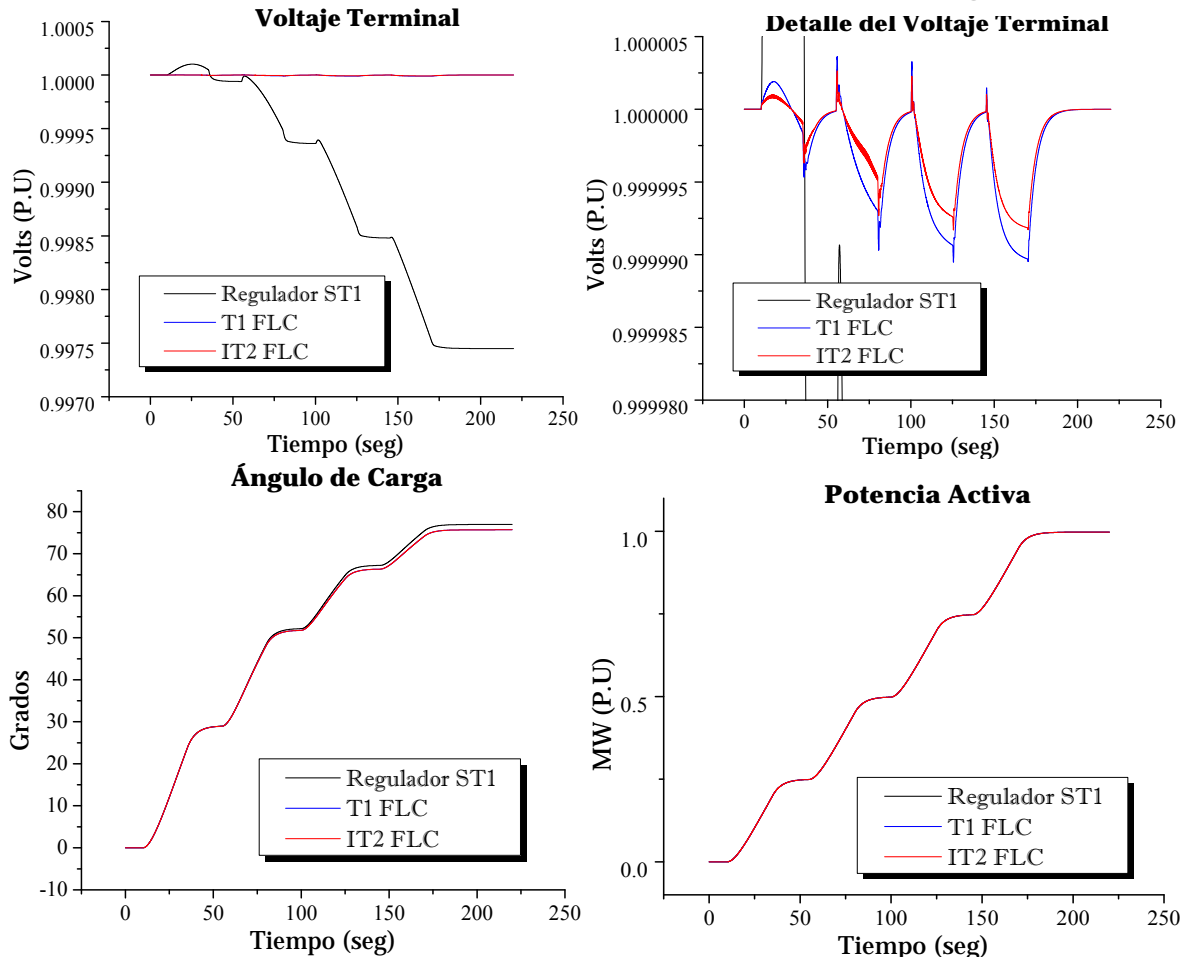
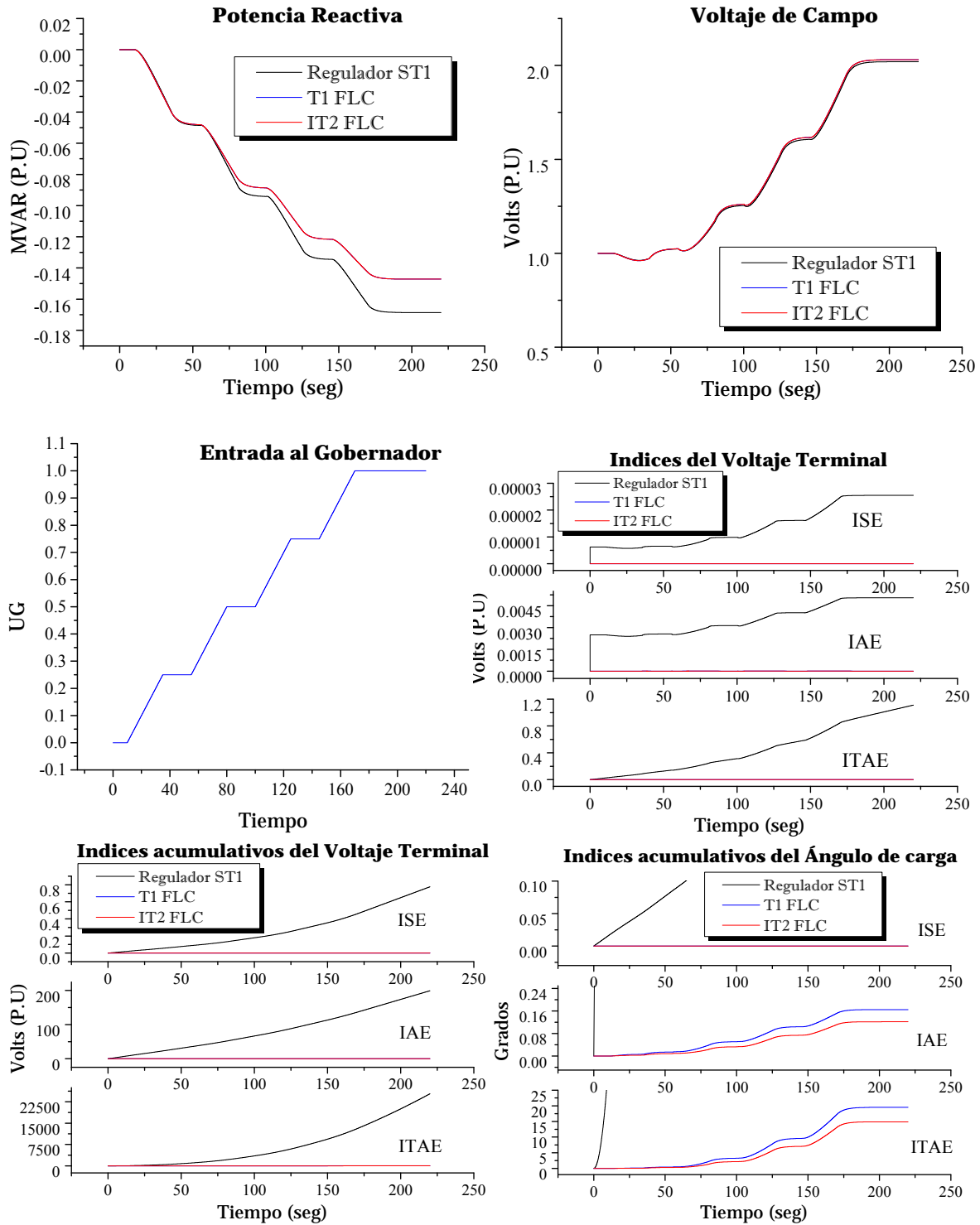


Figura 4.66 Respuesta del sistema sujeto a cambios de carga; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.



Continuación de Figura 4.66 Respuesta del sistema sujeto a cambios de carga; caso línea corta; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.

4.9.2 Controladores con funciones de membresía Gaussianas

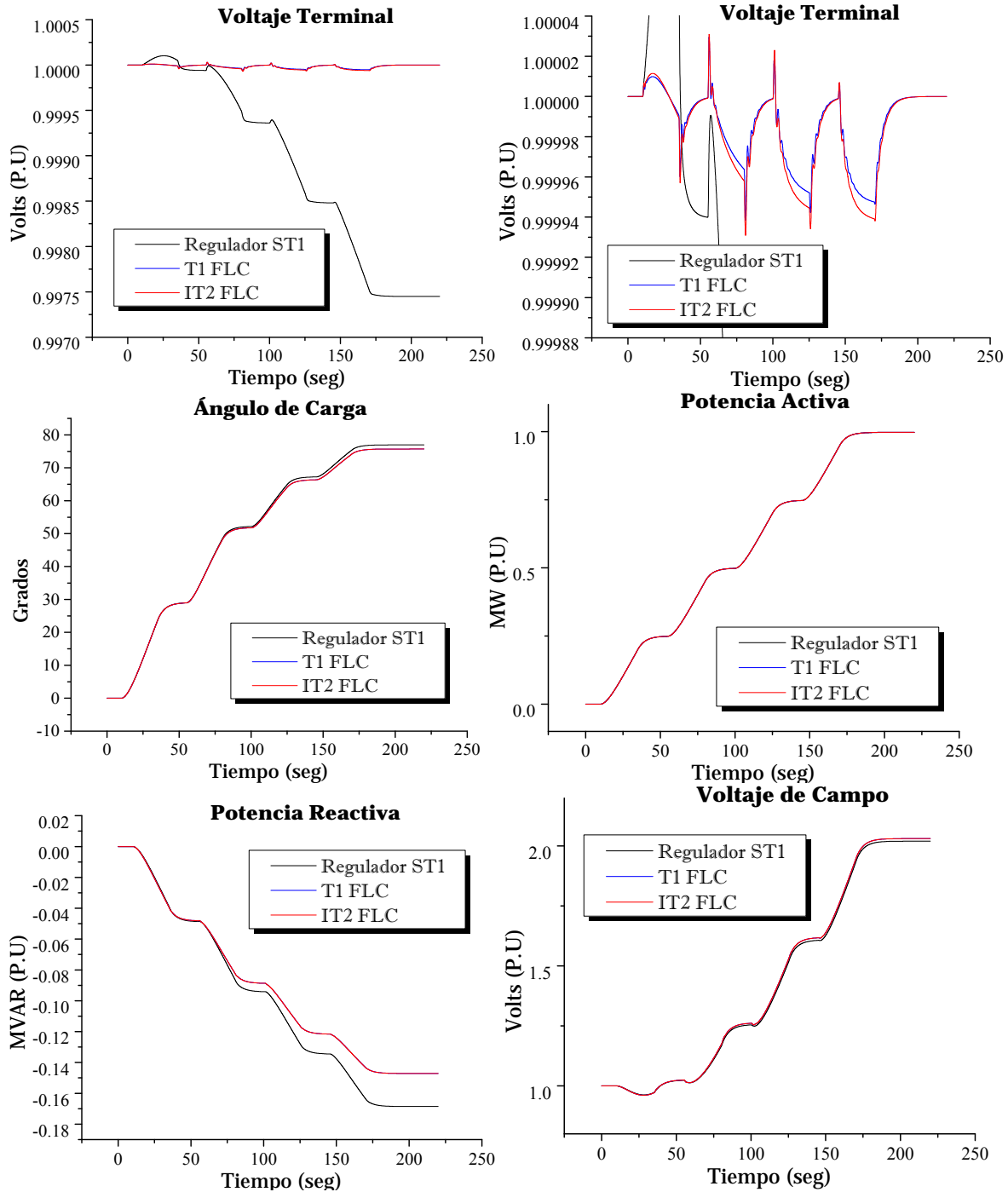
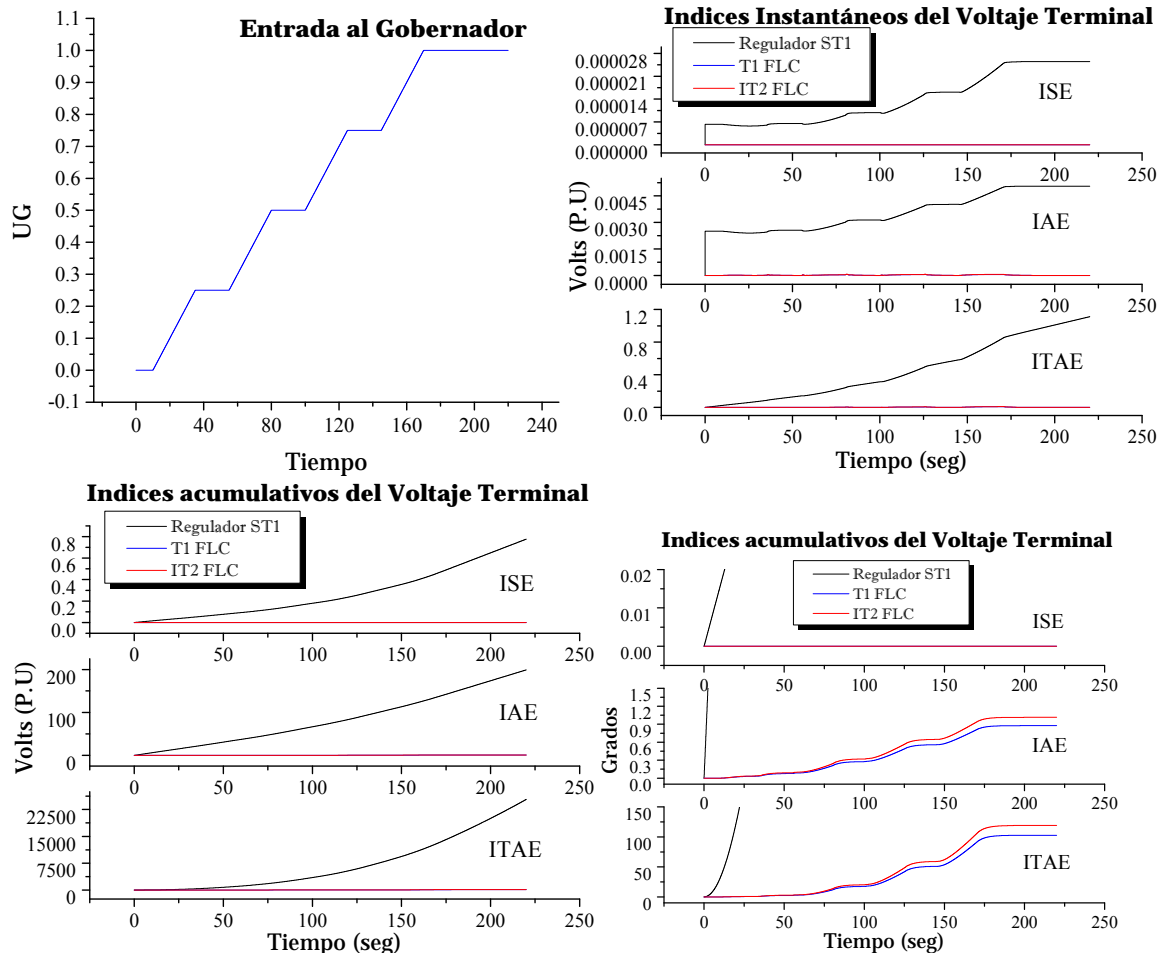


Figura 4.67 Respuesta del sistema sujeto a cambios de carga; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.



Continuación de Figura 4.67 Respuesta del sistema sujeto a cambios de carga; caso línea corta; controladores con funciones de membresía Gaussianas y 49 reglas de conocimiento.

4.9.3 Análisis de resultados

Para el caso línea corta y 9 reglas de conocimiento difuso y funciones de membresía triangulares, el controlador difuso tipo-2 presenta mejor control en el voltaje terminal ante los cambios de carga presentes. El índice acumulativo ITAE es menor, lo que significa que el controlador tipo-2 presenta menor error en el voltaje terminal a pesar de que este índice multa severamente en periodos largos de operación. Además las caídas en el voltaje terminal ante los cambios en el mando del gobernador, son menores que con el controlador difuso tipo-1, aunque cabe mencionar que estas son significativamente pequeñas.

Las respuestas en el ángulo de carga, potencia activa y reactiva se encuentran prácticamente sobrepuestas, lo que sugiere un comportamiento similar en los controladores.

Cuando se utilizan funciones de membresía Gaussianas y 49 reglas de conocimiento difuso, el controlador tipo-2 se ve superado por el controlador tipo-1. En lo que respecta a los índices de cuantificación del error.

El desempeño del regulador convencional ST1 es inferior a ambos controladores difusos. En ambos casos se ve superado por los controladores difusos

4.10 Prueba de Toma de Carga Línea Larga de 0 – 0.7 P.U.

En este apartado se realizan pequeños incrementos en el mando de la turbina desde 0 hasta 0.7 P.U. Los cambios no son súbitos, sino mediante rampas en tiempos preestablecidos. Al inicio el generador se encuentra sin carga y conectado al bus infinito a mediante de una línea de transmisión larga. Se compara el desempeño de los controladores tipo-1 y tipo-2 y el regulador convencional tipo St1.

Los parámetros de ambos controladores son tomados de la **Tabla 4.22** y **Tabla 4.35**.

4.10.1 Controladores con funciones de membresía triangulares

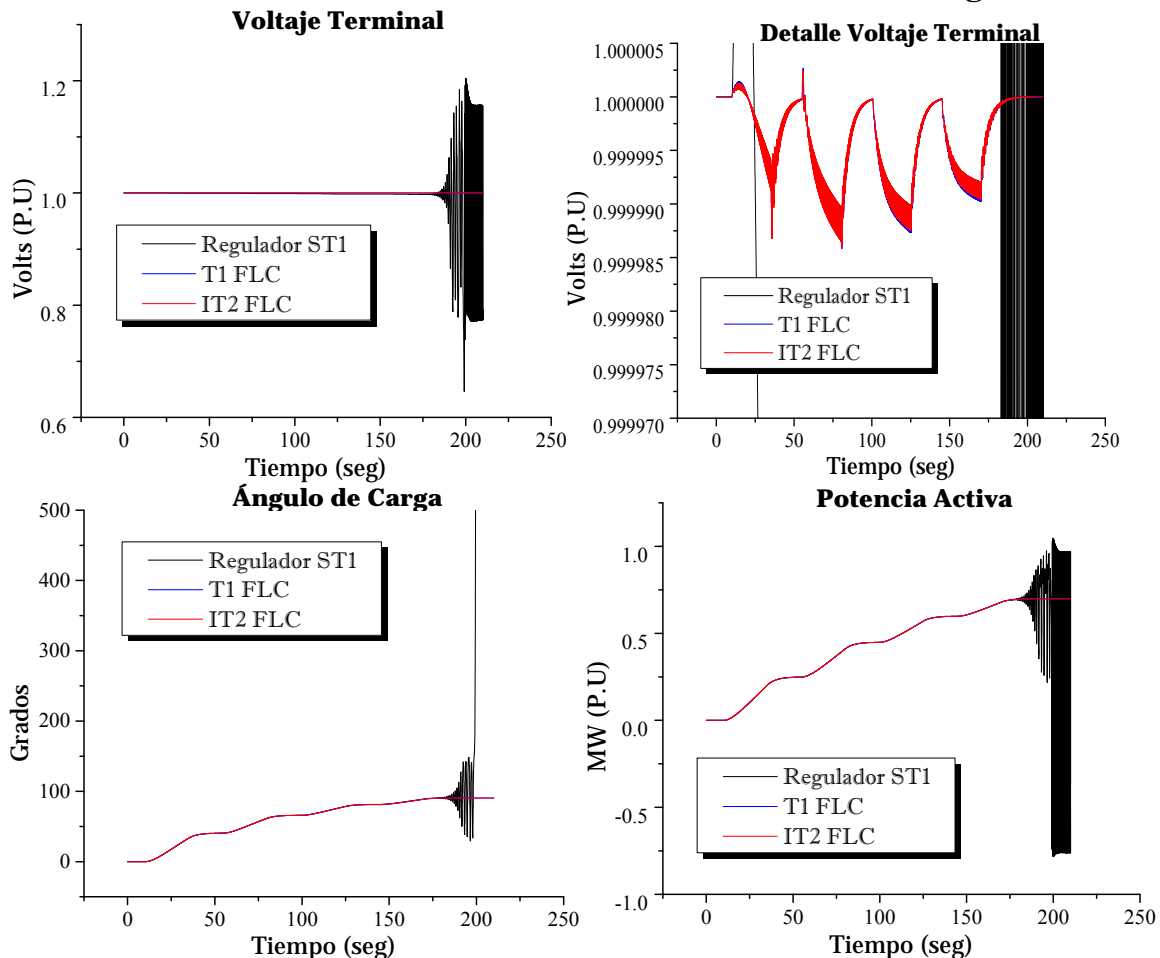
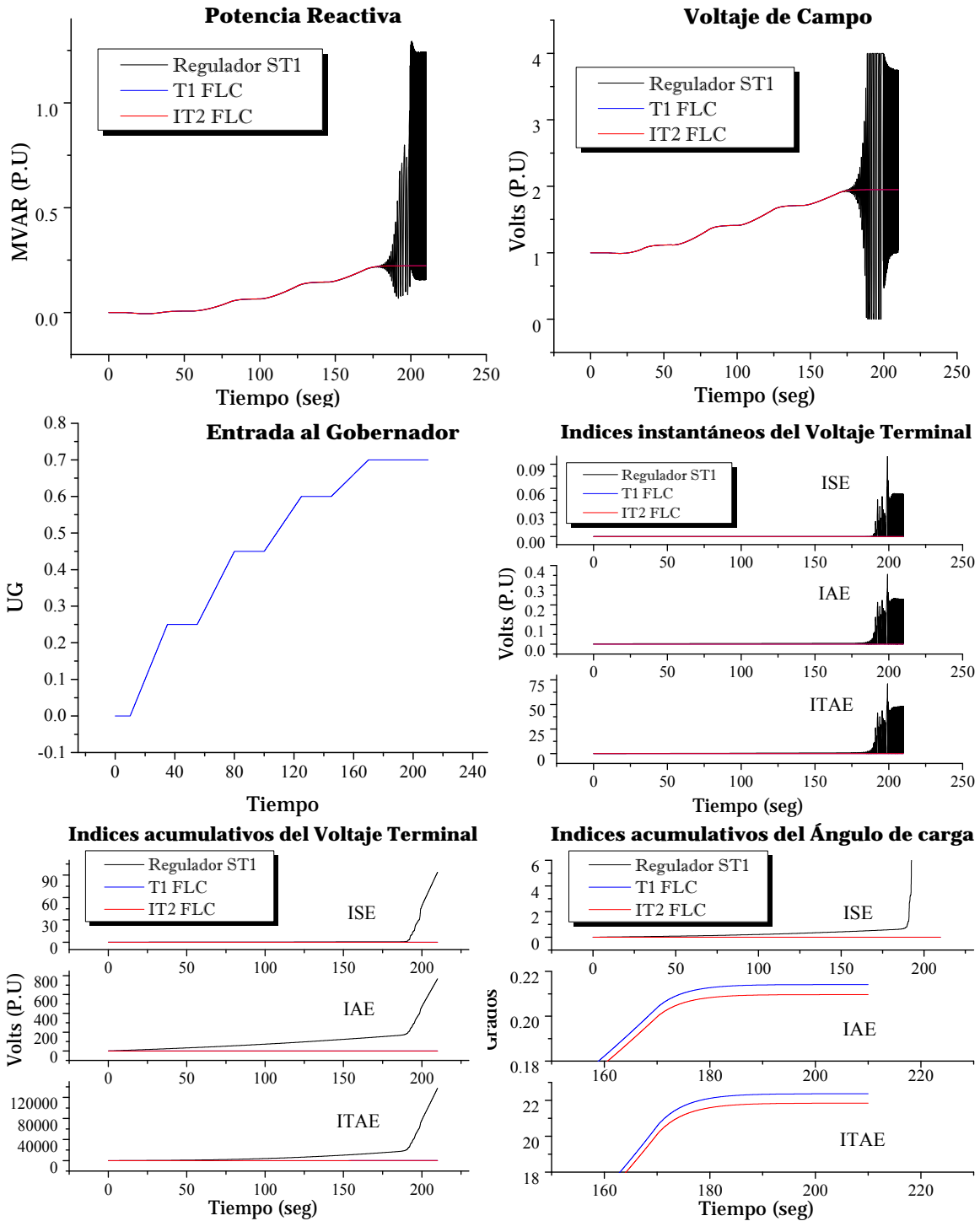


Figura 4.68 Respuesta del sistema sujeto a cambios de carga; caso línea larga; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.



Continuación de Figura 4.68 Respuesta del sistema sujeto a cambios de carga; caso línea larga; controladores con funciones de membresía triangulares y 9 reglas de conocimiento.

4.10.2 Controladores con funciones de membresía Gaussianas

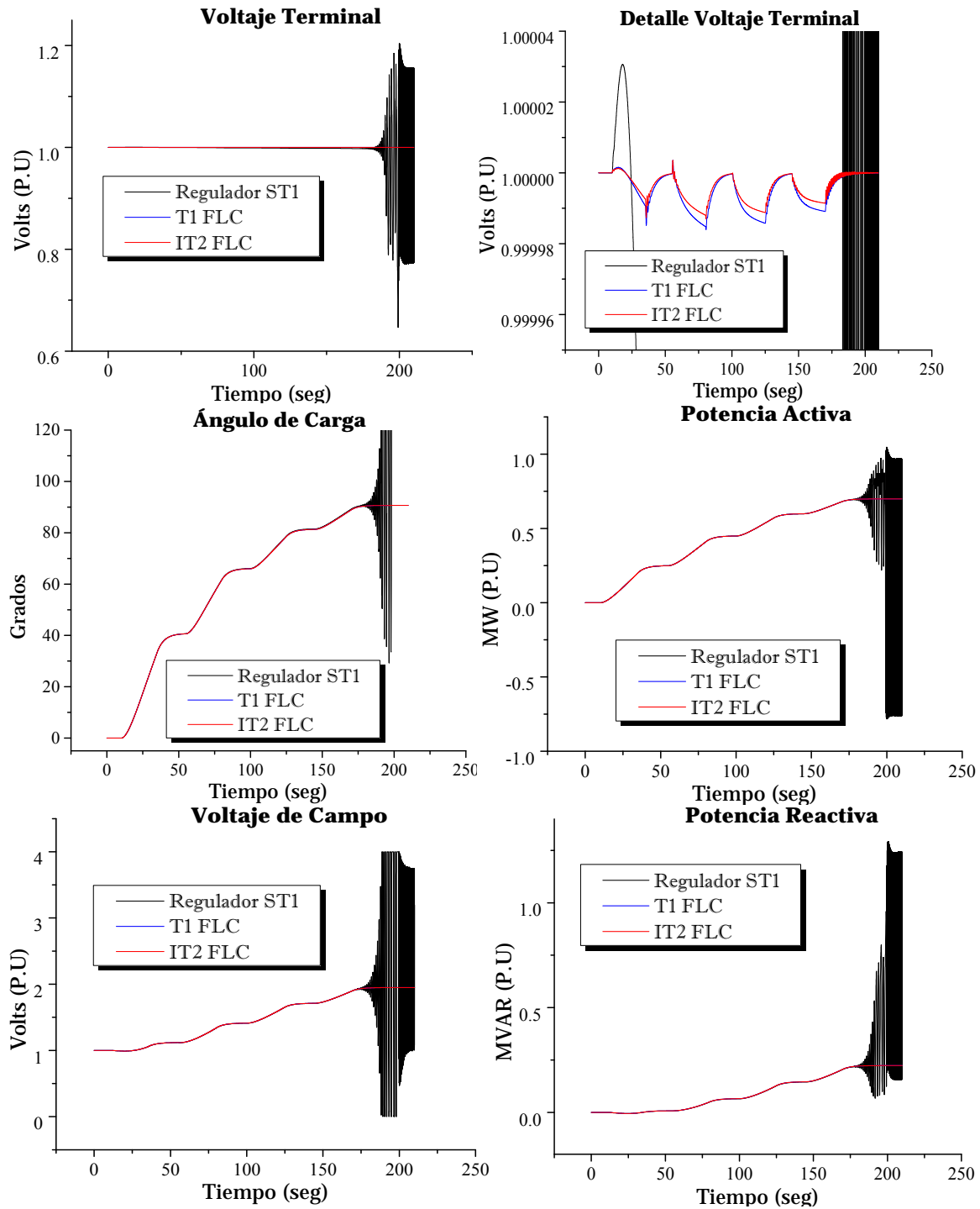
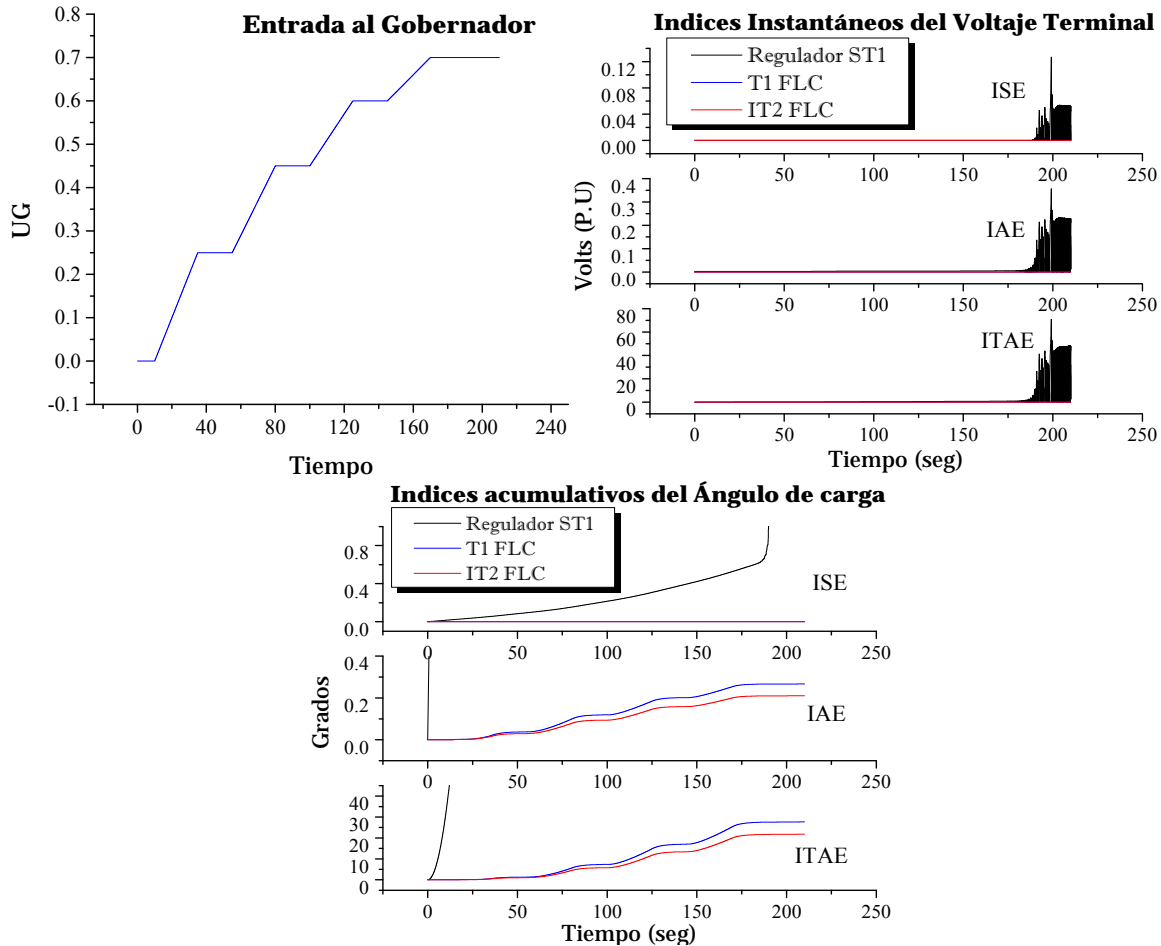


Figura 4.69 Respuesta del sistema sujeto a cambios de carga; caso línea larga; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.



Continuación de Figura 4.69 Respuesta del sistema sujeto a cambios de carga; caso línea larga; controladores con funciones de membresía Gaussianas y 9 reglas de conocimiento.

4.10.1 Análisis de resultados

Para el caso línea larga, el controlador difuso tipo-2 tiene menores caídas de voltaje con respecto al tipo-1 aunque es de notar que dichas caídas son pequeñas. Además el regulador ST1 se muestra inestable en el ultimo paso de toma de carga, y no logra regular el voltaje en terminales del generador.

El controlador difuso tipo-2 es superior tanto al controlador tipo-1 como al regulador ST1. Los índices IAE e ITAE permiten visualizar este hecho. Note que el índice ISE es pequeño en ambos controladores debido a que no existen sobretiros en la respuesta del voltaje terminal.

CAPÍTULO 5

*Vine, vi, vencí.
Julio Cesar. Emperador Romano.*

5. Conclusiones y Recomendaciones

5.1 Conclusiones Generales

En este trabajo se desarrolló un controlador difuso tipo-2 en intervalos. Este controlador muestra en la mayoría de las simulaciones ser superior al convencional controlador difuso tipo-1.

En general y para el caso línea corta y ante grandes disturbios, el controlador difuso tipo-2 que presentó mejores resultados fue el controlador con conjuntos difusos Gaussianos y 9 reglas de conocimiento. Ya que fue este controlador quien mostró mejor desempeño en las pruebas a gran disturbio, es decir corto circuito pleno en el bus a 6 y 10 ciclos de duración, tanto para el caso ideal (sin incertidumbre) como el caso no ideal (incertidumbre en la medición del voltaje en terminales), debido a ser esa una de las principales ventajas en este tipo de controladores (el manejo en la incertidumbre). Además el desempeño mejoró conforme se aumentaba el ancho de banda de variación aleatoria normal del voltaje terminal.

En el caso del generador conectado al bus mediante una línea de transmisión larga, no se pudieron realizar muchos análisis debido a que dadas las condiciones iniciales, el generador se encuentra operando cerca de sus límites teóricos de operación en los cuales ni el controlador propuesto logra evitar la inestabilidad del generador. Aunque en ciertas simulaciones se deja entrever la regulación del voltaje a pesar del disturbio presente.

Para el caso de pequeños disturbios y cambios en el voltaje de referencia caso ideal, el controlador difuso tipo-2 con conjuntos difusos Gaussianos mostró una buena respuesta dinámica en sus índices de comportamiento de sobretiro y tiempo de estabilización con respecto al controlador tipo-1 con conjuntos Gaussianos. Es difícil establecer un determinado porcentaje de mejora sobre el controlador tipo-1, puesto que cada caso de simulación es particular. Pero si se permiten observar mayor amortiguamiento en los sobretiros y menores oscilaciones. En el caso no ideal, es decir con incertidumbre en una banda de $\pm 1\%$ el controlador tipo-2 obtuvo menores índices de error con respecto al ISE, IAE e ITAE. La ventaja del controlador tipo-2 sobre el controlador tipo-1 es mas notoria cuando se utilizan funciones de membresía Gaussianas tipo-2.

Es posible que el mejor desempeño del controlador difuso tipo-2 con funciones de membresía Gaussianas se deba a la misma forma suave de las funciones de membresía, lo que permite una superficie de control mas suave a diferencia del controlador tipo-2 con funciones de membresía triangulares.

Las comparación entre el controlador tipo-2 y el regulador convencional ST1 no se menciona, puesto que gráficamente el controlador tipo-2 es superior en todas las pruebas a las que el sistema fué sometido.

También se concluye que el controlador difuso tipo-2 no es sencillo de comprender a diferencia del control difuso convencional y requiere de cierta experiencia y metodología para su sintonización y ajuste de las incertidumbres en las funciones de membresía.

Finalmente se ha presentado una novedosa alternativa de control inteligente prometedora para el control del voltaje en terminales de un generador síncrono.

5.2 Aportaciones

Las aportaciones del presente trabajo se enumeran de la forma siguiente:

- Aplicación de un Controlador Lógico Difuso tipo-2 con conjuntos difusos en intervalos (*“Interval Type 2 Fuzzy Logic Controller”* IT2 FLC), como regulador de voltaje al sistema de excitación de un generador síncrono, en vista de que no existe trabajo alguno reportado en la sección de SEPI-ESIME Zacatenco para el control de voltaje en terminales del generador síncrono utilizando un IT2 FLC. Aunque a nivel internacional existen algunos reportes de trabajos en bases como Science Direct [33][35] (<http://www.sciencedirect.com>) y la propia IEEE [34][7] (<http://ieeexplore.ieee.org/Xplore/home.jsp>).
- Las pruebas realizadas al sistema, dada las siguientes variables posibles en el IT2 FLC:
 - a) Número de conjuntos difusos tipo-2 posibles para las dos variables de entrada: 3, 5 y 7.
 - b) Tipos de funciones de membresía: Funciones triangulares tipo-2 (con 4 variantes en su incertidumbre) y Funciones Gaussianas tipo-2 con incertidumbre en la desviación estándar.
 - c) Algoritmos de reducción de tipo: Algoritmo Karnik y Mendel y Algoritmo mejorado de Karnik y Mendel.
 - d) Las pruebas con dos fuentes de incertidumbre en el sistema: en la lectura del voltaje terminal que sugiere ruido en el transductor y en la señal de referencia del voltaje.
- Las pruebas comparativas entre el IT2 FLC, el control difuso tipo 1 convencional y el regulador de voltaje convencional ST1.

Artículos

A. Villegas, E. Moo, R. Cortés, D. Ruiz; *“Medidor del Ángulo de Carga del Generador Síncrono basado en el Microcontrolador KL25Z”*; XIV Congreso Nacional de Ingeniería Electromecánica y de Sistemas, CNIES, México D.F. 2013.

E. Moo, R. Cortes, D. Romero; *“Control de Velocidad Difuso Sugeno con Parámetros Adaptables por Inferencia Tipo Mamdani para un Motor de CD”*; 7º Congreso Internacional de Ingeniería Electromecánica y de Sistemas, CIIES, México D.F. 2014.

E. Moo, D. Romero; *“Control de Velocidad Difuso Tipo 2 para un Motor CD Excitación Separada”*; XXVII Reunión Internacional de Verano de Potencia, Aplicaciones Industriales y Exposición Industrial, RVP-AI. Acapulco Guerrero, 2014.

5.3 Trabajos futuros

Varias propuestas se plantean para continuar desarrollando el controlador tipo-2 propuesto en este trabajo:

1. Se sugiere implementar un método de búsqueda inteligente (algoritmos genéticos, búsqueda por tabú, o algún otro método de optimización inteligente) para la sintonización de la incertidumbre presente en cada conjunto difuso tipo-2 en el universo de entrada, puesto que en este trabajo solo se realizó una sintonización manual.
2. Se sugiere utilizar universos de entrada no simétricos y con conjuntos difusos tipo-2 con incertidumbre independiente y no igual para todos. Ya que esto permitirá una superficie de control con mayor no linealidad. Y esto a su vez un control más robusto.
3. Se propone también, la implementación de este algoritmo de control a un microcontrolador para una aplicación en tiempo real del control del voltaje en terminales de un generador síncrono, a partir de las simulaciones realizadas en este trabajo.

REFERENCIAS

- [1] P. Kundur, *"Power System Stability and Control."* McGraw-Hill, 1993.
- [2] L. A. Zadeh, *"The Concept of a Linguistic Variable and its Application to Approximate Reasoning - I."* Information and Control 8, pp. 199-249, 1975.
- [3] J. Mendel, *"Uncertain Ruled-Based Fuzzy Logic Systems: Introduction and New Directions."* Prentice Hall, 2001.
- [4] R. Dunn, K. Bell and A. Daniels, *"Fuzzy Logic and its application to Power Systems."* The Institution of Electrical Engineerings, IEE, London UK, 1997.
- [5] Yong-Hua Song and Allan T. Johns, *"Tutorial: Applications of Fuzzy Logic in Power Systems Part 1 General Introduction to Fuzzy Logic."* Power Engineering Journal, October 1997.
- [6] G. J. and M. Wierman, *"Uncertain-Based Information."* Physica-Verlag, Heidelberg, Germany, 1998.
- [7] K. Vamma, D. Mohanta and M. Bharata, *"Applications of Type-2 Fuzzy Logic in Power Systems: A Literature Survey."* Environment and Electrical Engineering, International Conference, Wroclaw, IEEE 2013.
- [8] Daozi Xia y G.T Heydt, *"Self-Tunning Controller for Generator Excitation Control"*. IEEE Transactions on Power Apparatus and Systems, Vol. PAS-102, No. 6 June 1983.
- [9] J. Díaz, *"Desarrollo de un Regulador de Voltaje para un Generador Síncrono Usando la Teoría de Resonancia Adaptable."* Tesis de Maestría, SEPI-ESIME-ZACATENCO, Junio 2001.
- [10] R. Dorf and R. Bishop, *"Modern Control Systems."* Prentice Hall. 12th edition, 2011.
- [11] IEEE Power Engineering Society, *"IEEE Guide for Identification, Testing, and Evaluation of the Dynamic Performance of Excitation Control Systems."* IEEE Estandar 421.2-1990.
- [12] H. Rosas, *"Control Difuso Autosintonizable de Voltaje por medio de la Búsqueda por Tabú para un Genrador Síncrono."* Tesis de Maestría, SEPI-ESIME-ZACATENCO, Abril 2002.
- [13] P. Suppes; S. Hill, *"Introducción a la Lógica Matemática."* Reverté S.A 1975.
- [14] L. A. Zadeh, *"Fuzzy Sets."* Information and Control 8(3), pp. 338-353, 1965.
- [15] E.H. Mamdani, *"Application of fuzzy algorithm for simple dynamic plant."* IEEE Proceedings-Control Theory and Applications, Vol. 121, No.12, 1585-1588, 1974.
- [16] E. H. Mamdani, *"Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Synthesis."*
- [17] A. Rubio, *"Control Difuso Autosintonizable de Voltaje por Medio de la Búsqueda por tabú y Algoritmos para un Generador Síncrono."* Tesis de Maestría, SEPI-ESIME-ZACATENCO, 2006.
- [18] N. Pereyra, *"Control de Voltaje de un Compensador Estático de Vars Utilizando Lógica Difusa Sintonizada por Tabú."* Tesis de Maestría, SEPI-ESIME-ZACATENCO, 2006.
- [19] Wu and W. Tan, *"Type-2 Fuzzy Logic Controller for the Liquid-Level Process."* Proceedings of 2004 IEEE Int' 1. Conf. on Fuzzy Systems, Budapest, Hungary, pp. 248-253, 2004.
- [20] J. Mendel, H. Hagra, W. Tan, W. Melek, H. Ying, *"Introduction to Type-2 Fuzzy Logic Control Theory and Applications."* IEEE, Wiley, 2014.
- [21] M. Mizumo and K. Tanaka, *"Some Properties of Fuzzy Sets of Type-2."* Information and Control, Vol. 31, pp. 312-340, 1976.

- [22] N. Karnik, J. Mendel and Q. Liang, "Type-2 Fuzzy Logic Systems." IEEE Transactions on Fuzzy Systems, Vol. 7, No. 6, December 1999.
- [23] N. Karnik, J. Mendel, "Introduction to Type-2 Fuzzy Logic Systems." IEEE 1998.
- [24] D. Wu, "A Brief Tutorial on Interval Type-2 Fuzzy Sets and Systems."
- [25] J. Mendel and R. John, "Type-2 Fuzzy Sets Made Simple." IEEE Transactions on Fuzzy Systems, Vol. 10, No. 2, April 2002.
- [26] J. Mendel, R. John and F. Liu, "Interval Type-2 Fuzzy Logic Systems Made Simple." IEEE Transactions on Fuzzy Systems, Vol. 14, No. 6, December 2006.
- [27] C. Hernández, "Control difuso de Velocidad de un Motor de Corriente Directa." Tesis de Maestría, Sección de Estudios de Posgrado SEPI-ESIME-ZACATENCO, 1997.
- [28] R. Cortés, "Control de Excitación Difuso de un Generador Síncrono." Tesis de Maestría, Sección de Estudios de Posgrado SEPI-ESIME-ZACATENCO, 1997.
- [29] O. Ghanayem and L. Reznik, "Excitation Control of a Synchronous Generator Using an on-line Adaptive Fuzzy Logic Controller Structure." Sixth IEEE International Conference, Barcelona, July 1997.
- [30] G. Peiyusan, F. Shihong, P. Xiangnan, Z. Yongwang and B. Man, "Fuzzy Logic Controller for Generator Excitation Control." Power System Technology, IEEE International Conference, Beijing, August 1998.
- [31] D. Sumina, T. Idzotic and I. Erceg, "Fuzzy Logic Control of Synchronous Generator under the Condition of Transient Three Phase Short Circuit." Power Electronics and Motion Control Conference, IEEE, Portoroz Slovenia, 2006.
- [32] R. Ramya and K. Selvi, "A Simple Fuzzy Excitation Control System for Synchronous Generator." Proceedings of Emerging Trends in Electrical and Computer Technology (ICETECT), IEEE International Conference, Tamil Nadu, 2011.
- [33] I. Robandi and B. Kharisma, "Design of Interval Type-2 Fuzzy Logic Based Power Systems Stabilizer." International Journal of Electrical and Electronics Engineering, 2009.
- [34] M. Kumar, G. Pillai and V. Kumar, "Interval Type-2 Fuzzy Logic Controller as a Power System Stabilizer." Advances in Power Conversion and Energy Technologies, IEEE International conference, August 2012.
- [35] A. Abbadi, L. Nezli and D. Boukhetala, "A Nonlinear Voltage Controller Based on Interval Type-2 Fuzzy Logic Control System for Multimachine Power Systems." International Journal of Electrical Power & Energy Systems, Vol. 45, Issue 1, pp 456-457, 2012.
- [36] H. Fayek and I. Elamvazuthi, "Type-2 Fuzzy logic PI (T2FLPI) Based DC Servomotor Control." Journal of Applied Sciences Research, 8(5): pp. 2564-2574, 2012.
- [37] O. Castillo and P. Melin, "Type-2 Fuzzy Logic: Theory and Applications." Springer-Verlag, 2008.
- [38] C. Lynch, H. Hagra, and V. Callaghan, "Embedded Type-2 FLC for Real-Time Speed Control of Marine and Traction Diesel Engines." Proceedings of 2005 IEEE Int'l. Conf. on Fuzzy Systems, Reno, NV, pp. 347-352, 2005.
- [39] H. Hagra, "A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots." IEEE Trans. On Fuzzy Systems, vol. 12, pp. 524-539, Aug. 2004.
- [40] S. Zaheer and J. Kim, "Type-2 Fuzzy Airplane Altitude Control: A Comparative Study." Proceedings of 2011 IEEE Int'l. Conf. on Fuzzy Systems, Taipei, Taiwan pp. 2170-2176, 2011.
- [41] Dongrui Wu and Jerry Mendel, "Enhanced Karnik- Mendel Algorithms". IEEE Transactions on Fuzzy Systems, Vol. 17, No. 4, August 2009.

- [42] Dongrui Wu and Jerry Mendel, "Enhanced Karnik Mendel Algorithms for Interval Type-2 Fuzzy Sets and Systems". IEEE 2007.
- [43] J. Flore, L. Morones, "Simulación del Gobernador de la Turbina y del Control de Excitación de un Generador." *Conciencia Tecnológica*, num. 16 pp. 9-13, Instituto Tecnológico de Aguascaliente, Abril 2001.
- [44] G. Mann, B. Huand R. Gosine, "Analysis of Direct Action Fuzzy PID Controllers Structures." *IEEE Transactions on Systems Man, and Cybernetics*, Vol. 29, No. 3, June 1999.
- [45] K. Passino, "Fuzzy Control". Addison Wesley, 1998.
- [46] E. Gómez, P. Melin and O. Castillo, "Simple Tuning of Type-2 Controllers". *Soft Computing for Intelligent Control and Mobile Robot*, SCI 318, pp. 103-123, Springer Verlag 2010.
- [47] M. Aliasghary, I. Eksin, M. Güzelkaya and T. Kumbasar, "Design of an Interval Type-2 Fuzzy Logic Controller based on Conventional PI Controller." *Mediterranean Conference on Control & Automation*, Barcelona, Spain, July 2012.
- [48] T. Kumbasar, E. Ysil, O. Karasakal, "Self-tuning Interval type-2 Fuzzy PID Controllers Based on Online Rule Weighting." *Istanbul Technical University, Faculty of Electrical and Electronics Engineering*.
- [49] W. Dongrui, "Design and Analysis of Type-2 Fuzzy Logic Systems." A thesis submitted for the degree of master of engineering department of electrical and computer engineering national university of Singapore 2005.
- [50] R. Park and R. Robertson, "The Reactances of Synchronous Machines." Presented at the Winter Convention of the A.I.E.E., New York, Feb. 13-17, 1928.
- [51] D. Ruiz y D. Olguín, "Modelado de la Máquina Síncrona para Estudios de Estabilidad a Grandes Disturbios en Sistema Eléctricos de Potencia con Parámetros Obtenidos Mediante Pruebas Convencionales." *Reporte interno SEPI-ESIME-ZACATENCO*, Abril 1995.
- [52] F. Glover and M. Laguna, "Tabú search." Kluwer Academic, 1997.
- [53] D. Karaboga and D. T. Pham, "Intelligent Optimisation techniques." *Tesis de Maestría, SEPI-ESIME-ZACATENCO*, 2006.
- [54] EIA standard RS-232-C, "Interface between Data Terminal Equipment and Data Communication Equipment Employing Serial Binary Data Interchange." Washington: Electronic Industries Association. Engineering Dept. 1969.
- [55] R. Cano, "Implementación de un medidor inteligente con tecnología Arm." *Tesis de Maestría, SEPI-ESIME-ZACATENCO*, Julio 2014.
- [56] Freescale Semiconductors, "FRDM-KL25Z user's manual." Reference manual.
- [57] OpenSDA, User's guide. EUA. <http://www.pemicro.com/opensda/>.
- [58] Yong-Hua Song and Allan T. Johns, "Applications of Fuzzy Logic in Power Systems Part 3 Example Applications." *Power Engineering Journal*, April 1999.
- [59] M-Y Hsiao and T-H Li, "Design of Interval Type-2 Fuzzy Logic Controller." *International Conference on Systems, Man, and Cybernetics*, Taipei, Taiwan, October 2006.

APÉNDICE A

A.1 ÍNDICES DE FUNCIONAMIENTO

A.1.1 Introducción a los Índices de Funcionamiento

Es posible medir el desempeño de los sistemas de control mediante la respuesta transitoria que los sistemas presentan ante entradas tipo escalón, rampa, disturbios al sistema etc. De esta forma un índice de desempeño puede ser cuantificado con el fin de evaluar y analizar el desempeño del sistema de control. Estas medidas cuantitativas del comportamiento resultan útiles para el ajuste de parámetros de sistemas automáticos de control, así como el diseño de sistemas óptimos de control.

En términos generales, un índice de desempeño es una medición cuantitativa del desempeño de un sistema, el cual es escogido para enfatizar la importancia de ciertas especificaciones del sistema. [10]. El sistema es considerado óptimo cuando el ajuste de los parámetros del sistema de control, permiten obtener valores extremos para los índices de funcionamiento, comúnmente valores mínimos. La selección del índice de desempeño a medir, depende del tipo de respuesta deseada, debido a que el error en el sistema contribuye en diferente forma para cada índice de desempeño [37].

La forma general de la integral de funcionamiento es:

$$I = \int_0^T f[e(t), r(t), y(t), t] dt \quad (\text{A. 1})$$

Es decir, f es función de la señal del error $e(t)$, de la entrada o referencia $r(t)$, de la salida $y(t)$ y del tiempo, integrada desde un tiempo cero hasta un límite superior finito T elegido arbitrariamente tal que aproxime un valor en estado estable. Comúnmente suele seleccionarse el tiempo de estabilización T_s [10]. En la literatura existen varios índices de funcionamiento algunos de estos son:

A 1.2 Índices ISE, IAE e ITAE

ISE (*Integral Square Error*): La integral del cuadrado del error es un índice que permite identificar excesivos sobretiros en la respuesta del sistema.

$$ISE = \int_0^T e^2(t) dt \quad (\text{A. 2})$$

IAE (*Integral Absolute Error*): La integral del valor absoluto del error es utilizado para el análisis de estudios de simulación en computadoras. En suma este índice indica un amortiguamiento razonable y una respuesta transitoria satisfactoria [10]. El ITAE, proporciona una mejor selectividad en los índices de desempeño, esto es, el mínimo valor de la integral es fácilmente discernible conforme varían los parámetros del sistema [10].

$$IAE = \int_0^T |e(t)| dt \quad (\text{A. 3})$$

ITAE (*Integral Time Absolute Error*): La integral del tiempo por el valor absoluto del error es un índice que permite cuantificar el desempeño tal que errores grandes en la respuesta transitoria contribuyen poco, pero enfatiza los errores que se presenten más adelante

$$ITAE = \int_0^T t|e(t)|dt \quad (\text{A. 4})$$

ITSE (*Integral Time Square Error*): La integral del tiempo por el valor cuadrado del error es un índice similar al ITAE con la diferencia que utiliza el cuadrado del valor del error.

$$ITSE = \int_0^T te^2(t)dt \quad (\text{A. 5})$$

El diseño basado en un solo índice de funcionamiento presenta ventajas y desventajas según el criterio seleccionado, así para grandes errores el valor de ISE incrementa en mayor grado que para IAE. ISE es favorable con respuestas con pequeños sobretiros pero permitirá grandes tiempos de asentamiento. En ITAE el tiempo aparece como factor por lo que penaliza fuertemente los errores que se presenten en tiempos posteriores pero ignora los errores presentes en tiempos tempranos. El diseño en base a ITAE permitirá tiempos de asentamiento pequeño, pero permitirá grandes sobretiros. Para un diseño basado en IAE proporciona resultados intermedios entre ISE e ITAE es decir para el caso del tiempo de asentamiento, este será no tan grande que el permitido por ISE y a la vez no tan pequeño que el permitido por ITAE [37]. Por lo que para este trabajo se analiza la respuesta del sistema en base a los índices: ISE, IAE e ITAE.

APÉNDICE B

B.1 MODELOS MATEMÁTICOS

B 1.1 Introducción

El modelo de sistema bajo prueba es un generador síncrono conectado a un bus infinito mediante una línea de transmisión.

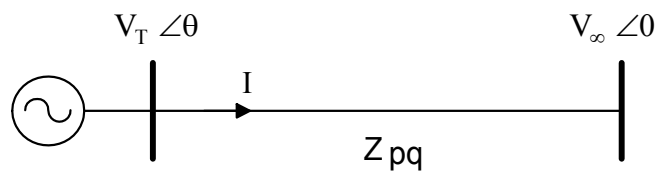


Figura B.1 Sistema máquina bus infinito.

El modelo general de la máquina síncrona se realiza en el marco ABC, el cual resulta en un modelo de octavo orden [1]. Queda completamente representado por tres ecuaciones diferenciales para los enlaces de flujo del estator; tres para los enlaces de flujo del rotor y dos para representar los transitorios electromecánicos. Este modelo contiene inductancias propias y mutuas que dependen de la posición del rotor, incluso con términos cosenoidales y sinusoidales al doble de la frecuencia fundamental [1]. Este hecho complica el computo de dichas ecuaciones por lo que generalmente se transforma el modelo en el marco dq (Transformación de Park [50]). El marco dq es un marco de referencia que gira a la velocidad del rotor. El eje d (eje directo) se encuentra situado sobre la cara polar del rotor y el eje q (eje de cuadratura) se encuentra 90° adelantado con respecto al eje d, esta convención de ejes es propuesto en [1]. La transformación de Park permite proyectar las variables del estator en el marco de referencia del rotor. Esta transformación transforma las inductancias variables en el marco ABC en constantes lo cual facilita el cómputo y solución del modelo. Para el planteamiento de las ecuaciones de tensión, se asume convención motor en los circuitos del rotor y convención generador en los circuitos del estator [1].

B 1.2 Modelo de Séptimo Orden en Variables de Park

El modelo de 7to orden en el marco dq (Modelo de Park) es llamado el modelo exacto [51]. El orden de modelo de la máquina síncrona está definido por el número de ecuaciones diferenciales y por ello el número de variables de estado que representan a la máquina en cuestión. El vector de estado para este modelo sin controles es $[\delta \ \dot{\delta} \ \psi_d \ \psi_q \ \psi_{fd} \ \psi_{kd} \ \psi_{kq}]$.

Partiendo del modelo exacto de 7to orden en el marco dq0 (Transformación de Park) en el sistema por unidad (PU) recíproco, en el cual todas las inductancias mutuas en los ejes d son iguales y de igual forma para el eje q. Es decir:

Para eje directo

$$L_{afd} = L_{akd} = L_{ad}$$

Para eje de cuadratura

$$L_{akq} = L_{aq}$$

Donde:

L_{afd} es la inductancia mutua entre el devanado del estator y el devanado de campo.

L_{akd} es la inductancia mutua entre el devanado del estator y el devanado de amortiguamiento en el eje directo.

L_{ad} es la inductancia mutua entre el rotor y el estator en el eje directo.

L_{akq} es la inductancia mutua entre el devanado del estator y el devanado de amortiguamiento en el eje de cuadratura.

L_{aq} es la inductancia mutua entre el rotor y el estator en el eje en cuadratura.

Las ecuaciones (B. 1) a (B. 5), describen la dinámica en variables de Park (flujos y corrientes) de la máquina síncrona. Este modelo considera dos ecuaciones diferenciales para los transitorios del estator; tres para los transitorios del rotor (considerando un solo devanado de amortiguamiento), y dos ecuaciones diferenciales para los transitorios de la parte electromecánica.

Ecuaciones de voltaje del estator en PU:

$$\begin{aligned} e_d &= -r_a i_d - \psi_q + \frac{d}{dt}(\psi_d) \\ e_q &= -r_a i_q + \psi_d + \frac{d}{dt}(\psi_q) \end{aligned} \quad (\text{B. 1})$$

Ecuaciones de voltaje en el rotor en PU:

$$\begin{aligned} e_{fd} &= r_{fd} i_{fd} + \frac{d}{dt}(\psi_{fd}) \\ 0 &= r_{kd} i_{kd} + \frac{d}{dt}(\psi_{kd}) \\ 0 &= r_{kq} i_{kq} + \frac{d}{dt}(\psi_{kq}) \end{aligned} \quad (\text{B. 2})$$

Ecuaciones de enlace de flujo en el estator en PU:

$$\begin{aligned} \psi_d &= -X_d i_d + X_{ad} i_{fd} + X_{ad} i_{kd} \\ \psi_q &= -X_q i_q + X_{aq} i_{kq} \end{aligned} \quad (\text{B. 3})$$

Ecuaciones de enlace de flujo en el rotor en PU:

$$\begin{aligned} \psi_{fd} &= -X_{ad} i_d + X_{ffd} i_{fd} + X_{ad} i_{kd} \\ \psi_{kd} &= -X_{ad} i_d + X_{ad} i_{fd} + X_{kkd} i_{kd} \\ \psi_{kq} &= -X_{aq} i_q + X_{kkq} i_{kq} \end{aligned} \quad (\text{B. 4})$$

Torque de entrehierro:

$$T_e = (\psi_d i_q - \psi_q i_d) \quad (\text{B. 5})$$

B 1.3 Modelo de 5to Orden en Variables de Park

Un modelo de quinto orden es llamado modelo aproximado [51]. Este modelo se obtiene al despreciar los transitorios del estator y la red, es decir, igualar a cero los términos $\frac{d}{dt}(\psi_d)$ y $\frac{d}{dt}(\psi_q)$ en (B. 1). De esta forma ahora se tiene:

$$\begin{aligned}
\frac{d}{dx}(\delta_r) &= \omega_0 s \\
M \frac{d}{dx}(s) &= -k_d + T_m - T_e \\
\frac{d}{dt}(\psi_{fd}) &= e_{fd} - r_{fd} i_{fd} \\
\frac{d}{dt}(\psi_{kd}) &= -r_{kd} i_{kd} \\
\frac{d}{dt}(\psi_{kq}) &= -r_{kq} i_{kq}
\end{aligned} \tag{B. 6}$$

$$\begin{aligned}
e_d &= -r_a i_d - \psi_q \\
e_q &= -r_a i_q + \psi_d \\
T_e &= (\psi_d i_q - \psi_q i_d) / 2
\end{aligned} \tag{B. 7}$$

$$\begin{aligned}
\psi_d &= X_d i_d + X_{ad} i_{fd} + X_{ad} i_{kd} \\
\psi_q &= X_q i_q + X_{aq} i_{kq}
\end{aligned} \tag{B. 8}$$

$$\begin{aligned}
\psi_{fd} &= -X_{ad} i_d + X_{ffd} i_{fd} + X_{ad} i_{kd} \\
\psi_{kd} &= -X_{ad} i_d + X_{ad} i_{fd} + X_{kkd} i_{kd} \\
\psi_{kq} &= -X_{aq} i_q + X_{kkq} i_{kq}
\end{aligned} \tag{B. 9}$$

En donde las ecuaciones de tensión para el estator en (B. 7) son algebraicas para este modelo. En lo siguiente se desarrolla el modelo de 5to orden en el sistema por unidad no recíproco.

B 1.4 Modelo de 5to Orden en el Sistema no Recíproco

Para este modelo se requiere realizar cambios de variables. Por lo que las siguientes ecuaciones son utilizadas:

$$e'_q = \frac{X_{ad}}{X_{ffd}} \psi_{fd} \tag{B. 10}$$

$$e''_q = \frac{X_{ad}}{X_{kkd}} \psi_{kd} \tag{B. 11}$$

$$e''_d = -\frac{X_{aq}}{X_{kkq}} \psi_{kq} \tag{B. 12}$$

$$e_{q1} = X_{ad} i_{fd} \tag{B. 13}$$

$$e_{q2} = X_{ad} \psi_{kd} \tag{B. 14}$$

$$e_{d2} = X_{aq} \psi_{kq} \tag{B. 15}$$

$$V_{fd} = \frac{X_{ad}}{r_{fd}} e_{fd} \quad (\text{B. 16})$$

$$X_d - X'_d = \frac{X_{ad}^2}{X_{ffd}} \quad (\text{B. 17})$$

$$X_q - X''_q = \frac{X_{aq}^2}{X_{kkq}} \quad (\text{B. 18})$$

$$X_d - X_d'' = \frac{X_{ad}^2(X_{ffd} + X_{kkd} - 2X_{ad})}{X_{ffd}X_{kkd} - X_{ad}^2} \quad (\text{B. 19})$$

$$X'_d - X_d'' = \frac{X_{ad}^2(X_{ffd} - X_{ad})}{X_{ffd}(X_{ffd}X_{kkd} - X_{ad}^2)} \quad (\text{B. 20})$$

$$T'_{d0} = \frac{X_{ad} + X_{fd}}{r_{fd}} = \frac{X_{ffd}}{r_{fd}} \quad (\text{B. 21})$$

$$T''_{d0} = \left(\frac{1}{r_{kd}} \right) \left(X_{kd} + \frac{X_{ad}X_{fd}}{X_{ffd}} \right) \quad (\text{B. 22})$$

$$T''_{q0} = \frac{X_{kq} + X_{aq}}{r_{kq}} = \frac{X_{kkq}}{r_{kq}} \quad (\text{B. 23})$$

En el modelo en el sistema no recíproco, las ecuaciones diferenciales para los transitorios electromecánicos no se ven afectadas.

Ecuación de flujo del devanado de campo

Tomando la ecuación del enlace de flujo del devanado de campo en (B. 9) y multiplicando ambos lados de la ecuación por $\frac{X_{ad}}{X_{ffd}}$

$$\begin{aligned} \frac{X_{ad}}{X_{ffd}} (\psi_{fd}) &= -\frac{X_{ad}}{X_{ffd}} (X_{ad}i_d) + \frac{X_{ad}}{X_{ffd}} (X_{ffd}i_{fd}) \\ \frac{X_{ad}}{X_{ffd}} (\psi_{fd}) &= -\frac{X_{ad}^2}{X_{ffd}} i_d + X_{ad}i_{fd} \end{aligned}$$

Sustituyendo (B. 10); (B. 13) y (B. 14) en la ecuación anterior por lo que obtenemos:

$$e'_q = -(X_d - X'_d)i_d + e_{q1} \quad (\text{B. 24})$$

Además

$$e_{q1} = e'_q + (X_d - X'_d)i_d \quad (\text{B. 25})$$

Para simplificar se asume que $X_{kkd} \approx X_{ad}$. De esta forma desarrollando (B. 17) y (B. 19) tenemos:

$$\frac{X_d - X'_d}{X_d - X''_d} = \left(\frac{X_{ad}^2}{X_{ffd}} \right) / \left(\frac{X_{ad}^2(X_{ffd} + X_{kkd} - 2X_{ad})}{X_{ffd}X_{kkd} - X_{ad}^2} \right) = \left(\frac{X_{ad}^2}{X_{ffd}} \right) / \left(\frac{X_{ad}^2(X_{ffd} - X_{ad})}{X_{ad}(X_{ffd} - X_{ad})} \right)$$

$$\frac{X_d - X'_d}{X_d - X''_d} = \left(\frac{X_{ad}^2}{X_{ffd}} \right) / \left(\frac{X_{ad}(X_{ffd} - X_{ad})}{(X_{ffd} - X_{ad})} \right) = (X_{ad}(X_{ffd} - X_{ad})) / (X_{ffd}(X_{ffd} - X_{ad}))$$

Por lo que:

$$\frac{X_d - X'_d}{X_d - X''_d} = \frac{X_{ad}}{X_{ffd}} \quad (\text{B. 26})$$

Sustituyendo ahora (B. 26) en (B. 24)

$$e'_q = -(X_d - X'_d)i_d + e_{q1} + \frac{X_d - X'_d}{X_d - X''_d} e_{q2} \quad (\text{B. 27})$$

Tomando ahora de (B. 6), la ecuación diferencial del enlace de flujo del campo y multiplicando ambos lados de la ecuación por $\frac{X_{ad}}{X_{ffd}}$

$$\left(\frac{X_{ad}}{X_{ffd}} \right) \frac{d}{dt} (\psi_{fd}) = \left(\frac{X_{ad}}{X_{ffd}} \right) e_{fd} - \left(\frac{X_{ad}}{X_{ffd}} \right) r_{fd} i_{fd}$$

Multiplicando el término del voltaje de campo en el lado derecho de la ecuación por $\frac{r_{fd}}{r_{fd}}$

$$\frac{d}{dt} \left(\frac{X_{ad}}{X_{ffd}} \psi_{fd} \right) = \left(\frac{r_{fd} X_{ad}}{X_{ffd} r_{fd}} \right) e_{fd} - \left(\frac{r_{fd}}{X_{ffd}} \right) X_{ad} i_{fd} = \left(\frac{r_{fd}}{X_{ffd}} \right) \left(\frac{X_{ad}}{r_{fd}} e_{fd} - X_{ad} i_{fd} \right)$$

Sustituyendo entonces (B. 10); (B. 13); (B. 16) y (B. 17) y (B. 21) en la ecuación anterior

$$\frac{d}{dt} (e'_q) = \left(\frac{1}{T'_{d0}} \right) (V_{fd} - e_{q1}) \quad (\text{B. 28})$$

Sustituyendo ahora (B. 25) en (B. 28)

$$\frac{d}{dt} (e'_q) = \frac{1}{T'_{d0}} [V_{fd} - e'_q - (X_d - X'_d)i_d] \quad (\text{B. 29})$$

Enlace de flujo en el devanado de amortiguamiento

Tomando de (B. 9), la ecuación correspondiente al enlace de flujo en el devanado de amortiguamiento en eje directo y multiplicándolo en ambos lados por $\frac{X_{ad}}{X_{kkd}}$

$$\begin{aligned} \left(\frac{X_{ad}}{X_{kkd}} \right) \psi_{kd} &= - \left(\frac{X_{ad}}{X_{kkd}} \right) X_{ad} i_d + \left(\frac{X_{ad}}{X_{kkd}} \right) X_{ad} i_{fd} + \left(\frac{X_{ad}}{X_{kkd}} \right) X_{kkd} i_{kd} \\ \left(\frac{X_{ad}}{X_{kkd}} \right) \psi_{kd} &= - \frac{X_{ad}^2}{X_{kkd}} i_d + \frac{X_{ad}^2}{X_{kkd}} i_{fd} + X_{ad} i_{kd} \end{aligned}$$

Sustituyendo (B. 11) y (B. 14) en la ecuación anterior

$$e''_q = - \frac{X_{ad}^2}{X_{kkd}} i_d + \frac{X_{ad}^2}{X_{kkd}} i_{fd} + e_{q2} \quad (\text{B. 30})$$

Considerando $X_{kkd} \approx X_{ad}$ de (B. 19) se obtiene

$$X_d - X''_d = \frac{X_{ad}^2 (X_{ffd} + X_{kkd} - 2X_{ad})}{X_{ffd} X_{kkd} - X_{ad}^2} = \frac{X_{ad}^2 (X_{ffd} - X_{ad})}{X_{ad} (X_{ffd} - X_{ad})} = X_{ad}$$

De esta forma sustituyendo (B. 13); (B. 17) y (B. 19) en (B. 30) y como $X_{kkd} \approx X_{ad}$

$$e''_q = -(X_d - X''_d)i_d + e_{q1} + e_{q2} \quad (\text{B. 31})$$

Tomando de (B. 6), la ecuación diferencial para el enlace de flujo en el devanado de amortiguamiento en el eje directo, y multiplicando ambos lados por $\frac{X_{ad}}{X_{kkd}}$.

$$\left(\frac{X_{ad}}{X_{kkd}}\right) \frac{d}{dt}(\psi_{kd}) = -\left(\frac{X_{ad}}{X_{kkd}}\right) r_{kd} i_{kd}$$

Sustituyendo (B. 11) y (B. 14) en la ecuación anterior

$$\frac{d}{dt}(e_q'') = -\left(\frac{r_{kd}}{X_{kkd}}\right) e_{q2} \quad (\text{B. 32})$$

Para simplificar aún más la ecuación (B. 32), se asume de nueva cuenta que $X_{kkd} \approx X_{ad}$

Se demostrara que $\left(\frac{1}{T''_{d0}}\right) \left(\frac{X'_d - X''_d}{X_d - X''_d}\right) = \frac{r_{kd}}{X_{kkd}}$

$$T''_{d0} = \left(\frac{1}{r_{kd}}\right) \left(X_{kd} + \frac{X_{ad} X_{fd}}{X_{ffd}}\right) = \left(\frac{1}{r_{kd}}\right) \left(\frac{X_{ad} X_{ffd} + X_{ad} X_{fd}}{X_{ffd}}\right)$$

Como la inductancia propia es igual a la inductancia mutua más la inductancia de dispersión, es decir para la inductancia del devanado de amortiguamiento tenemos: $X_{kkd} = X_{kd} + X_{ad}$; además $X_{kd} = X_{kkd} - X_{ad}$. Y de igual forma $X_{ffd} = X_{fd} + X_{ad}$.

$$T''_{d0} = \left(\frac{1}{r_{kd}}\right) \left(\frac{(X_{kkd} - X_{ad})(X_{fd} + X_{ad}) + X_{ad} X_{fd}}{X_{ffd}}\right)$$

Debido a $X_{kkd} \approx X_{ad}$, puede considerarse a $(X_{kkd} - X_{ad}) = 0$, entonces:

$$T''_{d0} = \frac{r_{fd} X_{ffd}}{X_{kkd} X_{fd}}$$

De (B. 20)

$$X'_d - X''_d = \frac{X_{ad}^2 (X_{ffd} - X_{ad}^2)}{X_{ffd} (X_{ffd} X_{kkd} - X_{ad}^2)} = \frac{X_{ad}^2 (X_{ffd} - X_{ad}^2)}{X_{ffd} X_{ad} (X_{ffd} - X_{ad})}$$

$$X'_d - X''_d = \frac{X_{ad}^2 (X_{ffd} - X_{ad}^2)}{X_{ffd} X_{ad}} = \frac{X_{ad} (X_{fd} + X_{ad} - X_{ad})}{X_{ffd}}$$

$$X'_d - X''_d = \frac{X_{ad} X_{fd}}{X_{ffd}}$$

Recordando que $X_d - X''_d = X_{ad}$

De aquí

$$\left(\frac{1}{T''_{d0}}\right) \left(\frac{X'_d - X''_d}{X_d - X''_d}\right) = \frac{r_{kd}}{X_{kkd}} \quad (\text{B. 33})$$

Para hallar e_{q1} y e_{q2} , se resuelven en forma simultánea (B. 31) y (B. 27) es decir

$$e_q' = -(X_d - X'_d)(i_d) + e_{q1} + \left(\frac{X_d - X'_d}{X_d - X''_d}\right) e_{q2}$$

$$e_q'' = -(X_d - X''_d)(i_d) + e_{q1} + e_{q2}$$

Lo que conduce a

$$e_{q1} = e_q'' \left(\frac{X'_d - X_d}{X'_d - X''_d}\right) + e_q' \left(\frac{X_d - X''_d}{X'_d - X''_d}\right)$$

$$e_{q2} = (X_d - X''_d) i_d + (e_q'' - e_q') \left(\frac{X_d - X''_d}{X'_d - X''_d}\right) \quad (\text{B. 34})$$

Sustituyendo ahora los resultados obtenidos en (B. 33) y (B. 34) en (B. 32)

$$\frac{d}{dt}(e_q'') = -\left(\frac{1}{T''_{d0}}\right) \left(\frac{X'_d - X''_d}{X_d - X''_d}\right) \left[(X_d - X''_d) i_d + (e_q'' - e_q') \left(\frac{X_d - X''_d}{X'_d - X''_d}\right) \right]$$

$$\frac{d}{dt}(e_q'') = e_q' - (X'_d - X''_d) i_d - e_q'' \quad (\text{B. 35})$$

Enlace de flujo q en el Devanado de Amortiguamiento

Tomando de (B. 9), la ecuación correspondiente al enlace de flujo en el devanado de amortiguamiento en eje de cuadratura y multiplicándolo en ambos lados por $\frac{X_{ad}}{X_{kkq}}$

$$\left(\frac{X_{ad}}{X_{kkq}}\right)\psi_{kq} = -\left(\frac{X_{ad}}{X_{kkq}}\right)X_{aq}i_q + \left(\frac{X_{ad}}{X_{kkq}}\right)X_{kkq}i_{kq}$$

$$\left(\frac{X_{ad}}{X_{kkq}}\right)\psi_{kq} = -\left(\frac{X_{ad}^2}{X_{kkq}}\right)i_q + X_{ad}i_{kq}$$

Sustituyendo (B. 12); (B. 15) y (B. 18) en la ecuación anterior tenemos:

$$e_d'' = -(X_q - X''_q)i_q + e_{d2}$$

O también

$$e_{d2} = -e_d'' + (X_q - X''_q)i_q \quad (\text{B. 36})$$

Tomando ahora de (B. 6), la ecuación diferencial correspondiente al enlace de flujo del devanado de amortiguamiento en eje de cuadratura y multiplicando ambos lados de la ecuación por $\frac{X_{ad}}{X_{kkq}}$

$$\left(\frac{X_{ad}}{X_{kkq}}\right)\frac{d}{dt}(\psi_{kq}) = -\left(\frac{X_{ad}}{X_{kkq}}\right)r_{kq}i_{kq}$$

Realizando el cambio de variable dado en (B. 12); (B. 15) y (B. 23)

$$\frac{d}{dt}(-e_d'') = -\left(\frac{1}{T''_{q0}}\right)e_{d2}$$

De aquí se sustituye (B. 36) en la ecuación anterior, lo que resulta

$$\frac{d}{dt}(e_d'') = \frac{1}{T''_{q0}}\left((X_q - X''_q)i_q - e_d''\right) \quad (\text{B. 37})$$

Ecuaciones Electromecánicas

Las ecuaciones diferenciales (B. 29), (B. 35) y (B. 37) describen el comportamiento eléctrico. La parte mecánica queda definida mediante la ecuación de oscilación para la máquina síncrona, esta ecuación describe matemáticamente el balance de energía mecánica y eléctrica en la máquina síncrona.

$$J\frac{d\omega_m}{dt} = T_m - T_e \quad (\text{B. 38})$$

En donde, J es el momento de inercia combinado del generador y turbina; ω_m es la velocidad angular del rotor; T_m es el par mecánico de la turbina; y T_e es el par electromagnético del generador. Ambos pares son considerados positivos cuando la máquina opera dentro del régimen generador. La resta de pares en el lado derecho de (B. 38) produce la aceleración del rotor, por lo que es conocido como par de aceleración.

Una normalización de (B. 38) es hecha tomando una constante de inercia H en por unidad definida como:

$$H = \frac{1}{2}\frac{J\omega_{0m}^2}{VA_{base}} \quad (\text{B. 39})$$

En donde VA_{base} son los Volt- Ampere base seleccionados en el sistema por unidad. De (B. 39) $J = \frac{2H}{\omega_{0m}^2} VA_{base}$ por lo que (B. 38) resulta:

$$\frac{2H}{\omega_{0m}^2} VA_{base} \frac{d\omega_m}{dt} = T_m - T_e$$

De aquí reagrupando términos tenemos:

$$2H \frac{d(\omega_m/\omega_{0m})}{dt} = \frac{T_m - T_e}{VA_{base}/\omega_{0m}} \quad (\text{B. 40})$$

Desde que $T_{base} = VA_{base}/\omega_{0m}$ la ecuación anterior se encuentra ya expresada en por unidad. Y $\hat{\omega}_r = \omega_m/\omega_{0m} = \frac{\omega_r/P_f}{\omega_0/P_f} = \omega_r/\omega_0$ es la velocidad del rotor en por unidad. De esta forma (B. 40) se reescribe considerando que la velocidad del rotor, el par de aceleración se encuentran ya en por unidad, lo cual se indica mediante:

$$2H \frac{d\hat{\omega}_r}{dt} = \hat{T}_m - \hat{T}_e \quad (\text{B. 41})$$

Si consideramos δ como la posición angular del rotor con respecto a un eje de referencia que gira a la velocidad síncrona, entonces:

$$\delta = \omega_r t - \omega_0 t + \delta_0 \quad (\text{B. 42})$$

Donde ω_r es la velocidad angular del rotor en rad/s, ω_0 es la velocidad síncrona y δ_0 es la posición en $t = 0$. Derivando con respecto al tiempo:

$$\frac{d\delta}{dt} = \omega_r - \omega_0 = \Delta\omega_r \quad (\text{B. 43})$$

Y en por unidad

$$\Delta\hat{\omega}_r = \frac{\Delta\omega_r}{\omega_0} = \frac{1}{\omega_0} \frac{d\delta}{dt} \quad (\text{B. 44})$$

Además

$$\frac{d\delta^2}{dt^2} = \frac{d\omega_r}{dt} = \omega_0 \frac{d\hat{\omega}_r}{dt} = \omega_0 \frac{d\Delta\hat{\omega}_r}{dt} \quad (\text{B. 45})$$

Por lo que al sustituir (B. 43) en (B. 41) y añadiendo un término de amortiguamiento (que es igual a la suma de todos los pares que se oponen a la rotación de la máquina, tales como fricción en rodamientos, pérdidas magnéticas, etc.) multiplicado por la velocidad de desviamiento.

$$2H \frac{d\delta^2}{dt^2} = \hat{T}_m - \hat{T}_e - k_D \Delta\hat{\omega}_r \quad (\text{B. 46})$$

Donde $\Delta\hat{\omega}_r = s = \omega_r - \omega_0/\omega_0$ se conoce como el deslizamiento lo cual se indica mediante el símbolo s . Además $T_M = 2H$ se conoce como la constante mecánica de arranque [1], la cual indica el tiempo requerido para que aun valor de torque nominal se

acelere el rotor desde un estado estático hasta la velocidad nominal. Dicha constante es indicada en este trabajo mediante el símbolo M .

La ecuación (B. 46) puede expresarse mediante dos ecuaciones diferenciales de primer grado, para esto y con auxilio de (B. 43) y (B. 45):

$$\frac{1}{\omega_0} \frac{d^2 \delta}{dt^2} = \frac{1}{\omega_0} \frac{d}{dt} \left(\frac{d\delta}{dt} \right) = \frac{d}{dt} (\Delta \hat{\omega}_r)$$

Por lo que (B. 46) puede expresarse como:

$$M \frac{ds}{dt} = \hat{T}_m - \hat{T}_e - k_D \Delta \hat{\omega}_r$$

Finalmente

$$\begin{aligned} \frac{d\delta}{dt} &= \omega_0 s \\ M \frac{ds}{dt} &= \hat{T}_m - \hat{T}_e - k_D s \end{aligned} \quad (\text{B. 47})$$

El diagrama de bloques resulta:

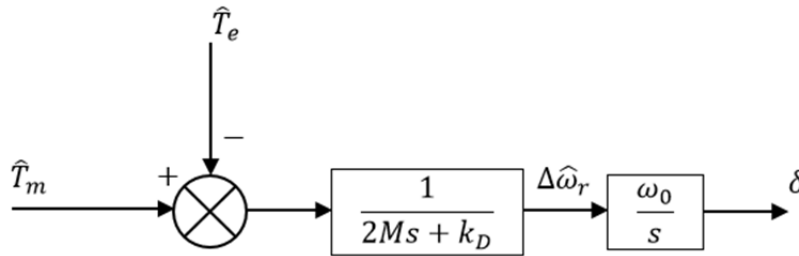


Figura B.2 Diagrama de bloques de la ecuación de oscilación de la máquina síncrona.

B 1.5 Modelo de la red

El modelo de red, es decir del sistema de transmisión se representa mediante parámetros concentrados en estado estable. Para el caso de una máquina simple bus infinito se considera para la línea de transmisión, una resistencia y reactancia concentradas, las cuales representan la resistencia y reactancia totales considerando el transformador y la línea de transmisión entre el generador y el bus infinito.

$$\begin{aligned} e_d &= e_{bus} \text{sen}(\delta) + r_{ex} i_d + x_{ex} \frac{d}{dt} i_d - x_{ex} i_q \\ e_q &= e_{bus} \text{cos}(\delta) + r_{ex} i_q + x_{ex} \frac{d}{dt} i_q + x_{ex} i_d \end{aligned}$$

Estas ecuaciones diferenciales representan los transitorios en la línea de transmisión, pero como el modelo de la máquina síncrona utilizado para este trabajo desprecia los transitorios del estator, los transitorios de la línea de transmisión también serán despreciados. Lo que nos conduce a dos ecuaciones algebraicas.

$$\begin{aligned} e_d &= e_{bus} \text{sen}(\delta) + r_{ex} i_d + x_{ex} i_q \\ e_q &= e_{bus} \text{cos}(\delta) + r_{ex} i_q + x_{ex} i_d \end{aligned} \quad (\text{B. 48})$$

B 1.6 Concepto Máquina Modificada

Las ecuaciones algebraicas para el voltaje en terminales de la máquina síncrona para este modelo de 5to orden, se obtiene al considerar los efectos subtransitorios en los enlaces de flujo en eje directo y de cuadratura [1]. De esta forma las ecuaciones resultan:

$$e''_d = e_d + r_a i_d - X''_q i_q$$

$$e''_q = e_q + r_a i_q + X''_d i_d \quad (\text{B. 49})$$

El concepto máquina modificada incluye el modelo de la línea de transmisión dentro del modelo de la máquina síncrona aquí planteado. Para esto se introducen las impedancias externas en las impedancias del estator de la máquina (armadura).

De esta forma el voltaje terminal se iguala al voltaje en el bus, por lo que sus componentes en sus respectivos ejes son idénticos, y es posible sustituir (B. 48) en (B. 49).

$$e''_d = e_{bus} \sin(\delta) + (r_a + r_{ex}) i_d + (x_{ex} - X''_q) i_q$$

$$e''_q = e_{bus} \cos(\delta) + (r_a + r_{ex}) i_q + (x_{ex} + X''_d) i_d \quad (\text{B. 50})$$

De (B. 50) se pueden despejar las variables i_d e i_q y resolver el sistema de dos ecuaciones. Con el objeto de encontrar el valor en cada iteración de las corrientes en eje directo y de cuadratura. Además se puede conocer el valor de la magnitud del voltaje en terminales mediante:

$$V_T^2 = V_d^2 + V_q^2 \quad (\text{B. 51})$$

Para el par eléctrico

$$T_e = e''_d i_d + e''_q i_q - (X''_d - X''_q) i_d i_q \quad (\text{B. 52})$$

B 1.7 Modelo del Sistema de Excitación

El sistema de excitación es el encargado de suministrar corriente directa al devanado de campo de la máquina síncrona. En este trabajo se utiliza un sistema de excitación del tipo AC rotatorio, el cual es representado mediante la siguiente ecuación:

$$T_x \frac{dV_f}{dt} = u - V_f \quad (\text{B. 53})$$

En donde V_f es el voltaje de campo, y u es la entrada de control, es decir el valor de la salida del regulador de voltaje o del controlador inteligente según sea el caso.

B 1.8 Modelo del Regulador de Voltaje

El modelo de regulador de Voltaje utilizado es del tipo ST1 (excitatriz con fuente de potencial a través de rectificadores controlados). En este tipo de reguladores, los rectificadores controlados proporcionan la corriente necesaria para el devanado de campo del generador. El regulador es alimentado a través de un transformador de potencial desde las terminales del generador y regulado mediante un rectificador controlado. Desde que la constante de tiempo del transformador de potencial es demasiado pequeña, esta constante es despreciada en el modelo. El máximo voltaje de excitación disponible se

encuentra relacionada directamente del nivel de voltaje en las terminales del generador. La **Figura B.3** muestra el diagrama de bloques del regulador tipo ST1.

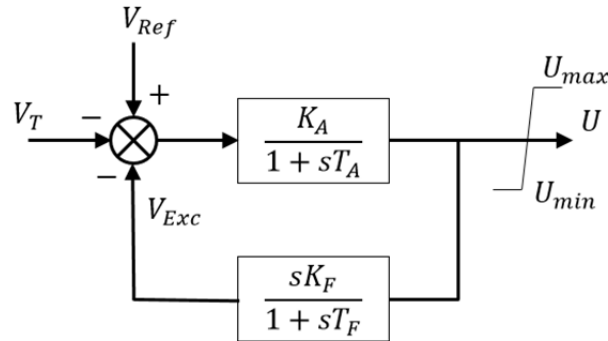


Figura B.3 Diagrama de bloques del regulador tipo ST1.

De este diagrama de bloques se deducen las siguientes ecuaciones diferenciales:

$$\begin{aligned} T_A \frac{du}{dt} &= K_A (V_{ref} - V_T - V_{Exc}) - u \\ T_F \frac{d}{dt} V_{Exc} &= K_F \frac{du}{dt} - V_{Exc} \end{aligned} \quad (\text{B. 54})$$

En donde u es la salida del regulador tipo ST1, el cual servirá de entrada de control para (B. 53).

B 1.9 Modelo del Gobernador y Turbina

La función primordial del gobernador es ejercer control sobre la velocidad y/o carga [1]. El modelo del gobernador utilizado es del tipo mecánico - hidráulico, el cual es idealizado, ya que no se considera el limitador de posición de la válvula de control, ni la función de transferencia que representa el relevador de velocidad. En base a estas consideraciones el diagrama de bloques es el siguiente:

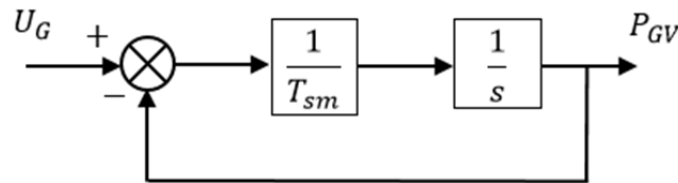


Figura B.4 Diagrama de bloques del gobernador de velocidad.

Este diagrama en lazo cerrado con retroalimentación unitaria puede ser expresado por un función de transferencia dada por

$$FT = \frac{G(s)}{1 + G(s)H(s)} \quad (\text{B. 55})$$

Donde $G(s)$ = Función de transferencia en lazo superior.

$H(s)$ = Función de transferencia en lazo de retroalimentación.

De esta forma el diagrama de bloques de la **Figura B.4** se reduce a:

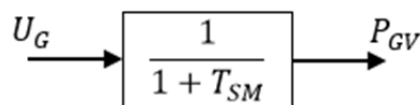


Figura B.5 Función de transferencia en lazo abierto del gobernador de velocidad.

El modelo de la turbina empleado es turbina térmica de simple recalentamiento modificada. El incremento o decremento de la potencia activa se logra mediante la manipulación de la variable U_g , que indica la entrada al gobernador, sin hacer control sobre el lazo primario. De esta forma el conjunto de ecuaciones diferenciales que representan el modelo del sistema gobernador - turbina resultan:

$$\begin{aligned} T_{SM} \frac{d}{dt} P_{GV} &= U_g - P_{GV} \\ T_{CH} \frac{d}{dt} P_{HP} &= P_{GV} - P_{HP} \\ T_{RH} \frac{d}{dt} T_m &= F_{HP} \frac{d}{dt} P_{HP} + P_{HP} - T_m \end{aligned} \quad (\text{B. 56})$$

B 1.10 Modelo Completo de 5to Orden de la Máquina Síncrona

En suma el modelo de 5to orden de la máquina síncrona utilizado en este trabajo, viene expresado por (B. 29), (B. 35), (B. 37), (B. 47), (B. 53), (B. 54) y (B. 56), de esta forma:

Generador y red

$$\begin{aligned} \frac{d\delta}{dt} &= \omega_0 s \\ M \frac{ds}{dt} &= \hat{T}_m - \hat{T}_e - k_D s \\ \frac{d}{dt} (e'_q) &= \frac{1}{T'_{d0}} [V_{fd} - e'_q - (X_d - X'_d) i_d] \\ \frac{d}{dt} (e''_q) &= e'_q - (X'_d - X''_d) i_d - e''_q \\ \frac{d}{dt} (e''_d) &= \frac{1}{T''_{q0}} ((X_q - X''_q) i_q - e''_d) \end{aligned} \quad (\text{B. 57})$$

Donde:

$$\begin{aligned} e''_d &= V_d + r_a i_d - X''_q i_q \\ e''_q &= V_q + r_a i_q + X''_d i_d \\ V_d &= V_{bus} \text{sen}(\delta) + r_e i_d - X_e i_q \\ V_q &= V_{bus} \text{cos}(\delta) + r_e i_q + X_e i_d \\ V_T^2 &= V_d^2 + V_q^2 \\ T_e &= e''_d i_d + e''_q i_q - (X''_d - X''_q) i_d i_q \end{aligned} \quad (\text{B. 58})$$

Sistema de Excitación

$$T_X \frac{dV_f}{dt} = u - V_f \quad (\text{B. 59})$$

Regulador de Voltaje

$$\begin{aligned} T_A \frac{du}{dt} &= K_A (V_{ref} - V_T - V_{Exc}) - u \\ T_F \frac{d}{dt} V_{Exc} &= K_F \frac{du}{dt} - V_{Exc} \end{aligned} \quad (\text{B. 60})$$

Turbina de Vapor y Gobernador

$$\begin{aligned}
 T_{SM} \frac{d}{dt} P_{GV} &= U_g - P_{GV} \\
 T_{CH} \frac{d}{dt} P_{HP} &= P_{GV} - P_{HP} \\
 T_{RH} \frac{d}{dt} T_m &= F_{HP} \frac{d}{dt} P_{HP} + P_{HP} - T_m
 \end{aligned} \tag{B. 61}$$

Este conjunto de ecuaciones representa el modelo de quinto orden de la máquina bus infinito operando con el regulador tipo ST1. Cuando el controlador inteligente es utilizado, las ecuaciones del regulador de voltaje no son ya necesarias y la salida del controlador inteligente se aplica sobre la entrada de control u en la ecuación del sistema de excitación.

Para la primera iteración se calculan las condiciones iniciales (ver sección B.2), pero para iteraciones posteriores se requiere conocer el valor de los voltajes subtransitorios en eje directo (e_d'') y cuadratura (e_q''); las corrientes en eje directo (i_d) y cuadratura (i_q); la Potencia Aparente, Activa, y Reactiva; así como el valor del par eléctrico (T_e); para ello se han de resolver por cada iteración el siguiente conjuntos de ecuaciones algebraicas, llamadas ecuaciones dinámicas [9].

Sustituyendo V_d en e_d'' y V_q en e_q'' en (B. 58)

$$\begin{aligned}
 e_d'' &= V_{bus} \sin(\delta) + (r_e + r_a)i_d - (X_e + X_q'')i_q \\
 e_q'' &= V_{bus} \cos(\delta) + (r_e + r_a)i_q + (X_e + X_d'')i_d
 \end{aligned}$$

Resolviendo este sistema de dos ecuaciones para las variables i_d e i_q , obtendremos:

$$i_d = \frac{(r_e + r_a)(e_d'' - V_{bus} \sin(\delta)) + (X_e + X_q'')(e_q'' - V_{bus} \cos(\delta))}{(r_e + r_a)^2 + (X_e + X_d'')(X_e + X_q'')} \tag{B. 62}$$

$$i_q = \frac{(r_e + r_a)(e_q'' - V_{bus} \cos(\delta)) - (X_e + X_d'')(e_d'' - V_{bus} \sin(\delta))}{(r_e + r_a)^2 + (X_e + X_d'')(X_e + X_q'')} \tag{B. 63}$$

Además:

$$I = i_q + ji_d \tag{B. 64}$$

Para los voltajes es necesario despejar V_d y V_q de (B. 58)

$$V_d = e_d'' - r_a i_d - X_q'' i_q \tag{B. 65}$$

$$V_q = e_q'' - r_a i_q + X_d'' i_d \tag{B. 66}$$

Para la Potencia

$$S = VI \tag{B. 67}$$

$$P = \text{Re}(S) \tag{B. 68}$$

$$Q = -\text{Im}(S) \tag{B. 69}$$

El par eléctrico viene en estado estable viene dado como en (B. 58)

$$T_e = e_d'' i_d + e_q'' i_q \tag{B. 70}$$

En suma de (B. 62) a (B. 70) son las ecuaciones algebraicas que deberán ser resueltas para cada iteración $k+1$ para la solución del modelo de la máquina síncrona.

B 1.11 Determinación de las Condiciones Iniciales del Sistema Máquina Síncrona Bus Infinito

La simulación de la máquina bus infinito se realiza a partir de un punto de operación definido, esto da lugar al cálculo de las condiciones iniciales para las variables del vector de estado del modelo con controles. Es decir:

$$[\delta(t_0) \quad s(t_0) \quad e'_q(t_0) \quad e''_q(t_0) \quad e''_d(t_0) \quad V_f(t_0) \quad u(t_0) \quad V_{Exc}(t_0) \quad P_{GV}(t_0) \quad P_{HP}(t_0) \quad T_m(t_0)]^T \quad (\text{B. 71})$$

B 1.12 Condiciones Iniciales de la Red

El voltaje de bus, magnitud del voltaje terminal, ángulo del voltaje terminal, potencia activa y la potencia reactiva, son datos conocidos de un estudio previo de flujos de potencia. En forma compleja los voltajes de bus y terminal se expresan como:

$$V_{bus} = |V_{bus}| \angle 0 \quad V_{Term} = |V_{Term}| \angle \theta = V_{Term_real} + jV_{Term_imag}$$

Donde:

$$\begin{aligned} V_{Term_real} &= |V_{Term}| \cos(\theta) \\ V_{Term_imag} &= |V_{Term}| \sin(\theta) \end{aligned}$$

Para la potencia eléctrica tenemos

$$S = VI = P - jQ$$

Por lo que la corriente de armadura de la máquina síncrona puede calcularse mediante

$$S = V_{Term} I_a$$

Con lo que I_a puede expresarse como

$$I_a = \frac{S}{V_{Term}} \quad (\text{B. 72})$$

Donde

$$S = \sqrt{P^2 + Q^2} \quad (\text{B. 73})$$

Y el ángulo de S esta dado por:

$$\phi = \tan^{-1}(Q/P) \quad (\text{B. 74})$$

B 1.13 Condiciones Iniciales del Generador

Para el generador es necesario determinar las condiciones iniciales de las variables ángulo de carga, deslizamiento, voltaje transitorio y subtransitorio en eje de cuadratura, voltaje subtransitorio en eje directo de (B. 71) así como el par mecánico y eléctrico. Para esto se igualan a cero las derivadas con respecto al tiempo en (B. 57)

$$s = 0 \quad (\text{B. 75})$$

$$T_e = T_m \quad (\text{B. 76})$$

$$V_{fd} = e'_q + (X_d - X'_d)i_d \quad (\text{B. 77})$$

$$e'_q = (X'_d - X''_d)i_d + e''_q \quad (\text{B. 78})$$

A partir del modelo de la máquina síncrona bus infinito para un modelo subtransitorio, se calculan los voltajes subtransitorios en eje directo y cuadratura mediante:

$$e''_d = V_{Ter} \text{sen}(\delta) + r_a i_d - X''_q i_q \quad (\text{B. 79})$$

$$e''_q = V_{Ter} \cos(\delta) + r_a i_q + X''_d i_d \quad (\text{B. 80})$$

El par mecánico es:

$$T_e = e''_d i_d + e''_q i_q \quad (\text{B. 81})$$

Para el cálculo del ángulo de carga se hace uso del diagrama vectorial de la máquina síncrona en estado subtransitorio.

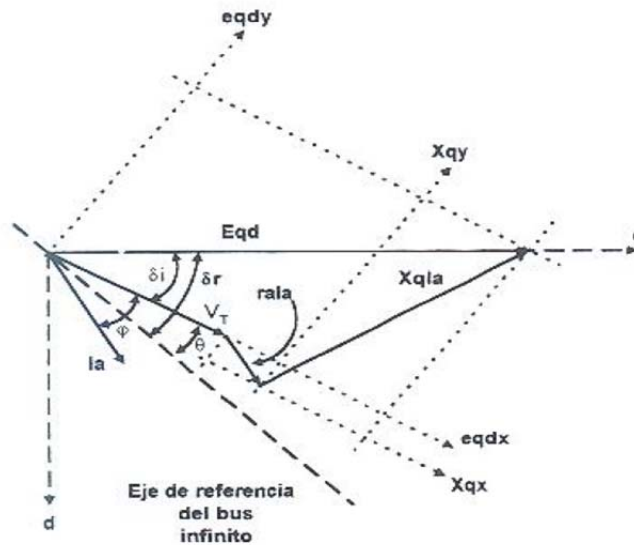


Figura B.6 Diagrama vectorial. Tomado de [9].

Tomando el eje x sobre el vector del voltaje terminal con lo que $\overline{V_{Ter}} = |V_{Ter}| \angle 0$; $\overline{I_a} = |I_a| \angle -\varphi$; las componentes del vector E_{qd} resultan:

$$e_{qdx} = V_{Ter} + r_a i_a \cos(\phi) + X_q i_a \text{sen}(\phi) \quad (\text{B. 82})$$

$$e_{qdy} = -r_a i_a \text{sen}(\phi) + X_q i_a \cos(\phi) \quad (\text{B. 83})$$

A partir de estos componentes

$$\delta_i = \tan^{-1}(e_{qdy}/e_{qdx}) \quad (\text{B. 84})$$

Por lo que

$$\delta_r = \theta + \delta_i \quad (\text{B. 85})$$

B 1.14 Condiciones Iniciales del Sistema de Excitación y Regulador de Voltaje

Para la obtención de las condiciones iniciales del sistema de excitación, se hace cero la derivada en (B. 59) con lo que

$$V_f = u \quad (\text{B. 86})$$

Siendo u la entrada de control. De esta forma cuando se utiliza el IT2FLC la condición inicial resulta únicamente en (B. 86).

Cuando se utiliza el regulador de voltaje ST1 se hace necesario, además de (B. 86), determinar las condiciones iniciales en (B. 60) por lo que:

$$0 = K_A(V_{ref} - V_T - V_{Exc}) - u$$

Como

$$V_{ref} = \frac{u}{K_A} + V_T \quad (\text{B. 87})$$

Por cuestiones de protección se utiliza un limitador en el voltaje de excitación por lo que:

$$V_{ref} = \frac{u}{K_A} + V_T \quad (\text{B. 88})$$

B 1.15 Condiciones Iniciales del Gobernador de Velocidad y Turbina de Vapor

Igualando a cero las derivadas en (B.61) tenemos:

$$\begin{aligned} 0 &= U_g - P_{GV} \\ 0 &= P_{GV} - P_{HP} \\ 0 &= 0 + P_{HP} - T_m \end{aligned}$$

Lo que conlleva a:

$$\begin{aligned} P_{GV} &= U_g \\ P_{HP} &= P_{GV} \\ T_m &= P_{HP} \end{aligned} \quad (\text{B. 89})$$

B 1.16 Datos del Sistema

Los datos del sistema máquina bus infinito, con un generador de 330 Kv y 400 Mw [8], son los siguientes:

Tabla B.1 Datos del Sistema bajo prueba.			
Variable	Valor	Variable	Valor
ω_0	377.0	X_e	0.115
M	5.5294	r_e	0.024
k_d	3.0	T_{SM}	0.1
T'_{d0}	5.66	T_{CH}	0.15
T''_{d0}	0.041	T_{RH}	0.15
T''_{q0}	0.065	F_{HP}	0.33
X_d	1.904	K_A	400.0
X'_d	0.312	T_A	0.02
X''_d	0.266	K_P	0.008
X_q	1.881	T_P	1.0
X''_q	0.260	T_X	0.025
r_a	0.0		

APÉNDICE C

C.1 ALGORITMO KM Y ALGORITMO EKM

Como se muestra en la sección 3.7 se deja entrever que ambos algoritmos convergen al mismo resultado. Es decir ambos algoritmos propondrán el mismo \dot{u} ante una dupla $[e \Delta e]$. La diferencia radica en el número de iteraciones ocupadas por cada algoritmo para converger al resultado y con esto el tiempo de cómputo requerido. Esta ventaja del EKM sobre el KM se debe a la mejor inicialización del método para converger al resultado.

La figura **Figura C.1** muestra el resultado de las iteraciones ocupadas por el IT2 FLC con el algoritmo de KM y el EKM. Las pruebas seleccionadas son con el controlador tipo-2 con funciones de membresía triangulares y Gaussianas tipo-2, con 9 y 25 reglas de conocimiento con el generador conectado al bus infinito mediante una línea transmisión corta y falla severa de corto circuito en el bus con duración de 10 ciclos. Estas fueron expuestas en la sección 4.4.

Como puede observarse, el algoritmo EKM utiliza para el cálculo de \underline{y} y \bar{y} entre 1 y 3 iteraciones en las simulaciones. Además que cuando se utilizan funciones de membresía Gaussianas tipo-2 se ocupa 1 sola iteración la mayor parte del tiempo. 3 iteraciones son utilizadas en el momento del disturbio y en los ciclos posteriores. Por su parte el KM utiliza entre 3 y 4 iteraciones en las simulaciones presentadas. Y solo 2 iteraciones en algunos instantes. Este desempeño de ambos algoritmos impacta en forma directa al tiempo de cómputo. Algo que es de suma importancia, por ejemplo, en aplicaciones de control en tiempo real basado en microcontroladores.

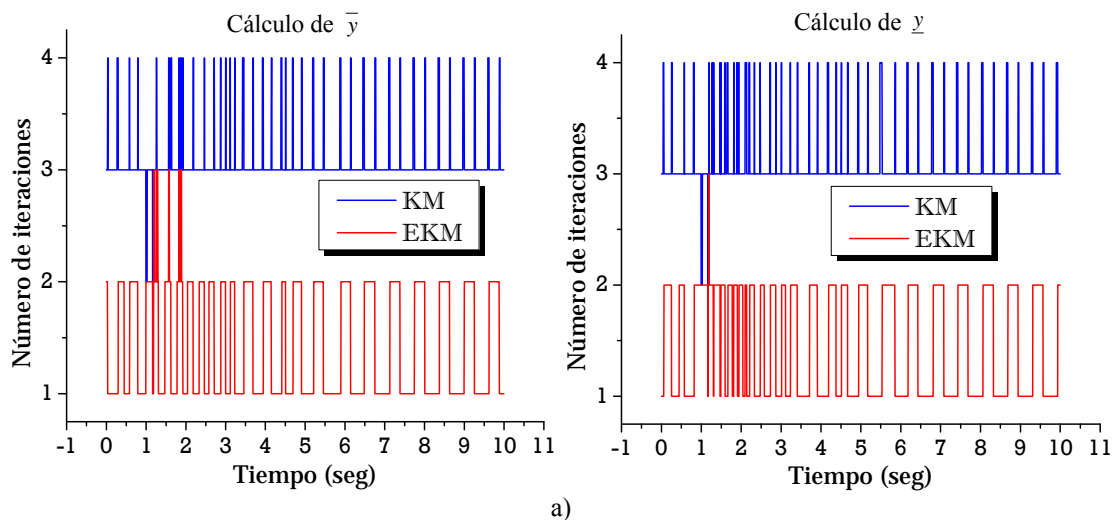
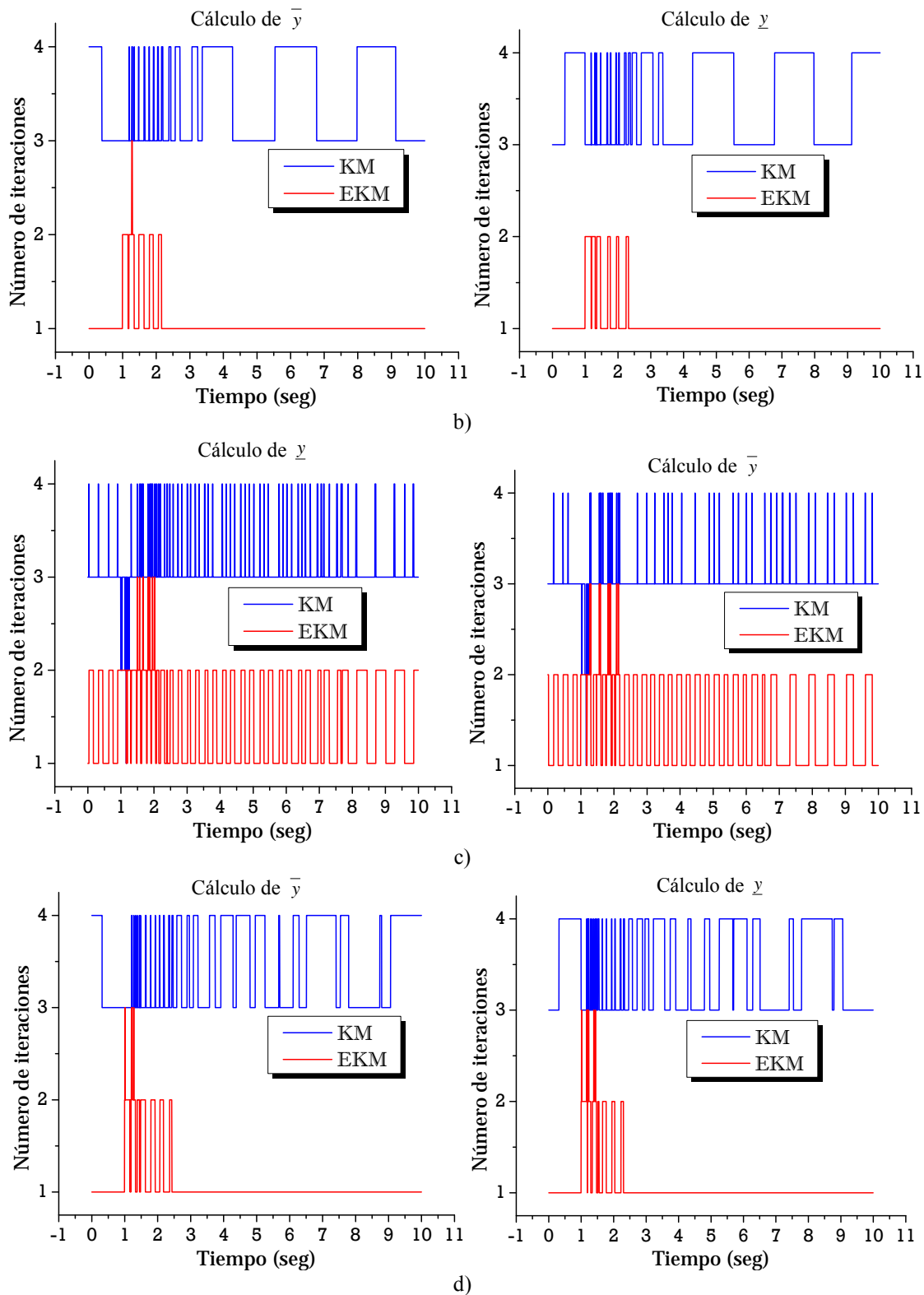


Figura C.1 Iteraciones ocupadas por KM y EKM ante distintos casos: a) funciones de membresía triangulares tipo-2 y 9 reglas; b)) funciones de membresía Gaussianas tipo-2 y 9 reglas; c)) funciones de membresía triangulares tipo-2 y 25 reglas; d)) funciones de membresía Gaussianas tipo-2 y 25 reglas.



Continuación de Figura C.1 Iteraciones ocupadas por KM y EKM ante distintos casos: a) funciones de membresía triangulares tipo-2 y 9 reglas; b)) funciones de membresía Gaussianas tipo-2 y 9 reglas; c)) funciones de membresía triangulares tipo-2 y 25 reglas; d)) funciones de membresía Gaussianas tipo-2 y 25 reglas.

La siguiente tabla muestra el promedio de iteraciones y el tiempo de cómputo utilizado por el IT2 FLC en las simulaciones anteriores. El IT2 FLC con funciones de membresía Gaussianas tipo-2 en intervalos y 9 reglas de conocimiento difuso, muestra el mejor desempeño en tiempo y número de iteraciones.

Tabla C.1 Comparación del desempeño de algoritmos reducción de tipo para el IT2 FLC.						
T2 FLC con: Funciones de membresía/número de reglas	Algoritmo					
	KM			EKM		
	Promedio de iteraciones		Tiempo de cómputo	Promedio de iteraciones		Tiempo de cómputo
	\underline{y}	\bar{y}		\underline{y}	\bar{y}	
Triangular 2/9	3	3	6.3774	1	1	4.2598
Gaussiana/9	3	3	5.8527	1	1	4.2393
Triangular 2/25	3	3	6.7381	1	2	5.4451
Gaussiana/25	3	3	6.9931	1	1	5.3056

El cálculo del tiempo de cómputo fue realizado en Matlab mediante los comandos tic y toc de la siguiente manera:

```
TSTART = tic;
      :
Simulación del control
      :
TSTOP = toc(TSTART);
```


APÉNDICE D

D.1 BÚSQUEDA POR TABÚ

D 1.1 Introducción

Se entiende por *heurística*, a una técnica inteligente que utiliza conocimiento experto de una tarea, para la realización de la misma [18]. El término *metaheurística*, fue propuesto por primera vez por Glover en el mismo artículo en el cual presentó el término búsqueda por tabú. Glover definió metaheurística como una estrategia maestra que guía y modifica procesos heurísticos para producir soluciones más allá de las que normalmente son generadas en la búsqueda de un óptimo local.

El algoritmo de búsqueda Tabú fue desarrollado por Glover y Hansen en 1986, con el propósito de resolver problemas de optimización combinatoria. El algoritmo es del tipo iterativo y se caracteriza por el uso de una memoria flexible [53]. La búsqueda Tabú es una metaheurística la cual guía una búsqueda local heurística a explorar el espacio de solución más allá del óptimo local [52]. El proceso de búsqueda se basa en operaciones denominadas movimientos sobre el vecindario definido para una solución dada.

La búsqueda tabú ha sido aplicada a problemas que implican decisiones, sin la necesidad de transformarlos en formulaciones matemáticas. Problemas de optimización en los cuales se busca maximizar o minimizar según el caso, una función objetivo $f(x)$ para toda $x \in \mathbf{X}$, donde $f(x)$ es una función del tipo lineal o no lineal, y el conjunto \mathbf{X} es el espacio de búsqueda, contiene ciertas restricciones sobre el vector de decisión x . En general, la búsqueda tabú inicia con una búsqueda local ordinaria sobre un vecindario de búsqueda definido. El algoritmo procederá en forma iterativa desde un punto inicial (solución) hasta otro punto en el cual se ha satisfecho un criterio de terminación o paro. Cada $x \in \mathbf{X}$ tiene un vecindario asociado $\mathbf{N}(x) \subset \mathbf{X}$, en donde los elementos del vecindario $\mathbf{N}(x)$ son puntos o elementos cercanos y/o alrededor del punto actual x , y cada solución $x' \in \mathbf{N}(x)$ se alcanza desde una solución x , mediante operaciones llamadas *movimientos*.

D 1.2 Estructura de Búsqueda por Tabú

La estructura de la memoria de búsqueda tabú contiene cuatro propiedades principales, consistentes en la propiedad de:

- Reciente
- Frecuencia
- Calidad
- Influencia

La memoria basada en lo reciente y la memoria basada en la frecuencia se complementan entre sí. La calidad se refiere a la habilidad para diferenciar sobre buenas soluciones visitadas durante la búsqueda. En este sentido, la memoria puede ser utilizada para identificar elementos que producen buenas soluciones o que conducen a caminos que conducen a ellas. Por último la influencia, considera el impacto de los elementos seleccionados durante la búsqueda.

La búsqueda por tabú utiliza dos tipos de memoria clasificadas como: *memoria corto plazo* y *memoria largo plazo* [12][52]. El efecto de ambos tipos de memoria es la de modificar

en alguna forma el vecindario $\mathbf{N}(x)$ de la actual solución x , y el vecindario modificado se denota mediante $\mathbf{N}^*(x)$. En lo siguiente se definen ambos tipos de memoria.

D 1.2.1 Memoria a Corto Plazo

La memoria a corto plazo, es comúnmente conocida como *memoria basada en lo reciente* [12]. La función de la memoria a corto plazo dentro de TS, es en forma semejante a la memoria a corto plazo en los seres humanos⁹. La función de esta memoria, es la de mantener un registro de los cambios realizados en la vecindad $\mathbf{N}(x)$ e identificar mediante atributos, las soluciones que han sido modificadas durante el pasado reciente, y así generar una vecindad modificada $\mathbf{N}^*(x)$. Esto permite mediante los atributos (etiquetas *tabú activo*), que ciertas soluciones del pasado reciente formen parte de $\mathbf{N}^*(x)$ y de esta forma algunas soluciones sean visitadas. Para su función, la memoria a corto plazo se auxilia de dos tipos de memorias denominadas, *memoria explícita* y *memoria atributiva*, aunque contiene también otros elementos. Los componentes que forman parte de la memoria a corto plazo son:

Base y atributo tabú: Base tabú, es el valor base para realizar prohibiciones a soluciones visitadas recientemente o soluciones con pésimo valor en su función objetivo, de la vecindad $\mathbf{N}(x)$ de la solución actual x . El valor de la base tabú puede ser modificado mediante la aplicación de multas e incentivos, lo que permite que la base tabú sea dinámica durante el proceso de búsqueda.

Por otra parte, el atributo tabú, es el valor por medio del cual, una solución sea evaluada de nueva cuenta, o que elemento no forme parte de la solución élite. Esta valor puede ser modificado únicamente mediante una nueva asignación de la base tabú a dicha solución.

Memoria Explícita: Esta memoria es la encargada de registrar las soluciones x de la vecindad $\mathbf{N}(x)$ que fueron visitadas durante la búsqueda, esto con el fin de evitar ciclos durante la búsqueda. De esta forma se tendrá una lista tabú, la cual contiene n soluciones, es decir $T = \{x_1, x_2, \dots, x_n\}$. En la lista tabú se registran las soluciones de la vecindad $\mathbf{N}(x)$ que contienen un atributo tabú, lo que indica que son soluciones prohibidas de ser visitadas por el proceso de búsqueda, una solución se convierte en tabú mediante la asignación de atributo tabú debido a la base tabú. Entonces la vecindad modificada $\mathbf{N}^*(x)$ se obtiene de las soluciones de la vecindad $\mathbf{N}(x)$ que no se encuentre dentro de la lista tabú T .

Memoria atributiva: Esta memoria lleva el registro mediante una lista atributiva, en la cual se almacena el periodo de tiempo en que cada solución x contenida en T será tabú activo. Esto significa, el tiempo en el cual una solución no puede ser visitada durante el proceso de búsqueda. La dimensión de esta memoria, necesariamente debe ser igual a la dimensión de la lista tabú T . El número de listas atributivas debe ser igual al número de listas tabú T que se tengan en particular. En general, una solución $x_i \in \mathbf{N}(x)$ puede ser: tabú activo si $x_i \in T$, o tabú no activo si $x_i \notin T$.

⁹ Se cree que la memoria a corto plazo en un ser humano típico, involucra los eventos que pueden recordarse los últimos seis días. Mientras que la memoria a largo plazo involucra eventos importantes capaces de ser recordados desde los 6 días hasta los seis meses [17].

Solución élite: es el registro de las mejores soluciones de las vecindades, obtenidas durante el proceso de búsqueda. La solución élite final, debe ser la mejor de las soluciones halladas dentro del proceso de búsqueda.

Lista de candidatos: Esta lista contiene las soluciones que se pueden sumar a la solución élite. Los elementos de la lista de candidatos no deben pertenecer a la lista tabú T . Esta lista también puede contener soluciones, que han permanecido durante largos periodos de tiempo en la solución élite o que han sido parte de la solución élite en varias ocasiones, durante el proceso de búsqueda hecha por TS.

Exploración sensible: se refiere, al tipo de búsqueda dirigida realizada por TS. Esta exploración utiliza el pasado reciente de la búsqueda, el cual se encuentra almacenado en una estructura de memoria. Con esta información, es posible obtener mayor información de una mala solución encontrada durante la búsqueda que la generación aleatoria de puntos para ser evaluados en la función objetivo [12].

Criterio de Aspiración: este es el criterio a considerar para que la solución de la vecindad $N(x)$, puedan aspirar a formar parte de la solución élite. En este trabajo se consideran dos tipos de criterio de aspiración: *baja influencia* y *alta influencia*.

D 1.2.2 Memoria a largo plazo

En algunas aplicaciones, la memoria a corto plazo es suficiente para producir buenas soluciones [52]. Sin embargo la búsqueda por tabú se fortalece a incluir una memoria a largo plazo. Con la estrategia de la memoria a largo plazo, el vecindario modificado producido por TS, podría contener soluciones diferentes a la original, generalmente consiste en la selección de soluciones élite de alta calidad encontradas en varios puntos dentro del proceso de búsqueda. Los elementos que conforman la estructura de la memoria a largo plazo son los siguientes:

Memoria basada en la frecuencia: La memoria basada en la frecuencia, proporciona información, que completa la información dada por la memoria basada en lo reciente, ampliando el criterio de selección de los movimientos [52].

Para la medición de la frecuencia, esta se realiza a través de una proporción, basada en numeradores y denominadores. Los numeradores expresan dos tipos de mediciones, *medición de la transición*, y *medición de residencia*. La medición de la transición se entiende como el número de iteraciones en las que un atributo cambia su estado tabú, es decir cuando un elemento entra o sale de la solución, en las soluciones visitadas dentro de la búsqueda. La medición de residencia expresa el número de iteraciones en las que un atributo ha sido parte de la solución durante el proceso de búsqueda. Los denominadores generalmente se expresan en uno de las siguientes formas:

1. El total de número de ocurrencias de todos los eventos presentados por los numeradores (tal como el número total de iteraciones asociadas).
2. La suma o promedio de los numeradores.
3. El numerador de mayor valor.
4. La media del valor de los numeradores.

En casos en los que los numeradores representan conteos ponderados, en los que se pudieran tener valores negativos, 3 y 4 deben ser considerados como valor absoluto. Y 2 debe ser expresado como una suma de valores absolutos.

Los cocientes entre los numeradores y denominadores producen *la frecuencia de transición* que sigue la pista de cuan a menudo cambia un atributo, y *la frecuencia de residencia* que sigue la pista de cuan a menudo los miembros forman parte de la solución generada.

Creación de multas e incentivos: La memoria basada en la frecuencia, es usualmente aplicada para la modificación del estado tabú, y definir multas e incentivos modificando de esta forma la evaluación de movimiento.

Estrategia de intensificación: Esta estrategia no se realiza únicamente en la memoria a largo plazo, ya que involucra a la memoria a corto plazo. La estrategia de intensificación tiene por objeto modificar las reglas de elección fortaleciendo la combinación de movimientos y las características de soluciones con un buen historial halladas. También permiten regresar a regiones atractivas para realizar una búsqueda más a fondo [17][52]

Se inicia aplicando el proceso de la memoria a corto plazo. Inmediatamente se aplica una estrategia de selección a la solución élite, esta estrategia se refiere a la creación de una lista de soluciones élite, que se encarga de grabar la solución o soluciones élites, obtenidas por la memoria a corto plazo, con el fin de mantenerlas registradas por un determinado tiempo [12].

Seguidamente, mientras el número de iteraciones no supere un valor máximo y además la lista de soluciones no se encuentre vacía, se elige una solución de la lista de soluciones élite y se continúa con el proceso de la memoria a corto plazo para la solución elegida. En todo el proceso de búsqueda, las soluciones son evaluadas por el criterio de aspiración. Cuando se encuentran soluciones mejores a las de la lista, la solución encontrada se suma a la lista y si la lista se encuentra llena, la solución encontrada reemplaza a la peor.

Estrategia de diversificación: Las estrategias de diversificación previenen que el proceso de búsqueda entre en un ciclo repetitivo [12][52], permitiendo salir de un mínimo o máximo local, cuando este mínimo o máximo han sido hallados en una búsqueda normal [18], de esta forma, dirige la búsqueda a regiones que no han sido visitadas. En la búsqueda tabú, la diversificación se realiza en cierta forma por la memoria a corto plazo, pero en particular es reforzada por la memoria a largo plazo [52]. La estrategia de diversificación es relevante en espacios de búsqueda grandes, en los cuales es posible tener varios óptimos locales. En este trabajo, la estrategia de diversificación se omite debido a que el espacio de búsqueda es relativamente pequeño.

D 1.3 Prueba Línea Corta con autosintonización por búsqueda Tabú

En esta sección se presentan los resultados al aplicar la autosintonización en línea a la tabla de reglas de conocimiento mediante búsqueda por tabú para el T1FLC y el IT2 FLC con funciones de membresía Gaussianas y 25 reglas de conocimiento. Se considera el generador síncrono conectado al bus infinito mediante una línea de transmisión corta, con condiciones iniciales

$$V_{ter} = 1 P.U. \quad \theta = 6.8077^\circ \quad P_A = 1 P.U. \quad P_R = -0.1474 P.U.$$

El sistema se sujeta a una falla trifásica en el bus del generador con duración de 10 ciclos. Se utilizan dos criterios de aspiración, alta y baja influencia. El controlador con búsqueda tabú es en línea por lo que la búsqueda por tabú siempre se encuentra ejecutándose.

Los datos para la simulación tabú son solicitados vía un archivo de entrada de datos al programa (ver Apéndice F). la función objetivo utilizada es [12][17][18]:

$$fobj = abs(error * \alpha + \Delta error * \beta)$$

Donde las constantes son $\alpha = 10$ y $\beta = 125$. Además

Tabla D.1 Parámetros para la búsqueda Tabú.

$caos = 0.24$
Tiempo base tabú eliminado = 1
Tiempo base tabú sumado = 0.125
Espacio de búsqueda = Barras de Salida
Lista de candidatos:
 [NG NM NP ZE PP PM PG]

D 1.3.1 Controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento, y autozintonización por búsqueda tabú con criterio de alta influencia

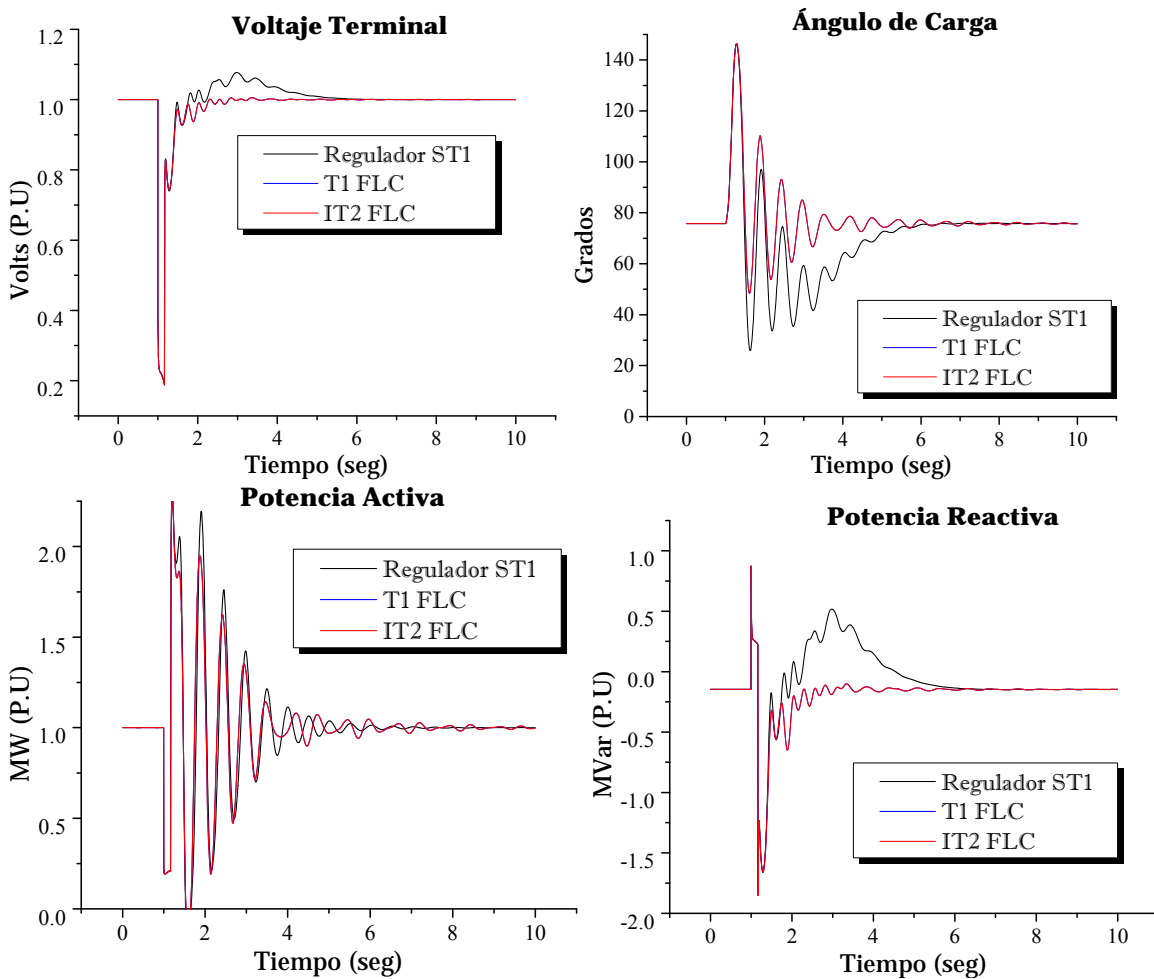
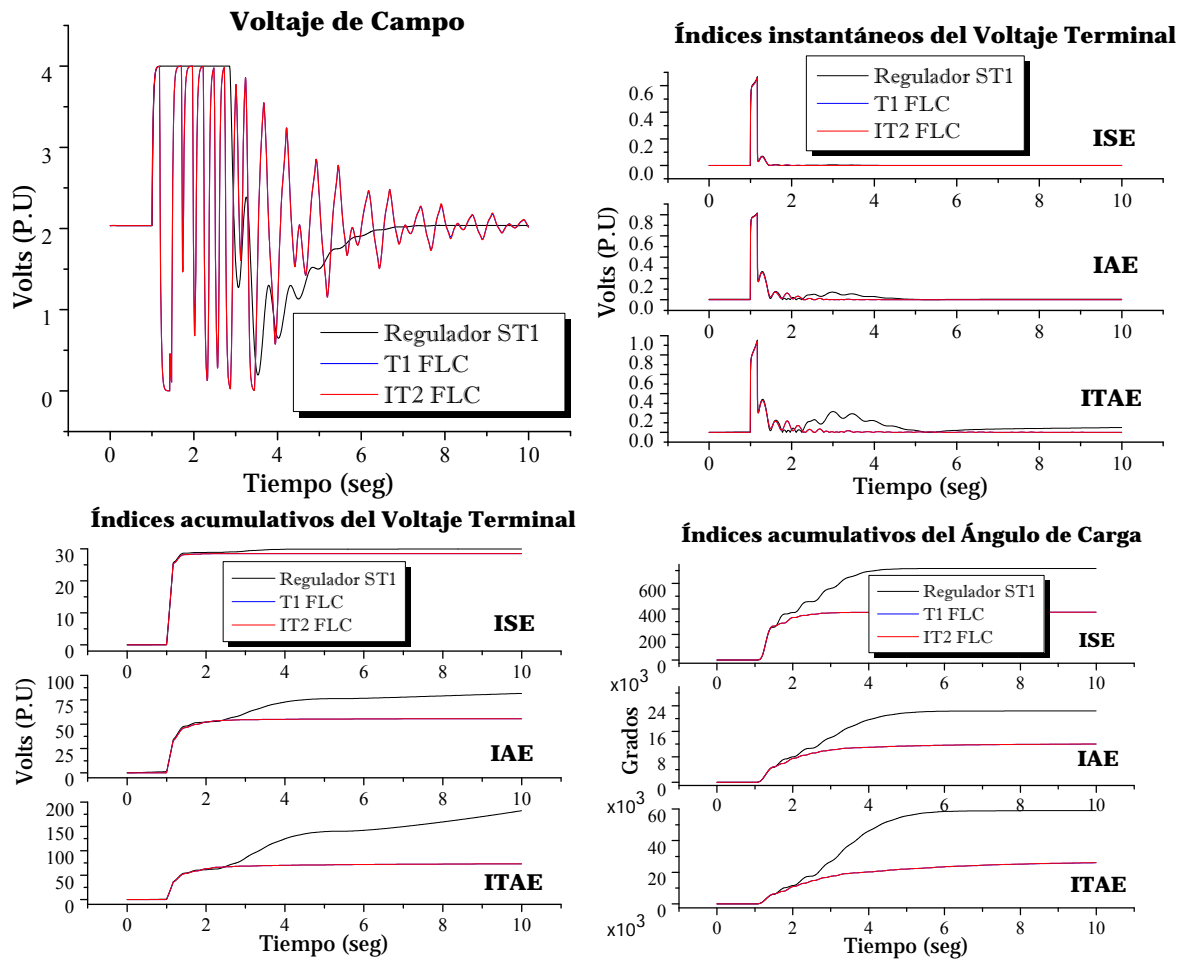


Figura D.1 Respuesta del sistema sujeto a falla trifásica en el bus del generador con duración de 10 ciclos; controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento y autosintonización por búsqueda tabú bajo el criterio de alta influencia.



Continuación de Figura D.1 Respuesta del sistema sujeto a falla trifásica en el bus del generador con duración de 10 ciclos; controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento y autosintonización por búsqueda tabú bajo el criterio de alta influencia.

D 1.3.2 Controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento, y autozintonización por búsqueda tabú con criterio de baja influencia

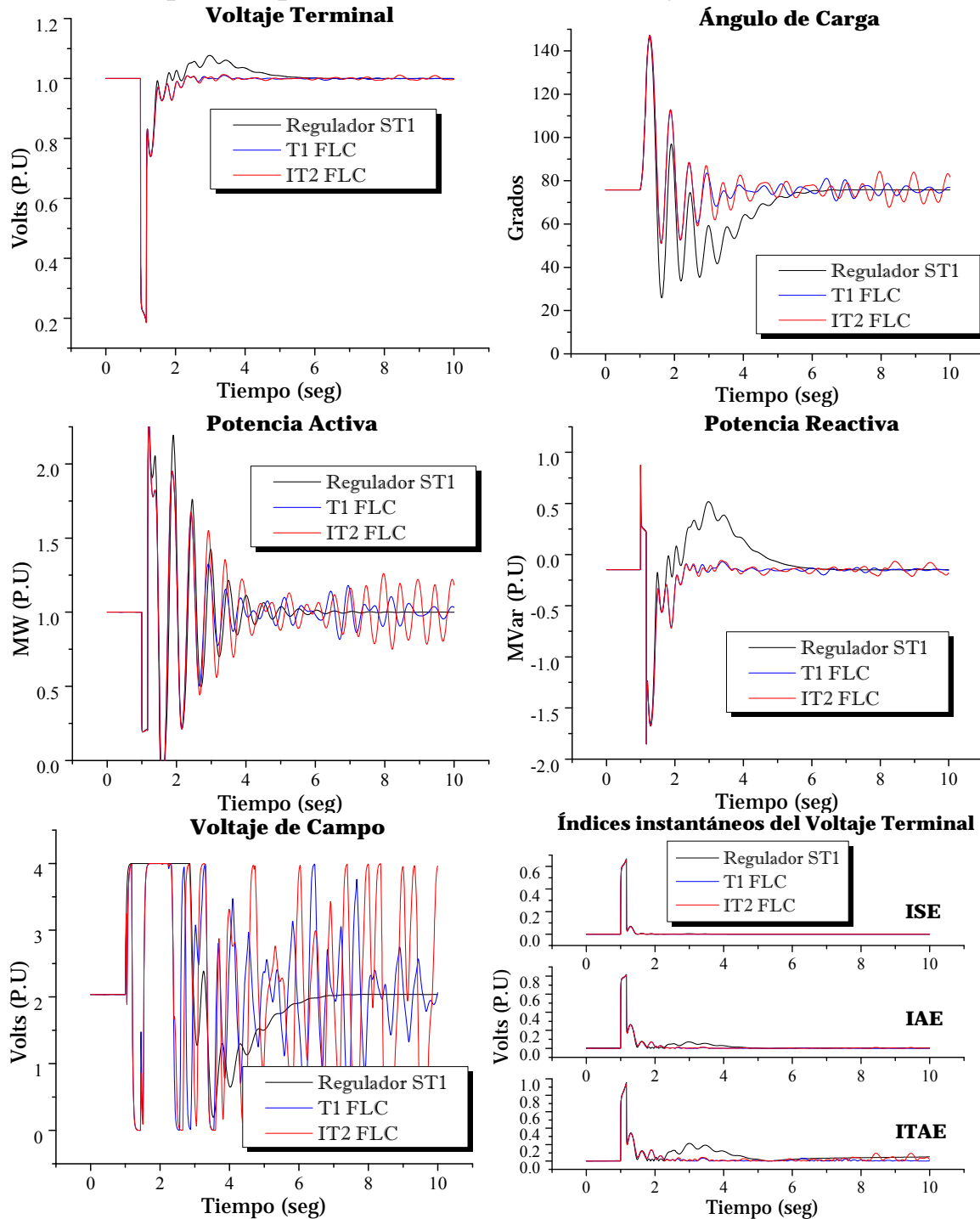
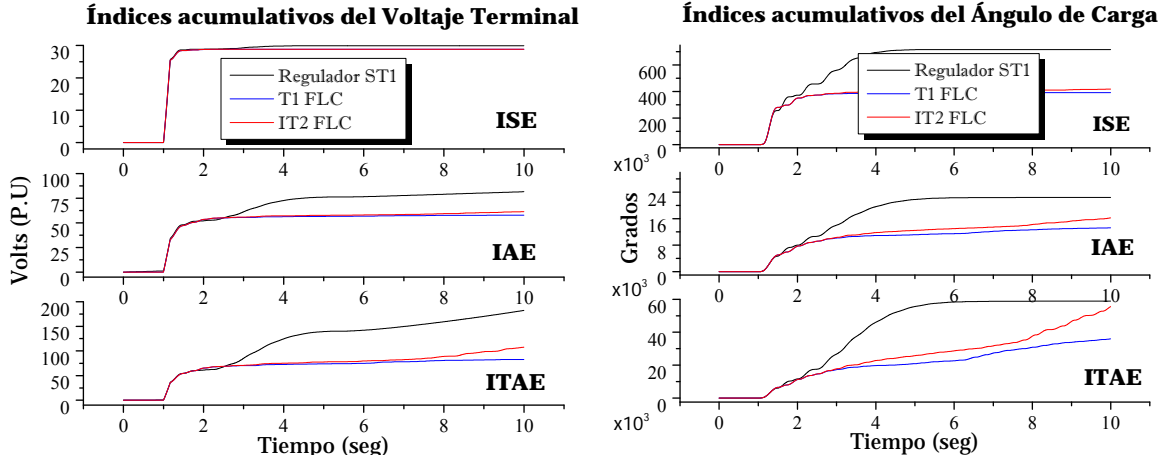


Figura D.2 Respuesta del sistema sujeto a falla trifásica en el bus del generador con duración de 10 ciclos; controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento y autosintonización por búsqueda tabú bajo el criterio de baja influencia.



Continuación de Figura D.2 Respuesta del sistema sujeto a falla trifásica en el bus del generador con duración de 10 ciclos; controladores con funciones de membresía Gaussianas, 25 reglas de conocimiento y autosintonización por búsqueda tabú bajo el criterio de baja influencia.

Tabla D.2 Matriz de 9 reglas difusas inicial para el T1FLC y el IT2 FLC, criterio baja y alta influencia.

e	Δe	NM	NP	CE	PP	PM
NM	NG	NG	NM	NP	CE	PM
NP	NG	NM	NP	CE	PP	PM
CE	NM	NP	CE	PP	PM	PG
PP	NP	CE	PP	PM	PG	PG
PM	CE	PP	PM	PG	PG	PG

Tabla D.3 Matriz de 9 reglas difusas final para el T1FLC a los 2.5s de simulación, criterio baja influencia

e	Δe	NM	NP	CE	PP	PM
NM	NG	PG	PG	PG	PG	PG
NP	PG	NM	NG	CE	PG	PG
CE	PG	NP	CE	CE	PG	PG
PP	PG	CE	CE	CE	PG	PG
PM	PG	PG	PG	PG	PG	PG

Tabla D.4 Matriz de 9 reglas difusas final para IT2 FLC a los 2.5s de simulación, criterio baja influencia

e	Δe	NM	NP	CE	PP	PM
NM	NG	PG	PG	PG	PG	PG
NP	PG	NG	NM	NP	PG	PG
CE	PG	NG	CE	NM	PG	PG
PP	PG	NP	NG	CE	PG	PG
PM	PG	PG	PG	PG	PG	PG

Tabla D.5 Matriz de 9 reglas difusas final para el T1FLC a los 2.5s de simulación, criterio baja influencia

e	Δe	NM	NP	CE	PP	PM
NM	NG	PG	PG	PG	PG	PG
NP	PG	CE	CE	PG	PG	PG
CE	PG	NG	CE	PG	PG	PG
PP	PG	NM	NP	PG	PG	PG
PM	PG	PG	PG	PG	PG	PG

Tabla D.6 Matriz de 9 reglas difusas final para el T1FLC a los 2.5s de simulación, criterio baja influencia

e	Δe	NM	NP	CE	PP	PM
NM	NG	PG	PG	PG	PG	PG
NP	PG	PG	PG	PG	PG	PG
CE	PG	PG	CE	PP	PG	PG
PP	PG	PP	PM	PG	PG	PG
PM	PG	PG	PG	PG	PG	PG

D 1.3.5 Análisis de resultados

Al utilizar la autosintonización de la matriz de reglas difusas mediante búsqueda por Tabú utilizando el criterio de alta influencia, la respuesta de ambos controladores es casi idéntica, razón por la cual las respuestas se encuentran sobrepuestas. Esto se debe a que se ignoran toda multa e incentivo, no se utiliza ninguna memoria y la selección de movimiento se basa únicamente en el valor de la función objetivo. Aunque las respuestas son estables, puede observarse que el voltaje de campo después de 10 segundos de simulación aun presenta oscilaciones.

Para el caso de criterio de baja influencia, los resultados no son satisfactorios debido a que el controlador difuso tipo-2 se ve superado en gran parte de las pruebas por el controlador tipo-1. Nótese que tanto el controlador tipo-1 y el tipo-2 no logran estabilizar al generador síncrono en su totalidad. Esto es debido a que el método de búsqueda aun requiere mejorarse, puesto que el voltaje terminal presenta leves oscilaciones en estado estable, lo que hace que las respuestas del ángulo de carga, potencia activa y reactiva también las presenten.

APÉNDICE E

E.1 CONTROL DIFUSO TIPO-1 ENTRE KL25Z Y SIMULINK MEDIANTE COMUNICACIÓN SERIAL

E 1.1 Introducción

En este apéndice se presenta la implementación de un control de velocidad difuso Mamdani en un microcontrolador Freescale Kinetis KL25Z para un motor de corriente directa (CD) excitación separada. El modelo lineal del motor CD se implementó en Simulink por lo que se hace uso de la comunicación serial para enlazar el controlador difuso y el modelo a controlar en simulink.

La programación del microcontrolador se realiza con la herramienta Code Warrior 10.4 utilizando el MQX Lite, el cual es una versión light del programa Freescale MQX™ Software solutions. Siendo este último un sistema operativo en tiempo real [55]. La ventaja de los sistemas operativos en tiempo real es el manejo del tiempo del microcontrolador teniendo como ventajas: el procesamiento multitarea; manejo de interrupciones; comunicación entre tareas y programación de estas mediante el uso de prioridades; sincronización de recursos de memoria, hardware, periféricos, etc. [55]. De esta forma el CPU puede ejecutar muchas tareas, manejar distintas interrupciones, pero solamente una tarea o interrupción se encuentra en ejecución [55].

MQX Lite es un sistema operativo en tiempo real para microcontroladores de bajos recursos en memoria RAM y Flash [55]. Este sistema gestiona los recursos tales como: memoria; periféricos, mediante actividades denominadas tareas. La ejecución de estas tareas siguen una política FIFO que define el orden en el cual son ejecutadas. MQX Lite cuenta con diferentes herramientas de programación: anejo de tareas, manejo de errores, política FIFO, eventos, semáforos, mutex, mensajes y timers.

E 1.2 Comunicación serial

La comunicación serial consiste en el envío de un bit de información de manera secuencial, es decir, un bit a la vez y a una velocidad definida entre el emisor y el receptor. Este tipo de comunicación sigue los estándares definidos por el RS-232 (Recommended Standard 232) que establece niveles de voltaje, velocidad de transmisión de los datos, etc.

El dispositivo encargado de controlar los puertos y dispositivos serie es denominado UART (Universal Asynchronous Receiver and Transmitter). En la comunicación asincrónica, la velocidad de envío de los datos es acordada entre el emisor y el receptor, y la conexión física es tal como lo muestra la **Figura E.1** Comunicación serial asíncrona.

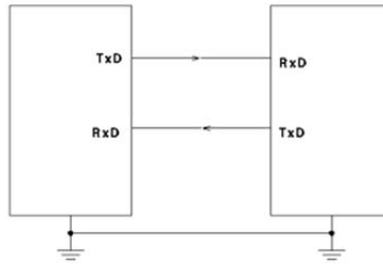


Figura E.1 Comunicación serial asíncrona.

E 1.3 Modelo en Simulink

La comunicación serial con el software comercial MatLab puede ser implementada mediante un modelo en Simulink. Para esto se hace uso de los bloques localizados en la herramienta *Instrument Control Toolbox*, dentro de la librería del propio Simulink.

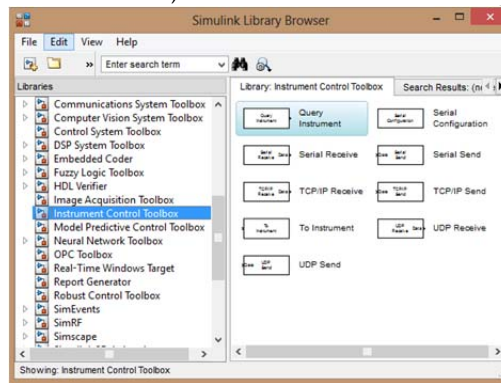


Figura E.2 Bloques para comunicación serial.

Los bloques denominados *To instrument* y *Query Instrument*, permiten realizar el envío y recepción de datos respectivamente, en forma serial. La configuración de estos bloques a 115200 Baudios, y direccionados al puerto COM4 se muestra en la figura siguiente.

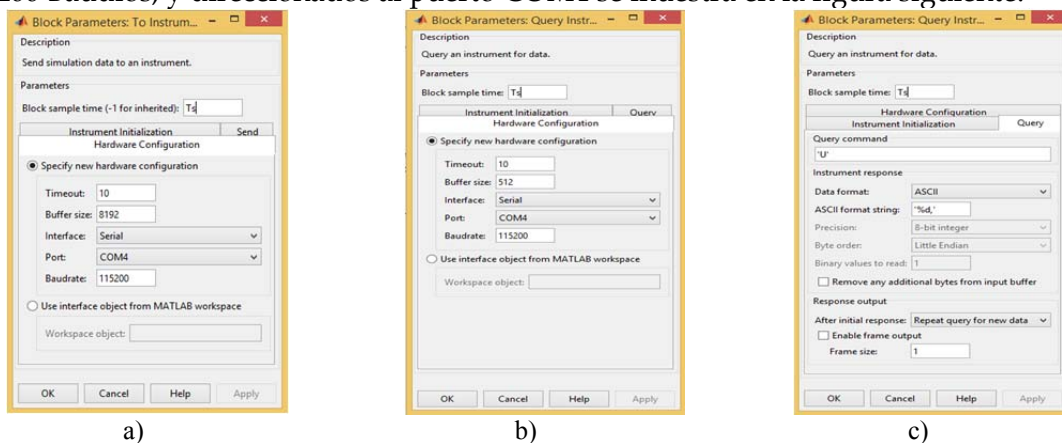


Figura E.3 Configuración de bloques de envío y recepción en Simulink: a) configuración del hardware del bloque de envío; b) configuración del hardware del bloque de recepción; c) configuración del comando de consulta del bloque de recepción.

Además el bloque de recepción *Query Instrument*, requiere de la configuración del terminador de la cadena de caracteres que recibirá, esto con el fin de determinar que una cadena de datos completa ha sido recibida. En este caso el terminador utilizado es el 0x55 hexadecimal, que corresponde al carácter U. Los datos son enviados y recibidos por medio de cadenas de caracteres en formato ASCII, los cuales también se especifican en las configuraciones del bloque en la pestaña “*query*”.

La variable T_s se utiliza para definir el tiempo de muestreo en los bloques utilizados en Simulink. Esta variable debe ser colocada en el apartado de tiempo de muestreo de cada bloque como lo muestra **Figura E.3** E.7. Es necesario definir esta variable en el espacio de trabajo del MatLab, para este caso el tiempo de muestreo utilizado fue $T_s = \frac{1}{64} \text{ seg.}$



Figura E.4 Variable T_s en espacio de trabajo.

El modelo implementado en Simulink es el siguiente.

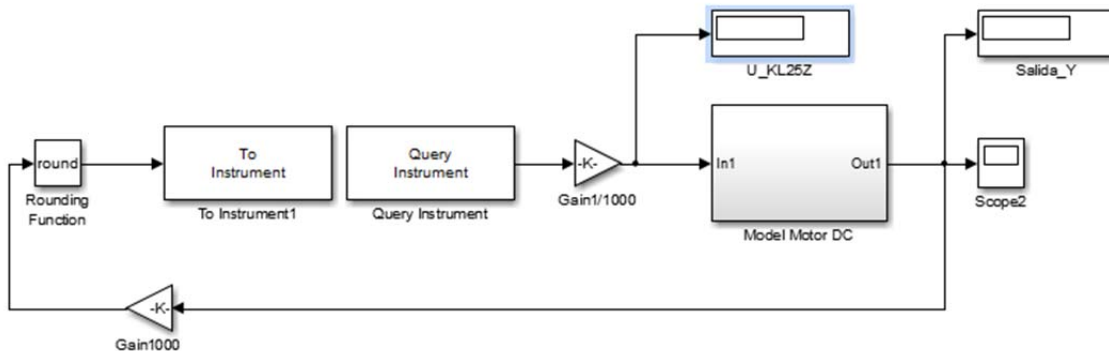


Figura E.5 Lazo de control para el modelo de un motor DC.

El subsistema Model Motor DC contiene lo siguiente:

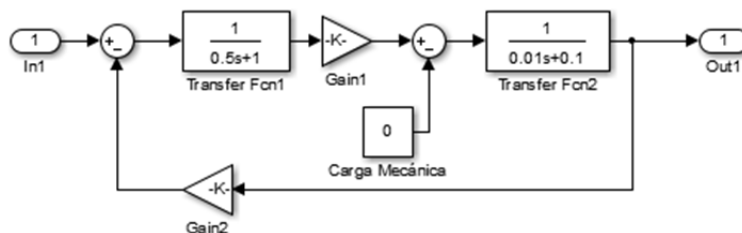


Figura E.6 Subsistema del modelo para un motor DC.

E 1.4 Implementación con Kinetis FRDM-KL25Z y CodeWarrior 10.4

La tarjeta de desarrollo FRDM- KL25Z es desarrollado por Freescale Semiconductor. Este dispositivo es de bajo costo y consumo de energía. Cuenta con un procesador MKL25Z128VLK4, touch slider del tipo capacitivo, cristal de 8 MHz, compartimento

disponible para pila CR2032, acelerómetro MMA8451Q, led RGB e interfaz Open SDA¹⁰ [55].

Los puertos seriales primarios están ubicados en PTA1 y PTA2, las cuales se encuentran conectados al OpenSDA y al conector del microcontrolador [56].

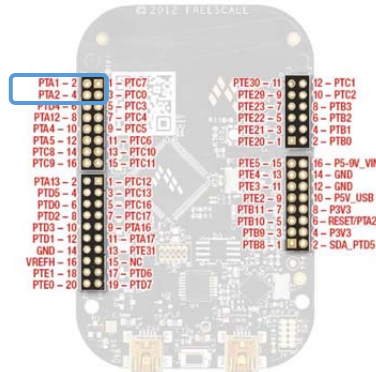


Figura E.7 Ubicación de pines PTA1 y PTA2 en conectores del Kinetis KL25Z.

En la **Figura E.5**, el sistema se encuentra abierto entre los bloques *to instrument* y *Query Instrument*, el microcontrolador es el encargado de cerrar ese lazo mediante la aplicación de un control difuso discreto tipo Mamdani.

El programa se basa en la interacción de 6 tareas y 3 eventos descritos en las tablas siguientes.

Tabla E.1 Eventos utilizados en el programa.

Nombre	Tamaño	Prioridad	Estado inicial de la tarea
Task1 (initialization)	512	8	HABILITADO
Task2 (main_task)	512	8	HABILITADO
Task3 (fuzzification_task)	512	8	DESHABILITADO
Task4 (norm_t_min_task)	512	8	DESHABILITADO
Task5 (maximum_task)	512	8	DESHABILITADO
Task6(defuzzification_task)	512	8	DESHABILITADO

Tabla E.2 Eventos utilizados en el programa.

Nombre	Descripción
AD1_OnMeasurementComplete	Evento que es invocado cuando una medición ha sido realizada.
AS1_OnRxChar	Evento que es invocado cuando un carácter ha sido recibido correctamente.
AS1_OnTxChar	Evento que es invocado después que un carácter ha sido transmitido.

Básicamente se lee el valor de referencia (*setpoint*) mediante un potenciómetro en el puerto del ADC (convertidor analógico digital), el cual es controlado por un temporizador

¹⁰ OpenSDA (Serial and Debug Adapter) es un software libre que se utiliza para la depuración de programas [55][57]. En la tarjeta FRDM-KL25Z está basado en un microcontrolador Kinetis K20, el cual cuenta con un controlador USB, lo que le permite ejecutar aplicaciones OpenSDA [55].

que actúa cada 10ms. La tarea 1 es la encargada de la inicialización de los periféricos, crear de un evento, semáforo para control de ejecución de las tareas y las cuatro tareas secuenciales para el proceso del control difuso tipo Mamdani, una vez realizadas estas acciones la tarea es destruida, mediante la instrucción `_task_abort(Initialization)`.

El algoritmo sigue los pasos siguientes:

1. El evento de recepción de datos se encarga de recibir los caracteres en formato ASCII y almacenarlos en un buffer circular de 16 posiciones, este evento habilita la tarea 2 (Main_task), mediante un evento.
2. La tarea 2, por su parte llama a la función `Str2Int` para convertir la cadena recibida a un valor entero, solicita un dato al ADC, calcula las variables error y cambio del error, aumenta el contador del semáforo para poder ejecutar las 4 tareas siguientes y se coloca en estado de tarea bloqueada, esperando se ejecuten las cuatro tareas restantes.
3. La tarea 3 (`difuzzification_task`), es la primera que se encuentra en la cola del semáforo por lo que es la primera a ejecutarse cuando el contador del semáforo es incrementado. Esta tarea es la encargada de llamar a función `membership_grade()`, la cual modifica dos arreglos del tipo globales (uno para cada variable de entrada), los cuales en sus primeras columnas se encuentran los grados de membresía y en sus segundas columnas el número del conjunto perteneciente de acuerdo con: (0) primer conjunto, (1) segundo conjunto, (2) tercer conjunto, y (3) ningún conjunto, este último indica que la variable correspondiente tiene un grado de membresía nulo para un determinado conjunto. El conteo de los conjuntos difusos es de izquierda a derecha. Por último la tarea 3, aumenta el contador del semáforo poder ejecutar la siguiente tarea con base a la cola de tareas en espera.

Grado de pertenencia	Numero de Conjunto Difuso	Grado de pertenencia	Número de Conjunto Difuso
--	--	--	--
--	--	--	--
--	--	--	--

a)

b)

Figura E.8 Arreglos de tarea 3. (a) variable del error, (b) variable del cambio del error.

4. La tarea 4 (`normT_min`), llama a la función `minimo()`, para obtener los valores mínimos al combinar los grados de membresía de los conjuntos de la variable del error con cada uno de los grados de membresía de los conjuntos de la variable cambio del error. Es decir aplica la norma t con la operación mínimo. Posteriormente aumenta el contador del semáforo para la ejecución de la tarea siguiente.

Grado de pertenencia	Numero de Conjunto Difuso	Grado de pertenencia	Número de Conjunto Difuso
--	--	--	--
--	--	--	--
--	--	--	--

a)

b)

Figura E.9 Arreglos: a) `normT_min` de tarea 4, b) `Tnorm` de la tarea 5.

5. La tarea 5 (`maximum_task`), llama a la función `máximo()`; para evaluar si existen reglas difusas con el mismo consecuente en base al proceso de fuzzificación del

valor actual de las variables de entrada. Es decir aplica la acción máximo o unión de conjuntos. Si existen reglas con el mismo consecuente toma el mayor grado de membresía. La función modifica un nuevo arreglo denominado Tnorm, arreglo de tipo global. Finalmente la tarea 5 incrementa el contador del semáforo para la siguiente tarea en espera.

6. La tarea 6 (`dedifuzzification_task`), invoca por su parte a la función `dedifusificacion()`; enviando como argumento al arreglo Tnorm, modificado en la tarea anterior. Esta función es la que realiza el proceso de dedifusificación mediante el método de barras, mediante la ecuación siguiente:

$$y = \frac{\sum_{i=1}^N x_i \mu_{\bar{A}}(x_i)}{\sum_{i=1}^N \mu_{\bar{A}}(x_i)} \quad (\text{E. 1})$$

Donde: y , es la valor de salida del controlador despues del proceso de dedifusificación. x_i son las distancias sobre el eje horizontal donde se encuentran hallados los centros de los conjuntos difusos, pudiendo ser estos negativos. $\mu_{\bar{A}}(x_i)$ es el grado de membresía final del conjunto difuso de salida obtenida en la tarea 5. Como es de observarse este método representa una aproximación del centroide del área de la unión de los conjuntos difusos de salida en estado discreto. El método de inferencia utilizado es min-max, lo que indica primero un operador de implicación mínimo (tarea 4) y posteriormente un operador de implicación máximo (tarea 5).

Es posible tomar N muestras según la ecuación (E.1). Tomando los centros de los conjuntos difusos de salida, N será igual al número de conjuntos difusos de salida. Esta última tarea habilita la tarea 2 (previamente bloqueada), mediante la instrucción `_task_ready()`.

7. La tarea 2 puede entonces continuar el proceso de ejecución, obteniendo el valor de la salida a enviar por el puerto serie, la salida del controlador difuso implementado es ΔU , por lo que la salida final será:

$$\Delta U = U_{actual} - U_{anterior} \quad (\text{E. 2})$$

Despejando tenemos:

$$U_{actual} = \Delta U + U_{anterior} \quad (\text{E. 3})$$

Posteriormente el dato U_{actual} , es convertido a cadena con el correspondiente terminador, se envía el primer dato al puerto serie mediante `AS1_SendChar()`; y el evento de transmisión se encarga de continuar el envío de datos. De esta forma el lazo de control difuso tipo Mamdani es realizado por el microcontrolador.

Para poder tener mejor precisión en las operaciones y en base a que el microcontrolador es de punto fijo. La señal de respuesta del modelo en el MatLab antes de ser enviada por el puerto serie es multiplicada por una ganancia, la cual debe ser compensada por otra en forma inversa al momento recibir los datos en el MatLab y poder darle entrada al modelo.

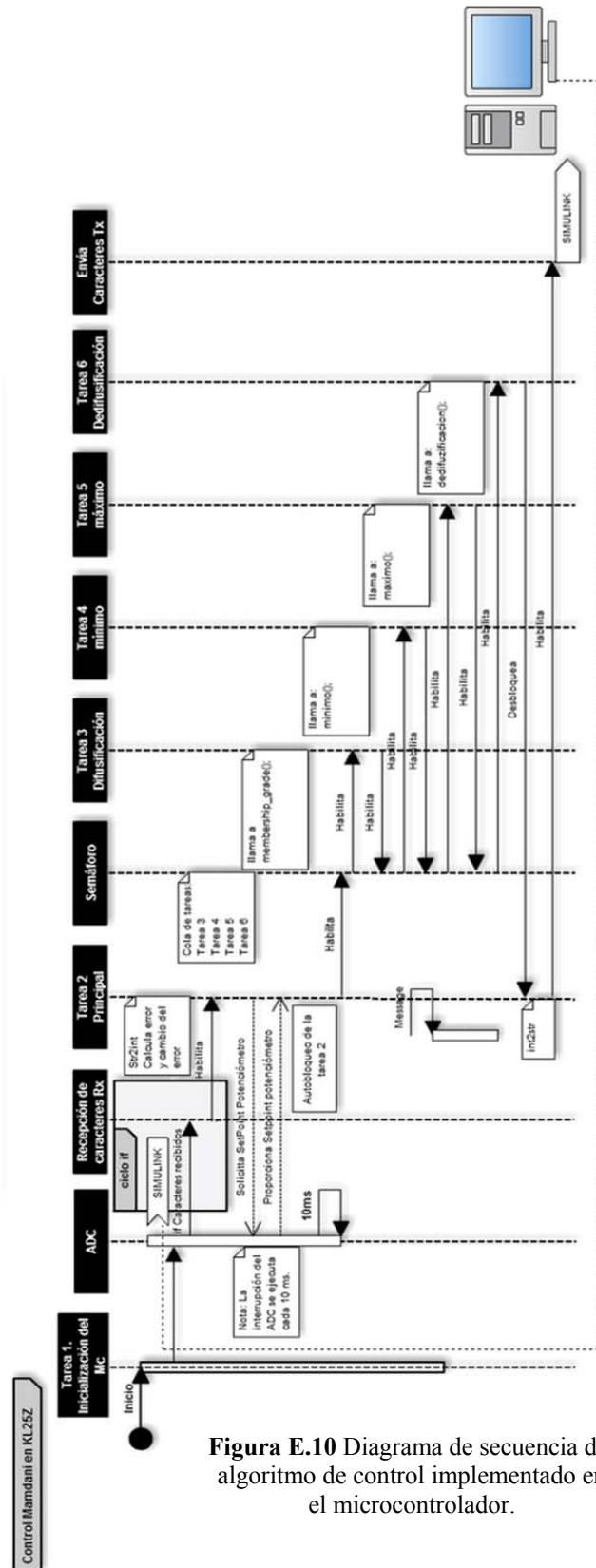


Figura E.10 Diagrama de secuencia del algoritmo de control implementado en el microcontrolador.

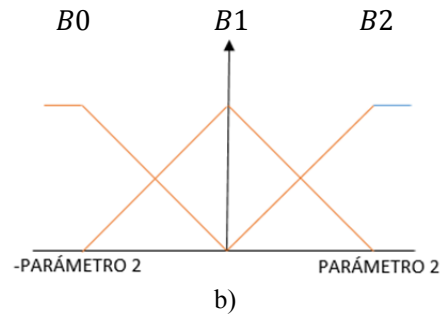
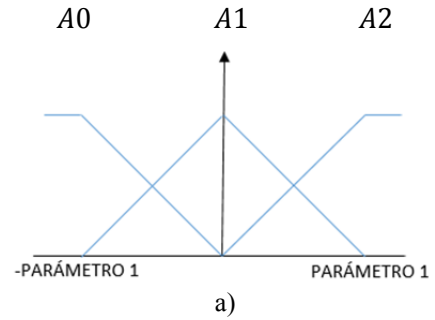


Figura E.11 Parámetros de ajuste de conjuntos difusos de las variables de entrada.

Tabla E.3 Matriz de conocimiento difuso para la superficie de control de la Figura 3.20

e	Δe	N	CE	P
N	NG	NM	NP	
CE	NP	CE	PP	
P	PP	PM	PG	

E 1.5 Simulaciones

El rango de valores del setpoint fue escalado a valores de 0-20, lo que representa para el modelo un rango de velocidades de ajuste de 0 a 20 rad/seg (0 a ≈ 191 RPM). Las figuras siguientes muestran la respuesta del control difuso con:

$$\text{Set point} = 20 \text{ rad/seg}$$

Para la simulación, el ajuste del tiempo de muestro (parámetro T_s en el modelo) es definido por el usuario, sin embargo la respuesta del control varía según el valor seleccionado para T_s . Las figuras E.12 y E.13, muestran la respuesta en el tiempo y el correspondiente diagrama de fase para las variables del error y el cambio del error.

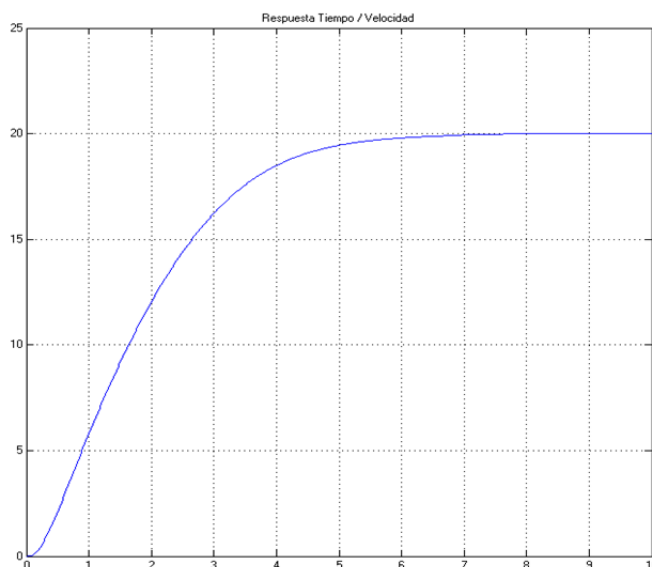


Figura E.12 Respuesta del modelo con control difuso con $T_s = 1/64$ seg.

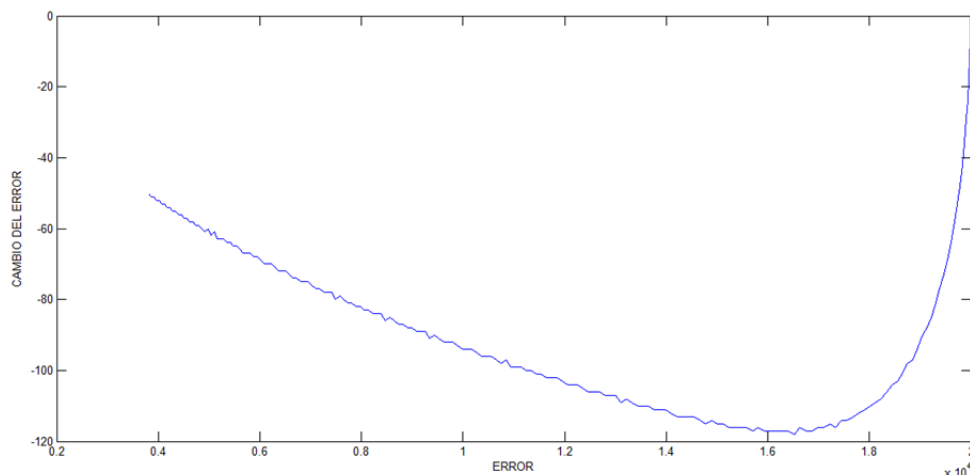


Figura E.13 Diagrama de fase, $T_s = 1/64$ seg.

Se observa una respuesta lenta del control, puesto que el tiempo de estabilización es aproximadamente 7 seg. El diagrama de fase muestra el cambio de las variables de entrada error y cambio del error. La **Figura E.13** muestra únicamente el cambio de dichas variables

en el lapso de aproximadamente 3 seg, razón por la cual no se observa el cero deseado en el diagrama de fase.

Disminuyendo el tiempo de muestreo para este caso, es posible obtener una respuesta más rápida.

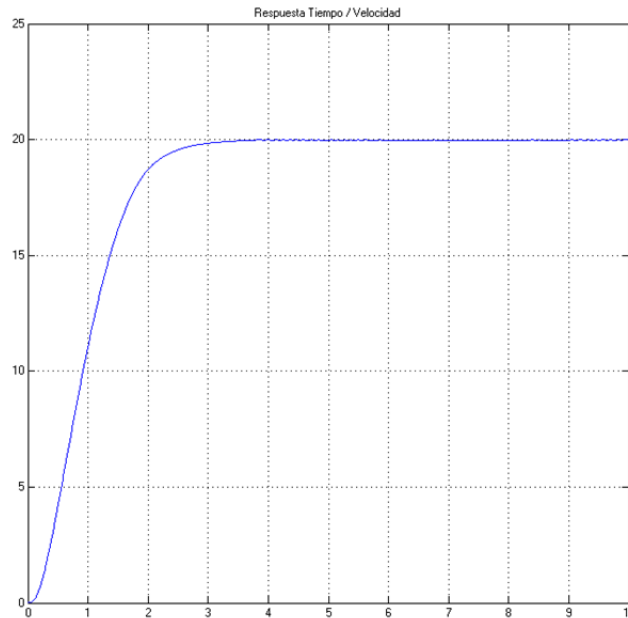


Figura E.14 Respuesta del modelo con $T_s = 1/128$ seg (0.0078 seg).

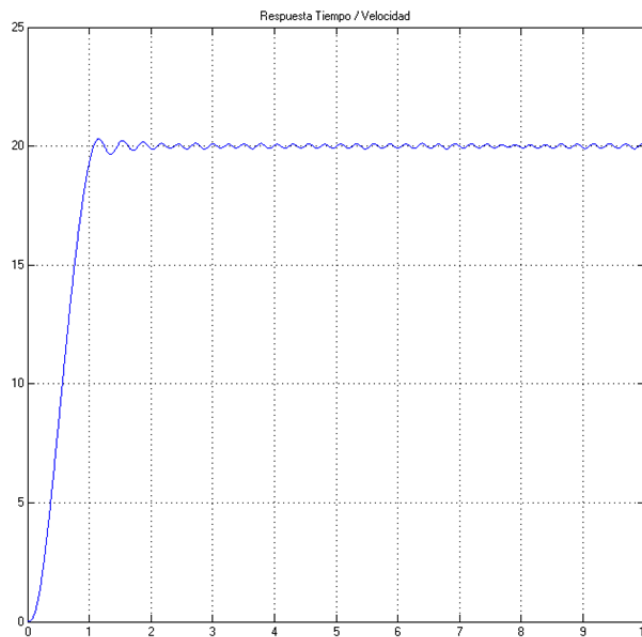


Figura E.15 Respuesta del modelo con $T_s = 1/256$ seg, (0.0039 seg).

Con un tiempo de muestreo de aproximadamente 40 ms, el tiempo de estabilización es aceptable. Sin embargo se presentan oscilaciones en estado estacionario. Se ajustan los parámetros de los conjuntos difusos de entrada para el error, según **Figura E.11**.

Los valores (ajustados con su respectiva ganancia en el microcontrolador) finales son:

PARÁMETRO 1 = 40; PARÁMETRO 2 = 0.2

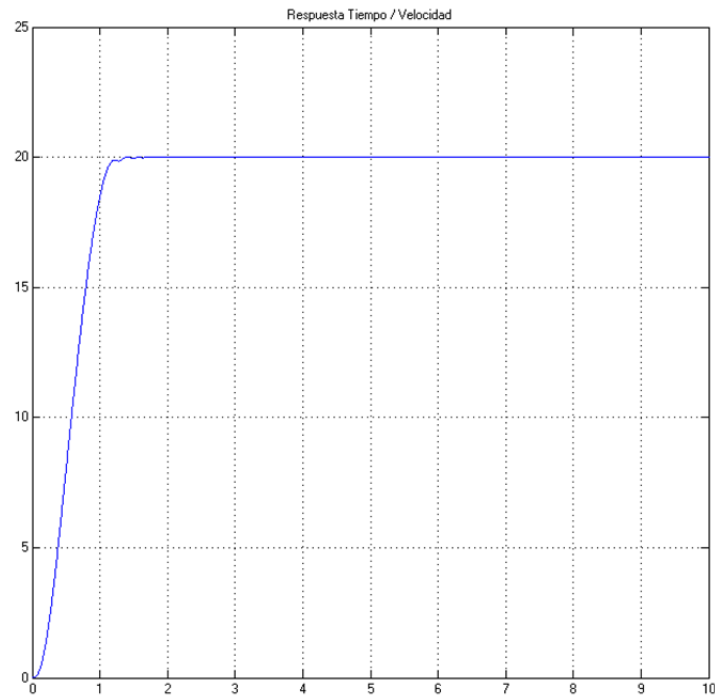


Figura E.16 Respuesta final en el tiempo con $T_s = 1/256$ (0.0039 seg).

La Figura E.16, muestra la respuesta final del modelo con el control difuso. La simulación a un pequeño disturbio dentro del modelo del motor de CD, se muestra en la Figura E.17. A manera de ejemplo se realiza un incremento de 0 a 2 N.m de carga mecánica en el modelo.

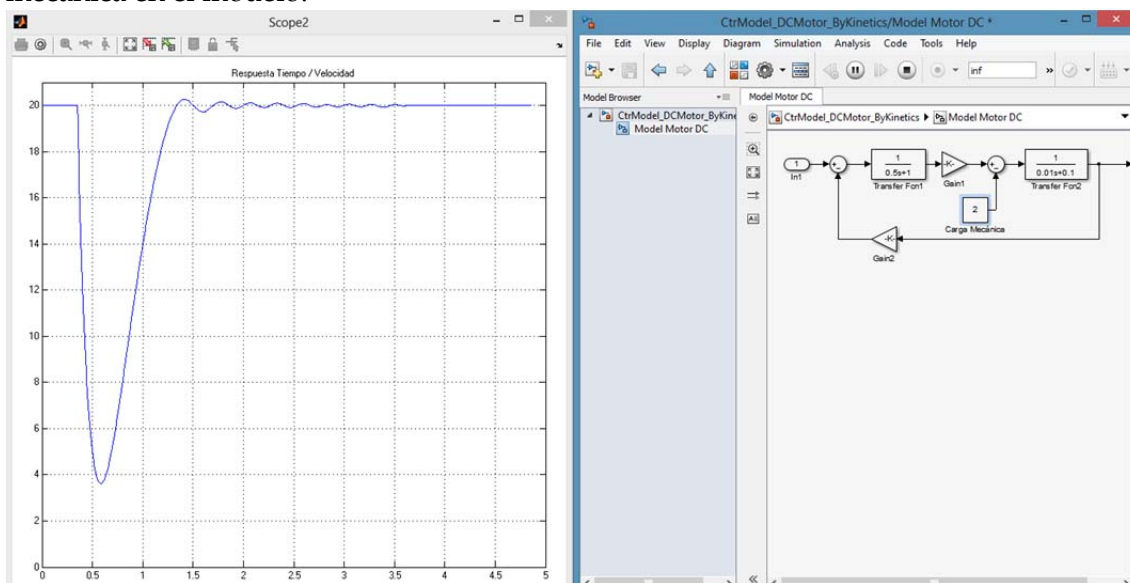


Figura E.17 Respuesta en el tiempo a una perturbación de 2 N.m, de carga mecánica.

E 1.6 Código del programa

```

#include "Cpu.h"
#include "Events.h"
#include "mqx_tasks.h"
#include "Functions.h"

#define DATARCV 1
#define GAIN 1000
#define NUMBEROFDATA 191
//-----
//      \   /   \
//      /   \   /
//      /   \   \
//      \   /   /
//      -PAR1  PAR1
//
#define PAR1 40000 //Error
#define PAR2 200 // Cambio del error
/*-----*/
/*REGLAS*/
#define NG -6000
#define NM -4000
#define NP -2000
#define ZE 0
#define PP 2000
#define PM 4000
#define PG 6000

int RULES[4][4]={{NG,NM,NP,0},{NP,ZE,PP,0},{PP,PM,PG,0},{0,0,0,0}};
/*-----*/
#ifdef __cplusplus
extern "C" {
#endif
/*Variables*/
LDD_TDeviceData *PTR_ADC;
pointer PTR_Initialization,PTR_Main_Task;
short position;
int Error_Store[NUMBEROFDATA + 1],Change_Error_Store[NUMBEROFDATA + 1];
int SetPoint,Y,U=0,Value,dU,MF1[3][2],MF2 3][2],Tnorm_min[9][2],Tnorm[4][2],
Error,Last_Error,Change_Error = 20;
unsignedchar MatLab_Data[16],ConvertedValues[16],*ConvertedValuesptr,*AP_Data;
/* Use light weight Events */
LWEVENT_STRUCT lwevent;

/* Use light weight semaphores */
LWSEM_STRUCT semaforo;

/* User includes (#include below this line is not maintained by Processor Expert) */

/*
** =====
** Event      : Initialization (module mqx_tasks)
**
** Component  : Task1 [MQXLite_task]
** Description :
**      MQX task routine. The routine is generated into mqx_tasks.c
**      file.
** Parameters  :
**      NAME          - DESCRIPTION
**      task_init_data -
** Returns       : Nothing
** =====
*/
void Initialization(uint32_t task_init_data)
{
    _task_id Initialization;
    Initialization = _task_get_id();
    PTR_Initialization = _task_get_td(Initialization);

    PTR_ADC = AD1_Init(NULL); //INICIALIZA ADC
    AD1_SelectSampleGroup(PTR_ADC,0U);

    //Create event to Main_Task
    _lwevent_create(&lwevent,LWEVENT_AUTO_CLEAR);

    _lwsem_create(&semaforo,0);

    //Create the fuzzification_task Task
    _task_create_at(0, FUZZIFICATION_TASK, 0, fuzzification_task_stack, FUZZIFICATION_TASK_STACK_SIZE);

    //Create the norm_t_min_task Task
    _task_create_at(0, NORM_T_MIN_TASK, 0, norm_t_min_task_stack, NORM_T_MIN_TASK_STACK_SIZE);

    //Create the maximum_task Task
    _task_create_at(0, MAXIMUM_TASK, 0, maximum_task_stack, MAXIMUM_TASK_STACK_SIZE);

    //Create the defuzzification_task Task
    _task_create_at(0, DEFUZZIFICATION_TASK, 0, defuzzification_task_stack, DEFUZZIFICATION_TASK_STACK_SIZE);

    MatLab_Data[15]='\n'; //Escribe '\n' enultimaposicionpara leer primer dato

```

```

_task_abort(Initialization);

while(1) {
}

/*
** =====
** Event      : Main_Task (module mxq_tasks)
**
** Component  : Task2 [MQXLite_task]
** Description :
**   MQX task routine. The routine is generated into mxq_tasks.c
**   file.
** Parameters :
**   NAME      - DESCRIPTION
**   task_init_data -
** Returns    : Nothing
** =====
*/
void Main_Task(uint32_t task_init_data)
{
    _task_id Main_Task;
    Main_Task = _task_get_id();
    PTR_Main_Task = _task_get_td(Main_Task);
    int counter = 0;
    //AD1_StartLoopMeasurement(PTR_ADC);

while(1) {
    _lwevent_wait_ticks(&lwevent,DATARCV,FALSE,0);

    /*ConvierteDatosde MatLab a Entero*/
    Y = str2int(position,MatLab_Data);

    /*Lee SetPoint*/
    AD1_StartSingleMeasurement(PTR_ADC);

    /*Varibale Error and Change_Error*/
    Error = (SetPoint*GAIN) - Y;
    Change_Error = Error - Last_Error;
    Last_Error = Error;

    /*Save*/
    if (counter <= NUMBEROFDATA){
        Error_Store[counter] = Error;
        Change_Error_Store[counter] = Change_Error;
        counter++;}

    _lwsem_post(&semaforo);
    _task_block();

    /*PI incremental output*/
    U += dU;
    int2str(U,8);
    ConvertedValuesptr=ConvertedValues;
    AS1_SendChar(*ConvertedValuesptr);
}
}

/*
** =====
** Event      : fuzzification_task (module mxq_tasks)
**
** Component  : Task3 [MQXLite_task]
** Description :
**   MQX task routine. The routine is generated into mxq_tasks.c
**   file.
** Parameters :
**   NAME      - DESCRIPTION
**   task_init_data -
** Returns    : Nothing
** =====
*/
void fuzzification_task(uint32_t task_init_data)
{
    int counter = 0;

while(1) {
    _lwsem_wait(&semaforo);
    GREEN_LED_NegVal(GREEN_LED_DeviceData);

    /*Membership Grade*/
    membership_grade(Error,Change_Error,PAR1,PAR2);
    _lwsem_post(&semaforo);
}
}

/*
** =====
** Event      : norm_t_min_task (module mxq_tasks)
**
** Component  : Task4 [MQXLite_task]
** Description :

```

```

**      MQX task routine.  The routine is generated into mqx_tasks.c
**      file.
**      Parameters  :
**      NAME        - DESCRIPTION
**      task_init_data -
**      Returns     : Nothing
** =====
*/
voidnorm_t_min_task(uint32_t task_init_data)
{
int counter = 0;

while(1) {
    _lwsem_wait(&semaforo);
    GREEN_LED_NegVal(GREEN_LED_DeviceData);

    minimo(MF1,MF2,RULES);
    _lwsem_post(&semaforo);
}
}

/* =====
**      Event      : maximum_task (module mqx_tasks)
**
**      Component  : Task5 [MQXLite_task]
**      Description:
**      MQX task routine. The routine is generated into mqx_tasks.c
**      file.
**      Parameters :
**      NAME        - DESCRIPTION
**      task_init_data -
**      Returns     : Nothing
** =====
*/
voidmaximum_task(uint32_t task_init_data)
{
int counter = 0;

while(1) {
    _lwsem_wait(&semaforo);
    GREEN_LED_NegVal(GREEN_LED_DeviceData);

    /*Ask for rules with same consequent*/
    maximo(Tnorm_min);
    _lwsem_post(&semaforo);
}
}

/* =====
**      Event      : defuzzification_task (module mqx_tasks)
**
**      Component  : Task6 [MQXLite_task]
**      Description:
**      MQX task routine. The routine is generated into mqx_tasks.c
**      file.
**      Parameters :
**      NAME        - DESCRIPTION
**      task_init_data -
**      Returns     : Nothing
** =====
*/
voiddefuzzification_task(uint32_t task_init_data)
{
int counter = 0;

while(1) {
    _lwsem_wait(&semaforo);
    GREEN_LED_NegVal(GREEN_LED_DeviceData);

    dU = defuzificacion(Tnorm);
    _task_ready(PTR_Main_Task);
}
}

/* END mqx_tasks */

#ifdef __cplusplus
} /* extern "C" */
#endif

#include"cpu.h"
#include"Events.h"
#include"mqx_tasks.h"

#define DATARC 1

#ifdef __cplusplus
extern"C" {
#endif

uint16 ValueADC;
externint SetPoint;

```

```

externshort position;
externLDD_TDeviceData *PTR_ADC;
externunsignedchar MatLab_Data[16];
byte Error,RecvChrError,SendChrError;
externunsignedchar ConvertedValues[16],*ConvertedValuesptr;

/* Use light weight Events */
LWEVENT_STRUCT lwevent;

** =====
** Event      : AS1_OnRxChar (module Events)
**
** Component  : AS1 [AsynchoSerial]
** Description:
**   This event is called after a correct character is received.
**   The event is available only when the <Interrupt
**   service/event> property is enabled and either the <Receiver>
**   property is enabled or the <SCI output mode> property (if
**   supported) is set to Single-wire mode.
** Parameters : None
** Returns    : Nothing
** =====
*/
voidAS1_OnRxChar(void)
{
    staticAS1_TComData *AP_BuffInput = MatLab_Data;
    staticint i=0;

    RecvChrError = AS1_RecvChar(AP_BuffInput); //Recibedatode canal derecepcion
    if (MatLab_Data[i] == '\n' && MatLab_Data[(i-1)&0xf] == 'U'){

        _lwevent_set(&lwevent,DATARCV); //Habilitaevento DATORECIBIDO
        position=i; //Guardaposiciónpara el apuntador
        AP_BuffInput++;
        i=(i+1)&0xf;
        if(!i)AP_BuffInput = MatLab_Data;
    }
}
/*
** =====
** Event      : AS1_OnTxChar (module Events)
**
** Component  : AS1 [AsynchoSerial]
** Description:
**   This event is called after a character is transmitted.
** Parameters : None
** Returns    : Nothing
** =====
*/
voidAS1_OnTxChar(void)
{
    ConvertedValuesptr++;
    if(*ConvertedValuesptr!=NULL){
        while ((SendChrError = AS1_SendChar(*ConvertedValuesptr))!=ERR_OK);
    }
}
/*
** =====
** Event      : AD1_OnMeasurementComplete (module Events)
**
** Component  : AD1 [ADC_LDD]
**
** @brief
**   Called after measurement is done, [Interrupt service/event]
**   is enabled, OnMeasurementComplete event is enabled and ADC
**   device is enabled. See [SetEventMask()] method or [Event
**   mask] property group to enable this event and [Enable]
**   method or [Enabled in init. code] property to enable ADC
**   device. If DMA is enabled , this event is called after the
**   configured number of measurements and DMA transfer is done.
**
** @param
**   UserDataPtr - Pointer to the user or
**               RTOS specific data. The pointer is passed
**               as the parameter of Init method.
**
** =====*/
voidAD1_OnMeasurementComplete(LDD_TUserData *UserDataPtr)
{
    AD1_GetMeasuredValues(PTR_ADC,&ValueADC); //Tomavalordel ADC
    SetPoint=(int)ValueADC*20/65000; //Escalavalorde 0 - 20
}

//PROTOTIPO DE FUNCIONES=====
//void int16toascii(int);
int str2int(int pos_actual, unsigned char MatLab_Data[16]);
int numpow(int,int);
void int2str(int,int);
void membership_grade(int,int,int,int);
void minimo(int MF1[3][2], int MF2[3][2], int RULES[4][4]);
void maximo(int Thorm_min[9][2]);
int defuzificacion(int Thorm[4][2]);

```

```

//=====
#ifndef FUNCTIONS_H_
#define FUNCTIONS_H_
#endif /* FUNCTIONS_H_ */

#include "Functions.h"

/*Labels*/
#define HIGH_SPEED 0
#define SPEED_OK 1
#define LOW_SPEED 2
#define WITHOUT_MEMBERSHIP 3
#define LOWERING_SPEED 0
#define STEADY_SPEED 1
#define INCREASING_SPEED 2
/*-----*/

#define GAIN 1000
extern int Value, MF1[3][2], MF2[3][2], Tnorm_min[9][2], Tnorm[4][2];
extern unsigned char ConvertedValues[16];
//=====
//Función convierte cadena de datos ASCII a entero.
//Parámetro 'posicion' es la posición i del arreglo donde se encuentra el
//terminador '\n'.
//Parámetro 'MatLab[]' es el arreglo global donde se están recibiendo los datos
//serial. Buffer circular de 16 posiciones.
//=====
int str2int(int posicion, unsigned char MatLab_Data[16])
{
short pos_actual, i;
unsigned char *AP_Data;

pos_actual=(posicion-2)&0xf; //Determina la posición a enviar al apuntador por posición-2 debido a 'U' y '\n'
AP_Data=&MatLab_Data[pos_actual]; //Envía al apuntador a la posición de las unidades del dato a leer
Value=*AP_Data-'0'; //Convierte ASCII a Decimal '0'= 30
i=10;
while ( (*AP_Data=&MatLab_Data[(-pos_actual)&0xf]) != '\n' ){ //Si el contenido en el Apuntador decrementado es diferente
a '\n'
if(*AP_Data == '-') Value = -Value; //Si el dato es negativo, entonces el
dato es negativo. //Si no sumas decenas, centenas y millares.
else{ Value=Value + (*AP_Data - '0')*i; //Multiplicador de decenas, centenas y millares.
i=i*10; }
}
return Value;
}

//=====
//Función convierte entero a cadena asignada.
//Parámetro value es el valor entero a convertir
//Parámetro p es el número de dígitos incluyendo el signo del número a convertir
//P. Ejemplo: value= -12905 p= 6.
//El arreglo que se modifica debe ser declarado global y debe ser de dimensión
//mínima p+2 la función no devuelve valor.
//=====
void int2str(int value, int p)
{
int Aux, i, n, r;
ConvertedValues[p+1]='\n'; //Escribe Terminador
ConvertedValues[p]='U'; //Escribe Terminador
if (value < 0){ //Pregunta por número negativo
value=abs(value); //Hace el valor positivo
ConvertedValues[0]='-'; //Coloca signo negativo en primera posición.
for(i=2;i<=p;i++){ //infiere número comenzando por el más significativo
Aux=numpow(10,(p-i)); //Llama a función numpow
ConvertedValues[i-1]=((value/Aux)%10)+48;
}
}
else{
for(i=1;i<=p;i++){ //infiere número comenzando por el más significativo
Aux=numpow(10,(p-i)); //Llama a función numpow
ConvertedValues[i-1]=((value/Aux)%10)+48;
}
}
}

//=====
//Función eleva 'num' a potencia 'n'
//=====
int numpow(int num, int n)
{
int Aux, i;
Aux=num;
if (n!=0) //Verifica que la potencia a elevar sea
diferente a cero
for(i=1;i<=n-1;i++) Aux=Aux*num; //Calcula el número a la potencia 'n'
else Aux=1; //si n=0 entonces el resultado es 1
return Aux; //regresa entero
}

//=====
/*Función grado de membresía*/
/*Calcula el grado de membresía según valor y parámetros*/
/*Los parámetros de fin de tres funciones de membresía tipo triangular*/

```

```

//=====
voidmembership_grade(int x1,int x2,int A,int B)
{
    #define LIMIT1 -100000
    #define LIMIT2 100000

    /*Universe 1*/
    if (x1 <= (-A) && x1 >= LIMIT1){
        MF1[0][0] = GAIN*1; MF1[1][0] = 0; MF1[2][0] = 0;
        MF1[0][1] = HIGH_SPEED; MF1[1][1] = WITHOUT_MEMBERSHIP; MF1[2][1] = WITHOUT_MEMBERSHIP;}
    if (x1 <= 0 && x1 > (-A)){
        MF1[0][0] = GAIN* (-x1)/ A; MF1[1][0] = GAIN*(x1 + A)/A; MF1[2][0] = 0;
        MF1[0][1] = HIGH_SPEED; MF1[1][1] = SPEED_OK; MF1[2][1] = WITHOUT_MEMBERSHIP;}
    if (x1 <= A && x1 > 0){
        MF1[0][0] = 0; MF1[1][0] = GAIN*(A - x1)/A; MF1[2][0] = GAIN*x1/A;
        MF1[0][1] = WITHOUT_MEMBERSHIP; MF1[1][1] = SPEED_OK; MF1[2][1] = LOW_SPEED;}
    if(x1 <= LIMIT2 && x1 > A){
        MF1[0][0] = 0; MF1[1][0] = 0; MF1[2][0] = GAIN*1;
        MF1[0][1] = WITHOUT_MEMBERSHIP; MF1[1][1] = WITHOUT_MEMBERSHIP; MF1[2][1] = LOW_SPEED;}

    /*Universe 2*/
    if (x2 <= (-B) && x2 >= LIMIT1){
        MF2[0][0] = GAIN*1; MF2[1][0] = 0; MF2[2][0] = 0;
        MF2[0][1] = LOWERING_SPEED; MF2[1][1] = WITHOUT_MEMBERSHIP; MF2[2][1] = WITHOUT_MEMBERSHIP;}
    if (x2 <= 0 && x2 > (-B)){
        MF2[0][0] = GAIN*(-x2)/B; MF2[1][0] = GAIN*(x2 + B)/B; MF2[2][0] = 0;
        MF2[0][1] = LOWERING_SPEED; MF2[1][1] = STEADY_SPEED; MF2[2][1] = WITHOUT_MEMBERSHIP;}
    if (x2 <= B && x2 > 0){
        MF2[0][0] = 0; MF2[1][0] = GAIN*(B - x2)/B; MF2[2][0] = GAIN*x2/B;
        MF2[0][1] = WITHOUT_MEMBERSHIP; MF2[1][1] = STEADY_SPEED; MF2[2][1] = INCREASING_SPEED;}
    if(x2 <= LIMIT2 && x2 > B){
        MF2[0][0] = 0; MF2[1][0] = 0; MF2[2][0] = GAIN*1;
        MF2[0][1] = WITHOUT_MEMBERSHIP; MF2[1][1] = WITHOUT_MEMBERSHIP; MF2[2][1] =
        INCREASING_SPEED;}
    }

//=====
/*Funciónquerealizaoperaciónmínimoparalacombinacióndelos
conjuntos, esdecirparalareglasdifusas.
PARAMETROS: MF1 y MF2 son losarregloscon el gradodemembresía*/
/*Modifica el arreglo Tnorm_min el cualdebeser global*/
//=====
voidminimo(int MF1[3][2], int MF2[3][2], int RULES[4][4])
{
    short i,j,k = 0;

    for(i = 0; i <= 2; i++){
        for(j = 0; j <= 2; j++){
            Tnorm_min[k][0] = MF1[i][0] < MF2[j][0] ? MF1[i][0] : MF2[j][0];
            Tnorm_min[k][1] = RULES[(MF1[i][1])][(MF2[j][1])];
            k++;
        }
    }
}

//=====
/*Funciónquerealizaoperación máximo paraconjuntosrepetidos
Esdecirparareglasconigualconsecuente
PARAMETRO Tnorm_min es el resultadodefunciónmínimo
/*Modifica el arregloTnorm el cualdebeser global*/
voidmaximo(int Tnorm_min[9][2])
{
    short i,j,k = 0;
    for(i = 0; i <= 8; i++){
        if (Tnorm_min[i][1] != 0){
            Tnorm[k][0] = Tnorm_min[i][0];
            Tnorm[k][1] = Tnorm_min[i][1];
            for(j = i+1; j <= 8; j++){
                if (Tnorm_min[j][1] != 0){
                    if (Tnorm_min[i][1] == Tnorm_min[j][1]){
                        Tnorm[k][0] = Tnorm[k][0] > Tnorm_min[j][0] ? Tnorm[k][0] :
                        Tnorm_min[j][0];
                        Tnorm_min[j][0] = 0; Tnorm_min[j][1] = 0;
                    }
                }
            }
            k++;
        }
    }
}

//=====
/*FunciónquerealizaprocesodeDefuzificaciónpormétodo debarras*/
//=====
intdefuzificacion(int Tnorm[4][2])
{
    int COG = 0,i,AUX = 0;
    for (i = 0; i <= 3; i++){
        AUX += Tnorm[i][0];
    }
    for (i = 0; i <= 3; i++){
        COG += Tnorm[i][0]*Tnorm[i][1]/AUX;
    }
    return COG;
}

```


APÉNDICE F

F 1 CÓDIGOS DE PROGRAMACIÓN

F 1.1 Introducción

Enseguida se muestran los códigos de programación desarrollados en lenguaje Matlab en su versión R2013a. El *programa principal* contiene las variables globales utilizadas, definición de vectores y matrices, y la simulación de sistema máquina bus infinito con el regulador ST1, el T1 FLC y el IT2 FLC. Nótese que para la simulación de las incertidumbres en el voltaje terminal y el voltaje de referencia simplemente se debe habilitar las instrucciones eliminando el comando "%". También se grafican algunos resultados mediante al final de la simulación. El programa principal genera un archivo de salida . DAT, el cual fue utilizado para graficar los resultados presentados.

En el apartado *funciones secundarias* se tienen las funciones para la lectura de los datos de entrada, la generación de condiciones iniciales del sistema, función de las ecuaciones dinámicas, funciones requeridas para la solución del sistema mediante el método de integración Runge Kutta 4to orden, la función de falla, y cambio de carga. Esta última es utilizada cuando se simula el sistema en vacío y tomas de carga en intervalos de tiempo definido por el usuario.

En el bloque de *funciones para los controladores difusos*, se muestran las funciones para los algoritmos KM y EKMT además de las cinco funciones de membresía tipo-2 utilizadas en este trabajo. Estas funciones simulan también funciones de membresía tipo-1 cuando las incertidumbres se hacen nulas. En *funciones auxiliares para los controladores difusos* se muestran las funciones requeridas por los algoritmos de control difuso tipo-2 KM y EKM.

Por último en el apartado *Funciones para búsqueda por tabú* se muestran el conjunto de funciones utilizadas para el desarrollo de la memoria a corto y largo plazo utilizadas por la búsqueda por tabú.

F 1.2 Datos de Simulación

A continuación se muestran los archivos para lectura de datos para la simulación de los controladores difusos y el generador síncrono. Se tienen dos archivos de datos, el primero solicita datos de simulación en general, mientras que el segundo se solicita en caso de realizar una simulación de toma de carga, los datos de tiempo e incrementos de carga deseados.

```
% ARCHIVO 1 - DATOS DE ENTRADA PARA SIMULACIÓN
% Línea de transmisión : Corta [1] Larga [2]
1
% Tiempo Final de Simulación
10
% Paso de integración
0.004
% Selección de disturbio
% Cambios de carga (pequeño disturbio) [1] Corto circuito trifásico (disturbio grande) [2]
```

```

2
% Para Falla de corto circuito en bus infinito, opción 2:
% Operación de la Máquina: En vacío [1] Con carga [2]
% Nota la cantidad de carga en la máquina esta dada de acuerdo a la condición inicial
2
% Tiempo inicial de falla en seg:
1
% Ciclos de duración de falla:
10
%*****
%Datos del Controlador difuso tipo 1:
% Universo de discurso del error
0.5
% Universo de discurso del cambio del error
0.01
% Número de reglas difusas: 9 reglas [9] 25 reglas [25] 49 reglas [49]
9
% Numero de barras de salida: 5 [5] 7 [7]
7
%Uniformemente distribuidas: Si [1] No [2]
1
% Universo de discurso de Salida
3
% Para las siguientes matrices, si Numero de Barras de salida es = 5, NG = NM y PG = PM
% Matriz de 9 reglas difusas
NG NM CE
NP CE PP
CE PM PG
% Matriz de 25 reglas difusas
NG NG NM NP CE
NG NM NP CE PP
NM NP CE PP PM
NP CE PP PM PG
CE PP PM PG PG
% Matriz de 49 reglas difusas
NG NG NG NM NM NP CE
NG NG NM NM NP CE PP
NG NM NM NP CE PP PM
NM NM NP CE PP PM PM
NM NP CE PP PM PM PG
NP CE PP PM PM PG PG
CE PP PM PM PG PG PG
% Autosintonización de reglas por Tabú: Si [1] No [2]
2
%*****
%Datos del Controlador difuso tipo 2:
% Universo de discurso del error
0.5
%Incertidumbre para conjuntos difusos del error
0.01
% Universo de discurso del cambio del error
0.01
%Incertidumbre para conjuntos difusos del cambio del error
0.0001
% Número de reglas difusas: 9 reglas [9] 25 reglas [25] 49 reglas [49]
9
% Numero de barras de salida: 5 [5] 7 [7]
7
%Uniformemente distribuidas: Si [1] No [2]
1
% Universo de discurso de Salida
3
% Matriz de 9 reglas difusas
NG NM CE
NP CE PP

```

```

CE PM PG
% Matriz de 25 reglas difusas
NG NG NM NP CE
NG NM NP CE PP
NM NP CE PP PM
NP CE PP PM PG
CE PP PM PG PG
% Matriz de 49 reglas difusas
NG NG NG NM NM NP CE
NG NG NM NM NP CE PP
NG NM NM NP CE PP PM
NM NM NP CE PP PM PM
NM NP CE PP PM PM PG
NP CE PP PM PM PG PG
CE PP PM PM PG PG
%Tipo de funciones de membresía: Triangular01 [1] Triangular02 [2] Triangular03 [3] Triangular diamante [4] Gaussiana
Tipo 2 [5]
5
%Algoritmo de reducción de tipo: KM [1] EKM [2]
2
% Autosintonización de reglas por Tabú: Si [1] No [2]
2
%*****
% Datos para la búsqueda Tabú
% Alfa
10
% Beta
125
% Caos
0.24
% Tiempo Base Tabú Eliminado
0.125
% Tiempo Base Tabú Eliminado
1
% Criterio de Aspiración : Baja Influencia [1] Alta Influencia [2]
1

%ARCHIVO-2 CAMBIOS DE CARGA
% Para Cambios de carga opción 1:
% Se efectuarán 4 cambios de carga como máximo o el número de cambios correspondiente antes de que suceda el tiempo
%final de simulación seleccionado
% El intervalo [Tiempo inicial estable Tiempo final estable] indica el tiempo en que la entrada al gobernador se mantiene
%constante.
% El intervalo [Tiempo inicial transición Tiempo final transición] indica el tiempo en el cual se realiza el incremento de carga
%en forma lineal desde un valor inicial hasta un valor final.
% Además Tiempo inicial 2 = Tiempo final 1 y así sucesivamente según:
% |
% |
% | 9 _____ 10
% | /
% | 7 _____ 8
% | /
% | 5 _____ 6
% | /
% | 3 _____ 4
% | /
% | 1 _____ 2
% 1 2
1.- Tiempo inicial estable 1
2.- Tiempo final estable 2 = Tiempo inicial transición 1
3.- Tiempo final transición 1 = Tiempo inicial estable 2
4.- Tiempo final estable 2 = Tiempo inicial transición 2
5.- Tiempo final transición 2 = Tiempo inicial estable 3
6.- Tiempo final estable 3 = Tiempo inicial transición 3
7.- Tiempo final transición 3 = Tiempo inicial estable 4
8.- Tiempo final estable 4 = Tiempo inicial transición 4
9.- Tiempo final transición 4 = Tiempo inicial estable 5
10.- Tiempo final estable 5

% Tiempo inicial estable 1 =
0
% Tiempo final estable 1 = Tiempo inicial transición 1
10
% Tiempo final transición 1 = Tiempo inicial estable 2
35
% Tiempo final estable 2 = Tiempo inicial transición 2

```

55
% Tiempo final transición 2 = Tiempo inicial estable 3
80
% Tiempo final estable 3 = Tiempo inicial transición 3
100
% Tiempo final transición 3 = Tiempo inicial estable 4
125
% Tiempo final estable 4 = Tiempo inicial transición 4
145
% Tiempo final transición 4 = Tiempo inicial estable 5
170
% Tiempo final estable 5
190
%Define incrementos de carga
%Incremento 1:
0.25
%Incremento 2:
0.25
%Incremento 3:
0.25
%Incremento 4:
0.25

F 1.3 Programa Principal

```

%-----
%SIMULACIÓN DE LA MAQUINA SINCRONA CON CONTROLES ST1, DIFUSO TIPO-1
% Y DIFUSO TIPO-2
%-----
clear all;
clc;
%-----
%VARIABLES GLOBALES
%Datos Constantes del Del Sistema
global ra re xe kd w0 M Tsm Tch Trh Fhp Tx Tq02p Td0p Td02p Xd Xdp
Xd2p
global Xq Xq2p
ra = 0.0;
kd = 3;
w0 = 377;
M = 5.5294;
Tsm = 0.1;
Tch = 0.15;
Trh = 5;
Fhp = 0.33;
Tx = 0.025;
Tq02p = 0.065;
Td0p = 5.66;
Td02p = 0.041;
Xd = 1.904;
Xdp = 0.312;
Xd2p = 0.266;
Xq = 1.881;
Xq2p = 0.260;

%Constantes del regulador de estado solido
global Ka Ta Kf Tf
Ka = 400;
Ta = 0.02;
Kf = 0.008;
Tf = 1;

%Datos del estudio de flujos
global Vter theta Vbus P Q
Corta = int8(1);
Larga = int8(2);
Base = 100;

%Variables globales generales
global U UG Te id iq Vref FID iter_CL_KM iter_CR_KM iter_CL_EKM
iter_CR_EKM

%Variables globales para datos de simulación
global Linea T_Final_Simulacion h DISTURBIO OPERACION
T_Inicial_Falla ...
CICLOS_FALLA T_Final_Falla T_Inicio_Carga A1 B1 C1 A2 B2 C2
...
DELTA_E DELTA_DE NUM_REGLAS_T1 NUM_BARRAS_T1 TABU_T1 ...
NUM_REGLAS_T2 NUM_BARRAS_T2 TABU_T2 TIPO_FUNCION_T2 ...
ALGORITM_RED_TIPO ALFA BETA CAOS TABU_SUMADO TABU_ELIMINADO
...
NUM_CANDIDATOS CRIT_ASP REGLAS_T1 REGLAS_T2 CANDIDATOS_T1
...
CANDIDATOS_T2 CENTROS_DE_BARRA_T1 CENTROS_DE_BARRA_T2
...
NUM_CONJ_ENTRADA_T1 NUM_CONJ_ENTRADA_T2
%-----
%DEFINICIÓN DE VARIABLES Y VECTORES

%Vector de Estado
dx_dt = zeros(11,1);

KM = int8(1);
EKM = int8(2);
SPREAD_LEFT = [0; 0; 0; 0; 0; 0; 0];
SPREAD_RIGHT = [0; 0; 0; 0; 0; 0; 0];

%Lee datos de Simulación de archivo de entrada
datos_simulacion();

%Dimensiona e inicializa vectores
DIM = cast(T_Final_Simulacion / h, 'int32') + 1;

%Vectores de salida
Tiempo = zeros(DIM,1);

%Vectores de salida para controlador ST1
UG_ST1 = zeros(DIM,1);
A_Carga_ST1 = zeros(DIM,1);
V_ter_ST1 = zeros(DIM,1);
P_Activa_ST1 = zeros(DIM,1);
P_Reactiva_ST1 = zeros(DIM,1);
V_ref_ST1 = zeros(DIM,1);
Error_ST1 = zeros(DIM,1);
Vector_ISE_ST1 = zeros(DIM,1);
Vector_ITAE_ST1 = zeros(DIM,1);
VectorAcum_ISE_ST1 = zeros(DIM,1);
VectorAcum_ITAE_ST1 = zeros(DIM,1);
Vector_ISE_ACarga_ST1 = zeros(DIM,1);
Vector_ITAE_ACarga_ST1 = zeros(DIM,1);
VectorAcum_ISE_ACarga_ST1 = zeros(DIM,1);
VectorAcum_ITAE_ACarga_ST1 = zeros(DIM,1);
UG_IT2FLC = zeros(DIM,1);
Salida_Regulador_ST1 = zeros(DIM,1);

%Vectores de salida para controlador difuso tipo 1
UG_T1FLC = zeros(DIM,1);
A_Carga_T1FLC = zeros(DIM,1);
V_ter_T1FLC = zeros(DIM,1);
P_Activa_T1FLC = zeros(DIM,1);
P_Reactiva_T1FLC = zeros(DIM,1);
Volt_Campo_T1FLC = zeros(DIM,1);
V_ref_T1FLC = zeros(DIM,1);
Error_T1FLC = zeros(DIM,1);
Cambio_del_Error_T1FLC = zeros(DIM,1);
Vector_ISE_T1FLC = zeros(DIM,1);
Vector_ITAE_T1FLC = zeros(DIM,1);
VectorAcum_ISE_T1FLC = zeros(DIM,1);
VectorAcum_ITAE_T1FLC = zeros(DIM,1);
Vector_ISE_ACarga_T1FLC = zeros(DIM,1);
Vector_ITAE_ACarga_T1FLC = zeros(DIM,1);
VectorAcum_ISE_ACarga_T1FLC = zeros(DIM,1);
VectorAcum_ITAE_ACarga_T1FLC = zeros(DIM,1);
Salida_Ctrl_T1FLC = zeros(DIM,1);

%Vectores de salida para controlador difuso tipo 2
A_Carga_IT2FLC = zeros(DIM,1);
V_ter_IT2FLC = zeros(DIM,1);
P_Activa_IT2FLC = zeros(DIM,1);
Volt_Campo_IT2FLC = zeros(DIM,1);
P_Reactiva_IT2FLC = zeros(DIM,1);
V_ref_IT2FLC = zeros(DIM,1);
Error_IT2FLC = zeros(DIM,1);
Cambio_del_Error_IT2FLC = zeros(DIM,1);
Vector_ISE_IT2FLC = zeros(DIM,1);
Vector_ITAE_IT2FLC = zeros(DIM,1);
VectorAcum_ISE_IT2FLC = zeros(DIM,1);
VectorAcum_ITAE_IT2FLC = zeros(DIM,1);
Vector_ISE_ACarga_IT2FLC = zeros(DIM,1);
Vector_ITAE_ACarga_IT2FLC = zeros(DIM,1);
VectorAcum_ISE_ACarga_IT2FLC = zeros(DIM,1);
VectorAcum_ITAE_ACarga_IT2FLC = zeros(DIM,1);
Salida_Ctrl_IT2FLC = zeros(DIM,1);

%Vector de almacenamiento de Vter con ruido
Vter_Ruido = zeros(DIM,1);

%Define los valores de resistencia y reactivancias segun tipo de linea
if (Linea == Corta)
    re = 0.024; xe = 0.115; Vbus = 1;
    if (OPERACION == 1)
        Vter = 1; theta = 0; P = 0; Q = 0;
    else
        Vter = 1; theta = 6.807699; P = 100/Base; Q = -14.74/Base;
    end
elseif (Linea == Larga)
    re = 0.12; xe = 1.1; Vbus = 1;
    if (OPERACION == 1)
        Vter = 1; theta = 0; P = 0; Q = 0;
    else
        Vter = 1; theta = 52.399673; P = 75/Base; Q = 27.26/Base;
    end
end

%Abre archivo de salida
[FID,msj_error] =
fopen('LC_KMvSEKM_T1(TKM)_T2(TEKM)_A=0.3_0.5_B=0.003_0.5_C=3_25R.DAT', 'w');
if msj_error == -1
    disp('No se pudo abrir fichero solicitado');
end

%Almacena copia de variables globales generales
Vter_aux = Vter;
theta_aux = theta;
P_aux = P;
Q_aux = Q;

%Calculo de condiciones iniciales
dx_dt = Condiciones_Iniciales();

%Inicializa contador

```

```

K = 1;
Error_ant = 0;

%Almacena los primeros datos resultados de las consiciones iniciales
UG_STL(K) = UG;
A_Carga_STL(K)= dx_dt(1) * 180 / pi;
V_ter_STL(K)= Vter;
P_Activa_STL(K)= P;
P_Reactiva_STL(K) = Q;
V_ref_STL(K) = (U / Ka) + Vter;
Salida_Regulador_STL(K) = U;

Vref_aux = Vref;
%Simulación de Máquina Síncrona con Regulador tipo ST1
for Tiempo_Actual = 0 : h : T_Final_Simulacion

    %if (Tiempo_Actual <= 20 && Tiempo_Actual >= 1)
    %   Vref = 1.05 + 0.005 * randn;
    %else
    %   Vref = Vref_aux + 0.005 * randn;
    %end

    %Pregunta por simulacion de falla
    if (DISTURBIO == 2)
        Vbus = Falla(Tiempo_Actual);
    else
        Cambio_De_Carga(Tiempo_Actual);
    end

    %Ingresa Ruido retroalimentacion
    %Vter = Vter + Vter * 0.03 * randn;

    %Calcula Error y Cambio del error
    Error = Vref - Vter;
    Cambio_del_Error = (Error - Error_ant);
    Error_ant = Error;

    %LLama funcion Runge Kutta
    dx_dt = Runge_Kutta_4to(dx_dt);

    %Entrada de control
    U = dx_dt(10);

    %Limitador de la salida del controlador
    if U >= 4, U = 4; end
    if U <= 0, U = 0; end

    %Resuelve Ecuaciones Dinamicas
    Ecs_Dinamicas(dx_dt);

    %Almacena variables de salida en valores reales
    Tiempo(K+1) = Tiempo_Actual;
    V_ref_STL(K+1) = Vref;
    UG_STL(K+1) = UG;
    A_Carga_STL(K+1)= dx_dt(1) * 180 / pi;
    V_ter_STL(K+1)= Vter;
    Volt_Campo_STL(K+1) = dx_dt(6);
    P_Activa_STL(K+1)= P;
    P_Reactiva_STL(K+1) = Q;
    Salida_Regulador_STL(K+1) = U;
    Error_STL(K+1) = Error;
    Cambio_del_Error_STL(K+1) = Cambio_del_Error;

    %Vectores de índices instantáneos del error del voltaje terminal
    Vector_ISE_STL(K+1) = Error*Error;
    Vector_IAE_STL(K+1) = abs(Error);
    Vector_ITAE_STL(K+1) = abs(Error)*Tiempo_Actual;
    %Vectores de índices acumulativos del error del voltaje terminal
    VectorAcum_ISE_STL(K+1) = VectorAcum_ISE_STL(K) +
    Vector_ISE_STL(K+1);
    VectorAcum_IAE_STL(K+1) = VectorAcum_IAE_STL(K) +
    Vector_IAE_STL(K+1);
    VectorAcum_ITAE_STL(K+1) = VectorAcum_ITAE_STL(K) +
    Vector_ITAE_STL(K+1);

    %Calcula el error del ángulo de carga
    error_ACarga = A_Carga_STL(1) - A_Carga_STL(K+1);
    %if (Tiempo_Actual <= 20 && Tiempo_Actual >= 1)
    %   error_ACarga = 54.491 - A_Carga_STL(K+1);
    % end
    %Vectores de índices instantáneos del error del ángulo de carga
    Vector_ISE_ACarga_STL(K+1) = error_ACarga*error_ACarga;
    Vector_IAE_ACarga_STL(K+1) = abs(error_ACarga);
    Vector_ITAE_ACarga_STL(K+1) = abs(error_ACarga)*Tiempo_Actual;
    %Vectores de índices acumulativos del error del ángulo de carga
    VectorAcum_ISE_ACarga_STL(K+1) = VectorAcum_ISE_ACarga_STL(K) +
    Vector_ISE_ACarga_STL(K+1);
    VectorAcum_IAE_ACarga_STL(K+1) = VectorAcum_IAE_ACarga_STL(K) +
    Vector_IAE_ACarga_STL(K+1);
    VectorAcum_ITAE_ACarga_STL(K+1) = VectorAcum_ITAE_ACarga_STL(K) +
    Vector_ITAE_ACarga_STL(K+1);

    %Incrementa contador
    K = K + 1;
end

%Calcula índices del error
ISE_ST1 = Fnc_Suma_Trap(Vector_ISE_ST1,h);
IAE_ST1 = Fnc_Suma_Trap(Vector_IAE_ST1,h);

ITAE_ST1 = Fnc_Suma_Trap(Vector_ITAE_ST1,h);

%Reinicia variables globales y vector de estado
Vter = Vter_aux;
theta = theta_aux;
P = P_aux;
Q = Q_aux;

%Calculo de condiciones iniciales
dx_dt = Condiciones_Iniciales();

%Voltaje de referencia para controlador difuso
Vref = Vter;
Vref_aux = Vref;

%Inicializa contador
K = 1;

%Almacena los primeros datos resultados de las consiciones iniciales
UG_T1FLC(K) = UG;
A_Carga_T1FLC(K)= dx_dt(1) * 180 / pi;
V_ter_T1FLC(K)= Vter;
P_Activa_T1FLC(K)= P;
P_Reactiva_T1FLC(K) = Q;
Volt_Campo_T1FLC(K) = dx_dt(6);
Salida_Ctrl_IT2FLC(K) = U;
Error_T1FLC(K) = Vref - Vter;

Error_ant = 0;
TSTART = tic

%Simulación de Máquina Síncrona con Regulador Inteligente T1FLC
for Tiempo_Actual = 0 : h : T_Final_Simulacion
    %if (Tiempo_Actual <= 30 && Tiempo_Actual >= 10)
    %   Vref = 1.05 + 0.0033 * randn;
    %else
    %   Vref = Vref_aux + 0.0033 * randn;
    %end
    %Pregunta por simulacion de falla
    if (DISTURBIO == 2)
        Vbus = Falla(Tiempo_Actual);
    else
        Cambio_De_Carga(Tiempo_Actual);
    end

    %LLama funcion Runge Kutta
    dx_dt = Runge_Kutta_4to(dx_dt);

    %Resuelve Ecuaciones Dinamicas
    Ecs_Dinamicas(dx_dt);

    %Ingresa Ruido retroalimentacion
    RUIDO = 0;%Vter * 0.03 * randn;

    %Calcula Error y Cambio del error
    Error = Vref - (Vter + RUIDO);
    Cambio_del_Error = (Error - Error_ant);
    Error_ant = Error;

    REGLAS = REGLAS_T1;
    if TABU_T1 == 1
        %Llama a función búsqueda tabu, memoria corto plazo
        REGLAS_T1
        Busqueda_Tabu_Mem_CortoPlazo_Prueba02(Tiempo_Actual,...
        A1,B1,0,0,REGLAS,CENTROS_DE_BARRA_T1,NUM_CONJ_ENTRADA_T1,...
        CANDIDATOS_T1,dx_dt,Error,Cambio_del_Error);
    end

    %Controlador Difuso tipo 1
    if ALGORITM_RED_TIPO == KM
        U = Control_Intervalo_Tipo2_KM(A1,B1,0,0,0,0,Error,...
        Cambio_del_Error,REGLAS_T1,CENTROS_DE_BARRA_T1,...
        TIPO_FUNCION_T2,NUM_CONJ_ENTRADA_T1,U);
    elseif ALGORITM_RED_TIPO == EKM
        U = Control_Intervalo_Tipo2_EKM(A1,B1,0,0,0,0,Error,...
        Cambio_del_Error,REGLAS_T1,CENTROS_DE_BARRA_T1,...
        TIPO_FUNCION_T2,NUM_CONJ_ENTRADA_T1,U);
    end

    %Limitador de salida del controlador
    if U >= 4, U = 4; end
    if U <= 0, U = 0; end

    %Almacena variables de salida en valores reales
    Tiempo(K+1) = Tiempo_Actual;
    UG_T1FLC(K+1) = UG;
    A_Carga_T1FLC(K+1)= dx_dt(1) * 180 / pi;
    V_ter_T1FLC(K+1)= Vter;
    P_Activa_T1FLC(K+1)= P;
    P_Reactiva_T1FLC(K+1) = Q;
    Volt_Campo_T1FLC(K+1) = dx_dt(6);
    Salida_Ctrl_T1FLC(K+1) = U;
    Error_T1FLC(K+1) = Error;
    Cambio_del_Error_T1FLC(K+1) = Cambio_del_Error;

    %Vectores de índices instantáneos del error del voltaje terminal
    Vector_ISE_T1FLC(K+1) = Error*Error;
    Vector_IAE_T1FLC(K+1) = abs(Error);

```

```

Vector_ITAE_T1FLC(K+1) = abs(Error)*Tiempo_Actual;
%Vectores de índices acumulativos del error del voltaje terminal
VectorAcum_ISE_T1FLC(K+1) = VectorAcum_ISE_T1FLC(K) +
Vector_ISE_T1FLC(K+1);
VectorAcum_IAE_T1FLC(K+1) = VectorAcum_IAE_T1FLC(K) +
Vector_IAE_T1FLC(K+1);
VectorAcum_ITAE_T1FLC(K+1) = VectorAcum_ITAE_T1FLC(K) +
Vector_ITAE_T1FLC(K+1);

%Calcula el error del ángulo de carga
error_ACarga = A_Carga_T1FLC(1) - A_Carga_T1FLC(K+1);
%if (Tiempo_Actual <= 20 && Tiempo_Actual >= 1)
%   error_ACarga = 54.491 - A_Carga_T1FLC(K+1);
% end
%Vectores de índices instantáneos del error del ángulo de carga
Vector_ISE_ACarga_T1FLC(K+1) = error_ACarga*error_ACarga;
Vector_IAE_ACarga_T1FLC(K+1) = abs(error_ACarga);
Vector_ITAE_ACarga_T1FLC(K+1) = abs(error_ACarga)*Tiempo_Actual;
%Vectores de índices acumulativos del error del ángulo de carga
VectorAcum_ISE_ACarga_T1FLC(K+1) =
VectorAcum_ISE_ACarga_T1FLC(K) + Vector_ISE_ACarga_T1FLC(K+1);
VectorAcum_IAE_ACarga_T1FLC(K+1) =
VectorAcum_IAE_ACarga_T1FLC(K) + Vector_IAE_ACarga_T1FLC(K+1);
VectorAcum_ITAE_ACarga_T1FLC(K+1) =
VectorAcum_ITAE_ACarga_T1FLC(K) + Vector_ITAE_ACarga_T1FLC(K+1);

%Incrementa contador
K = K + 1;

end

TSTOP = toc(TSTART);

%Calcula índices del error
ISE_T1 = Fnc_Suma_Trap(Vector_ISE_T1FLC,h);
IAE_T1 = Fnc_Suma_Trap(Vector_IAE_T1FLC,h);
ITAE_T1 = Fnc_Suma_Trap(Vector_ITAE_T1FLC,h);

%Reinicia variables globales y vector de estado
Vter = Vter_aux;
theta = theta_aux;
P = P_aux;
Q = Q_aux;

%Calculo de condiciones iniciales
dx_dt = Condiciones_Iniciales();

%Voltaje de referencia para controlador difuso
Vref = Vter;
Vref_aux = Vref;

%Inicializa contador
K = 1;

%Almacena los primeros datos resultados de las consiciones iniciales
UG_IT2FLC(K) = UG;
A_Carga_IT2FLC(K)= dx_dt(1) * 180 / pi;
V_ter_IT2FLC(K)= Vter;
P_Activa_IT2FLC(K)= P;
P_Reactiva_IT2FLC(K) = Q;
Volt_Campo_IT2FLC(K) = dx_dt(6);
Salida_Ctrl_T1FLC(K) = U;
Error_IT2FLC(K) = Vref - Vter;

Vter_Ruido(K) = Vter;
Vref_ruido(K+1) = Vref;

Error_ant = 0;
TSTART = tic;
%Simulación de Máquina Síncrona con Regulador Inteligente IT2FLC
for Tiempo_Actual = 0 : h : T_Final_Simulacion
%if (Tiempo_Actual <= 30 && Tiempo_Actual >= 10)
%   Vref = 1.05 + 0.0033 * randn;
%else
%   Vref = Vref_aux + 0.0033 * randn;
%end

Vref_ruido(K+1) = Vref;
%Pregunta por simulacion de falla
if (DISTURBIO == 2)
    Vbus = Falla(Tiempo_Actual);
else
    Cambio_De_Carga(Tiempo_Actual);
end

%Llama funcion Runge Kutta
dx_dt = Runge_Kutta_4to(dx_dt);

%Resuelve Ecuaciones Dinamicas
Ecs_Dinamicas(dx_dt);

%Ingresa Ruido retroalimentacion
RUIDO = 0;%Vter * 0.03 *randn;

Vter_Ruido(K+1) = Vter + RUIDO;

%Calcula Error y Cambio del error
Error = Vref - (Vter + RUIDO);
Cambio_del_Error = (Error - Error_ant);

Error_ant = Error;

REGLAS = REGLAS_T2;
if TABU_T2 == 1
    %Llama a función búsqueda tabu, memoria corto plazo
    REGLAS_T2 =
Busqueda_Tabu_Mem_CortoPlazo_Prueba02(Tiempo_Actual,...
    A2,B2,DELTA_E,DELTA_DE,REGLAS,CENTROS_DE_BARRA_T2,
    ...
NUM_CONJ_ENTRADA_T2,CANDIDATOS_T2,dx_dt,Error,Cambio_del_Error);
end

%Controlador Difuso tipo 2
if ALGORITM_RED_TIPO == KM
    U = Control_Intervalo_Tipo2_KM(A2,B2,DELTA_E,DELTA_DE,
    ...
SPREAD_LEFT,SPREAD_RIGHT,Error,Cambio_del_Error,REGLAS_T2,...
CENTROS_DE_BARRA_T2,TIPO_FUNCION_T2,NUM_CONJ_ENTRADA_T2,U);
elseif ALGORITM_RED_TIPO == EKM
    U = Control_Intervalo_Tipo2_EKM(A2,B2,DELTA_E,DELTA_DE,
    ...
SPREAD_LEFT,SPREAD_RIGHT,Error,Cambio_del_Error,REGLAS_T2,...
CENTROS_DE_BARRA_T2,TIPO_FUNCION_T2,NUM_CONJ_ENTRADA_T2,U);
end

%Limitador de salida del controlador
if U >= 4, U = 4; end
if U <= 0, U = 0; end

%Almacena variables de salida en valores reales
Tiempo(K+1) = Tiempo_Actual;
UG_IT2FLC(K+1) = UG;
A_Carga_IT2FLC(K+1)= dx_dt(1) * 180 / pi;
V_ter_IT2FLC(K+1)= Vter;
P_Activa_IT2FLC(K+1)= P;
P_Reactiva_IT2FLC(K+1) = Q;
Volt_Campo_IT2FLC(K+1) = dx_dt(6);
Salida_Ctrl_IT2FLC(K+1) = U;
Error_IT2FLC(K+1) = Error;
Cambio_del_Error_IT2FLC(K+1) = Cambio_del_Error;

%Vectores de índices instantáneos del error del voltaje terminal
Vector_ISE_IT2FLC(K+1) = Error*Error;
Vector_IAE_IT2FLC(K+1) = abs(Error);
Vector_ITAE_IT2FLC(K+1) = abs(Error)*Tiempo_Actual;
%Vectores de índices acumulativos del error del voltaje terminal
VectorAcum_ISE_IT2FLC(K+1) = VectorAcum_ISE_IT2FLC(K) +
Vector_ISE_IT2FLC(K+1);
VectorAcum_IAE_IT2FLC(K+1) = VectorAcum_IAE_IT2FLC(K) +
Vector_IAE_IT2FLC(K+1);
VectorAcum_ITAE_IT2FLC(K+1) = VectorAcum_ITAE_IT2FLC(K) +
Vector_ITAE_IT2FLC(K+1);

%Calcula el error del ángulo de carga
error_ACarga = A_Carga_IT2FLC(1) - A_Carga_IT2FLC(K+1);
%if (Tiempo_Actual <= 20 && Tiempo_Actual >= 1)
%   error_ACarga = 54.491 - A_Carga_IT2FLC(K+1);
% end
%Vectores de índices instantáneos del error del ángulo de carga
Vector_ISE_ACarga_IT2FLC(K+1) = error_ACarga*error_ACarga;
Vector_IAE_ACarga_IT2FLC(K+1) = abs(error_ACarga);
Vector_ITAE_ACarga_IT2FLC(K+1) =
abs(error_ACarga)*Tiempo_Actual;
%Vectores de índices acumulativos del error del ángulo de carga
VectorAcum_ISE_ACarga_IT2FLC(K+1) =
VectorAcum_ISE_ACarga_IT2FLC(K) + Vector_ISE_ACarga_IT2FLC(K+1);
VectorAcum_IAE_ACarga_IT2FLC(K+1) =
VectorAcum_IAE_ACarga_IT2FLC(K) + Vector_IAE_ACarga_IT2FLC(K+1);
VectorAcum_ITAE_ACarga_IT2FLC(K+1) =
VectorAcum_ITAE_ACarga_IT2FLC(K) + Vector_ITAE_ACarga_IT2FLC(K+1);

%Incrementa contador
K = K + 1;

end

TSTOP = toc(TSTART);

%Calcula índices del error
ISE_T2 = Fnc_Suma_Trap(Vector_ISE_IT2FLC,h)
IAE_T2 = Fnc_Suma_Trap(Vector_IAE_IT2FLC,h)
ITAE_T2 = Fnc_Suma_Trap(Vector_ITAE_IT2FLC,h)

%Escribe resultados en archivo de salida
fprintf(FID,'Tiempo\titer\tVolt. Terminal ST1\tVolt. Terminal
T1\tVolt. Terminal T2\tA. Carga ST1\tA. Carga T1\tA. Carga T2\tV.
Campo ST1\tV. Campo T1\tV. Campo T2\tP. Activa ST1\tP. Activa T1\tP.
Activa T2\tP. Reactiva ST1\tP. Reactiva T1\tP. Reactiva T2\tU.
Control ST1\tU. Control T1\tU. Control T2\tError ST1\tCambio derror
ST1\tError T1\tCambio derror T1\tError T2\tCambio derror
T2\tUGT1\tISE ST1\tIAE ST1\tITAE ST1\tISEAcum ST1\tIAEAcum
ST1\tITAEAcum ST1\tISE T1\tIAE T1\tITAE T1\tISEAcum T1\tIAEAcum
T1\tITAEAcum T1\tISE T2\tIAE T2\tITAE T2\tISEAcum T2\tIAEAcum
T2\tITAEAcum T2\tISE ACarga ST1\tIAE ACarga ST1\tITAE ACarga
ST1\tISE ACarga Acum ST1\tIAE ACarga Acum ST1\tITAE ACarga Acum

```



```

plot(Tiempo,VectorAcum_ISE_ST1,'green:',Tiempo,VectorAcum_ISE_T1FLC,
....
'blue--',Tiempo,VectorAcum_ISE_IT2FLC,'red-', 'lineWidth',1.5);
title('ACUMULATIVO ISE','FontName','timesnewroman','FontSize',18);
xlabel('Tiempo
(Segundos)','FontName','timesnewroman','FontSize',16);
ylabel('ISE','FontName','timesnewroman','FontSize',16);
grid on;
axis square;

figure(14);
plot(Tiempo,VectorAcum_IAE_ST1,'green:',Tiempo,VectorAcum_IAE_T1FLC,
....
'blue--',Tiempo,VectorAcum_IAE_IT2FLC,'red-', 'lineWidth',1.5);
title('ACUMULATIVO IAE','FontName','timesnewroman','FontSize',18);
xlabel('Tiempo
(Segundos)','FontName','timesnewroman','FontSize',16);
ylabel('IAE','FontName','timesnewroman','FontSize',16);
grid on;
axis square;

figure(15);
plot(Tiempo,VectorAcum_ITAE_ST1,'green:',Tiempo,VectorAcum_ITAE_T1FLC,
C,....
'blue--',Tiempo,VectorAcum_ITAE_IT2FLC,'red-', 'lineWidth',1.5);
title('ACUMULATIVO ITAE','FontName','timesnewroman','FontSize',18);
xlabel('Tiempo
(Segundos)','FontName','timesnewroman','FontSize',16);
ylabel('ITAE','FontName','timesnewroman','FontSize',16);
grid on;
axis square;

figure(16);
plot(Tiempo,Vter_Ruido,'red-', 'lineWidth',1.5);
title('Vter','FontName','timesnewroman','FontSize',18);
xlabel('Tiempo
(Segundos)','FontName','timesnewroman','FontSize',16);
ylabel('Volts','FontName','timesnewroman','FontSize',16);
grid on;
axis square;

function [ ] = datos_simulacion( )
% Solicita los datos de tiempos requeridos de simulación
% La función retorna la variable h = paso de simulación.
% Los argumentos de entrada son variables globales que se modifican
global Linea T_Final_Simulacion h DISTURBIO OPERACION
T_Inicial_Falla ...
CICLOS_FALLA T_Final_Falla T_Inicio_Carga T_Final_Carga
Cambios_Carga A1 ...
B1 C1 A2 B2 C2 DELTA_E DELTA_DE NUM_REGLAS_T1 NUM_BARRAS_T1
UNIF_C1 TABU_T1 NUM_REGLAS_T2 ...
NUM_BARRAS_T2 UNIF_C2 TABU_T2 TIPO_FUNCION_T2
ALGORITM_RED_TIPO ALFA BETA ...
CAOS TABU_SUMADO TABU_ELIMINADO NUM_CANDIDATOS CRIT_ASP ...
REGLAS_T1 REGLAS_T2 NUM_CONJ_ENTRADA_T1 NUM_CONJ_ENTRADA_T2

% Abre archivo de datos de entrada
FID = fopen('Datos_Entrada.txt','r+');

% Almacena datos en una celda
DATOS(1) = textscan(FID,'%d8',1,'CommentStyle','%');
DATOS(2) = textscan(FID,'%f64',2,'CommentStyle','%');
DATOS(3) = textscan(FID,'%d8',2,'CommentStyle','%');
DATOS(4) = textscan(FID,'%f64',4,'CommentStyle','%');
DATOS(5) = textscan(FID,'%f32',2,'CommentStyle','%');
DATOS(6) = textscan(FID,'%d8',1,'CommentStyle','%');
DATOS(7) = textscan(FID,'%f64',1,'CommentStyle','%');
DATOS(8) = textscan(FID,'%s',9,'CommentStyle','%');
DATOS(9) = textscan(FID,'%s',25,'CommentStyle','%');
DATOS(10) = textscan(FID,'%s',49,'CommentStyle','%');
DATOS(11) = textscan(FID,'%d8',1,'CommentStyle','%');
DATOS(12) = textscan(FID,'%f64',4,'CommentStyle','%');
DATOS(13) = textscan(FID,'%f32',2,'CommentStyle','%');
DATOS(14) = textscan(FID,'%d8',1,'CommentStyle','%');
DATOS(15) = textscan(FID,'%f64',1,'CommentStyle','%');
DATOS(16) = textscan(FID,'%s',9,'CommentStyle','%');
DATOS(17) = textscan(FID,'%s',25,'CommentStyle','%');
DATOS(18) = textscan(FID,'%s',49,'CommentStyle','%');
DATOS(19) = textscan(FID,'%d8',3,'CommentStyle','%');
DATOS(20) = textscan(FID,'%f32',5,'CommentStyle','%');
DATOS(21) = textscan(FID,'%d8',1,'CommentStyle','%');

% Copia datos de la celda DATOS a vectores
Y1 = DATOS{1,2};
Y2 = DATOS{1,3};
Y3 = DATOS{1,4};
Y4 = DATOS{1,5};
Y5 = DATOS{1,8};
Y6 = DATOS{1,9};
Y7 = DATOS{1,10};
Y8 = DATOS{1,12};
Y9 = DATOS{1,13};
Y10 = DATOS{1,16};
Y11 = DATOS{1,17};

Y12 = DATOS{1,18};
Y13 = DATOS{1,19};
Y14 = DATOS{1,20};

% Copia datos de vectores a variables
Linea = int8(DATOS{1,1});
T_Final_Simulacion = Y1(1);
h = Y1(2);
DISTURBIO = int8(Y2(1));
OPERACION = int8(Y2(2));
T_Inicial_Falla = Y3(1);
CICLOS_FALLA = Y3(2);
% Calcula el tiempo final de la falla
T_Duracion_Falla = CICLOS_FALLA * 1 / 60;
T_Final_Falla = T_Inicial_Falla + T_Duracion_Falla;

A1 = Y3(3);
B1 = Y3(4);
NUM_REGLAS_T1 = Y4(1);
NUM_CONJ_ENTRADA_T1 = sqrt(NUM_REGLAS_T1);
NUM_BARRAS_T1 = Y4(2);
UNIF_C1 = int8(DATOS{1,6});
C1 = DATOS{1,7};
% Llama a función Fam_T1 para definir la matriz de control para el
control
% tipo 1
Fam_T1(Y5,Y6,Y7);
TABU_T1 = int8(DATOS{1,11});
A2 = Y8(1);
DELTA_E = Y8(2);
B2 = Y8(3);
DELTA_DE = Y8(4);
NUM_REGLAS_T2 = Y9(1);
NUM_CONJ_ENTRADA_T2 = sqrt(NUM_REGLAS_T2);
NUM_BARRAS_T2 = Y9(2);
UNIF_C2 = int8(DATOS{1,14});
C2 = DATOS{1,15};
% Llama a función Fam_T1 para definir la matriz de control para el
control
% tipo 2
Fam_T2(Y10,Y11,Y12);
TIPO_FUNCION_T2 = int8(Y13(1));
ALGORITM_RED_TIPO = int8(Y13(2));
TABU_T2 = int8(Y13(3));
ALFA = Y14(1);
BETA = Y14(2);
CAOS = Y14(3);
TABU_SUMADO = Y14(4);
TABU_ELIMINADO = Y14(5);
CRIT_ASP = int8(DATOS{1,21});

% Cierra archivo de datos de entrada
fclose(FID);

% Abre archivo de datos de cambios de carga
FID = fopen('Datos_Cambios_de_Carga.txt','r+');

% Almacena datos en una celda
DATOS2(1) = textscan(FID,'%f32',14,'CommentStyle','%');

% Copia datos de la celda DATOS2 a un vector
Y16 = DATOS2{1,1};
T_Inicio_Carga(1) = Y16(1);
T_Final_Carga(1) = Y16(2);
T_Inicio_Carga(2) = Y16(3);
T_Final_Carga(2) = Y16(4);
T_Inicio_Carga(3) = Y16(5);
T_Final_Carga(3) = Y16(6);
T_Inicio_Carga(4) = Y16(7);
T_Final_Carga(4) = Y16(8);
T_Inicio_Carga(5) = Y16(9);
T_Final_Carga(5) = Y16(10);
Cambios_Carga(1) = Y16(11);
Cambios_Carga(2) = Y16(12);
Cambios_Carga(3) = Y16(13);
Cambios_Carga(4) = Y16(14);

% Cierra archivo de datos de entrada
fclose(FID);

end

function [ dx0_dt0 ] = Condiciones_Iniciales()
% Función que calcula las condiciones iniciales para la operación en
estado
% permanente. Estos son calculados al hacer todas las derivadas de
las
% ecuaciones del modelo, iguales a cero.
% Vector de estado:
% dx_dt(1) = delta_r
% dx_dt(2) = s
% dx_dt(3) = eqp
% dx_dt(4) = eq2p
% dx_dt(5) = ed2p
% dx_dt(6) = Vf
% dx_dt(7) = Pgv

```

F 1.4 Funciones secundarias

```

% dx_dt(8) = Php
% dx_dt(9) = Tm
% dx_dt(10) = U
% dx_dt(11) = Vexc

%Variables globales
global Vbus id iq Te Vref UG ra Xq Xd Xdp Xd2p Xq2p Vter theta P Q
Ka U

%Variable de salida
dx0_dt0 = zeros(11,1);

%Convierte angulo en radianes
theta = theta * pi / 180;

%Calcula ia
ia = sqrt(P^2 + Q^2) / Vter;
if (ia > 0.000001)
    %Calcula ángulo phi en radianes
    phi = atan(Q/P);
else
    phi = 0;
end

e_qx = Vter+(ra*ia*cos(phi)) + (Xq*ia*sin(phi));
e_qy = -(ra*ia*sin(phi)) + (Xq*ia*cos(phi));
delta = atan(e_qy/e_qx);

%Calcula id, iq
iq = ia * cos(phi + delta);
id = ia * sin(phi + delta);

ed2p = (Vter*sin(delta)) + (ra*id) - (Xq2p*iq);
eq2p = (Vter*cos(delta)) + (ra*iq) + (Xd2p*id);
eqp = eq2p + ((Xd - Xdp)*id);

%Calcula Angulo de carga
delta_r = (theta + delta);

%Calcula Par Eléctrico
Te = (id*ed2p) + (iq*eq2p);

Vf = eqp + ((Xd - Xdp)*id)
U = Vf; %%%%%%%%%

Vref = (U / Ka) + Vter

Tm = Te;
Php = Tm;
Pgv = Php;
UG = Pgv;

Vexc = 0;
s = 0;

%Copia resultados a vector xi
dx0_dt0(1) = delta_r;
dx0_dt0(2) = s;
dx0_dt0(3) = eqp;
dx0_dt0(4) = eq2p;
dx0_dt0(5) = ed2p;
dx0_dt0(6) = Vf;
dx0_dt0(7) = Pgv;
dx0_dt0(8) = Php;
dx0_dt0(9) = Tm;
dx0_dt0(10) = U; %Ka*(Vref - Vter)
dx0_dt0(11) = Vexc;
end

function [ ] = Fam_T1(Y5,Y6,Y7)
global CANDIDATOS_T1 CENTROS_DE_BARRA_T1 REGLAS_T1 NUM_REGLAS_T1 ...
    NUM_BARRAS_T1 UNIF_C1 C1
%Define matriz de asociación difusa (matriz de reglas) en base a
datos del
%archivo de entrada.
Si = 1;
No = 2;
m = 1;
if NUM_REGLAS_T1 == 9
    if NUM_BARRAS_T1 == 5
        if UNIF_C1 == Si
            Cy = C1 / 2; %para i = 2, Cy es el mismo para toda i
            for i = -2 : 1 : 2
                CENTROS_DE_BARRA_T1(m,1) = Cy * i;
                m = m + 1;
            end
        else
            Cy = C1 / 4; %para i = 2, Cy es el mismo para toda i
            for i = -2 : 1 : 2
                CENTROS_DE_BARRA_T1(m,1) = Cy * sign(i) * i^2;
                m = m + 1;
            end
        end
    else
        k = 0;
        for i = 1 : sqrt(NUM_REGLAS_T1)
            for j = 1 : sqrt(NUM_REGLAS_T1)
                k = k + 1;
                if strcmp(Y5{k,1}, 'NG')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
                if strcmp(Y5{k,1}, 'NM')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
                if strcmp(Y5{k,1}, 'NP')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
                if strcmp(Y5{k,1}, 'CE')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(2); end
                if strcmp(Y5{k,1}, 'PP')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(3); end
                if strcmp(Y5{k,1}, 'PM')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
                if strcmp(Y5{k,1}, 'PG')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
                if strcmp(Y5{k,1}, 'PG')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
                end
            end
        end
    end
elseif NUM_REGLAS_T1 == 25
    if NUM_BARRAS_T1 == 5
        if UNIF_C1 == Si
            Cy = C1 / 2; %para i = 2, Cy es el mismo para toda i
            for i = -2 : 1 : 2
                CENTROS_DE_BARRA_T1(m,1) = Cy * i;
                m = m + 1;
            end
        else
            Cy = C1 / 4; %para i = 2, Cy es el mismo para toda i
            for i = -2 : 1 : 2
                CENTROS_DE_BARRA_T1(m,1) = Cy * sign(i) * i^2;
                m = m + 1;
            end
        end
    else
        k = 0;
        for i = 1 : sqrt(NUM_REGLAS_T1)
            for j = 1 : sqrt(NUM_REGLAS_T1)
                k = k + 1;
                if strcmp(Y6{k,1}, 'NG')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
                if strcmp(Y6{k,1}, 'NM')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
                if strcmp(Y6{k,1}, 'NP')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
                if strcmp(Y6{k,1}, 'CE')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(2); end
                if strcmp(Y6{k,1}, 'PP')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(3); end
                if strcmp(Y6{k,1}, 'PM')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
                if strcmp(Y6{k,1}, 'PM')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
                if strcmp(Y6{k,1}, 'PG')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
                if strcmp(Y6{k,1}, 'PG')
                    REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
                end
            end
        end
    end
end
if NUM_BARRAS_T1 == 7
    if UNIF_C1 == Si
        Cy = C1 / 3; %para i = 2, Cy es el mismo para toda i
        for i = -3 : 1 : 3
            CENTROS_DE_BARRA_T1(m,1) = Cy * i;
            m = m + 1;
        end
    else
        Cy = C1 / 9; %para i = 2, Cy es el mismo para toda i
        for i = -3 : 1 : 3
            CENTROS_DE_BARRA_T1(m,1) = Cy * sign(i) * i^2;
            m = m + 1;
        end
    end
end
k = 0;
for i = 1 : sqrt(NUM_REGLAS_T1)
    for j = 1 : sqrt(NUM_REGLAS_T1)
        k = k + 1;
        if strcmp(Y5{k,1}, 'NG')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
        if strcmp(Y5{k,1}, 'NM')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
        if strcmp(Y5{k,1}, 'NP')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
        if strcmp(Y5{k,1}, 'CE')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(2); end
        if strcmp(Y5{k,1}, 'PP')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(3); end
        if strcmp(Y5{k,1}, 'PM')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
        if strcmp(Y5{k,1}, 'PM')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
        if strcmp(Y5{k,1}, 'PG')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
        if strcmp(Y5{k,1}, 'PG')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
        end
    end
end
end
if NUM_BARRAS_T1 == 7
    if UNIF_C1 == Si
        Cy = C1 / 3; %para i = 2, Cy es el mismo para toda i
        for i = -3 : 1 : 3
            CENTROS_DE_BARRA_T1(m,1) = Cy * i;
            m = m + 1;
        end
    else
        Cy = C1 / 9; %para i = 2, Cy es el mismo para toda i
        for i = -3 : 1 : 3
            CENTROS_DE_BARRA_T1(m,1) = Cy * sign(i) * i^2;
            m = m + 1;
        end
    end
end
k = 0;
for i = 1 : sqrt(NUM_REGLAS_T1)
    for j = 1 : sqrt(NUM_REGLAS_T1)
        k = k + 1;
        if strcmp(Y5{k,1}, 'NG')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
        if strcmp(Y5{k,1}, 'NM')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
        if strcmp(Y5{k,1}, 'NP')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
        if strcmp(Y5{k,1}, 'CE')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(2); end
        if strcmp(Y5{k,1}, 'PP')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(3); end
        if strcmp(Y5{k,1}, 'PM')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
        if strcmp(Y5{k,1}, 'PM')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
        if strcmp(Y5{k,1}, 'PG')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
        if strcmp(Y5{k,1}, 'PG')
            REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
        end
    end
end
end

```

```

CENTROS_DE_BARRA_T1(m,1) = Cy * sign(i) * i^2;
m = m + 1;
end
end
k = 0;
for i = 1 : sqrt(NUM REGLAS_T1)
for j = 1 : sqrt(NUM REGLAS_T1)
k = k + 1;
if strcmp(Y6{k,1},'NG')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
if strcmp(Y6{k,1},'NM')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(2); end
if strcmp(Y6{k,1},'NP')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(3); end
if strcmp(Y6{k,1},'CE')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
if strcmp(Y6{k,1},'PP')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
if strcmp(Y6{k,1},'PM')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(6); end
if strcmp(Y6{k,1},'PG')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(7); end
end
end
elseif NUM REGLAS_T1 == 49
if NUM BARRAS_T1 == 5
if UNIF_C1 == S1
Cy = C1 / 2; %para i = 2, Cy es el mismo para toda i
for i = -2 : 1 : 2
CENTROS_DE_BARRA_T1(m,1) = Cy * i;
m = m + 1;
end
else
Cy = C1 / 4; %para i = 2, Cy es el mismo para toda i
for i = -2 : 1 : 2
CENTROS_DE_BARRA_T1(m,1) = Cy * sign(i) * i^2;
m = m + 1;
end
end
k = 0;
for i = 1 : sqrt(NUM REGLAS_T1)
for j = 1 : sqrt(NUM REGLAS_T1)
k = k + 1;
if strcmp(Y7{k,1},'NG')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
if strcmp(Y7{k,1},'NM')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
if strcmp(Y7{k,1},'NP')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(2); end
if strcmp(Y7{k,1},'CE')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(3); end
if strcmp(Y7{k,1},'PP')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
if strcmp(Y7{k,1},'PM')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
if strcmp(Y7{k,1},'PG')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
end
end
if NUM BARRAS_T1 == 7
if UNIF_C1 == S1
Cy = C1 / 3; %para i = 2, Cy es el mismo para toda i
for i = -3 : 1 : 3
CENTROS_DE_BARRA_T1(m,1) = Cy * i;
m = m + 1;
end
else
Cy = C1 / 9; %para i = 2, Cy es el mismo para toda i
for i = -3 : 1 : 3
CENTROS_DE_BARRA_T1(m,1) = Cy * sign(i) * i^2;
m = m + 1;
end
end
k = 0;
for i = 1 : sqrt(NUM REGLAS_T1)
for j = 1 : sqrt(NUM REGLAS_T1)
k = k + 1;
if strcmp(Y7{k,1},'NG')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(1); end
if strcmp(Y7{k,1},'NM')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(2); end
if strcmp(Y7{k,1},'NP')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(3); end
if strcmp(Y7{k,1},'CE')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(4); end
if strcmp(Y7{k,1},'PP')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(5); end
if strcmp(Y7{k,1},'PM')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(6); end
if strcmp(Y7{k,1},'PG')
REGLAS_T1(i,j) = CENTROS_DE_BARRA_T1(7); end
end
end
end
end
end

CANDIDATOS_T1 = CENTROS_DE_BARRA_T1;

end

function [ ] = Fam_T2(Y10,Y11,Y12)
global CANDIDATOS_T2 CENTROS_DE_BARRA_T2 REGLAS_T2 NUM REGLAS_T2 ...
NUM BARRAS_T2 UNIF_C2 C2
%Define matriz de asociación difusa (matriz de reglas) en base a
datos del
%archivo de entrada.
S1 = 1;
No = 2;
m = 1;

if NUM REGLAS_T2 == 9
if NUM BARRAS_T2 == 5
if UNIF_C2 == S1
Cy = C2 / 2; %para i = 2, Cy es el mismo para toda i
for i = -2 : 1 : 2
CENTROS_DE_BARRA_T2(m,1) = double(Cy * i);
m = m + 1;
end
else
Cy = C2 / 4; %para i = 2, Cy es el mismo para toda i
for i = -2 : 1 : 2
CENTROS_DE_BARRA_T2(m,1) = Cy * sign(i) * i^2;
m = m + 1;
end
end
k = 0;
for i = 1 : sqrt(NUM REGLAS_T2)
for j = 1 : sqrt(NUM REGLAS_T2)
k = k + 1;
if strcmp(Y10{k,1},'NG')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(1); end
if strcmp(Y10{k,1},'NM')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(1); end
if strcmp(Y10{k,1},'NP')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(2); end
if strcmp(Y10{k,1},'CE')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(3); end
if strcmp(Y10{k,1},'PP')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(4); end
if strcmp(Y10{k,1},'PM')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(5); end
if strcmp(Y10{k,1},'PG')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(5); end
end
end
end
if NUM BARRAS_T2 == 7
if UNIF_C2 == S1
Cy = C2 / 3; %para i = 2, Cy es el mismo para toda i
for i = -3 : 1 : 3
CENTROS_DE_BARRA_T2(m,1) = double(Cy * i);
m = m + 1;
end
else
Cy = C2 / 9; %para i = 2, Cy es el mismo para toda i
for i = -3 : 1 : 3
CENTROS_DE_BARRA_T2(m,1) = Cy * sign(i) * i^2;
m = m + 1;
end
end
k = 0;
for i = 1 : sqrt(NUM REGLAS_T2)
for j = 1 : sqrt(NUM REGLAS_T2)
k = k + 1;
if strcmp(Y10{k,1},'NG')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(1); end
if strcmp(Y10{k,1},'NM')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(2); end
if strcmp(Y10{k,1},'NP')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(3); end
if strcmp(Y10{k,1},'CE')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(4); end
if strcmp(Y10{k,1},'PP')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(5); end
if strcmp(Y10{k,1},'PM')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(6); end
if strcmp(Y10{k,1},'PG')
REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(7); end
end
end
end
elseif NUM REGLAS_T2 == 25
if NUM BARRAS_T2 == 5
if UNIF_C2 == S1
Cy = C2 / 2; %para i = 2, Cy es el mismo para toda i
for i = -2 : 1 : 2
CENTROS_DE_BARRA_T2(m,1) = Cy * i;
m = m + 1;
end
end
else
Cy = C2 / 4; %para i = 2, Cy es el mismo para toda i
for i = -2 : 1 : 2
CENTROS_DE_BARRA_T2(m,1) = Cy * sign(i) * i^2;
m = m + 1;
end
end

```

```

        end
    end
    k = 0;
    for i = 1 : sqrt(NUM_REGLAS_T2)
        for j = 1 : sqrt(NUM_REGLAS_T2)
            k = k + 1;
            if strcmp(Y11{k,1},'NG')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(1); end
            if strcmp(Y11{k,1},'NM')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(1); end
            if strcmp(Y11{k,1},'NP')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(2); end
            if strcmp(Y11{k,1},'CE')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(3); end
            if strcmp(Y11{k,1},'PP')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(4); end
            if strcmp(Y11{k,1},'PM')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(5); end
            if strcmp(Y11{k,1},'PG')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(5); end
        end
    end
    end
    if NUM_BARRAS_T2 == 7
        if UNIF_C2 == S1
            Cy = C2 / 3; %para i = 2, Cy es el mismo para toda i
            for i = -3 : 1 : 3
                CENTROS_DE_BARRA_T2(m,1) = Cy * i;
                m = m + 1;
            end
        else
            Cy = C2 / 9; %para i = 2, Cy es el mismo para toda i
            for i = -3 : 1 : 3
                CENTROS_DE_BARRA_T2(m,1) = Cy * sign(i) * i^2;
                m = m + 1;
            end
        end
    end
    k = 0;
    for i = 1 : sqrt(NUM_REGLAS_T2)
        for j = 1 : sqrt(NUM_REGLAS_T2)
            k = k + 1;
            if strcmp(Y11{k,1},'NG')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(1); end
            if strcmp(Y11{k,1},'NM')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(2); end
            if strcmp(Y11{k,1},'NP')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(3); end
            if strcmp(Y11{k,1},'CE')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(4); end
            if strcmp(Y11{k,1},'PP')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(5); end
            if strcmp(Y11{k,1},'PM')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(6); end
            if strcmp(Y11{k,1},'PG')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(7); end
        end
    end
    end
    elseif NUM_REGLAS_T2 == 49
        if NUM_BARRAS_T2 == 5
            if UNIF_C2 == S1
                Cy = C2 / 2; %para i = 2, Cy es el mismo para toda i
                for i = -2 : 1 : 2
                    CENTROS_DE_BARRA_T2(m,1) = Cy * i;
                    m = m + 1;
                end
            else
                Cy = C2 / 4; %para i = 2, Cy es el mismo para toda i
                for i = -2 : 1 : 2
                    CENTROS_DE_BARRA_T2(m,1) = Cy * sign(i) * i^2;
                    m = m + 1;
                end
            end
        end
    end
    k = 0;
    for i = 1 : sqrt(NUM_REGLAS_T2)
        for j = 1 : sqrt(NUM_REGLAS_T2)
            k = k + 1;
            if strcmp(Y12{k,1},'NG')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(1); end
            if strcmp(Y12{k,1},'NM')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(1); end
            if strcmp(Y12{k,1},'NP')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(2); end
            if strcmp(Y12{k,1},'CE')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(3); end
            if strcmp(Y12{k,1},'PP')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(4); end
            if strcmp(Y12{k,1},'PM')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(5); end
            if strcmp(Y12{k,1},'PG')
                REGLAS_T2(i,j) = CENTROS_DE_BARRA_T2(5); end
        end
    end
    end
    if NUM_BARRAS_T2 == 7
        if UNIF_C2 == S1
            Cy = C2 / 3; %para i = 2, Cy es el mismo para toda i
            for i = -3 : 1 : 3
                CENTROS_DE_BARRA_T2(m,1) = Cy * i;
                m = m + 1;
            end
        else
            Cy = C2 / 9; %para i = 2, Cy es el mismo para toda i
            for i = -3 : 1 : 3
                CENTROS_DE_BARRA_T2(m,1) = Cy * sign(i) * i^2;
                m = m + 1;
            end
        end
    end
    CANDIDATOS_T2 = CENTROS_DE_BARRA_T2;

    end

function [ dx_dt ] = Runge_Kutta_4to( dx )
%Runge Kutta 4to Orden para sistemas de ecuaciones diferenciales de
ler
%orden Xk+1 = Xk + h / 6 * (K1 + 2*K2 + 2*K3 + K4)
% Detailed explanation goes here
global h T_Final_Simulacion

%Inicializa vector de salida y vectores auxiliares
DIM = size(dx,1);
dx_dt = zeros(DIM,1);
Aux = zeros(DIM,1);
K1 = zeros(DIM,1);
K2 = zeros(DIM,1);
K3 = zeros(DIM,1);
K4 = zeros(DIM,1);

%Calcula numero de iteraciones a realizar
n = T_Final_Simulacion / h;

%Calcula Constantes K1
K1 = Ec_diff(dx);

%Calcula Constantes K2
Aux = dx + K1 * h / 2;
K2 = Ec_diff(Aux);

%Calcula Constantes K3
Aux = dx + K2 * h / 2;
K3 = Ec_diff(Aux);

%Calcula Constantes K4
Aux = dx + K3 * h;
K4 = Ec_diff(Aux);

%Calcula variables de estado en el tiempo t + h
for i = 1 : size(dx,1)
    dx_dt(i) = dx(i) + h / 6 * (K1(i) + 2*K2(i) + 2*K3(i) + K4(i));
end

function [ dx_dt ] = Ec_diff( X )
%FUNCION ECUACIONES DIFERENCIALES
% Contiene las ecuaciones de estado para la maquina sincrona, red,
% sistema de excitacion, turbina de vapor y gobernador.
% El argumento de salida es el vector KX con los valores
correspondientes
% a la constante KX donde X = 1,...,4.
% El argumento de entrada es el vector con las variables de estado
segun:
% XK(1) = delta_r
% XK(2) = s
% XK(3) = eqp
% XK(4) = eq2p
% XK(5) = ed2p
% XK(6) = Vf
% XK(7) = Pgv
% XK(8) = Pfp
% XK(9) = Tm
% XK(10) = U

```

```

% XK(11)= Vexc

%Variables globales
global w0 kd M Te Xd Xdp Td0p id Xd2p Td02p Xq Xq2p Tq02p iq UG Tsm
Tch
global Trh Fhp Vref Vter Ka Ta Kf Tf Tx U

%Inicializa Vector de resultados
dx_dt = zeros(11,1);

%Ecuaciones de estado
dx_dt(1) = w0 * X(2);
dx_dt(2) = ((-kd * X(2)) + X(9) - Te) / M;
dx_dt(3) = (X(6) - (Xd - Xdp) * id - X(3)) / Td0p;
dx_dt(4) = (X(3) - (Xdp - Xd2p) * id - X(4)) / Td02p;
dx_dt(5) = ((Xq - Xq2p) * iq - X(5)) / Tq02p;
dx_dt(6) = (U - X(6)) / Tx;
dx_dt(7) = (UG - X(7)) / Tsm;
dx_dt(8) = (X(7) - X(8)) / Tch;
dx_dt(9) = (Fhp * dx_dt(8) + X(8) - X(9)) / Trh;
dx_dt(10) = (Ka * (Vref - Vter - X(11)) - X(10)) / Ta;
dx_dt(11) = (Kf * dx_dt(10) - X(11)) / Tf;

end

function [ ] = Ecs_Dinamicas_ST1( dx_dt )
%Ecuaciones Dinamicas del modelo de la máquina síncrona.

global re ra Vbus xe Xd2p Xq2p %Variables Globales de entrada
global P Q Te Vter id iq %Variabes Globales que se modifican

X = zeros(2,1);

%Resuelve sistema para iq, id
% X = inv(A) * B ; X(1) = id, X(2) = iq.
%a = re + ra;
%b = -(xe + Xq2p);
%c = xe + Xd2p;
%d = re + ra;
%A = [ a , b ; c , d];
%B = [XK(5) - Vbus * sin(XK(1)) ; XK(4) - Vbus * cos(XK(1))];

%X = inv(A) * B;
%X = A\B;
%id = X(1);
%i = X(2);

a = re + ra;
b = dx_dt(5) - Vbus * sin(dx_dt(1));
c = dx_dt(4) - Vbus * cos(dx_dt(1));
d = xe + Xq2p;
e = xe + Xd2p;
id = (a*b + c*d) / (a^2 + e*d);
iq = (a*c - e*b) / (a^2 + e*d);

Vd = dx_dt(5) - ra * id + Xq2p * iq;
Vq = dx_dt(4) - ra * iq - Xd2p * id;

Vter = sqrt(Vd^2 + Vq^2);
V_complex = complex(Vq,Vd);
I = complex(iq,-id);
S = V_complex * I;
P = real(S);
Q = -imag(S);
Te = dx_dt(5) * id + dx_dt(4) * iq;

End

function [ Vbus ] = Falla( Tiempo_Actual )
%Simulacion de falla en el sistema
% La funcion solicita los argumentos del tiempo actual como
argumento
% y tiempo de duracion de la falla como variable global.
% La variable devuelta es el voltaje del bus.
global T_Inicial_Falla T_Final_Falla

if (T_Inicial_Falla <= Tiempo_Actual) && (Tiempo_Actual <=
T_Final_Falla)
Vbus = 0.1;
else
Vbus = 1;
end

end

function [ ] = Cambio_De_Carga( Tiempo )
%Calcula los cambios de carga
% Los cambios de carga son calculados mediante la ecuacion de la
recta
% dependiendo del tiempo actual de simulacion y los valores de
carga y
% tiempo definidos en los argumentos de entrada.
global UG %Variable global modificada
global T_Inicio_Carga T_Final_Carga Cambios_Carga
persistent Carga_Anterior Carga_Actual Carga_Posterior

if (Tiempo >= T_Inicio_Carga(1)) && (Tiempo < T_Final_Carga(1))
Carga_Anterior = UG;
Carga_Posterior = Carga_Anterior + Cambios_Carga(1);

```

```

elseif (Tiempo >= T_Final_Carga(1)) && (Tiempo < T_Inicio_Carga(2))
M = (Carga_Posterior - Carga_Anterior) / (T_Inicio_Carga(2) -
T_Final_Carga(1));
UG = (Tiempo - T_Final_Carga(1)) * M + Carga_Anterior;
Carga_Actual = Carga_Posterior;

elseif (Tiempo >= T_Inicio_Carga(2)) && (Tiempo < T_Final_Carga(2))
UG = Carga_Actual;
Carga_Anterior = Carga_Actual;
Carga_Posterior = Carga_Anterior + Cambios_Carga(2);

elseif (Tiempo >= T_Final_Carga(2)) && (Tiempo < T_Inicio_Carga(3))
M = (Carga_Posterior - Carga_Anterior) / (T_Inicio_Carga(3) -
T_Final_Carga(2));
UG = (Tiempo - T_Final_Carga(2)) * M + Carga_Anterior;
Carga_Actual = Carga_Posterior;

elseif (Tiempo >= T_Inicio_Carga(3)) && (Tiempo < T_Final_Carga(3))
UG = Carga_Actual;
Carga_Anterior = Carga_Actual;
Carga_Posterior = Carga_Anterior + Cambios_Carga(3);

elseif (Tiempo >= T_Final_Carga(3)) && (Tiempo < T_Inicio_Carga(4))
M = (Carga_Posterior - Carga_Anterior) / (T_Inicio_Carga(4) -
T_Final_Carga(3));
UG = (Tiempo - T_Final_Carga(3)) * M + Carga_Anterior;
Carga_Actual = Carga_Posterior;

elseif (Tiempo >= T_Inicio_Carga(4)) && (Tiempo < T_Final_Carga(4))
UG = Carga_Actual;
Carga_Anterior = Carga_Actual;
Carga_Posterior = Carga_Anterior + Cambios_Carga(4);

elseif (Tiempo >= T_Final_Carga(4)) && (Tiempo < T_Inicio_Carga(5))
M = (Carga_Posterior - Carga_Anterior) / (T_Inicio_Carga(5) -
T_Final_Carga(4));
UG = (Tiempo - T_Final_Carga(4)) * M + Carga_Anterior;
Carga_Actual = Carga_Posterior;

elseif (Tiempo >= T_Inicio_Carga(5)) && (Tiempo < T_Final_Carga(5)),
UG = Carga_Actual;

end

end

```

F 1.5 Funciones para los controladores difusos

```

function [ U_act ] =
Control_Intervalo_Tipo2_KM(A,B,DELTA_E,DELTA_DE,Spread_left,Spread_r
ight,Error,Cambio_del_Error,REGLAS,CENTROS_SALIDA,TIPO_FUNCION,NUM_C
ONJUNTOS,U_ant)
%Función que realiza un control PI difuso Mamdani con conjuntos
difusos
%Intervalos de 2do tipo. Las variables de entrada son los argumentos
Error
%y cambio del Error
% Argumentos: Error = Tensión referencia - Tensión actual
global Tnorn_min iter_CL_RM iter_CR_KM
persistent K
if isempty(K)
K = 1;
end;

BANDERA = 10000;
%Matrices para controlador Tipo 2
MF_1 = zeros(NUM_CONJUNTOS,3);
MF_2 = zeros(NUM_CONJUNTOS,3);
YL = zeros(10,1); YR = zeros(10,1);
NUM_FILAS = 1;
CL = 0; CR = 0;
if size(CENTROS_SALIDA,1) == 7
SPREAD_LEFT = [0.0; 0.0; 0.0; 0.0; 0.0; 0.0; 0.0];
SPREAD_RIGHT = [0.0; 0.0; 0.0; 0.0; 0.0; 0.0; 0.0];
elseif size(CENTROS_SALIDA,1) == 5
SPREAD_LEFT = [0.0; 0.0; 0.0; 0.0; 0.0];
SPREAD_RIGHT = [0.0; 0.0; 0.0; 0.0; 0.0];
end
if TIPO_FUNCION == 1 || TIPO_FUNCION == 2 || TIPO_FUNCION == 3 ||
TIPO_FUNCION == 4
if NUM_CONJUNTOS == 3
PARAMETROS_ERROR = [BANDERA, -A, 0; ...
-A, 0, A; ...
0, A, BANDERA];
PARAMETROS_DERROR = [BANDERA, -B, 0; ...
-B, 0, B; ...
0, B, BANDERA];
end
if NUM_CONJUNTOS == 5
PARAMETROS_ERROR = [BANDERA, -A, -A/2; ...
-A, -A/2, 0; ...

```

```

-A/2, 0, A/2; ...
0, A/2, A; ...
A/2, A BANDERA];
PARAMETROS_DERROR = [BANDERA, -B, -B/2; ...
-B, -B/2, 0; ...
-B/2, 0, B/2; ...
0, B/2, B; ...
B/2, B BANDERA];
end
if NUM_CONJUNTOS == 7
PARAMETROS_ERROR = [BANDERA, -A, -2*A/3; ...
-A, -2*A/3, -A/3; ...
-2*A/3, -A/3, 0; ...
-A/3, 0, A/3; ...
0, A/3, 2*A/3; ...
A/3, 2*A/3, A; ...
2*A/3, A, BANDERA];
PARAMETROS_DERROR = [BANDERA, -B, -2*B/3; ...
-B, -2*B/3, -B/3; ...
-2*B/3, -B/3, 0; ...
-B/3, 0, B/3; ...
0, B/3, 2*B/3; ...
B/3, 2*B/3, B; ...
2*B/3, B, BANDERA];
end
end
%SELECCIONE INCERTIDUMBRE DE CONJUNTOS DIFUSOS DE SALIDA
SPREAD = 0.0;

if TIPO_FUNCION == 5
if NUM_CONJUNTOS == 3
PARAMETROS_ERROR = [BANDERA, -A, A/3; ...
A/3, 0, A/3; ...
A/3, A, BANDERA];
PARAMETROS_DERROR = [BANDERA, -B, B/3; ...
B/3, 0, B/3; ...
B/3, B, BANDERA];
end
if NUM_CONJUNTOS == 5
PARAMETROS_ERROR = [BANDERA, -A, A/6; ...
A/6, -A/2, A/6; ...
A/6, 0, A/6; ...
A/6, A/2, A/6; ...
A/6, A, BANDERA];
PARAMETROS_DERROR = [BANDERA, -B, B/6; ...
B/6, -B/2, B/6; ...
B/6, 0, B/6; ...
B/6, B/2, B/6; ...
B/6, B, BANDERA];
end
if NUM_CONJUNTOS == 7
PARAMETROS_ERROR = [BANDERA, -A, A/9; ...
A/9, -2*A/3, A/9; ...
A/9, -A/3, A/9; ...
A/9, 0, A/9; ...
A/9, A/3, A/9; ...
A/9, 2*A/3, A/9; ...
A/9, A, BANDERA];
PARAMETROS_DERROR = [BANDERA, -B, B/9; ...
B/9, -2*B/3, B/9; ...
B/9, -B/3, B/9; ...
B/9, 0, B/9; ...
B/9, B/3, B/9; ...
B/9, 2*B/3, B/9; ...
B/9, B, BANDERA];
end
end

if TIPO_FUNCION == 1
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
MF_1(i,:) = Triangular01_Tipo2(PARAMETROS_ERROR(i,:),...
DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
MF_2(i,:) = Triangular01_Tipo2(PARAMETROS_DERROR(i,:),...
DELTA_DE,Cambio_del_Error,i);
end
elseif TIPO_FUNCION == 2
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
MF_1(i,:) = Triangular02_Tipo2(PARAMETROS_ERROR(i,:),...
DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
MF_2(i,:) = Triangular02_Tipo2(PARAMETROS_DERROR(i,:),...
DELTA_DE,Cambio_del_Error,i);
end
elseif TIPO_FUNCION == 3
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
MF_1(i,:) = Triangular03_Tipo2(PARAMETROS_ERROR(i,:),...
DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
MF_2(i,:) = Triangular03_Tipo2(PARAMETROS_DERROR(i,:),...
DELTA_DE,Cambio_del_Error,i);
end
elseif TIPO_FUNCION == 4
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
MF_1(i,:) = Triangular_Diamante_Tipo2(PARAMETROS_ERROR(i,:),...
DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
MF_2(i,:) = Triangular_Diamante_Tipo2(PARAMETROS_DERROR(i,:),...
DELTA_DE,Cambio_del_Error,i);
end
elseif TIPO_FUNCION == 5
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
MF_1(i,:) = Gaussian_Incert_DesvEst_Tipo2(PARAMETROS_ERROR(i,:),...
DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
MF_2(i,:) = Gaussian_Incert_DesvEst_Tipo2(PARAMETROS_DERROR(i,:),...
DELTA_DE,Cambio_del_Error,i);
end
end
%-----
%-----
Tnorm_min = minimo_tipo2(MF_1,MF_2,REGLAS); %Norma-T mínimo
Tnorm = maximo_tipo2(Tnorm_min,CENTROS_SALIDA); %Maximo

if size(Tnorm,NUM_FILAS) ~= size(SPREAD_LEFT,NUM_FILAS) || ...
size(SPREAD_RIGHT,NUM_FILAS)
disp('El num de filas del vector SPREAD debe ser de la
misma dimension que Tnorm');
Dim_Tnorm = size(Tnorm,NUM_FILAS)
SPREAD_LEFT
Dim_SPREAD_LEFT = size(SPREAD_LEFT,NUM_FILAS)
SPREAD_RIGHT
Dim_SPREAD_RIGHT = size(SPREAD_RIGHT,NUM_FILAS)
pause;
end
%-----
%-----
hk = calcula_h(Tnorm);
h_corregida = hk;
dh = calcula_dh(Tnorm);
%Iteraciones de Algoritmo
KM*****
%Calcula CL
for i=1:9
for j = 1 : size(Tnorm,NUM_FILAS)
YL(i+1) = YL(i+1) + (Tnorm(j,3)-
SPREAD_LEFT(j))*h_corregida(j);
end
YL(i+1) = YL(i+1)/sum(h_corregida(:,1));
%Verifica Condición de Parada
if (YL(i) == YL(i+1))
CL = YL(i+1);
Iter_CL=i-1;
break
end
%Hace Correcciones
J = localiza(Tnorm(:,3)-SPREAD_LEFT,YL(i+1));
h_corregida = corrige_hl(hk,Tnorm,J);
end
iter_CL_KM(K) = i;
*****
%Calcula CR
hk = calcula_h(Tnorm);
h_corregida = hk;
for i = 1 : 9
for j = 1 : size(Tnorm,NUM_FILAS)
YR(i+1) = YR(i+1) +
(Tnorm(j,3)+SPREAD_RIGHT(j))*h_corregida(j);
end
YR(i+1) = YR(i+1)/sum(h_corregida(:,1));
%Verifica Condición de Parada
if (YR(i) == YR(i+1))
CR = YR(i+1);
Iter_CR=i-1;
end
end
end

```

```

        break
    end
%Hace Correcciones
    J = localiza(Tnorm(:,3)+SPREAD_RIGHT,YR(i+1));
    h_corregida = corrige_hr(hk,Tnorm,J);
end
iter_CR_KM(K) = i;
%*****
%Defuzificación
dU = (CL + CR)/2;
%*****
%SALIDA TIPO PI
U_act = dU + U_ant;
K = K + 1;
end

function [ U_act ] =
Control_Intervalo_Tipo2_EKM(A,B,DELTA_E,DELTA_DE,spread1,spread2,Error,
Cambio_del_Error,REGLAS,CENTROS_SALIDA,TIPO_FUNCION,NUM_CONJUNTOS,
,U_ant)
%Función que realiza un control PI difuso Mamdani con conjuntos
difusos
%intervalos de 2do tipo. Las variables de entrada son los argumentos
Error
%y cambio del Error
% Argumentos: Error = Tensión referencia - Tensión actual
global du_ctrl Tnorm_min iter_CL_EKM iter_CR_EKM
persistent K
if isempty(K)
    K = 1; end;

BANDERA = 10000;
%Numero de conjuntos de salida
N = size(CENTROS_SALIDA,1);

%Matrices para controlador Tipo 2
MF_1 = zeros(NUM_CONJUNTOS,3);
MF_2 = zeros(NUM_CONJUNTOS,3);
YL = zeros(10,1); YR = zeros(10,1);
NUM_FILAS = 1;
CL = 0; CR = 0;

if size(CENTROS_SALIDA,1) == 7
    SPREAD_LEFT = [0.0; 0.0; 0.0; 0; 0.0; 0.0; 0.0];
    SPREAD_RIGHT = [0.0; 0.0; 0.0; 0.0; 0.0; 0.0; 0.0];
elseif size(CENTROS_SALIDA,1) == 5
    SPREAD_LEFT = [0.0; 0.0; 0; 0.0; 0.0];
    SPREAD_RIGHT = [0.0; 0.0; 0.0; 0.0; 0.0];
end
if TIPO_FUNCION == 1 || TIPO_FUNCION == 2 || TIPO_FUNCION == 3 ||
TIPO_FUNCION == 4
    NUM_CONJUNTOS == 3
        PARAMETROS_ERROR = [BANDERA, -A, 0; ...
            -A, 0, A; ...
            0, A, BANDERA];
        PARAMETROS_DERROR = [BANDERA, -B, 0; ...
            -B, 0, B; ...
            0, B, BANDERA];
    end
if NUM_CONJUNTOS == 5
    PARAMETROS_ERROR = [BANDERA, -A, -A/2; ...
        -A, -A/2, 0; ...
        -A/2, 0, A/2; ...
        0, A/2, A; ...
        A/2, A, BANDERA];
    PARAMETROS_DERROR = [BANDERA, -B, -B/2; ...
        -B, -B/2, 0; ...
        -B/2, 0, B/2; ...
        0, B/2, B; ...
        B/2, B, BANDERA];
    end
if NUM_CONJUNTOS == 7
    PARAMETROS_ERROR = [BANDERA, -A, -2*A/3; ...
        -A, -2*A/3, -A/3; ...
        -2*A/3, -A/3, 0; ...
        -A/3, 0, A/3; ...
        0, A/3, 2*A/3; ...
        A/3, 2*A/3, A; ...
        2*A/3, A, BANDERA];
    PARAMETROS_DERROR = [BANDERA, -B, -2*B/3; ...
        -B, -2*B/3, -B/3; ...
        -2*B/3, -B/3, 0; ...
        -B/3, 0, B/3; ...
        0, B/3, 2*B/3; ...
        B/3, 2*B/3, B; ...
        2*B/3, B, BANDERA];
    end
end
end
%SELECCIONE INCERTIDUMBRE DE CONJUNTOS DIFUSOS DE SALIDA
SPREAD = 0.0;

if TIPO_FUNCION == 5
    if NUM_CONJUNTOS == 3
        PARAMETROS_ERROR = [BANDERA, -A, A/3; ...
            A/3, 0, A/3; ...
            A/3, A, BANDERA];
        PARAMETROS_DERROR = [BANDERA, -B, B/3; ...
            B/3, 0, B/3; ...
            B/3, B, BANDERA];
    end
    if NUM_CONJUNTOS == 5
        PARAMETROS_ERROR = [BANDERA, -A, A/6; ...
            A/6, -A/2, A/6; ...
            A/6, 0, A/6; ...
            A/6, A/2, A/6; ...
            A/6, A, BANDERA];
        PARAMETROS_DERROR = [BANDERA, -B, B/6; ...
            B/6, -B/2, B/6; ...
            B/6, 0, B/6; ...
            B/6, B/2, B/6; ...
            B/6, B, BANDERA];
    end
    if NUM_CONJUNTOS == 7
        PARAMETROS_ERROR = [BANDERA, -A, A/9; ...
            A/9, -2*A/3, A/9; ...
            A/9, -A/3, A/9; ...
            A/9, 0, A/9; ...
            A/9, A/3, A/9; ...
            A/9, 2*A/3, A/9; ...
            A/9, A, BANDERA];
        PARAMETROS_DERROR = [BANDERA, -B, B/9; ...
            B/9, -2*B/3, B/9; ...
            B/9, -B/3, B/9; ...
            B/9, 0, B/9; ...
            B/9, B/3, B/9; ...
            B/9, 2*B/3, B/9; ...
            B/9, B, BANDERA];
    end
end
end
if TIPO_FUNCION == 1
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
    MF_1(i,:) = Triangular01_Tipo2(PARAMETROS_ERROR(i,:),...
        DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
    MF_2(i,:) = Triangular01_Tipo2(PARAMETROS_DERROR(i,:),...
        DELTA_DE,Cambio_del_Error,i);
end
elseif TIPO_FUNCION == 2
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
    MF_1(i,:) = Triangular02_Tipo2(PARAMETROS_ERROR(i,:),DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
    MF_2(i,:) = Triangular02_Tipo2(PARAMETROS_DERROR(i,:),...
        DELTA_DE,Cambio_del_Error,i);
end
elseif TIPO_FUNCION == 3
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
    MF_1(i,:) = Triangular03_Tipo2(PARAMETROS_ERROR(i,:),DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
    MF_2(i,:) = Triangular03_Tipo2(PARAMETROS_DERROR(i,:),...
        DELTA_DE,Cambio_del_Error,i);
end
elseif TIPO_FUNCION == 4
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
    MF_1(i,:) = Triangular_Diamante_Tipo2(PARAMETROS_ERROR(i,:),...
        DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS
    MF_2(i,:) = Triangular_Diamante_Tipo2(PARAMETROS_DERROR(i,:),...
        DELTA_DE,Cambio_del_Error,i);
end
elseif TIPO_FUNCION == 5
%Calcula Función de membresía primaria para variable de entrada
error
for i = 1 : NUM_CONJUNTOS
    MF_1(i,:) = Gaussiana_Incert_DesvEst_Tipo2(PARAMETROS_ERROR(i,:),...
        DELTA_E,Error,i);
end
%Calcula Función de membresía primaria para variable cambio del
error
for i = 1 : NUM_CONJUNTOS

```

```

MF_2(i,:) = b_aux + (Tnorm(i,1)-Tnorm(i,2));
Gaussiana_Incert_DesvEst_Tipo2(PARAMETROS_DERROR(i,:),...
    DELTA_DE,Cambio_del_Error,i);
end
end
%-----
Tnorm_min = minimo_type2(MF_1,MF_2,REGLAS); %Norma-T mínimo
Tnorm = maximo_type2(Tnorm_min,CENTROS_SALIDA); %Maximo

if size(Tnorm,NUM_FILAS) ~= size(SPREAD_LEFT,NUM_FILAS) || ...
    size(Tnorm,NUM_FILAS) ~= size(SPREAD_RIGHT,NUM_FILAS)
    disp('El num de filas del vector SPREAD debe ser de la
    misma dimension que Tnorm');
    Dim_Tnorm = size(Tnorm,NUM_FILAS);
    Dim_SPREAD_LEFT = size(SPREAD_LEFT,NUM_FILAS);
    Dim_SPREAD_RIGHT = size(SPREAD_RIGHT,NUM_FILAS);
    pause;
end
%-----
%Iteraciones de Algoritmo
EKM*****
%Calcula CL
r = round(N / 2.4);
a = 0; b = 0;
for i = 1 : r
    a = a + (Tnorm(i,3)-SPREAD_LEFT(i)) * Tnorm(i,1);
end
for i = r + 1 : N
    a = a + (Tnorm(i,3)-SPREAD_LEFT(i)) * Tnorm(i,2);
end
for i = 1 : r
    b = b + Tnorm(i,1);
end
for i = r + 1 : N
    b = b + Tnorm(i,2);
end
end
CL_aux = a / b;
R = localiza(Tnorm(:,3)-SPREAD_LEFT,single(CL_aux));

%inicializa contador de iteraciones
j = 1;
while (R ~= r)
    a_aux = 0; b_aux = 0;
    s = sign(R - r);
    for i = min(R,r) + 1 : max(R,r)
        a_aux = a_aux + (Tnorm(i,3)-SPREAD_LEFT(i)) * (Tnorm(i,1)-
Tnorm(i,2));
    end
    for i = min(R,r) + 1 : max(R,r)
        b_aux = b_aux + (Tnorm(i,1)-Tnorm(i,2));
    end
    a = a + s * a_aux;
    b = b + s * b_aux;
    CL_aux = a / b;
    r = R;
    R = localiza(Tnorm(:,3)-SPREAD_LEFT,single(CL_aux));

    %Incrementa contador de iteraciones
    j = j + 1;
end
CL = CL_aux;
iter_CL_EKM(K) = j;
*****
%Calcula CR
r = round(N / 1.7);
a = 0; b = 0;
for i = 1 : r
    a = a + (Tnorm(i,3)+SPREAD_RIGHT(i)) * Tnorm(i,2);
end
for i = r + 1 : N
    a = a + (Tnorm(i,3)+SPREAD_RIGHT(i)) * Tnorm(i,1);
end
for i = 1 : r
    b = b + Tnorm(i,2);
end
for i = r + 1 : N
    b = b + Tnorm(i,1);
end
end
CR_aux = a / b;
R = localiza(Tnorm(:,3)+SPREAD_RIGHT,single(CR_aux));
%Inicializa contador de iteraciones
j = 1;
while (R ~= r)
    a_aux = 0; b_aux = 0;
    s = sign(R - r);
    for i = min(R,r) + 1 : max(R,r)
        a_aux = a_aux + (Tnorm(i,3)+SPREAD_RIGHT(i)) * (Tnorm(i,1)-
Tnorm(i,2));
    end
    for i = min(R,r) + 1 : max(R,r)
        b_aux = b_aux + (Tnorm(i,1)-Tnorm(i,2));
    end
    a = a + s * a_aux;
    b = b + s * b_aux;
    CR_aux = a / b;
    R = localiza(Tnorm(:,3)+SPREAD_RIGHT,single(CR_aux));

    %Incrementa contador de iteraciones
    j = j + 1;
end
CR = CR_aux;
iter_CR_EKM(K) = j;
*****
%Dedifuzificación
dU = (CL + CR)/2;
*****
%SALIDA TIPO PI
U_act = dU + U_ant;
du_ctrl(K) = dU;
K = K + 1;
end

function [ Interval_MF ] = Triangular01_Tipo2(PARAM,DELTA,x,NUM_SET)
%FUNCIÓN DE MEMBRESIA TIPO 2
% Función de membresía triangular de tipo 2. Requiere de 4
parámetros
% para el conjunto difuso y dos argumentos mas correspondientes a
el valor
% de la entrada y el numero del conjunto difuso, con el objeto de
devolver
% una etiqueta numérica según el conjunto.
% Los parametros P1,P2,P3 definen un conjunto difuso tipo 1. para
este
% caso Delta = 0.

%Inicializa vector de salida
Interval_MF = zeros(1,3);
BANDERA = 10000;
%Parametros*****
A = PARAM(1) - DELTA; B = PARAM(1) + DELTA;
C = PARAM(2);
D = PARAM(3) - DELTA; E = PARAM(3) + DELTA;
Y = 1;
*****
UMF = 0;
LMF = 0;
if (PARAM(1) == BANDERA) %Triangular Izquierda
    if (x <= C)
        UMF = 1;
        LMF = 1;
    elseif C < x && x < E %UMF PENDIENTE DECRECIENTE
        UMF = (x-E) / (C-E);
    elseif (x >= E)
        UMF = 0;
    end
    if C < x && x < D %LMF PENDIENTE DECRECIENTE
        LMF = (x-D) / (C-D);
    elseif (x >= D)
        LMF = 0;
    end
elseif (PARAM(3) == BANDERA) %Triangula Derecha
    if (x >= C)
        UMF = 1;
        LMF = 1;
    elseif A < x && x < C %UMF PENDIENTE CRECIENTE
        UMF = (x-A) / (C-A);
    elseif (x <= A)
        UMF = 0;
    end
    if B < x && x < C %LMF PENDIENTE CRECIENTE
        LMF = (x-B) / (C-B);
    elseif (x <= B)
        LMF = 0;
    end
else %Triangular
    if (x <= A) || (x >= E)
        UMF = 0;
    elseif A < x && x <= C %UMF PENDIENTE CRECIENTE
        UMF = (x-A)/(C-A);
    elseif C < x && x < E %UMF PENDIENTE DECRECIENTE
        UMF = (x-E)/(C-E);
    end
    if (x <= B) || (x >= D)
        LMF = 0;
    elseif B < x && x <= C %LMF PENDIENTE CRECIENTE
        LMF = (x-B)/(C-B);
    elseif C < x && x < D %LMF PENDIENTE DECRECIENTE
        LMF = (x-D)/(C-D);
    end
end
end

```



```

%Guarda funciones de membresia superior e inferior
Interval_MF(1,1) = UMF;
Interval_MF(1,2) = LMF;

%Almacena Numero del conjunto
Interval_MF(1,3) = NUM_SET;
end

function [ Interval_MF ] = Triangular02_Tipo2(PARAM,DELTA,x,NUM_SET)
%UFUNCIÓN DE MEMBRESIA TIPO 2
% Función de membresía triangular de tipo 2. Requiere de 4
parámetros
% para el conjunto difuso y dos argumentos mas correspondientes a
el valor
% de la entrada y el numero del conjunto difuso, con el objeto de
devolver
% una etiqueta numérica según el conjunto.
% Los parametros P1,P2,P3 definen un conjunto difuso tipo 1. para
este
% caso Delta = 0.

%Inicializa vector de salida
Interval_MF = zeros(1,3);
BANDERA = 10000;
%Parametros*****
*****
A = PARAM(1); B = PARAM(2);
C = PARAM(3); D = 1 - DELTA;
%*****
*****
UMF = 0;
LMF = 0;
if (PARAM(1) == BANDERA) %Triangular Izquierda
    if (x <= B)
        UMF = 1;
        LMF = D;
        elseif B < x && x < C %UMF PENDIENTE DECRECIENTE
            UMF = (C-x) / (C-B);
            LMF = D*(C-x) / (C-B); %LMF PENDIENTE DECRECIENTE
        elseif (x >= C)
            UMF = 0;
            LMF = 0;
        end
    elseif (PARAM(3) == BANDERA) %Triangula Derecha
        if (x >= B)
            UMF = 1;
            LMF = D;
            elseif A < x && x < B %UMF PENDIENTE CRECIENTE
                UMF = (x-A) / (B-A);
                LMF = D*(x-A) / (B-A);
            elseif (x <= A)
                UMF = 0;
                LMF = 0;
            end
        else %Triangular
            if (x <= A) || (x >= C)
                UMF = 0;
                LMF = 0;
            elseif A < x && x <= B %UMF PENDIENTE CRECIENTE
                UMF = (x-A)/(B-A); %UMF PENDIENTE CRECIENTE
                LMF = D*(x-A)/(B-A); %LMF PENDIENTE CRECIENTE
            elseif B < x && x < C %UMF PENDIENTE DECRECIENTE
                UMF = (C-x)/(C-B); %UMF PENDIENTE DECRECIENTE
                LMF = D*(C-x)/(C-B); %LMF PENDIENTE DECRECIENTE
            end
        end
    end

%Guarda funciones de membresia superior e inferior
Interval_MF(1,1) = UMF;
Interval_MF(1,2) = LMF;

%Almacena Numero del conjunto
Interval_MF(1,3) = NUM_SET;
end

function [ Interval_MF ] = Triangular03_Tipo2(PARAM,DELTA,x,NUM_SET)
%UFUNCIÓN DE MEMBRESIA TIPO 2
% Función de membresía triangular de tipo 2. Requiere de 4
parámetros
% para el conunto difuso y dos argumentos mas correspondientes a
el valor
% de la entrada y el numero del conjunto difuso, con el objeto de
devolver
% una etiqueta numérica según el conjunto.
% Los parametros P1,P2,P3 definen un conjunto difuso tipo 1. para
este
% caso Delta = 0.

%Inicializa vector de salida
Interval_MF = zeros(1,3);
%Parametros*****
*****
A = PARAM(1) - DELTA; B = PARAM(1) + DELTA;
C = PARAM(2) - DELTA; D = PARAM(2) - DELTA;
E = PARAM(2) + DELTA; F = PARAM(2) + DELTA;
G = PARAM(3) - DELTA; H = PARAM(3) + DELTA;
Y = 1;% - 2*DELTA;
%*****
*****
UMF = 0;
LMF = 0;
if (PARAM(1) == BANDERA) %Triangular Diamante Izquierda
    if (x <= C)
        UMF = 1;
        LMF = 1;
        %Intervalo P2 < x < (P2 + P3)/2
        elseif C < x && x < E
            UMF = ((x - C)/(E - C)) * (Y1 - 1) + 1;
        end
    end
end

UMF = 0;
LMF = 0;
if (PARAM(1) == 10000) %Triangular Izquierda
    if (x < F)
        UMF = 1;
        elseif F <= x && x <= H %UMF PENDIENTE DECRECIENTE
            UMF = (H-x)/(H-F);
            elseif (x > H)
                UMF = 0;
            end
    if (x < E)
        LMF = Y;
        elseif E <= x && x <= G %LMF PENDIENTE DECRECIENTE
            LMF = -Y*(x-E)/(G-E)+Y;
            elseif (x > G)
                LMF = 0;
            end
    elseif (PARAM(3) == 10000) %Triangula Derecha
        if (x < A)
            UMF = 0;
            elseif A <= x && x <= C %UMF PENDIENTE CRECIENTE
                UMF = (x-A)/(C-A);
                elseif (x > C)
                    UMF = 1;
                end
            if (x < B)
                LMF = 0;
                elseif B <= x && x <= D %LMF PENDIENTE CRECIENTE
                    LMF = Y*(x-B)/(D-B);
                    elseif (x > D)
                        LMF = Y;
                    end
            else %Triangular
                if (x <= A) || (x >= H)
                    UMF = 0;
                    elseif A < x && x < C %UMF PENDIENTE CRECIENTE
                        UMF = (x-A)/(C-A);
                        elseif C <= x && x < F
                            UMF = 1;
                        elseif F <= x && x < H %UMF PENDIENTE DECRECIENTE
                            UMF = (H-x)/(H-F);
                        end
                    if (x <= B) || (x >= G)
                        LMF = 0;
                        elseif B < x && x < D %LMF PENDIENTE CRECIENTE
                            LMF = Y*(x-B)/(D-B);
                            elseif D <= x && x < E
                                LMF = Y;
                                elseif E <= x && x < G %LMF PENDIENTE DECRECIENTE
                                    LMF = -Y * (x-E)/(G-E) + Y;
                                end
                    end
                end
            end

%Guarda funciones de membresia superior e inferior
Interval_MF(1,1) = UMF;
Interval_MF(1,2) = LMF;

%Almacena Numero del conjunto
Interval_MF(1,3) = NUM_SET;
end

function [ Interval_MF ] = Triangular04_Tipo2(PARAM,DELTA,x,NUM_SET)
%UFUNCIÓN DE MEMBRESIA TIPO 2
% Función de membresía triangular de tipo 2. Requiere de 4
parámetros
% para el conunto difuso y dos argumentos mas correspondientes a
el valor
% de la entrada y el numero del conjunto difuso, con el objeto de
devolver
% una etiqueta numérica según el conjunto.
% Los parametros P1,P2,P3 definen un conjunto difuso tipo 1. para
este
% caso Delta = 0.

BANDERA = 10000;
%Inicializa vector de salida
Interval_MF = zeros(1,3);
%Parametros*****
*****
A = PARAM(1); B = (PARAM(1) + PARAM(2))/2;
C = PARAM(2); D = (PARAM(1) + PARAM(2))/2;
E = (PARAM(2) + PARAM(3))/2; F = PARAM(3);
G = (PARAM(2) + PARAM(3))/2;
Y1 = 0.5 + DELTA;
Y2 = 0.5 - DELTA;
%*****
*****
UMF = 0;
LMF = 0;
if (PARAM(1) == BANDERA) %Triangular Diamante Izquierda
    if (x <= C)
        UMF = 1;
        LMF = 1;
        %Intervalo P2 < x < (P2 + P3)/2
        elseif C < x && x < E
            UMF = ((x - C)/(E - C)) * (Y1 - 1) + 1;
        end
    end
end

```

```

    LMF = ((x - C)/(G - C)) * (Y2 - 1) + 1;
    %Intervalo (P2 + P3)/2 <= x < P3
    elseif E <= x && x < F
    UMF = ((x - E)/(F - E)) * (-Y1) + Y1;
    LMF = ((x - G)/(F - G)) * (-Y2) + Y2;
    elseif (x >= F)
    UMF = 0;
    LMF = 0;
    end
elseif (PARAM(3) == BANDERA) %Triangula Diamante Derecha
    if (x <= A)
    UMF = 0;
    LMF = 0;
    %Intervalo P1 < x < (P1 + P2)/2
    elseif A < x && x < B
    UMF = ((x - A)/(B - A)) * Y1;
    LMF = ((x - A)/(D - A)) * Y2;
    %Intervalo (P1 + P2)/2 <= x < P2
    elseif B <= x && x < C
    UMF = ((x - B)/(C - B)) * (1 - Y1) + Y1;
    LMF = ((x - D)/(C - D)) * (1 - Y2) + Y2;
    elseif (x >= C)
    UMF = 1;
    LMF = 1;
    end
else %Triangular
    %Fuera del conjunto difuso
    if (x <= A) || (x >= F)
    UMF = 0;
    LMF = 0;
    %Intervalo P1 < x < (P1 + P2)/2
    elseif A < x && x < B
    UMF = ((x - A)/(B - A)) * Y1;
    LMF = ((x - A)/(D - A)) * Y2;
    %Intervalo (P1 + P2)/2 <= x < P2
    elseif B <= x && x < C
    UMF = ((x - B)/(C - B)) * (1 - Y1) + Y1;
    LMF = ((x - D)/(C - D)) * (1 - Y2) + Y2;
    %Intervalo P2 <= x < (P2 + P3)/2
    elseif C <= x && x < E
    UMF = ((x - C)/(E - C)) * (Y1 - 1) + 1;
    LMF = ((x - C)/(G - C)) * (Y2 - 1) + 1;
    %Intervalo (P2 + P3)/2 <= x < P3
    elseif E <= x && x < F %LMF PENDIENTE CRECIENTE
    UMF = ((x - E)/(F - E)) * (-Y1) + Y1;
    LMF = ((x - G)/(F - G)) * (-Y2) + Y2;
    end
end

%Guarda funciones de membresia superior e inferior
Interval_MF(1,1) = UMF;
Interval_MF(1,2) = LMF;

%Almacena Numero del conjunto
Interval_MF(1,3) = NUM_SET;
end

function [ Tnorm_min Interval_MF ] =
Gaussiana_Incert_DesvEst_Tipo2(PARAM,DELTA,x,NUM_SET)
%FUNCION DE MEMBRESIA TIPO 2
% Función de membresia triangular de tipo 2. Requiere de 4
parámetros
% para el conunto difuso y dos argumentos mas correspondientes a
el valor
% de la entrada y el numero del conjunto difuso, con el objeto de
devolver
% una etiqueta numérica según el conjunto.
% Los parametros P1,P2,P3 definen un conjunto difuso tipo 1. para
este
% caso Delta = 0.

%Inicializa vector de salida
Interval_MF = zeros(1,3);
BANDERA = 10000;
m = PARAM(2);
*****
UMF = 0;
LMF = 0;
if (PARAM(1) == BANDERA) %Gaussiana Izquierda
    sigma_sup = PARAM(3) + DELTA;
    sigma_inf = PARAM(3) - DELTA;
    if (x <= PARAM(2))
    UMF = 1;
    LMF = 1;
    else
    UMF = exp(-1/2*((x-m)/sigma_sup)^2);
    LMF = exp(-1/2*((x-m)/sigma_inf)^2);
    end
elseif (PARAM(3) == BANDERA) %Gausiana Derecha
    sigma_sup = PARAM(1) + DELTA;
    sigma_inf = PARAM(1) - DELTA;
    if (x >= PARAM(2))
    UMF = 1;
    LMF = 1;
    else
    UMF = exp(-1/2*((x-m)/sigma_sup)^2);
    LMF = exp(-1/2*((x-m)/sigma_inf)^2);
    end
end

```

```

else %Gaussiana
    sigma_sup = PARAM(3) + DELTA;
    sigma_inf = PARAM(3) - DELTA;
    UMF = exp(-1/2*((x-m)/sigma_sup)^2);
    LMF = exp(-1/2*((x-m)/sigma_inf)^2);
end

%Guarda funciones de membresia superior e inferior
Interval_MF(1,1) = UMF;
Interval_MF(1,2) = LMF;

%Almacena Numero del conjunto
Interval_MF(1,3) = NUM_SET;
end

function [ Tnorm_min ] = minimo_type2( MF_1,MF_2,Rules)
%FUNCION QUE REALIZA OPERACION MINIMO PARA LA COMBINACIÓN DE LOS
CONJUNTOS
% ES DECIR PARA LAS REGLAS DIFUSAS.
%PARAMETROS: MF1 y MF2 son los arreglos con las función de membresia
% primaria MF1=[UMFi LMF1] donde i=1,...,n n=numero de
% conjuntos difusos asi el arreglo tiene tantas filas
como
% conjuntos difusos hayan en el universo

%Calcula la dimension de MF_1. MF_1 = MF_2
NUM_FILAS = 1;
DIM = size(MF_1,NUM_FILAS);

%Dimensiona matriz de salida
Tnorm_min = zeros(DIM*2,3);

%OPERACIÓN MINIMO PARA UMF-----
-----
k = 1;
for i = 1 : DIM
    for j = 1 : DIM
        Tnorm_min(k,1) = min(MF_1(i,1),MF_2(j,1));%Guarda valores de
membresia mínimos
        k = k + 1;
    end
end
%-----
%OPERACIÓN MINIMO PARA LMF
k = 1;
for i = 1 : DIM
    for j = 1 : DIM
        Tnorm_min(k,2) = min(MF_1(i,2),MF_2(j,2));%Guarda valores de
membresia mínimos
        k = k + 1;
    end
end
%-----
%Almacena centro del conjunto de salida correspondiente
k = 1;
for i = 1 : DIM
    for j = 1 : DIM
        Tnorm_min(k,3) = Rules(MF_1(i,3),MF_2(j,3));
        k = k + 1;
    end
end
end

function [ YK ] = maximo_type2( Tnorm_min,
Centros_Conjuntos_de_Salida )
%Verifica conjuntos repetidos y aplica operacion máximo
%Tnorm_min es el arreglo que contiene por cada fila los intervalos y
el
%centro de conjunto de salida correspondiente en su 3ra Columna

NUM_FILAS = 1;

%Dimensiona matriz de salida en base al número de conjuntos de
salida
YK = double(zeros(size(Centros_Conjuntos_de_Salida,1),3));

%Copia los centros de conjuntos de salida en la tercera columna de
YK
YK(:,3) = double(Centros_Conjuntos_de_Salida);

for i = 1 : size(YK,NUM_FILAS)
    for j = 1 : size(Tnorm_min,NUM_FILAS)
        if YK(i,3) == Tnorm_min(j,3)
            YK(i,1) = max(YK(i,1),Tnorm_min(j,1));
            YK(i,2) = max(YK(i,2),Tnorm_min(j,2));
        end
    end
end
end

```

```

end

function [ K ] = localiza(Tnorm,YL )
%LOCALIZA EL VALOR DE K
% y(K) < y' < y(K+1)
NUM_FILAS = 1;
K = 0;
for i = 1 : size(Tnorm,NUM_FILAS)-1 %el 6 indica zize(Tnorm)-1
    if YL >= Tnorm(size(Tnorm,NUM_FILAS))
        K = size(Tnorm,NUM_FILAS);
        break
    else
        if (Tnorm(i) <= YL) && (YL < Tnorm(i+1))
            K = i;
            break;
        end
    end
end
end

function [ h_corregida ] = corrige_hr( h,Tnorm,K )
%CORRIGE H para Cr
% K es constante calculada por KM... y(K) < y' < y(k+1)
% dh es la incertidumbre cuando un conjunto se expresa en la forma
% [c+s , c-s] donde dh=s.
NUM_FILAS = 1;
%Dimensiona vector de salida
h_corregida = zeros(size(Tnorm,NUM_FILAS),1);

for i = 1 : K
    h_corregida(i) = Tnorm(i,2);
end

for i = (K+1) : size(Tnorm,NUM_FILAS)
    h_corregida(i) = Tnorm(i,1);
end

end

function [ h_corregida ] = corrige_hl( h,Tnorm,K )
%CORRIGE H para Cl
% K es constante calculada por KM... y(K) < y' < y(k+1)
% dh es la incertidumbre cuando un conjunto se expresa en la forma
% [c+s , c-s] donde dh=s.
NUM_FILAS = 1;
%Dimensiona vector de salida
h_corregida = zeros(size(Tnorm,NUM_FILAS),1);

for i = 1 : K
    h_corregida(i) = Tnorm(i,1);
end

for i = (K+1) : size(Tnorm,NUM_FILAS) %el 7 indica zize(Tnorm)
    h_corregida(i) = Tnorm(i,2);
end

end

end

Tiempo_Base_Sumado = TABU_SUMADO;%Aproximadamente 32 iter con h
= 0.004
Tiempo_Base_Eliminado = TABU_ELIMINADO; %Aproximadamente 60 iter
con h = 0.004
Mem_Explicita_Accion_Eliminada = REGLAS;
Tabu_Sumado_Mem_Atributiva =
zeros(size(REGLAS,1),size(REGLAS,2));
Tabu_Eliminado_Mem_Atributiva =
zeros(size(REGLAS,1),size(REGLAS,2));
Tabu_Eliminado_Candidato = zeros(size(Candidato,1),1);
Tabu_Eliminado_Candidato = Tabu_Eliminado_Candidato +
Tiempo_Base_Eliminado;
Residencia = zeros(size(REGLAS,1),size(REGLAS,2));
Cont_Permanencia_Candidato = zeros(size(Candidato,1),1);
Num_Candidato = zeros(size(Candidato,1),1);
end

if Tiempo_Actual >= Caos
%Contador = Contador + 1;
%fprintf(FID,'CortoPlazo');
Caos = Tiempo_Actual + CAOS;
Tiempo_Base_Sumado = TABU_SUMADO;%Aproximadamente 32 iter con h
= 0.004
Tiempo_Base_Eliminado = TABU_ELIMINADO; %Aproximadamente 60 iter
con h = 0.004
Mem_Explicita_Accion_Eliminada = REGLAS;
Tabu_Sumado_Mem_Atributiva =
zeros(size(REGLAS,1),size(REGLAS,2));
Tabu_Eliminado_Mem_Atributiva =
zeros(size(REGLAS,1),size(REGLAS,2));
Tabu_Eliminado_Candidato = zeros(size(Candidato,1),1);
Tabu_Eliminado_Candidato = Tabu_Eliminado_Candidato +
Tiempo_Base_Eliminado;
Residencia = zeros(size(REGLAS,1),size(REGLAS,2));
Residencia = Residencia + Tiempo_Actual;
Cont_Permanencia_Candidato = zeros(size(Candidato,1),1);
Num_Candidato = zeros(size(Candidato,1),1);
end

%Guarda copia de variables del modelo
Vter_aux = Vter;
P_aux = P;
Q_aux = Q;
id_aux = id;
iq_aux = iq;
Te_aux = Te;
U_aux = U;
UG_aux = UG;
Vref_aux = Vref;
dx_dt_aux = dx_dt;
REGLAS_aux = REGLAS;

%Dimensiona variables y arreglos
Reglas_Optimo = zeros(size(REGLAS,1),size(REGLAS,2));
Continua = 0;
Num_Regla = 0;

%Controlador Difuso tipo 2
if ALGORITM_RED_TIPO == KM
    Control_Intervalo_Tipo2_KM(A,B,DELTA_E,DELTA_DE,...
=
0,0,Error,Cambio_del_Error,REGLAS,CENTROS_DE_BARRA,...
TIPO_FUNCION_T2,NUM_CONJ_ENTRADA,U);
elseif ALGORITM_RED_TIPO == EKM
    Control_Intervalo_Tipo2_EKM(A,B,DELTA_E,DELTA_DE,...
=
0,0,Error,Cambio_del_Error,REGLAS,CENTROS_DE_BARRA,...
TIPO_FUNCION_T2,NUM_CONJ_ENTRADA,U);
end

if (Error == 0)
if Caos > Tiempo_Actual
for i = 1 : size(REGLAS,1)
for j = 1 : size(REGLAS,2)
Num_Regla = Num_Regla + 1;
if (solicita_búsqueda(Num_Regla) == Continua)

fprintf(FID,'MEMBRÉSIA DE REGLAS\r\n');
%for s = 1 : size(Tnorm_min,1)
% for t = 1 : size(Tnorm_min,2)
% fprintf(FID,'\t%f',Tnorm_min(s,t));
% end
% fprintf(FID,'\r\n');
%end

%Prueba candidato r en la posición i,j
for r = 1 : size(Candidato,1)
REGLAS(i,j) = Candidato(r);

%Controlador Difuso tipo 2
if ALGORITM_RED_TIPO == KM
    Control_Intervalo_Tipo2_KM(A,B,DELTA_E,DELTA_DE,...
=

```

F 1.7 Funciones para búsqueda por Tabú

```

function [ Reglas_Optimo ] = Busqueda_Tabu_Mem_CortoPlazo_Prueba02(Tiempo_Actual,...
=
A,B,DELTA_E,DELTA_DE,REGLAS,CENTROS_DE_BARRA,NUM_CONJ_ENTRADA,...
Candidato,dx_dt,Error,Cambio_del_Error)

% Los vectores Error_Cand y Cambio_del_Error_Cand guardan los
valores del
% error y cambio del error respectivamente, de todos los
candidatos.
global Vter Vref UG Te id iq Vbus P Q U FID Tnorm_min TABU_SUMADO
ALGORITM_RED_TIPO ...
TABU_ELIMINADO NUM_CANDIDATOS CRIT_ASP CAOS ALFA BETA
REGLAS_T2 TIPO_FUNCION_T2
persistent Tiempo_Base_Sumado Tiempo_Base_Eliminado
Tabu_Sumado_Mem_Atributiva
persistent Tabu_Eliminado_Mem_Atributiva Tabu_Eliminado_Candidato
Mem_Explicita_Accion_Eliminada Caos
persistent Residencia Cont_Permanencia_Candidato Num_Candidato

BAJA_INF = int8(1);
ALTA_INF = int8(2);
KM = int8(1);
EKM = int8(2);

%Parámetros dinámicos base Tabú
if isempty(Caos) || Tiempo_Actual == 0
    Caos = CAOS;

```

```

0,0,Error,Cambio_del_Error,REGLAS,CENTROS_DE_BARRA,...
    TIPO_FUNCION_T2,NUM_CONJ_ENTRADA,U);
elseif ALGORITM_RED_TIPO == EKM
    U
Control_Intervalo_Tipo2_EKM(A,B,DELTA_E,DELTA_DE,...
0,0,Error,Cambio_del_Error,REGLAS,CENTROS_DE_BARRA,...
    TIPO_FUNCION_T2,NUM_CONJ_ENTRADA,U);
end

%LLama funcion Runge Kutta
dx_dt = Runge_Kutta_4to(dx_dt);

%Resuelve Ecuaciones Dinamicas
Ecs_Dinamicas(dx_dt);

%Calcula Error y Cambio del error
Error_Cand(r) = Vref - Vter;
Cambio_del_Error_Cand(r) = (Error_Cand(r) - Error);

%Reinicia variables que fueron modificadas
Vter = Vter_aux;
P = P_aux;
Q = Q_aux;
id = id_aux;
iq = iq_aux;
U = U_aux;
Te = Te_aux;
Vref = Vref_aux;
UG = UG_aux;
dx_dt = dx_dt_aux;
REGLAS(i,j) = REGLAS_aux(i,j);
end

func_obj_NG = abs(Error_Cand(1) * ALFA +
Cambio_del_Error_Cand(1) * BETA);
func_obj_NM = abs(Error_Cand(2) * ALFA +
Cambio_del_Error_Cand(2) * BETA);
func_obj_NP = abs(Error_Cand(3) * ALFA +
Cambio_del_Error_Cand(3) * BETA);
func_obj_CE = abs(Error_Cand(4) * ALFA +
Cambio_del_Error_Cand(4) * BETA);
func_obj_PP = abs(Error_Cand(5) * ALFA +
Cambio_del_Error_Cand(5) * BETA);
func_obj_PM = abs(Error_Cand(6) * ALFA +
Cambio_del_Error_Cand(6) * BETA);
func_obj_PG = abs(Error_Cand(7) * ALFA +
Cambio_del_Error_Cand(7) * BETA);

func_obj = [func_obj_NG;func_obj_NM;func_obj_NP;...
func_obj_CE;func_obj_PP;func_obj_PM;func_obj_PG];

% Candidato 1
if (func_obj_NG == min(func_obj))
%Evite se analice la acción actual
if REGLAS(i,j) ~= Candidato(1)
if CRIT_ASP == ALTA_INF
%Realiza movimiento
REGLAS(i,j) = Candidato(1);
elseif CRIT_ASP == BAJA_INF
%Pregunta por atributo tabu activo sumado
if Tiempo_Actual >=
Tabu_Sumado_Mem_Atributiva(i,j)
%Pregunta por atributo tabu activo eliminado de
la memoria
%explicita para saber si se puede realizar
movimiento
if Mem_Explicita_Accion_Eliminada(i,j) ==
Candidato(1)
if Tiempo_Actual >=
Tabu_Eliminado_Mem_Atributiva(i,j)
%Cumplido el criterio de aspiración
entonces, realiza movimiento
REGLAS(i,j) = Candidato(1);
end
else
%Realiza movimiento
REGLAS(i,j) = Candidato(1);
end
end
end
end
% Candidato 2
if (func_obj_NM == min(func_obj))
%Evite se analice la acción actual
if REGLAS(i,j) ~= Candidato(2)
if CRIT_ASP == ALTA_INF
%Realiza movimiento
REGLAS(i,j) = Candidato(2);
elseif CRIT_ASP == BAJA_INF
%Pregunta por atributo tabu activo sumado
if Tiempo_Actual >=
Tabu_Sumado_Mem_Atributiva(i,j)
%Pregunta por atributo tabu activo eliminado de
la memoria
%explicita para saber si se puede realizar
movimiento
if Mem_Explicita_Accion_Eliminada(i,j) ==
Candidato(2)
if Tiempo_Actual >=
Tabu_Eliminado_Mem_Atributiva(i,j)
%Cumplido el criterio de aspiración
entonces, realiza movimiento
REGLAS(i,j) = Candidato(2);
end
else
%Realiza movimiento
REGLAS(i,j) = Candidato(2);
end
end
end
end
% Candidato 3
if (func_obj_NP == min(func_obj))
%Evite se analice la acción actual
if REGLAS(i,j) ~= Candidato(3)
if CRIT_ASP == ALTA_INF
%Realiza movimiento
REGLAS(i,j) = Candidato(3);
elseif CRIT_ASP == BAJA_INF
%Pregunta por atributo tabu activo sumado
if Tiempo_Actual >=
Tabu_Sumado_Mem_Atributiva(i,j)
%Pregunta por atributo tabu activo eliminado de
la memoria
%explicita para saber si se puede realizar
movimiento
if Mem_Explicita_Accion_Eliminada(i,j) ==
Candidato(3)
if Tiempo_Actual >=
Tabu_Eliminado_Mem_Atributiva(i,j)
%Cumplido el criterio de aspiración
entonces, realiza movimiento
REGLAS(i,j) = Candidato(3);
end
else
%Realiza movimiento
REGLAS(i,j) = Candidato(3);
end
end
end
end
% Candidato 4
if (func_obj_CE == min(func_obj))
%Evite se analice la acción actual
if REGLAS(i,j) ~= Candidato(4)
if CRIT_ASP == ALTA_INF
%Realiza movimiento
REGLAS(i,j) = Candidato(4);
elseif CRIT_ASP == BAJA_INF
%Pregunta por atributo tabu activo sumado
if Tiempo_Actual >=
Tabu_Sumado_Mem_Atributiva(i,j)
%Pregunta por atributo tabu activo eliminado de
la memoria
%explicita para saber si se puede realizar
movimiento
if Mem_Explicita_Accion_Eliminada(i,j) ==
Candidato(4)
if Tiempo_Actual >=
Tabu_Eliminado_Mem_Atributiva(i,j)
%Cumplido el criterio de aspiración
entonces, realiza movimiento
REGLAS(i,j) = Candidato(4);
end
else
%Realiza movimiento
REGLAS(i,j) = Candidato(4);
end
end
end
end
% Candidato 5
if (func_obj_PP == min(func_obj))
%Evite se analice la acción actual
if REGLAS(i,j) ~= Candidato(5)
if CRIT_ASP == ALTA_INF
%Realiza movimiento
REGLAS(i,j) = Candidato(5);
elseif CRIT_ASP == BAJA_INF
%Pregunta por atributo tabu activo sumado
if Tiempo_Actual >=
Tabu_Sumado_Mem_Atributiva(i,j)
%Pregunta por atributo tabu activo eliminado de
la memoria
%explicita para saber si se puede realizar
movimiento
if Mem_Explicita_Accion_Eliminada(i,j) ==
Candidato(5)
if Tiempo_Actual >=
Tabu_Eliminado_Mem_Atributiva(i,j)

```

```

                                %Cumplido el criterio de aspiración
entonces, realiza movimiento
                                REGLAS(i,j) = Candidato(5);
                                end
                                else
                                %Realiza movimiento
                                REGLAS(i,j) = Candidato(5);
                                end
                                end
                                end
                                end
                                %Candidato 6
                                if (func_obj_PM == min(func_obj))
                                %Evite se analice la acción actual
                                if REGLAS(i,j) ~= Candidato(6)
                                if CRIT_ASP == ALTA_INF
                                %Realiza movimiento
                                REGLAS(i,j) = Candidato(6);
                                elseif CRIT_ASP == BAJA_INF
                                %Pregunta por atributo tabu activo sumado
                                if
                                Tiempo_Actual >=
                                Tabu_Sumado_Mem_Atributiva(i,j)
                                %Pregunta por atributo tabu activo eliminado de
                                la memoria
                                %explicita para saber si se puede realizar
                                movimiento
                                if
                                Mem_Explicita_Accion_Eliminada(i,j) ==
                                Candidato(6)
                                if
                                Tiempo_Actual >=
                                Tabu_Eliminado_Mem_Atributiva(i,j)
                                %Cumplido el criterio de aspiración
                                entonces, realiza movimiento
                                REGLAS(i,j) = Candidato(6);
                                end
                                else
                                %Realiza movimiento
                                REGLAS(i,j) = Candidato(6);
                                end
                                end
                                end
                                end
                                end
                                %Candidato 7
                                if (func_obj_PG == min(func_obj))
                                %Evite se analice la acción actual
                                if REGLAS(i,j) ~= Candidato(7)
                                if CRIT_ASP == ALTA_INF
                                %Realiza movimiento
                                REGLAS(i,j) = Candidato(7);
                                elseif CRIT_ASP == BAJA_INF
                                %Pregunta por atributo tabu activo sumado
                                if
                                Tiempo_Actual >=
                                Tabu_Sumado_Mem_Atributiva(i,j)
                                %Pregunta por atributo tabu activo eliminado de
                                la memoria
                                %explicita para saber si se puede realizar
                                movimiento
                                if
                                Mem_Explicita_Accion_Eliminada(i,j) ==
                                Candidato(7)
                                if
                                Tiempo_Actual >=
                                Tabu_Eliminado_Mem_Atributiva(i,j)
                                %Cumplido el criterio de aspiración
                                entonces, realiza movimiento
                                REGLAS(i,j) = Candidato(7);
                                end
                                else
                                %Realiza movimiento
                                REGLAS(i,j) = Candidato(7);
                                end
                                end
                                end
                                end
                                end
                                %Asegura accion Cero en el centro de la tabla de reglas
                                if size(REGLAS,1) == 3
                                REGLAS(2,2) = Candidato(4);
                                end
                                %Asegura accion Cero en el centro de la tabla de reglas
                                if size(REGLAS,1) == 5
                                REGLAS(3,3) = Candidato(4);
                                end
                                if CRIT_ASP == BAJA_INF
                                %Principio Tabu
                                %Pregunta si no se realizo movimiento
                                if REGLAS(i,j) == REGLAS_aux(i,j)
                                if Tiempo_Actual >= Tabu_Sumado_Mem_Atributiva(i,j)
                                Tabu_Sumado_Mem_Atributiva(i,j) = Tiempo_Actual;
                                end
                                if
                                Tiempo_Actual >=
                                Tabu_Eliminado_Mem_Atributiva(i,j)
                                =
                                Tabu_Eliminado_Mem_Atributiva(i,j)
                                =
                                Tiempo_Actual;
                                end
                                else %Si no: inidica que la accion actual Rules_T2(i,j)
                                es diferente a la almacenada
                                %es decir se realizó movimiento de acción en matriz
                                de reglas.
                                %Asigna Multa a tiempo de tabu sumado en la
                                posicion (i,j)
                                Tabu_Sumado_Mem_Atributiva(i,j) = Tiempo_Actual +
                                Tiempo_Base_Sumado;
                                %Identifica y asigna tiempo de tabu eliminado en la
                                posicon (i,j)
                                for r = 1 : size(Candidato,1)
                                if REGLAS_aux(i,j) == Candidato(r)
                                %Asigna Multa de tiempo tabú eliminado en la
                                posición (i,j)
                                Tabu_Eliminado_Mem_Atributiva(i,j)
                                =
                                Tiempo_Actual + Tabu_Eliminado_Candidato(r);
                                %Coloca candidato en la memoria explicita de
                                %acciones eliminadas
                                Mem_Explicita_Accion_Eliminada(i,j)
                                =
                                Candidato(r);
                                end
                                end
                                end
                                end
                                end
                                %Llama a memoria largo plazo para odificar Tiempo base tabú
                                sumado
                                %Y el tiempo tabu eliminado propio de cada candidato
                                [Tiempo_Base_Sumado,Tabu_Eliminado_Candidato,Residencia,...
                                Cont_Permanencia_Candidato,Num_Candidato] = ...
                                Busqueda_Tabu_Mem_LargoPlazo_Prueba02(REGLAS_aux,REGLAS,...
                                Tiempo_Actual,Candidato,Tabu_Eliminado_Candidato,...
                                Tiempo_Base_Sumado,Caos,Residencia,Cont_Permanencia_Candidato,...
                                Num_Candidato);
                                end
                                end
                                %Copia Matriz de regla a devolver
                                Reglas_Optimo = REGLAS;
                                end
                                function
                                [Tabu_Base_Sumado_aux,Tabu_Eliminado_Cand_aux,Residencia,...
                                Cont_Permanencia_Candidato,Num_Candidato] = ...
                                Busqueda_Tabu_Mem_LargoPlazo_Prueba02(Rules_ant,Rules_act,...
                                Tiempo_Simulacion,Candidato,Tabu_Eliminado_Cand,...
                                Tabu_Base_Sumado,Caos,Residencia,...
                                Cont_Permanencia_Candidato,Num_Candidato)
                                %Función memoria largo plazo. El argumento Rules_ant es la matriz de
                                reglas
                                %almacenada como respaldo de memoria corto plazo, la matriz
                                Rules_act, es
                                %la matriz modificada y obtenida como solución elite de la memoria
                                corto
                                %plazo.
                                %Los argmentos de salida corresponden a:
                                %Tabu_Base_Sumado - Contiene el tiempo base modificado para la base
                                tabu
                                %sumado para un candidato en general q servira para la siguiente
                                iteración.
                                %Tabu_Eliminado_Cand - Es un vector de la misma dimensión que el
                                vector de
                                %candidatos que contienen el tiempo base tabu eliminado para cada
                                candidato
                                %en particular, para la siguiente iteración.
                                global FID
                                %persistent Residencia Cont_Permanencia_Candidato Num_Candidato
                                Num_Regla = int8(0);
                                Continua = int8(1);
                                %Contador de Frecuencia de Residencia
                                for i = 1 : size(Rules_act,1)
                                for
                                j = 1 : size(Rules_act,2)
                                Num_Regla = Num_Regla + 1;
                                if (solicita_búsqueda(Num_Regla) ~= Continua)
                                if Rules_ant(i,j) == Rules_act(i,j)
                                %Localiza q candidato ha permanecido para aumentar
                                contador de
                                %permanencia. Estos contadores son los numeradores para
                                la
                                %asignacion de multas e incentivos
                                for r = 1 : size(Candidato,1)
                                if Rules_act(i,j) == Candidato(r)
                                %Aumenta contador de permanencia. Numeradores
                                Cont_Permanencia_Candidato(r)
                                =
                                Cont_Permanencia_Candidato(r) + 1;
                                end
                                end
                                end
                                %Si la acción ha cambiado
                                if Rules_act(i,j) ~= Rules_ant(i,j)
                                %Localiza cual candidato dejó de ser parte de la
                                solución élite
                                for r = 1 : size(Candidato,1)
                                if Rules_ant(i,j) == Candidato(r)
                                %Calcula cuanto tiempo duró en la solución

```

```

Residencia(i,j);      Num_Candidato(r) = Tiempo_Simulacion - else
                    %Guarda tiempo de inicio de residencia Alto_Busqueda= 0;
                    Residencia(i,j) = Tiempo_Simulacion; end
                    end end
                    end end
                    end end
                    end
end

%Calcula el denominador en base al maximo de los numeradores
Denomin_TabuSumado = max(Cont_Permanencia_Candidato);
%Calcula numerador de las acciones sumadas
Num_TabuSumado = max(Num_Candidato);

Num_Regla = 0;

%Modifica Principio Tabú
for i = 1 : size(Rules_act,1)
    for j = 1 : size(Rules_act,2)
        Num_Regla = Num_Regla + 1;
        if (solicita_búsqueda(Num_Regla) ~= Continua)
            %Si no hubo movimiento realizado en la matriz de reglas.
Realiza
            %incentivos
            if Rules_act(i,j) == Rules_ant(i,j)
                %Incentiva el tiempo base tabu sumado, restando tiempo
de
                %duracion de una acción sumada
                %Tabu_Base_Sumado = abs(Tabu_Base_Sumado -
Num_TabuSumado / Denomin_TabuSumado);
                Tabu_Base_Sumado = Tabu_Base_Sumado - Num_TabuSumado /
Denomin_TabuSumado;
                if Tabu_Base_Sumado < 0
                    Tabu_Base_Sumado = 0;
                end
                for r = 1 : size(Tabu_Eliminado_Cand,1)
                    %Incentiva tiempo tabu eliminado de los candidatos
restando
                    %tiempo tabu eliminado para poder entrar en la
solución.
                    if Cont_Permanencia_Candidato(r) ~= 0
                        %Tabu_Eliminado_Cand(r) =
                        abs(Tabu_Eliminado_Cand(r) - Num_Candidato(r) /
Cont_Permanencia_Candidato(r));
                        Tabu_Eliminado_Cand(r) = Tabu_Eliminado_Cand(r)
- Num_Candidato(r) / Cont_Permanencia_Candidato(r);
                        if Tabu_Eliminado_Cand(r) < 0
                            Tabu_Eliminado_Cand(r) = 0;
                        end
                    end
                end
                %Si hubo movimiento, aplica multas
            else
                %Multa tiempo base tabu sumado, aumentando el tiempo
base que
                %tardara una acción dentro de la solución.
                for r = 1 : size(Tabu_Eliminado_Cand,1)
                    if Rules_ant(i,j) == Candidato(r)
                        %Tabu_Base_Sumado = abs(Tabu_Base_Sumado +
Num_TabuSumado / Denomin_TabuSumado);
                        Tabu_Base_Sumado = Tabu_Base_Sumado +
Num_TabuSumado / Denomin_TabuSumado;
                        %Multa a la acción con mayor tiempo fuera de la
%solución aumentando el tiempo tabu eliminado
del
                        %candidato
                        if Cont_Permanencia_Candidato(r) ~= 0
                            %Tabu_Eliminado_Cand(r) =
                            abs(Tabu_Eliminado_Cand(r) + Num_Candidato(r) /
Cont_Permanencia_Candidato(r));
                            Tabu_Eliminado_Cand(r) =
                            Tabu_Eliminado_Cand(r) +
                            Num_Candidato(r) /
                            Cont_Permanencia_Candidato(r);
                        end
                    end
                end
            end
        end
    end
end
end

%Datos devueltos por la función
Tabu_Eliminado_Cand_aux = Tabu_Eliminado_Cand;
Tabu_Base_Sumado_aux = Tabu_Base_Sumado;
end

function [ Alto_Busqueda ] = solicita_búsqueda( Numero_de_Regla )
%Si la regla en cuestión se encuentra ponderada (con grado de
pertenencia),
%se devuelve un 1; si no, se devuelve 0, con objeto de que TS (Tabú
search
%suspenda la búsqueda.
global Tnorm_min

if Tnorm_min(Numero_de_Regla,1) == 0 && Tnorm_min(Numero_de_Regla,2)
== 0
    Alto_Busqueda = 1;

```

**Señor concédeme:
Trabajar honestamente
Estudiar dedicadamente
Y amar desinteresadamente.**

