

**UNIVERSIDAD DE ORIENTE
CONSEJO DE ESTUDIOS DE POSTGRADO
NÚCLEO DE ANZOÁTEGUI
POSTGRADO DE INGENIERÍA ELÉCTRICA**



**OPTIMIZACIÓN DE UN CONTROLADOR
PID POR MÉTODOS CLÁSICOS Y
MEDIANTE LÓGICA DIFUSA**

Por:

Ing. Ignacio José Domínguez Mata

Trabajo Especial de Grado como requisito para optar al Título de Especialista en
Automatización e Informática Industrial.

Barcelona, junio de 2009

UNIVERSIDAD DE ORIENTE
CONSEJO DE ESTUDIOS DE POSTGRADO
NÚCLEO DE ANZOÁTEGUI
POSTGRADO DE INGENIERÍA ELÉCTRICA



**OPTIMIZACIÓN DE UN CONTROLADOR
PID POR MÉTODOS CLÁSICOS Y
MEDIANTE LÓGICA DIFUSA**

JURADO CALIFICADOR

Ing. Esp. Margarita Heraoui

ASESOR ACADÉMICO

JURADO PRINCIPAL

JURADO PRINCIPAL

Barcelona, junio de 2009

ENUNCIADO DEL ARTÍCULO 44

Los trabajos especiales de grado son de exclusiva propiedad de la Universidad de Oriente y sólo podrán ser utilizados para otros fines con el consentimiento del Consejo de Núcleo respectivo quién lo participará al Consejo Universitario.

DEDICATORIA

A mi familia

AGRADECIMIENTOS

A Dios Todopoderoso.

A mi familia.

A los Profesores de la VI Cohorte del Postgrado en Ingeniería Eléctrica de la Universidad de Oriente.

A la Profesora Margarita Heraoui, cuya asesoría fue de mucha ayuda en la realización de este trabajo.

ÍNDICE GENERAL

| | |
|---|-------------|
| ENUNCIADO DEL ARTÍCULO 44 | III |
| DEDICATORIA..... | IV |
| AGRADECIMIENTOS..... | V |
| ÍNDICE DE FIGURAS..... | IX |
| ÍNDICE DE TABLAS..... | XIV |
| NOMENCLATURA | XVII |
| RESUMEN | XIX |
| CAPÍTULO 1. INTRODUCCIÓN | 1 |
| 1.1 ANTECEDENTES | 1 |
| 1.2 PLANTEAMIENTO DEL PROBLEMA..... | 2 |
| 1.3 JUSTIFICACIÓN..... | 3 |
| 1.4 OBJETIVO DEL TRABAJO | 4 |
| 1.4.1 <i>Objetivo General</i> | 4 |
| 1.4.2 <i>Objetivos Específicos</i> | 4 |
| 1.5 ALCANCE | 5 |
| 1.6 METODOLOGÍA UTILIZADA | 5 |
| CAPÍTULO 2. MARCO TEÓRICO | 7 |
| 2.1 IDENTIFICACIÓN DEL PROCESO..... | 7 |
| 2.1.1 <i>Método de la Tangente de Ziegler y Nichols</i> | 8 |
| 2.1.2 <i>Método de la Tangente Modificado de Miller</i> | 9 |
| 2.1.3 <i>Método de Smith</i> | 10 |
| 2.1.4 <i>Método de Ho et al. (POMTM)</i> | 11 |
| 2.1.5 <i>Método de Ho et al. (Polo Doble Más Tiempo Muerto)</i> | 12 |
| 2.1.6 <i>Método de Jahanmiri Y Fallahi</i> | 13 |
| 2.1.7 <i>Método de Mikles y Fikar</i> | 14 |
| 2.2 AJUSTE DE CONTROLADORES PID | 16 |
| 2.2.1 <i>Método de Ziegler y Nichols con Razón de Asentamiento $1/4$</i> | 17 |
| 2.2.2 <i>Método de Cohen y Coon</i> | 17 |

| | | |
|----------|---|----|
| 2.2.3 | <i>Métodos de Sung, O, Lee, Lee y Yi</i> | 18 |
| 2.2.4 | <i>Método de Rovira, Murrill y Smith</i> | 19 |
| 2.2.5 | <i>Método de Rivera, Morari y Stogestad</i> | 20 |
| 2.3 | LÓGICA DIFUSA | 22 |
| 2.3.1 | <i>Conjuntos Difusos</i> | 23 |
| 2.3.2 | <i>Funciones de Pertenencia</i> | 23 |
| 2.3.3 | <i>Soporte de un Conjunto Difuso en un Universo de Discurso U</i> | 26 |
| 2.3.4 | <i>Centro de un Conjunto Difuso</i> | 26 |
| 2.3.5 | <i>Variable Lingüística</i> | 26 |
| 2.3.6 | <i>Estructura Básica de un Sistema Difuso</i> | 27 |
| 2.3.7 | <i>Difusificación</i> | 28 |
| 2.3.8 | <i>Base de Reglas Difusas</i> | 28 |
| 2.3.9 | <i>Inferencia Difusa</i> | 29 |
| 2.3.10 | <i>Desdifusificación</i> | 32 |
| 2.3.10.1 | <i>Desdifusificación por Centro de Áreas</i> | 32 |
| 2.3.10.2 | <i>Desdifusificación por Media de Centros</i> | 33 |

CAPÍTULO 3. DESARROLLO DEL PROGRAMA PARA EL CÁLCULO Y AJUSTE DE CONTROLADORES PID..... 34

| | | |
|-------|--|----|
| 3.1 | DEFINICIÓN DEL SISTEMA..... | 36 |
| 3.2 | MODELADO DEL SISTEMA..... | 36 |
| 3.3 | CÁLCULO DE PARÁMETROS DEL CONTROLADOR..... | 38 |
| 3.3.1 | <i>Modos de Operación: Servo y Regulador</i> | 39 |
| 3.4 | AJUSTE DE PARÁMETROS DEL CONTROLADOR..... | 39 |
| 3.4.1 | <i>Algoritmo de Ajuste</i> | 39 |
| 3.4.2 | <i>Sistema Difuso</i> | 40 |
| 3.4.3 | <i>Variable Difusa de Entrada</i> | 41 |
| 3.4.4 | <i>Variable Difusa de Salida</i> | 42 |
| 3.4.5 | <i>Base De Reglas Difusas</i> | 44 |

CAPÍTULO 4. IMPLEMENTACIÓN 45

| | | |
|-----|---|----|
| 4.1 | DEFINICIÓN DEL SISTEMA A CONTROLAR..... | 45 |
| 4.2 | MODELAJE DEL SISTEMA..... | 47 |
| 4.3 | CÁLCULO DEL CONTROLADOR PID..... | 52 |
| 4.4 | OPTIMIZACIÓN DE LA GANANCIA PROPORCIONAL DEL CONTROLADOR..... | |

| | | |
|--|---|-----------|
| 4.5 | INTERFAZ DE USUARIO | 55 |
| 4.5.1 | <i>Definición del Sistema a Controlar</i> | 56 |
| 4.5.2 | <i>Cálculo del Modelo del Sistema</i> | 58 |
| 4.5.3 | <i>Cálculo del Controlador PID</i> | 59 |
| 4.5.4 | <i>Optimización del Controlador PID</i> | 60 |
| CAPÍTULO 5. PRUEBA | | 62 |
| 5.1 | METODOLOGÍA DE PRUEBA | 62 |
| 5.2 | PRUEBA CON FUNCIÓN DE TRANSFERENCIA | 63 |
| 5.2.1 | PLANTA DE PRUEBA | 63 |
| 5.2.2 | IDENTIFICACIÓN DEL MODELO DE LA PLANTA | 63 |
| 5.2.3 | CÁLCULO DEL CONTROLADOR PID..... | 68 |
| 5.2.4 | AJUSTE DE LA CONSTANTE PROPORCIONAL DEL CONTROLADOR PID | 73 |
| 5.3 | PRUEBA CON CURVA DE REACCIÓN | 80 |
| 5.3.1 | PLANTA DE PRUEBA | 80 |
| 5.3.2 | IDENTIFICACIÓN DEL MODELO DE LA PLANTA | 82 |
| 5.3.3 | CÁLCULO DEL CONTROLADOR PID..... | 86 |
| 5.3.4 | AJUSTE DE LA CONSTANTE PROPORCIONAL DEL CONTROLADOR PID | 91 |
| CONCLUSIONES Y RECOMENDACIONES..... | | 93 |
| CONCLUSIONES | | 93 |
| RECOMENDACIONES | | 94 |
| BIBLIOGRAFÍA | | 95 |
| METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:..... | | 1 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 2.1. Lazo de Control Realimentado | 7 |
| Figura 2.2. Ejemplo de Curva de Reacción Utilizada por Ziegler y Nichols..... | 9 |
| Figura 2.3. Ejemplo de Curva de Reacción Utilizada por Miller..... | 10 |
| Figura 2.4. Ejemplo de Curva de Reacción Utilizada por Smith..... | 11 |
| Figura 2.5. Ejemplo de Curva de Reacción Utilizada por Ho <i>et al</i> | 12 |
| Figura 2.6. Ejemplo de Curva de Reacción Utilizada por Jahanmiri y Fallahi..... | 14 |
| Figura 2.7. Ejemplo de Curva de Reacción de un Sistema de Segundo Orden Subamortiguado Utilizada por Mikles y Fikar..... | 15 |
| Figura 2.8. Estructura básica de un controlador IMC | 21 |
| Figura 2.9. Representación Gráfica de una Función de Pertenencia..... | 24 |
| Figura 2.10. Estructura Básica de un Sistema Difuso | 27 |
| Figura 2.11. Implicación por la Regla del Mínimo | 31 |
| Figura 2.12. Implicación por la Regla del Producto | 31 |
| Figura 3.1. Diagrama de Flujo del Sistema Propuesto..... | 35 |
| Figura 3.2. Estructura del Sistema Difuso Propuesto | 41 |
| Figura 3.3. Funciones de Pertenencia para la Variable de Entrada..... | 42 |
| Figura 3.4. Funciones de Pertenencia para la Variable Salida..... | 44 |
| Figura 4.1. Pantalla Inicial del Programa CYRCPID | 56 |
| Figura 4.2. Definición de la Planta Mediante Función de Transferencia..... | 57 |

| | |
|---|----|
| Figura 4.3. Definición de la Planta Mediante Curva de Reacción | 58 |
| Figura 4.4. Opciones para el Cálculo del Modelo del Sistema | 59 |
| Figura 4.5. Opciones para el Cálculo del Controlador PID | 60 |
| Figura 4.6. Opciones para la Optimización del Controlador PID | 61 |
| Figura 5.1. Definición de la Función de Transferencia de la Planta de Prueba | 64 |
| Figura 5.2. Cálculo del Modelo según Smith..... | 65 |
| Figura 5.1. Comparación de la Curva de Reacción Original y la del Modelo según Smith..... | 66 |
| Figura 5.2. Comparación de la Curva de Reacción Original y la del Modelo según Ho <i>et al.</i> POMTM..... | 66 |
| Figura 5.4. Comparación de la Curva de Reacción Original y la del Modelo según Jahanmiri y Fallahi..... | 67 |
| Figura 5.5. Cálculo del Controlador según Rovira <i>et al.</i> IAE..... | 68 |
| Figura 5.6. Desempeño del Sistema con el Controlador Calculado por el método de Rovira | 69 |
| Figura 5.7. Desempeño del Sistema con el Controlador Calculado por el método de Sung para Servos | 70 |
| Figura 5.8. Desempeño del Sistema con el Controlador Calculado por el método de Sung para Reguladores..... | 70 |
| Figura 5.9. Desempeño del Sistema con el Controlador Calculado por el método de Rivera | 71 |
| Figura 5.10. Desempeño del Sistema con el Controlador Calculado por el | |

| | |
|---|----|
| Calculado según Rivera, Optimizado para Sobrenivel Máximo del 2 % como Regulador | 78 |
| Figura 5.19. Sistema Definido por Función de Transferencia, Controlador Calculado según Ziegler y Nichols, Optimizado para Sobrenivel Máximo del 2 % como Regulador | 78 |
| Figura 5.20. Planta de Prueba Definida por su Curva de Reacción | 81 |
| Figura 5.21. Definición de la Planta de Prueba desde Archivo de Datos de Curva de Reacción | 82 |
| Figura 5.22. Curva de Reacción de la Planta de Prueba Definida desde un Archivo de Datos..... | 83 |
| Figura 5.23. Comparación entre la Curva de Reacción de la Planta de Prueba y el Modelo de Primer Orden más Tiempo Muerto según Smith | 84 |
| Figura 5.24. Comparación entre la Curva de Reacción de la Planta de Prueba y el Modelo de Primer Orden más Tiempo Muerto según Ho <i>et</i> <i>al.</i> | 85 |
| Figura 5.25. Comparación entre la Curva de Reacción de la Planta de Prueba y el Modelo de Segundo Orden más Tiempo Muerto según Ho <i>et</i> <i>al.</i> | 85 |
| Figura 5.26. Comparación entre la Curva de Reacción de la Planta de Prueba y el Modelo de Segundo Orden más Tiempo Muerto según Jahanmiri y Fallahi..... | 86 |
| Figura 5.27. Desempeño del Sistema Definido por Curva de Reacción, | |

| | |
|--|----|
| Controlador Calculado según Rovira | 87 |
| Figura 5.28. Desempeño del Sistema Definido por Curva de Reacción, Controlador Calculado según Sung (Servos) | 87 |
| Figura 5.29. Desempeño del Sistema Definido por Curva de Reacción, Controlador Calculado según Sung (Reguladores) | 88 |
| Figura 5.30. Desempeño del Sistema Definido por Curva de Reacción, Controlador Calculado según Rivera | 88 |
| Figura 5.31. Desempeño del Sistema definido por Curva de Reacción, Controlador Calculado según Ziegler y Nichols | 89 |
| Figura 5.32. Desempeño del Sistema Definido por Curva de Reacción, Controlador Calculado según Cohen y Coon | 89 |
| Figura 5.33. Sistema Definido por Curva de Reacción, Controlador Calculado según Sung (Servos), Optimizado para Sobrenivel Máximo del 0,3 % como Regulador | 92 |
| Figura 5.34. Definido por Curva de Reacción, Controlador Calculado según Cohen y Coon, Optimizado para Sobrenivel Máximo del 0,3 % como Regulador | 92 |

ÍNDICE DE TABLAS

| | |
|--|----|
| Tabla 2.1. Constantes para las ecuaciones del método de Rovira et al..... | 20 |
| Tabla 2.2. Parámetros PID-IMC de Rivera et al..... | 22 |
| Tabla 5.1. Parámetros de los Modelos Calculados para la Planta de Prueba mediante Función de Transferencia | 65 |
| Tabla 5.2. Parámetros del Controlador PID Calculado según los Distintos Métodos Utilizados | 69 |
| Tabla 5.3. Desempeño del Sistema Definido por Función de Transferencia, en Modo Servo..... | 72 |
| Tabla 5.4. Desempeño del Sistema Definido por Función de Transferencia, en Modo Regulador..... | 73 |
| Tabla 5.5. Parámetros de los Controladores PID Optimizados para Modo Servo, con un sobrenivel máximo del 15 %..... | 73 |
| Tabla 5.6. Desempeño del Sistema Definido por Función de Transferencia, Optimizado para un sobrenivel máximo del 15 % como Servo, Funcionando como Servo..... | 74 |
| Tabla 5.7. Desempeño del Sistema Definido por Función de Transferencia, Optimizado para un sobrenivel máximo del 15 % como Servo, Funcionando como Regulador | 74 |

| | |
|--|----|
| Tabla 5.8. Parámetros de los Controladores PID Optimizados para Modo Regulador | 79 |
| Tabla 5.9. Desempeño del Sistema Definido por Función de Transferencia, Optimizado para Modo Regulador, Funcionando como Servo, con un sobrenivel máximo del 2 %..... | 79 |
| Tabla 5.10. Desempeño del Sistema Definido por Función de Transferencia, Optimizado para Modo Regulador, Funcionando como Regulador, con un sobrenivel máximo del 2 %..... | 80 |
| Tabla 5.11. Datos de la Curva de Reacción de la Planta de Prueba..... | 81 |
| Tabla 5.10. Parámetros de los Modelos Calculados para la Planta de Prueba..... | 84 |
| Tabla 5.11. Parámetros del Controlador PID Calculado según los Distintos Métodos Utilizados, para el Sistema Definido por Curva de Reacción..... | 90 |
| Tabla 5.12. Desempeño del Sistema Definido por Curva de Reacción, en Modo Servo | 90 |
| Tabla 5.13. Desempeño del Sistema Definido por Curva de Reacción, en Modo Regulador | 90 |
| Tabla 5.14. Parámetros de los Controladores PID para la Planta Definida por Curva de Reacción, Optimizados para un Sobrenivel Máximo del 0,3 % como Regulador..... | 91 |
| Tabla 5.15. Desempeño del Sistema Optimizado para un Sobrenivel Máximo del 0,3 % como Regulador, Funcionando como Servomecanismo..... | 91 |

Tabla 5.16. Desempeño del Sistema Optimizado para un Sobrenivel Máximo

del 0,3 % como Regulador, Funcionando como Regulador..... 92

NOMENCLATURA

| | |
|------------|---|
| Δu | Cambio en la entrada de un sistema |
| Δy | Cambio en la salida de un sistema |
| $E(s)$ | Señal de error de un lazo de control realimentado, definida como $E(s) = R(s) - Y(s)$ |
| $G_c(s)$ | Función de Transferencia del controlador PID. |
| $Go(u)$ | Universo de discurso de la variable de salida de un sistema difuso |
| $G_p(s)$ | Función de Transferencia de la planta. |
| IAEP | Integral del error absoluto de predicción del modelo de un sistema. |
| K_c | Ganancia proporcional de un controlador PID |
| k_p | Ganancia de estado estacionario del modelo de un proceso |
| $\mu(x)$ | Función de pertenencia de una variable difusa |
| $R(s)$ | Entrada de referencia de un lazo de control realimentado |
| t | Constante de tiempo de un proceso |
| t_2 | Tiempo en el que la respuesta de un sistema alcanza el 2% del valor final |
| t_{28} | Tiempo en el que la respuesta de un sistema alcanza el 28,3% del valor final |
| t_{35} | Tiempo en el que la respuesta de un sistema alcanza el 35% del valor final |
| t_5 | Tiempo en el que la respuesta de un sistema alcanza el 5% del valor final |
| t_{63} | Tiempo en el que la respuesta de un sistema alcanza el 63,2% del valor final |
| t_{70} | Tiempo en el que la respuesta de un sistema alcanza el 70% del valor final |
| t_{85} | Tiempo en el que la respuesta de un sistema alcanza el 85% del valor final |

| | |
|----------|--|
| t_{90} | Tiempo en el que la respuesta de un sistema alcanza el 90% del valor final |
| T_d | Constante de derivación de un controlador PID |
| T_i | Constante de integración de un controlador PID |
| t_m | Tiempo muerto de un proceso |
| u_i | Valor discreto de la variable u en su universo de discurso |
| w_o | Frecuencia de oscilación de un sistema de segundo orden |
| $Y(s)$ | Salida de un lazo de control realimentado |
| $y_0(t)$ | Respuesta del sistema original a una entrada escalón |
| $y_1(t)$ | Respuesta del modelo del sistema original a una entrada escalón |
| ζ | Coefficiente de amortiguamiento de un sistema de segundo orden |
| $Z(s)$ | Perturbación |

RESUMEN

En este trabajo se desarrolla un programa que identifica el modelo de una planta, calcula un controlador PID y modifica su ganancia proporcional para que cumpla con un sobrenivel máximo establecido.

La identificación del modelo de la planta se realiza mediante el cálculo de varios modelos de primer y segundo orden y la selección del que mejor representa el comportamiento de la planta, mediante la comparación de la Integral del Error Absoluto de cada modelo, respecto a la curva de reacción original.

El cálculo del controlador PID se realiza por varios métodos, lo que permite la comparación del desempeño del sistema con cada uno de ellos. Finalmente, la constante proporcional del controlador PID seleccionado se modifica con ayuda de lógica difusa para lograr que el sistema en lazo cerrado se comporte con el sobrenivel máximo que se desea.

Los resultados obtenidos muestran que el programa desarrollado cumple con el objetivo de lograr un controlador PID adecuado para un sobrenivel definido.

CAPÍTULO 1. INTRODUCCIÓN

1.1 ANTECEDENTES

OPTIMUM SETTINGS FOR AUTOMATIC CONTROLLERS

Publicado por J.G. Ziegler y N. B. Nichols en la edición de noviembre de 1942 de la revista Transactions of the American Society of Mechanical Engineers, este trabajo desarrolla una metodología para el ajuste de controladores PID, basada en una serie de ensayos y simulaciones [1].

MÉTODOS DE SINTONIZACIÓN DE CONTROLADORES PID QUE OPERAN COMO REGULADORES.

Presentado por Víctor M. Alfaro Ruiz en el volumen 12 de la revista costarricense Ingeniería en el año 2002. Alfaro realiza una comparación de los métodos de sintonización de controladores PID propuestos por Ziegler y Nichols, Cohen y Coon, López *et al.*, Sung *et al.*, Kaya y Sheik, para sintonizar controladores que actúen como reguladores. Alfaro utilizó el programa VisSimTM para realizar simulaciones del comportamiento de una planta de prueba más un controlador PID sintonizado con cada uno de los métodos a evaluar. La evaluación de las prestaciones de cada método la realizó apoyándose en índices de desempeño y robustez definidos por él mismo, y en el desempeño de un controlador ajustado con el criterio IAE. El trabajo concluyó que: 1) el método de Sung *et al.* era el mejor para sintonizar controladores PID ideales; y 2) los métodos de Kaya y Sheik IAE son los más robustos [2].

MÉTODOS DE SINTONIZACIÓN DE CONTROLADORES PID QUE OPERAN COMO SERVOMEKANISMOS.

Víctor M. Alfaro Ruiz presentó este trabajo en el volumen 13 de la revista Costarricense Ingeniería en el año 2003. Alfaro realiza una comparación de los métodos de sintonización de controladores PID propuestos por Rovira, Murril y Smith, Kaya y Sheib, y otros, para sintonizar controladores que actúen como servomecanismos. Alfaro utilizó el programa VisSimTM para realizar simulaciones del comportamiento de una planta de prueba más un controlador PID sintonizado con cada uno de los métodos a evaluar. La evaluación de las prestaciones de cada método la realizó apoyándose en índices de desempeño y robustez definidos por él mismo, y en el desempeño de un controlador ajustado con el criterio IAE. El trabajo concluyó que el método de Sung *et al.* era el mejor para sintonizar controladores PID ideales cuando se busca una respuesta rápida y bajos sobreniveles [3].

DESIGN OF PI AND PID CONTROLLERS WITH TRANSIENT PERFORMANCE SPECIFICATION.

Presentado en IEEE Transactions on Education, Volumen 45, Número 4, Noviembre de 2002, por J.C. Basilio y S. R. Matos. En este trabajo, los autores proponen una metodología para el ajuste de controladores PI y PID, la cual funciona tanto para sistemas con respuesta monotónica como para sistemas subamortiguados. La metodología se basa en la asignación de polos y permite un ajuste sistemático de la ganancia para cumplir con especificaciones de régimen transitorio, incluyendo la no existencia de sobrepasos [4].

1.2 PLANTEAMIENTO DEL PROBLEMA

Los valores de ajuste para un controlador PID calculados mediante cualquiera de los métodos clásicos (Ziegler y Nichols, Cohen y Coon, etc.) con frecuencia no permiten que el sistema en lazo cerrado satisfaga requerimientos de bajos sobrepasos y/o tiempos de asentamiento cortos, por lo que es necesario efectuar manualmente ajustes adicionales en campo o en el laboratorio mediante ensayo y error. En este

sentido se han realizado varios trabajos con miras a mejorar la precisión en los ajustes calculados.

Alfaro realizó comparaciones de varios métodos de sintonización de controladores PID, tanto reguladores [2] como servomecanismos [3], tomando como base un controlador PID “óptimo” ajustado por el criterio de IAE. En todos los casos, los sobrepasos y/o tiempos de asentamiento obtenidos con cualquiera de los métodos utilizados fueron superiores a los obtenidos con el controlador “óptimo”.

En este trabajo se desarrolla un programa que calcula el modelo que mejor representa el sistema a controlar y utiliza métodos clásicos y lógica difusa para calcular un controlador PID que hace que el sistema cumpla con un sobrenivel máximo establecido.

1.3 JUSTIFICACIÓN

En la actualidad, el modelado y la simulación son actividades indispensables cuando se realizan análisis y diseño de sistemas de cierta complejidad. El objetivo es ayudar o dar el soporte necesario al diseñador durante este proceso. Mediante el uso de software es posible establecer una valoración final antes de que los sistemas sean implementados, aliviando la necesidad de experimentos caros.

El proceso de sintonización de un controlador PID para un sistema industrial consta de dos etapas: la identificación y el cálculo de los parámetros del controlador. Para la identificación del sistema existen varios métodos, por lo que se hace necesario calcular varios modelos y seleccionar el que mejor represente el sistema a controlar. De igual manera sucede con el cálculo de los parámetros de un controlador PID, existen varios métodos que dependen del modelo del sistema.

Desde el punto de vista académico es conveniente disponer de una herramienta

que permita calcular el modelo que mejor identifica a un sistema, calcule los parámetros de un controlador PID, y que también simule la operación del sistema en lazo cerrado para verificar los parámetros de ajuste calculado.

Hasta donde se conoce, tal herramienta no está disponible, por lo que se hace necesario el desarrollo de un programa que identifique el sistema a controlar, y calcule un controlador PID para que el sobrenivel no exceda de un valor preestablecido.

1.4 OBJETIVO DEL TRABAJO

1.4.1 Objetivo General

Optimizar los parámetros de ajustes para un controlador PID por métodos clásicos y mediante lógica difusa.

1.4.2 Objetivos Específicos

Definir el modelo que mejor represente el sistema a controlar, entre varios modelos de primero y segundo orden, a partir de los datos de la curva de reacción o de la función de transferencia del sistema.

Desarrollar un algoritmo para el cálculo de los valores de ajuste para la constante proporcional y los tiempos de integración y derivación de un controlador PID utilizando métodos clásicos.

Desarrollar un algoritmo basado en lógica difusa que permita ajustar la ganancia proporcional calculada para que el sistema en lazo cerrado cumpla con el máximo sobrenivel deseado.

Desarrollar una Interfaz Hombre-Máquina que permita al usuario la utilización de los algoritmos de cálculo y refinamiento.

1.5 ALCANCE

En este trabajo se desarrolla un programa cuyas funciones son:

Definir la planta para la cual se calculará un controlador PID, mediante una función de transferencia o mediante la curva de reacción obtenida al realiza una prueba escalón.

Identificar el modelo de la planta, para lo cual se calculan modelos de primer y segundo orden según Smith, Ho *et al.*, Jahanmiri y Fallahi, y Mikles y Fikar, y se selecciona aquél que mejor represente la curva de reacción o la función de transferencia utilizada en la definición.

Cálculo de un controlador PID mediante uno de los siguientes métodos clásicos: Ziegler y Nichols con razón de asentamiento $\frac{1}{4}$, Cohen y Coon, Sung *et al.*, Rovira *et al.* y Rivera *et al.*

Utilización de un algoritmo basado en lógica difusa para ajustar la constante proporcional del controlador PID calculado, para que el sistema alcance un sobrenivel que no exceda un valor establecido por el usuario.

1.6 METODOLOGÍA UTILIZADA

El trabajo se realizó utilizando una metodología investigativa y se estructuró en cinco capítulos. El capítulo 1 es un capítulo introductorio en el que se presentan los antecedentes, el problema a resolver, los objetivos generales y específicos, así como también la justificación y el alcance del trabajo.

El capítulo 2 presenta el marco teórico utilizado en este trabajo: se presentan en

forma breve los conceptos sobre identificación de procesos, ajustes de controladores PID y lógica difusa utilizados.

En los capítulos 3 y 4 se desarrolla e implementa el programa, mientras que en el capítulo 5 se muestra un resumen de las pruebas realizadas para validar su funcionamiento.

CAPÍTULO 2. MARCO TEÓRICO

Para poder sintonizar el controlador $G_c(s)$ de un lazo de control, como el mostrado en la figura 2.1, es necesario identificar primero la dinámica del proceso que se va a controlar, $G_p(s)$, para luego obtener los parámetros del controlador, empleando el método de sintonización seleccionado. El proceso de sintonización del controlador consta así de dos etapas: identificación y sintonización.

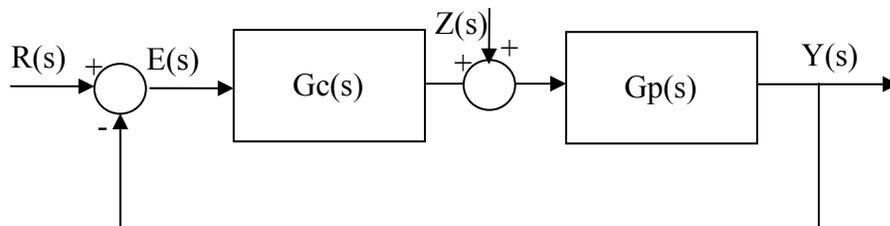


Figura 2.1. Lazo de Control Realimentado

2.1 IDENTIFICACIÓN DEL PROCESO

Para poder sintonizar el controlador PID de un lazo de control realimentado, es indispensable contar con información del comportamiento dinámico del proceso controlado. Dicho comportamiento puede estar dado por la función de transferencia del proceso a controlar o por la función de transferencia de un modelo adecuado, el cual es generalmente uno de los siguientes:

Primer Orden más Tiempo Muerto (POMTM):

$$G_p(s) = \frac{k_p e^{-t_m s}}{\tau s + 1} \quad (2.1)$$

Segundo Orden más Tiempo Muerto (SOMTM) con Polo Doble:

$$Gp(s) = \frac{k_p e^{-t_m s}}{(\tau s + 1)^2} \quad (2.2)$$

Segundo Orden más Tiempo Muerto (SOMTM):

$$Gp(s) = \frac{k_p e^{-t_m s}}{\tau^2 s^2 + 2\tau\zeta s + 1} \quad (2.3)$$

Segundo Orden sin Tiempo Muerto (SOSTM):

$$Gp(s) = \frac{Z\omega_0^2}{s^2 + 2\omega_0\zeta s + \omega_0^2} \quad (2.4)$$

Los parámetros de estos modelos se pueden identificar a partir de la curva de reacción del proceso (respuesta a una entrada escalón), conocidos los tiempos requeridos para alcanzar dos o tres puntos sobre la misma.

2.1.1 Método de la Tangente de Ziegler y Nichols

El método de la tangente de Ziegler y Nichols [1] es un modelo gráfico que se basa en la curva de reacción del proceso a un cambio en escalón en la salida del controlador, con el fin de determinar un modelo de primer orden más tiempo muerto (POMTM) de la forma:

$$Gp(s) = \frac{k_p e^{-t_m s}}{\tau s + 1} \quad (2.5)$$

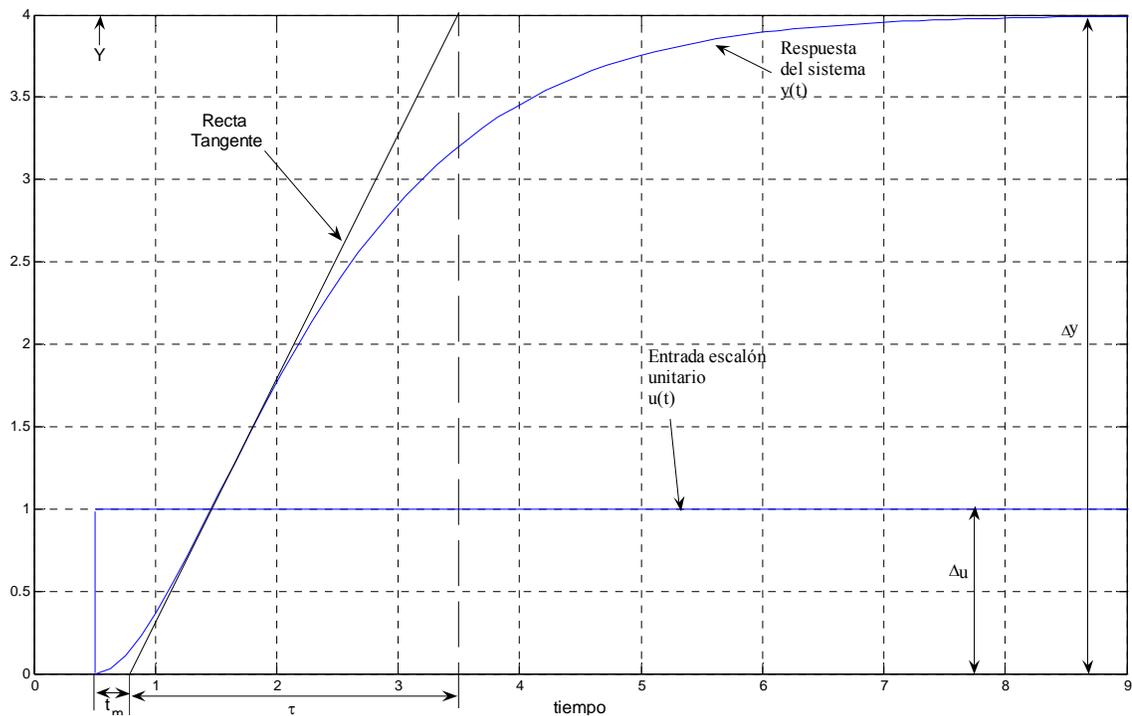


Figura 2.2. Ejemplo de Curva de Reacción Utilizada por Ziegler y Nichols

La recta tangente se traza en el punto de inflexión, donde la pendiente es máxima. El punto de corte con el eje de las ordenadas (tiempo) es el tiempo muerto aparente del sistema (t_m), mientras que el punto donde la recta tangente corta la línea del valor final es la constante de tiempo (τ). La ganancia de estado estacionario (k_p) se obtiene como la relación entre el cambio en la salida respecto al cambio en la entrada:

$$k_p = \frac{\Delta y}{\Delta u} \quad (2.6)$$

2.1.2 Método de la Tangente Modificado de Miller

Este método es muy similar al de Ziegler y Nichols, con la diferencia de que la constante de tiempo del sistema se calcula como el tiempo en el que la salida alcanza

un valor del 63,2% del valor final, como se observa en la figura 2.3. El resto de los parámetros se calcula igual que el método de Ziegler y Nichols [5].

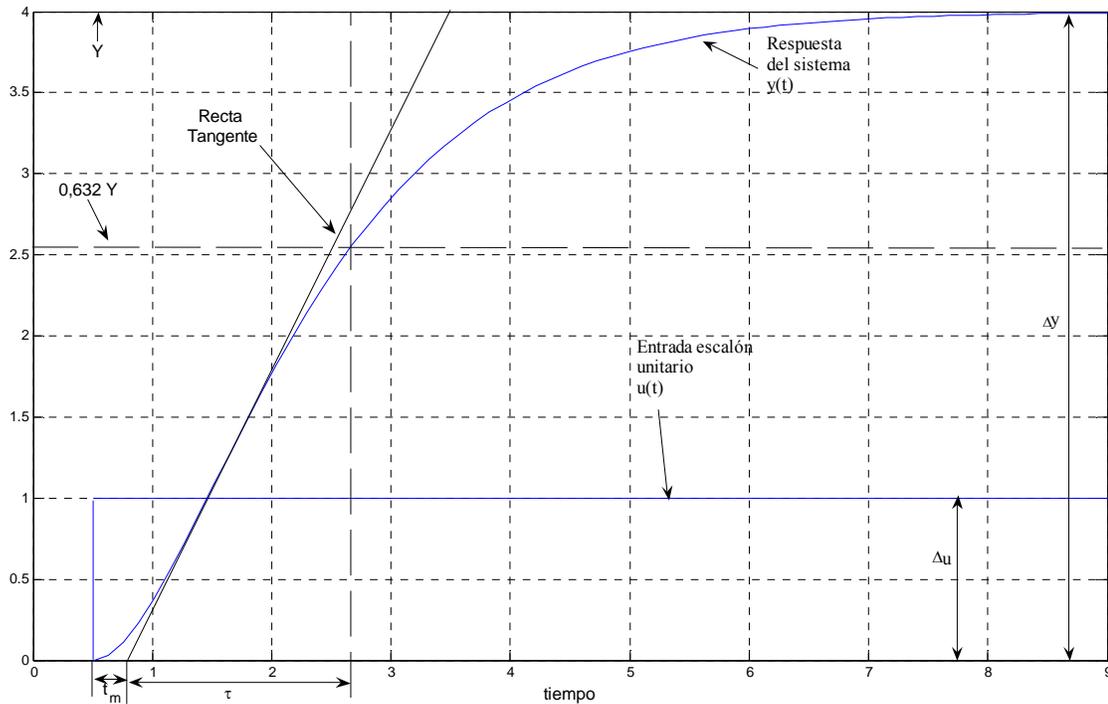


Figura 2.3. Ejemplo de Curva de Reacción Utilizada por Miller

2.1.3 Método de Smith

Smith utilizó una curva de reacción del proceso como la mostrada en la figura 2.4, e identificó los puntos en los que la respuesta del sistema alcanza 28,3% (t_{28}) y el 63,2% (t_{63}) del valor final, que corresponden a:

$$t_{28} = t_m + \tau/3 \quad (2.7)$$

$$t_{63} = t_m + \tau \quad (2.8)$$

Resolviendo el sistema de ecuaciones anterior para t_m y τ , se tiene que:

$$\tau = 1,5(t_{63} - t_{28}) \quad (2.9)$$

$$t_m = t_{63} - \tau \quad (2.10)$$

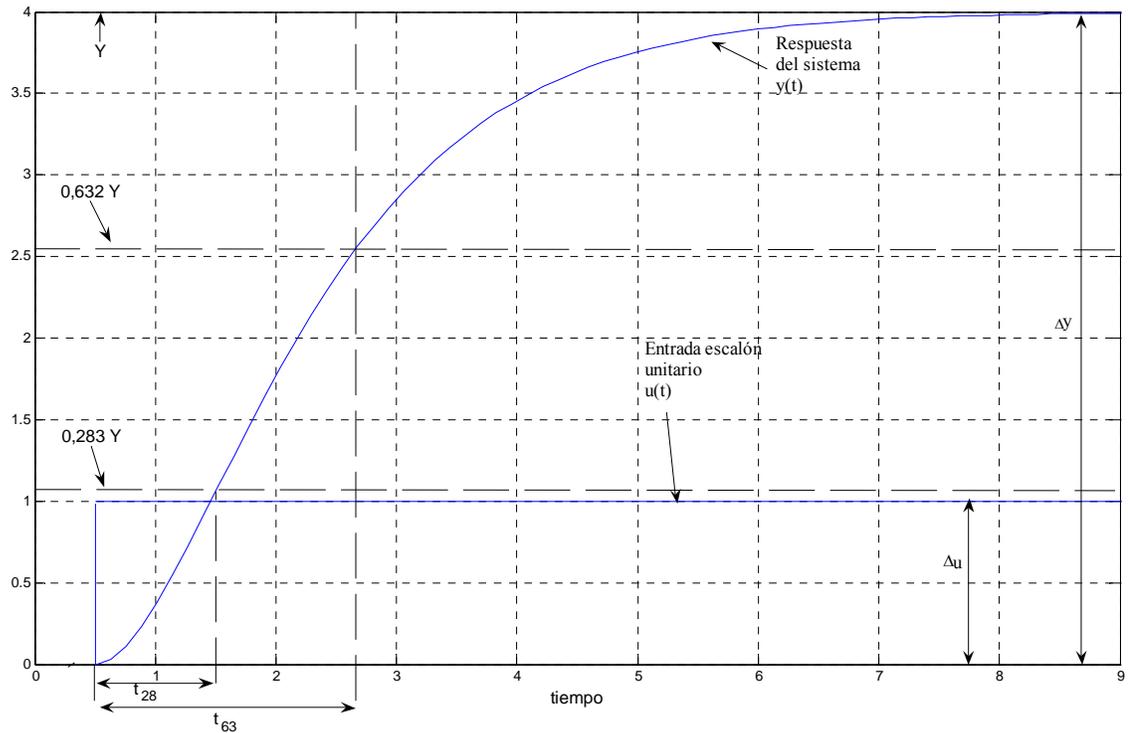


Figura 2.4. Ejemplo de Curva de Reacción Utilizada por Smith

2.1.4 Método de Ho *et al.* (POMTM)

Ho, Hang y Cao (Ho *et al.*) [5] desarrollaron un modelo que coincide con la curva de la respuesta escalón del sistema en dos puntos, que son el 35% y el 85% del valor final., como muestra la figura 2.5 Ho *et al.* establecieron que los parámetros de un modelo POMTM se pueden calcular como:

$$\tau = -0,670t_{35} + 0,670t_{85} \quad (2.11)$$

$$t_m = 1,300t_{35} - 0,290t_{85} \quad (2.12)$$

La ganancia de estado estacionario se calcula mediante (2.6)

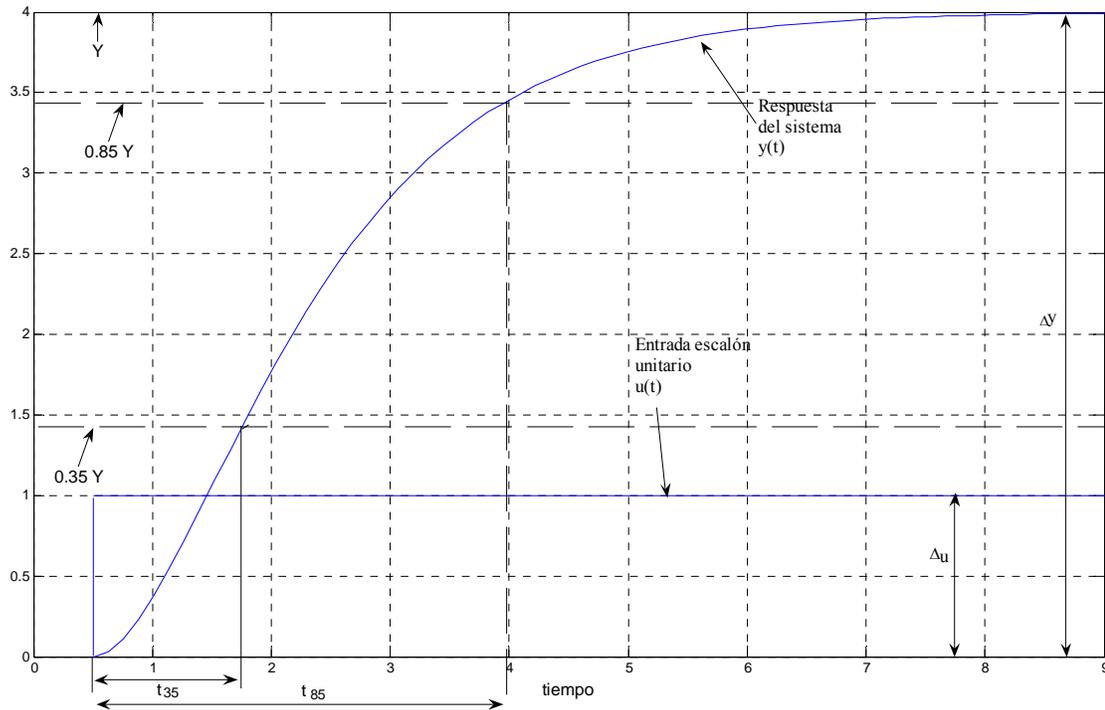


Figura 2.5. Ejemplo de Curva de Reacción Utilizada por Ho *et al.*

El modelo de POMTM de Ho *et al.* tiene, sin embargo, una limitante: debe cumplirse que $1,300t_{35} - 0,290t_{85} \geq 0$, ó $t_{35} \geq 0,223t_{85}$. De lo contrario, el tiempo muerto resultante sería negativo.

2.1.5 Método de Ho *et al.* (Polo Doble Más Tiempo Muerto)

Ho *et al.* también desarrollaron un método para el cálculo de modelos de segundo orden más tiempo muerto (SOMTM) de la forma:

$$Gp(s) = \frac{k_p e^{-t_m s}}{(\tau s + 1)^2} \quad (2.12)$$

Al igual que el método para calcular modelos de POMTM, este método permite calcular la constante de tiempo y el tiempo muerto aparente del sistema sobre la base

de dos puntos de la curva de reacción del proceso en los que la respuesta del sistema tiene valores del 35% y el 85% del valor final [5]. De acuerdo con este método,

$$\tau = -0,463t_{35} + 0,463t_{85} \quad (2.13)$$

$$t_m = 1,574t_{35} - 0,574t_{85} \quad (2.14)$$

La limitante del modelo de POMTM de Ho *et al.*, relativa a la relación entre t_{85} y t_{35} también aplica para este caso, solamente que aquí se debe cumplir que $1,574t_{35} - 0,574t_{85} \geq 0$, ó $t_{35} \geq 0,365t_{85}$.

2.1.6 Método de Jahanmiri Y Fallahi

Este método permite calcular un modelo SOMTM de la forma mostrada en la ecuación (2.3), basado en tres puntos de la curva de reacción del proceso: el 2% o el 5% (el que resulte en menor IAE), el 70% y el 90% del valor final [5], como se muestra en la figura 2.6.

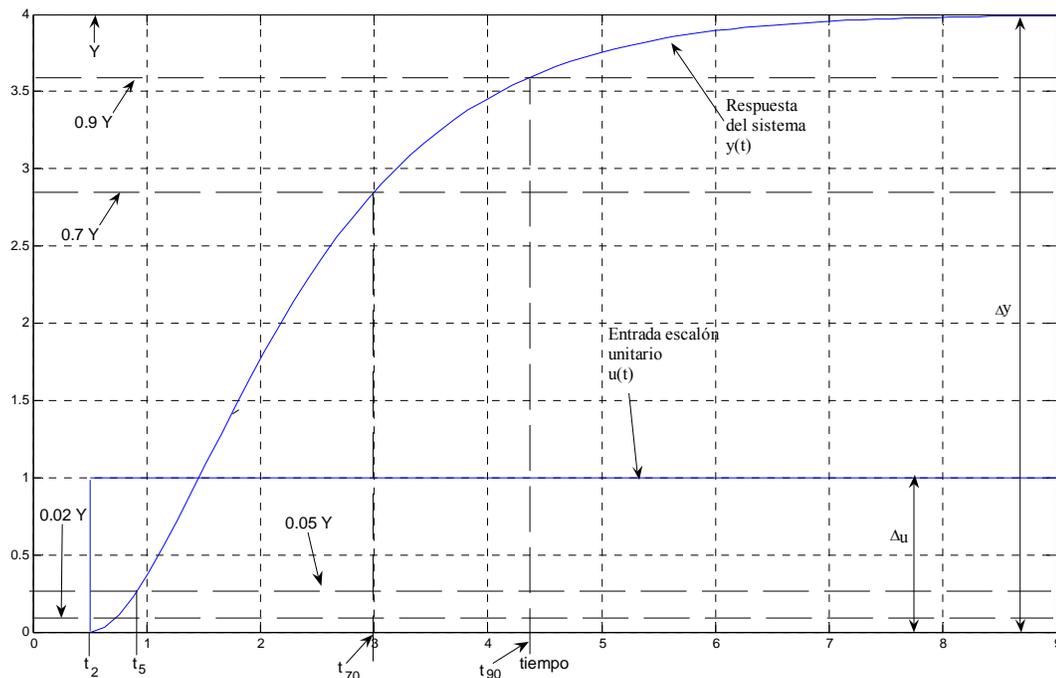


Figura 2.6. Ejemplo de Curva de Reacción Utilizada por Jahanmiri y Fallahi

De acuerdo con Jahanmiri y Fallahi, el modelo se calcula con las siguientes ecuaciones:

$$t_m = t_2 \text{ ó } t_5, \text{ el que resulte en una menor IAE} \quad (2.18)$$

$$\eta = \frac{t_{90} - t_{70}}{t_{90} - t_m} \quad (2.19)$$

$$\zeta = \sqrt{\frac{0,4844651 - 0,75323499\eta}{1 - 2,0946444\eta}}, \quad \eta \leq 0,4771 \quad (2.20)$$

$$\zeta = 13,9352, \quad \eta \geq 0,4771 \quad (2.21)$$

$$\tau = \frac{t_{90} - t_m}{0,424301 + 4,62533\zeta - 2,65412e^{-\zeta}} \quad (2.22)$$

La ganancia de estado estacionario se calcula mediante la ecuación 2.6.

2.1.7 Método de Mikles y Fikar

Para sistemas con comportamiento subamortiguado, como el mostrado en la figura 2.3, se puede identificar un modelo de segundo orden de la forma:

$$G(s) = \frac{Z\omega_0^2}{s^2 + 2\omega_0\zeta s + \omega_0^2} \quad (2.23)$$

Mikles y Fikar [6] determinaron que los parámetros del modelo se pueden calcular con las ecuaciones:

$$Z = y(\infty) \quad (2.24)$$

$$M = \frac{y_1 - y_2}{y_1} \quad (2.25)$$

$$\zeta = \left| \frac{\ln M}{\sqrt{\pi^2 + \ln^2 M}} \right| \quad (2.26)$$

$$\omega_0 = \frac{\pi}{(t_2 - t_1)\sqrt{1 - \zeta^2}} \quad (2.27)$$

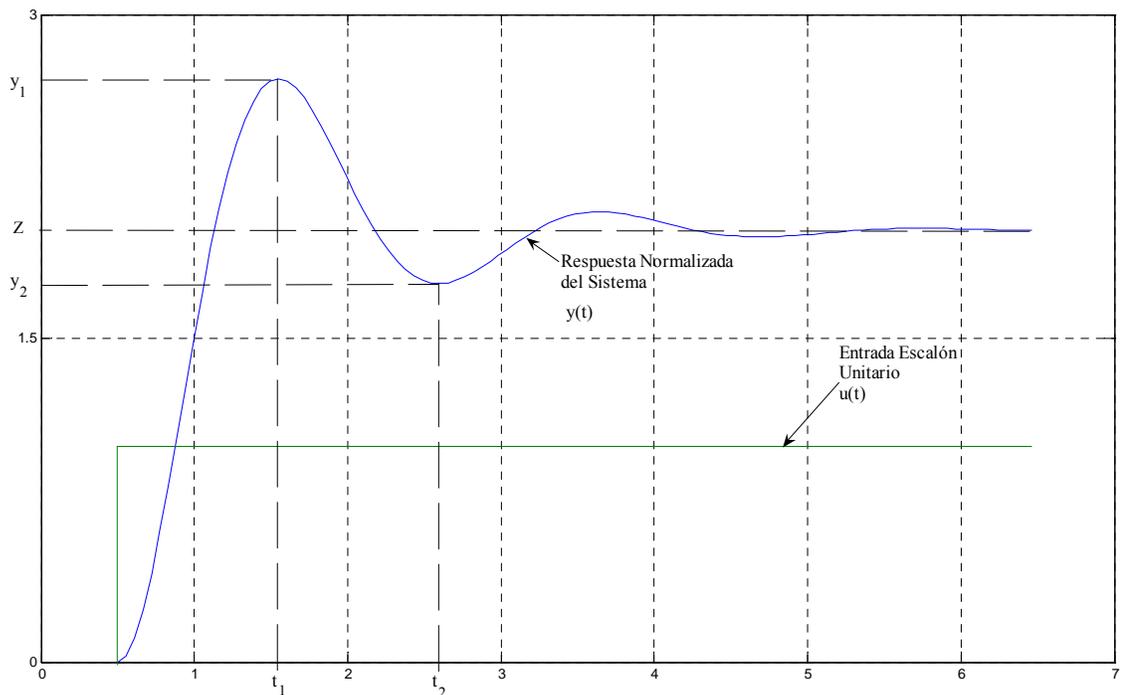


Figura 2.7. Ejemplo de Curva de Reacción de un Sistema de Segundo Orden Subamortiguado Utilizada por Mikles y Fikar

En el caso de Mikles y Fikar, la respuesta escalón del sistema se debe normalizar a una respuesta escalón unitario, para lo cual se divide entre la magnitud del escalón aplicado para generar la curva de reacción, es decir:

$$y(t) = y_0(t)/\Delta u(t) \quad (2.28)$$

donde $y_0(t)$ es la curva de reacción del sistema, $\Delta u(t)$ es la magnitud del escalón

aplicado, $y(t)$ es la curva de reacción normalizada. $y(\infty)$ es el valor final de la curva de reacción normalizada.

2.2 AJUSTE DE CONTROLADORES PID

Los métodos para el cálculo de los ajustes de un controlador Proporcional-Integral-Derivativo (PID) propuestos por Ziegler y Nichols en 1942 son los de uso más extendido debido a su sencillez, tanto el método de la ganancia última y período de oscilación (lazo cerrado) como el de la curva de reacción (lazo abierto). En ambos casos el controlador usado por Ziegler y Nichols es un controlador PID ideal, actuando como regulador, y los parámetros de ajuste se calcularon para una relación de decaimiento o razón de asentamiento de $\frac{1}{4}$.

Con el argumento de que Ziegler y Nichols no consideraron procesos autorregulados, Cohen y Coon propusieron su método de ajuste basados también en un controlador PID ideal, e introdujeron un índice de autorregulación definido como la relación entre el tiempo muerto y la constante de tiempo del modelo de primer orden más tiempo muerto del sistema a controlar [2].

En el año 2005, Alfaro [7] realizó una actualización de las ecuaciones desarrolladas por Ziegler y Nichols, para su utilización con los modelos de primer orden más tiempo muerto identificados a partir de la curva de reacción del proceso.

Sin embargo, los controladores PID ajustados por estos métodos, y que funcionan tanto como reguladores como servomecanismos, siempre presentan un sobrepaso que en algunos procesos no puede ser tolerado, como por ejemplo en la fabricación de guantes de plástico. En este proceso se requiere el posicionamiento de dos películas del material; si ocurre un sobre paso las películas de plástico se arrugan y se inutilizan [4].

2.2.1 Método de Ziegler y Nichols con Razón de Asentamiento $\frac{1}{4}$

Ziegler y Nichols establecieron su método basado en un modelo POMTM determinado por el método de la tangente, para una razón de asentamiento de $\frac{1}{4}$ y un lazo de control funcionando como regulador, y un controlador PID ideal [2]. Los parámetros de ajuste calculados por Ziegler y Nichols son:

$$K_c = 1,2 \frac{\tau}{k_p t_m} \quad \text{a} \quad K_c = 2,0 \frac{\tau}{k_p t_m} \quad (2.29)$$

$$T_i = 2t_m \quad (2.30)$$

$$T_d = 0,5t_m \quad (2.31)$$

2.2.2 Método de Cohen y Coon

Cohen y Coon desarrollaron su método para procesos autorregulados y basados en el mejor modelo POMTM que se pueda obtener para lazos de control que funcionen como reguladores con controladores PID ideales [2].

Las ecuaciones de ajuste desarrolladas por Cohen y Coon, en las que introdujeron el concepto de *índice de regulación* ($\mu = t_m/\tau$) son:

$$K_c = \frac{\tau}{k_p t_m} \left(\frac{4}{3} + \frac{t_m}{4\tau} \right) \quad (2.32)$$

$$T_i = t_m \left(\frac{32 + 6t_m / \tau}{13 + 8t_m / \tau} \right) \quad (2.33)$$

$$T_d = t_m \left(\frac{4}{11 + 2t_m / \tau} \right) \quad (2.34)$$

2.2.3 Métodos de Sung, O, Lee, Lee y Yi

El método de Sung *et al.* está basado en el mejor modelo de segundo orden más tiempo muerto que se pueda obtener para lazos de control reguladores, con criterio de desempeño correspondiente a la minimización de la ITAE y un controlador PID-Ideal [2]. El ajuste de las ecuaciones se realizó para $0,05 \leq t/\tau \leq 2,0$.

Sung *et al.* desarrollaron ecuaciones tanto para lazos reguladores como para servomecanismos. Las ecuaciones de sintonización son:

Reguladores:

$$K_c = \frac{1}{k_p} \left[-0,67 + 0,297 \left(\frac{t_m}{\tau} \right)^{-2,001} + 2,189 \zeta \left(\frac{t_m}{\tau} \right)^{-0,766} \right], \quad \frac{t_m}{\tau} \leq 0,9 \quad (2.39)$$

$$K_c = \frac{1}{k_p} \left[-0,365 + 0,260 \left(\frac{t_m}{\tau} - 1,4 \right)^{-2,0} + 2,189 \zeta \left(\frac{t_m}{\tau} \right)^{-0,766} \right], \quad \frac{t_m}{\tau} \geq 0,9 \quad (2.40)$$

$$T_i = \tau \left[2,212 \left(\frac{t_m}{\tau} \right)^{0,520} - 0,3 \right], \quad \frac{t_m}{\tau} < 0,4 \quad (2.41)$$

$$\frac{T_i}{\tau} = -90,975 + 0,91 \left(\frac{t_m}{\tau} - 1,845 \right)^2 + \left[1 - e^{-\frac{\zeta}{0,15+0,33(t_m/\tau)}} \right] \left[5,25 - 0,88 \left(\frac{t_m}{\tau} - 2,8 \right)^2 \right] \quad (2.42)$$

$$, \quad \frac{t_m}{\tau} \geq 0,4$$

$$\frac{\tau}{T_d} = -1,9 + 1,576 \left(\frac{t_m}{\tau} \right)^{-0,530} + \left[1 - e^{\frac{-\zeta}{0,15+0,939(t_m/\tau)^{-1,121}}} \right] \left[1,45 - 0,969 \left(\frac{t_m}{\tau} \right)^{-1,171} \right] \quad (2.43)$$

Servomecanismos:

$$K_c k_p = -0,04 + \left[0,333 + 0,949 \left(\frac{t_m}{\tau} \right)^{-0,983} \right] \zeta, \quad \zeta \leq 0,9 \quad (2.44)$$

$$K_c k_p = -0,544 + 0,308 \left(\frac{t_m}{\tau} \right) + 1,408 \zeta \left(\frac{t_m}{\tau} \right)^{-0,832}, \quad \zeta \geq 0,9 \quad (2.45)$$

$$T_i = \tau \zeta \left[2,055 + 0,072 \left(\frac{t_m}{\tau} \right) \right], \quad \frac{t_m}{\tau} \leq 1 \quad (2.46)$$

$$T_i = \tau \zeta \left[1,768 + 0,329 \left(\frac{t_m}{\tau} \right) \right], \quad \frac{t_m}{\tau} > 1 \quad (2.47)$$

$$\frac{\tau}{T_d} = \left[1 - e^{\frac{(t_m/\tau)^{-1,060} \zeta}{0,870}} \right] \left[0,55 - 1,683 \left(\frac{t_m}{\tau} \right)^{-1,090} \right] \quad (2.48)$$

2.2.4 Método de Rovira, Murrill y Smith

Rovira, Murrill y Smith se basaron en criterios integrales para presentar en 1969 el primer método de ajuste de controladores PID actuando como servomecanismos [3]. Rovira *et al.* utilizaron para el ajuste dos criterios de desempeño distintos: Integral del error absoluto (IAE) e Integral del error absoluto por el tiempo (ITAE), y se basaron en el mejor modelo POMTM para lazos que funcionen como servomecanismos con controladores PID ideales. Al igual que en el método de López, el método de Rovira se basa en la minimización de una función de costo, $\Phi = \int_0^{\infty} F[e(t), t] dt$, cuya minimización conduce a un desempeño óptimo del sistema.

Las ecuaciones de ajuste desarrolladas por Rovira *et al.* son:

$$K_c = \frac{a}{k_p} \left(\frac{t_m}{\tau} \right)^b \quad (2.49)$$

$$T_i = \frac{\tau}{c + d(t_m / \tau)} \quad (2.50)$$

$$T_d = e\tau \left(\frac{t_m}{\tau} \right)^f \quad (2.51)$$

Los valores de las constantes a , b , c , d , e y f dependen del criterio integral utilizado y se muestran en la tabla 2.1.

Tabla 2.1. Constantes para las ecuaciones del método de Rovira *et al.*

| Criterio de Desempeño | a | b | c | d | e | F |
|-----------------------|----------|----------|----------|----------|----------|----------|
| IAE | 1,086 | -0,869 | 0,740 | -0,130 | 0,348 | 0,914 |
| ITAE | 0,965 | -0,850 | 0,796 | -0,1465 | 0,308 | 0,929 |

El método de Rovira *et al.* está limitado, sin embargo, por la relación t_m/τ . Para el caso del criterio de desempeño IAE, debe cumplirse que $\frac{t_m}{\tau} < 5,6923$, mientras que para utilizar el criterio ITAE debe cumplirse que $\frac{t_m}{\tau} < 5,249$.

2.2.5 Método de Rivera, Morari y Stogstad

Rivera *et al.* basaron sus cálculos en controladores PID-IMC, es decir,

controladores PID cuyos parámetros se calculan utilizando técnicas de Control con Modelo Interno (*Internal Model Control* o IMC). El concepto IMC incorpora dentro del controlador un modelo del proceso y fue establecido por García *et al.* en 1985, primero para sistemas de una entrada y una salida (SISO) y luego para sistemas multivariados (MIMO) [3]. La figura 2.4 muestra la estructura básica de un controlador IMC.

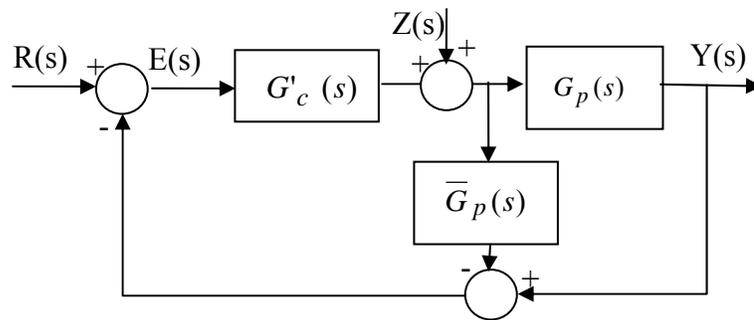


Figura 2.8. Estructura básica de un controlador IMC

La ecuación del controlador $G'_c(s)$, equivalente al controlador de la figura 2.1, está dada por:

$$G'_c(s) = \frac{G'_c(s)}{1 - G'_c(s)\bar{G}_p(s)} \quad (2.52)$$

Rivera, Morari y Stogestad redefinieron el controlador IMC como:

$$G'_c(s) = \bar{G}_p^{-1}(s)F(s) \quad (2.53)$$

$F(s)$ es un filtro paso bajo, el cual debe ser seleccionarse de manera tal para garantizar que la función de transferencia del controlador IMC sea propia. El filtro es de la forma:

$$F(s) = \frac{1}{(\lambda s + 1)^n} \quad (2.54)$$

Rivera *et al.*, en 1985, desarrollaron un procedimiento para el cálculo de controladores IMC que permitan lograr un desempeño determinado. También demostraron que para modelos simples los controladores IMC calculados conducen a controladores tipo PID. Las ecuaciones de ajuste desarrolladas Rivera *et al.*, para un controlador PID ideal y modelos sin tiempo muerto se muestran en la tabla 2.2:

Tabla 2.2. Parámetros PID-IMC de Rivera et al.

| Modelo $\bar{G}_p(s)$ | $\frac{y(s)}{r(s)}$ | Controlador $G_c(s)$ | Parámetros | | |
|--|---------------------------|--|----------------------------------|--------------|-----------------------|
| | | | K_c | T_i | T_d |
| $\frac{k_p}{\tau^2 s^2 + 2\delta s + 1}$ | $\frac{1}{\lambda s + 1}$ | $\frac{\tau^2 s^2 + 2\delta s + 1}{\lambda k_p s}$ | $\frac{2\zeta\tau}{\lambda k_p}$ | $2\zeta\tau$ | $\frac{2\zeta}{\tau}$ |

2.3 LÓGICA DIFUSA

Se considera que el concepto de lógica difusa fue introducido por Lotfi A. Zadeh en la Universidad de California en Berkeley, en 1965 [8].

La lógica difusa es técnica de inteligencia artificial que se basa en el concepto de raciocinio aproximado y en la capacidad de extraer conclusiones y generar respuestas usando información vaga, ambigua, incompleta, cualitativa y/o imprecisa. Es una extensión de la lógica clásica que permite asociar incertidumbre a la estructura de un conjunto de datos. Mientras la lógica clásica trata solamente con valores de *falso* o *verdadero*, la lógica difusa trata de emular una parte del razonamiento humano que trata con valores de verdad no necesariamente tan deterministas. Por

ejemplo, se puede decir que una persona es “alta” o “baja” sin que esto signifique que exista una frontera exacta entre las dos categorías. Una premisa basada en la lógica tradicional posee solamente dos valores de verdad, verdadera o falsa, en lógica difusa una premisa puede tener valores de verdad que van desde el completamente falso (0%) hasta el completamente verdadero (100%).

2.3.1 Conjuntos Difusos

Se entiende por un conjunto a una colección definida de elementos, de tal manera que siempre es posible determinar si un objeto cualquiera en un universo dado pertenece o no pertenece al conjunto. Los conjuntos clásicos (con valores específicos o *crisp*) surgen de la necesidad del ser humano de clasificar todas las cosas a su alrededor. En estos conjuntos un elemento simplemente *pertenece* o *no pertenece*.

En los conjuntos difusos, a cada elemento se le asocia un *grado de pertenencia* al conjunto. Este grado de pertenencia es un número entre 0 y 1; mientras más cerca de 1 esté, más pertenece el elemento al conjunto.

Los conjuntos difusos pueden representarse como pares ordenados que indican el valor del elemento y el grado en el que pertenece al conjunto. También se pueden representar mediante *funciones de pertenencia*, especialmente cuando el *universo de discurso* es continuo.

2.3.2 Funciones de Pertenencia

Como ya se dijo, un conjunto difuso F en un universo U puede representarse también como una función. La función que define la pertenencia o no de un objeto a un conjunto se denomina *Función de Pertenencia*. Por ejemplo, en el universo X de temperaturas, se puede definir la función de pertenencia Alta Temperatura como se

muestra en la figura 2.9.

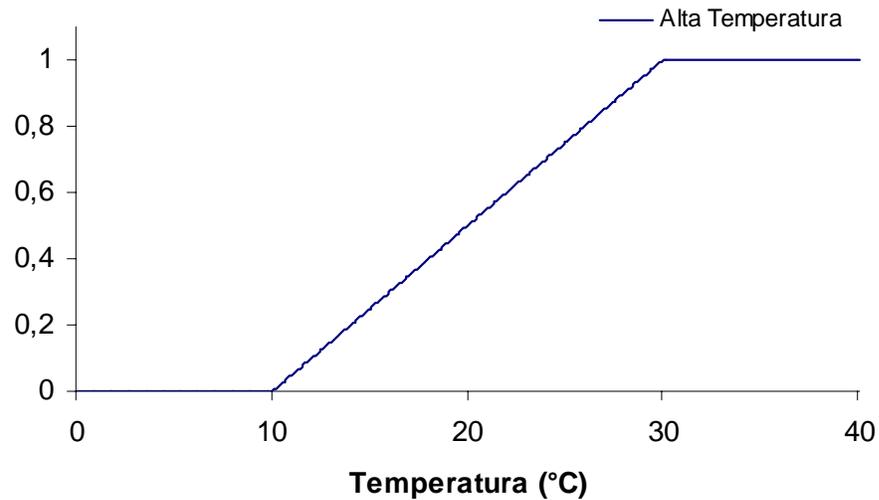


Figura 2.9. Representación Gráfica de una Función de Pertenencia

Las funciones de pertenencia más utilizadas son:

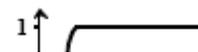
Función Triangular: Definida por sus límites inferior a y superior b , y el valor modal m , tal que $a < m < b$.

$$\mu(x) = \begin{cases} 0 & \text{si } x \leq a \\ (x-a)/(m-a) & \text{si } a < x \leq m \\ (b-x)/(b-m) & \text{si } m < x < b \\ 0 & \text{si } x \geq b \end{cases}$$



Función Gamma (Γ): Definido por sus límites inferior a y superior b , y el valor modal m , tal que $a < m < b$.

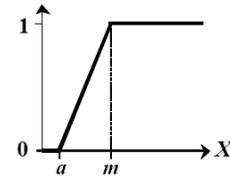
$$\begin{cases} 0 & \text{si } x \leq a \end{cases}$$



$$1 - e^{-k(x-a)^2} \quad \text{si } x > a$$

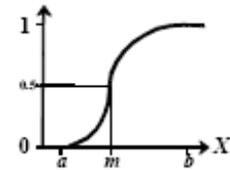
Esta función se caracteriza por un rápido crecimiento a partir de a . Cuanto mayor es el valor de k , el crecimiento es más rápido aún. Nunca toma el valor 1, aunque tienen una asíntota horizontal en 1. Se aproxima linealmente por:

$$\mu(x) = \begin{cases} 0 & \text{si } x \leq a \\ (x-a)/(m-a) & \text{si } a < x < m \\ 1 & \text{si } x \geq m \end{cases}$$



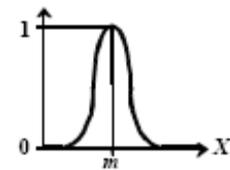
Función S: Definida por sus límites inferior a y superior b , y el valor m , o punto de inflexión tal que $a < m < b$. Un valor típico es: $m = (a+b)/2$. El crecimiento es más lento cuanto mayor sea la distancia $a - b$.

$$\mu(x) = \begin{cases} 0 & \text{si } x \leq a \\ 2[(x-a)/(b-a)]^2 & \text{si } a < x \leq m \\ 1 - 2[(x-b)/(b-a)]^2 & \text{si } m < x < b \\ 1 & \text{si } x \geq b \end{cases}$$



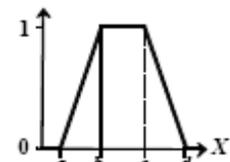
Función Gaussiana: Definida por su valor medio m y el valor $k > 0$. Es la típica campana de Gauss. Cuanto mayor es k , más estrecha es la campana.

$$\mu(x) = e^{-k(x-m)^2}$$



Función Trapezoidal: Definida por sus límites inferior a y superior d , y los límites de su soporte, b y c , inferior y superior respectivamente.

$$\mu(x) = \begin{cases} 0 & \text{si } (x \leq a) \text{ ó } (x \geq d) \\ (x-a)/(b-a) & \text{si } a < x \leq b \\ 1 & \text{si } b < x < c \end{cases}$$



$$(d-x)/(d-c) \quad \text{si } c < x < d$$

2.3.3 Soporte de un Conjunto Difuso en un Universo de Discurso U

Es el conjunto de elementos $u \in U$, tales que $\mu_F(u) > 0$.

2.3.4 Centro de un Conjunto Difuso

Es el conjunto de elementos $u \in U$, en los que $\mu_F(u)$ alcanza su máximo valor.

2.3.5 Variable Lingüística

Es una variable cuyos posibles valores son palabras y pueden ser representados mediante conjuntos difusos.

Por ejemplo:

La variable lingüística “velocidad”: Admite valores lingüísticos: lenta, moderada y rápida. Admite valores numéricos: números reales en $[0, V_{\max}]$. Se pueden proyectar los valores lingüísticos sobre el intervalo $[0, V_{\max}]$ mediante funciones de pertenencia.

Una variable lingüística admite que sus valores sean *Etiquetas Lingüísticas*, que son términos lingüísticos definidos como conjuntos difusos (sobre cierto dominio subyacente). Ejemplos: Temperatura “Cálida”, o “aproximadamente 25°C”. El dominio subyacente es un dominio numérico: Los grados centígrados.

Un valor concreto (25°C, por ejemplo) es, en general, más específico que una etiqueta lingüística; es un punto del conjunto, mientras que una etiqueta lingüística es una colección de puntos (temperaturas posibles).

Las Variables Lingüísticas son de gran utilidad pues:

- Son una forma de comprimir información llamada *granulación* (*granulation*): Una etiqueta incluye muchos valores posibles.
- Ayudan a caracterizar fenómenos que o están mal definidos o son complejos de definir o ambas cosas.
- Son un medio de trasladar conceptos o descripciones lingüísticas a descripciones numéricas que pueden ser tratadas automáticamente: Relaciona o traduce el proceso simbólico a proceso numérico.

2.3.6 Estructura Básica de un Sistema Difuso

La estructura básica de un sistema difuso se muestra en la figura 2.10.

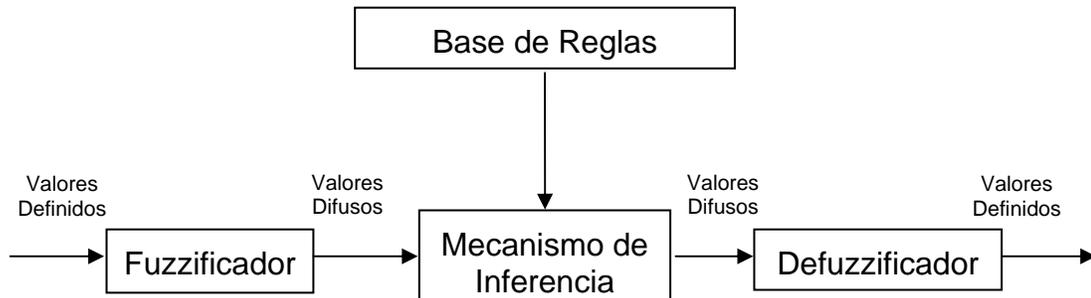


Figura 2.10. Estructura Básica de un Sistema Difuso

Un sistema difuso está formado básicamente por una base de reglas difusas (que contienen el conocimiento de los expertos consultados para desarrollar el sistema) y un mecanismo de inferencia que aplica estas reglas a los valores difusos de entrada para generar los valores difusos de salida.

En aplicaciones de Ingeniería, los valores utilizados como entrada y salida de los sistemas difusos son valores definidos o *crisp*, por lo que se hace necesario incluir mecanismos para convertir los valores de entrada a valores difusos y convertir los

valores de salida a valores definidos. Esto se hace con ayuda del *fuzzificador* y el *defuzzificador*, respectivamente.

2.3.7 Difusificación

El proceso de difusificación consiste en establecer una relación entre los valores numéricos de una variable y sus correspondientes conjuntos difusos. El método de difusificación más utilizado en sistemas de control es el *Fuzzificador Singleton*, en el cual se considera a los propios valores discretos como conjuntos difusos. En otras palabras, se define un conjunto difuso $A(x')$, para el cual $\mu_A(x') = 1$ si $x' = x$, y $\mu_A(x') = 0$ si $x' \neq x$.

2.3.8 Base de Reglas Difusas

Las reglas difusas permiten expresar el conocimiento que se tiene sobre la operación de un proceso. Una regla difusa consiste en la combinación de uno o varios conjuntos difusos de entrada (antecedentes o premisas) y la asociación con un conjunto difuso de salida (consecuente o consecuencia). Las reglas generalmente tienen la forma IF *antecedente* THEN *consecuente*, y la combinación entre los conjuntos difusos de los antecedentes se realiza mediante operadores lógicos (AND, OR, etc.).

Una base de reglas es la agrupación de varias reglas que expresan el conocimiento sobre determinado proceso, y se puede representar mediante una tabla de reglas o mediante una *Memoria Asociativa Difusa* o *FAM* (Fuzzy Associative Memory); ésta última no es más que una tabla o matriz en la cual se representa el consecuente de cada regla definida para la combinación de dos entradas.

Una base de reglas difusas se define formalmente como una colección de reglas R_i , donde cada regla está compuesta de uno o varios antecedentes y un consecuente. Cuando existen varios antecedentes, éstos se combinan mediante operadores *fuzzy*,

como AND, OR y NOT, y tienen el formato:

$$R_i = \text{IF } x_1 \text{ es } F_{1i} \text{ OPE } x_2 \text{ es } F_{2i} \text{ OPE } \dots \text{ OPE } x_n \text{ es } F_{ni} \text{ THEN } y \text{ es } G_i$$

En este formato, F_{ji} y G_{ji} son conjuntos difusos y x e y son variables lingüísticas, y *OPE* es cualquiera de los operadores difusos. A este formato se le conoce como *Formato Difuso Puro o de Tipo Mamdani*. Otro formato utilizado es el *Formato de Tipo Sugeno*, en cuyo caso la salida y es una función de las variables de entrada.

2.3.9 Inferencia Difusa

Por inferencia difusa se entiende la interpretación de las reglas contenidas en una base de reglas con el fin de obtener valores de salida a partir de los valores de las variables lingüísticas de entrada. El resultado de cada regla depende de la definición de los operadores empleados en éstas. Entre las definiciones más utilizadas se encuentran:

- AND: mínimo valor entre dos funciones de pertenencia:

$$\mu(A \text{ AND } B) = \min(\mu(A), \mu(B))$$

- OR : máximo valor entre dos funciones de pertenencia:

$$\mu(A \text{ OR } B) = \max(\mu(A), \mu(B))$$

- NOT: complemento a 1 de la función de pertenencia afectada.

Las reglas pueden ser evaluadas en paralelo, mediante *hardware*, o secuencialmente, mediante *software*. Los resultados obtenidos de la evaluación de las reglas activadas se utilizan para calcular el resultado global, mediante un operador que generalmente es el operador de agregación OR. Posteriormente, el resultado

global se debe transformar a un valor definido (*crisp*) mediante la desfusificación.

Durante el proceso de evaluación de reglas, los valores son computados con base en los niveles de activación alcanzados o grados de cumplimiento de cada una de las funciones de pertenencia definidas para cada una de las entradas, para luego asociarlos a conjuntos difusos de salida.

Cuando se activa más de una función de pertenencia, éstas se deben procesar mediante un operador. El operador más utilizado es el *operador Mamdani* (\wedge), el cual utiliza la función *min*, en lo que se conoce como *implicación por la regla del mínimo*. En ésta el grado de cumplimiento de la *i-ésima* regla está dado por el mínimo grado de cumplimiento de los antecedentes de la misma. Por ejemplo, sea la regla:

$$R_i = \text{IF } x_1 \text{ es } F_{1i} \text{ AND } x_2 \text{ es } F_{2i} \text{ AND } \dots \text{ AND } x_n \text{ es } F_{ni} \text{ THEN } y \text{ es } G_i$$

La implicación por la regla del mínimo establece que el grado de cumplimiento del consecuente G_i viene dado por:

$$\mu_i(x_1, x_2, \dots, x_n) = \min(\mu_{F_1}(x_1), \mu_{F_2}(x_2), \dots, \mu_{F_n}(x_n)) \quad (2.55)$$

En la práctica, la implicación por la regla del mínimo “recorta” la función de pertenencia del consecuente al mínimo valor entre los grados de cumplimiento de las funciones de pertenencia del antecedente.

Otra forma de implicación es la *implicación por la regla del producto*, la cual utiliza el *operador Larsen* (\bullet) u *operador producto*. En esta forma de implicación, el grado de cumplimiento de la *i-ésima* regla está dado por el producto de los grados de cumplimiento de los antecedentes de la misma.

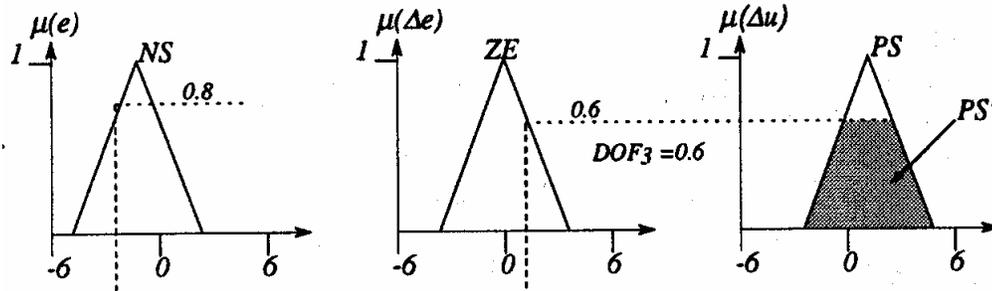


Figura 2.11. Implicación por la Regla del Mínimo

La implicación por la regla del producto “escala” la función de pertenencia del consecuente al producto de los grados de cumplimiento de las funciones de pertenencia del antecedente.

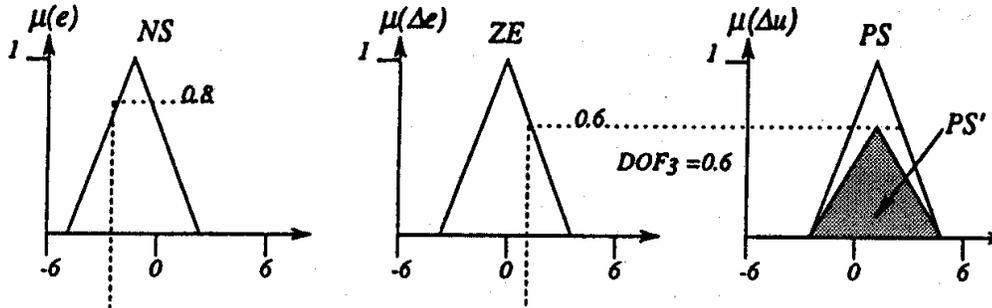


Figura 2.12. Implicación por la Regla del Producto

El conjunto difuso de salida, G_o , es la combinación de los conjuntos difusos G_i activados en la aplicación de cada regla de la base de reglas. La combinación se realiza generalmente mediante la suma o agregación de dichos conjuntos difusos, en lo que se conoce como *agregación por máximo*. La combinación de implicación por la regla del mínimo y la agregación por el máximo se conoce como *mecanismo de inferencia min-max o max-min*, y es el más utilizado en sistemas de tipo Mamdani

2.3.10 Desdifusificación

La desdifusificación tiene por objeto transformar este conjunto difuso en un valor no difuso. Para esto existen diversos métodos, tales como *desdifusificación por centro de área* y *desdifusificación por media de centros*.

2.3.10.1 Desdifusificación por Centro de Áreas

En la desdifusificación por centro de áreas, también conocida como centroide o centro de masas, la salida y es el centroide del área del conjunto difuso G_o , y se obtiene como:

$$y = \frac{\sum_{i=1}^N u_i \cdot G_o(u_i)}{\sum_{i=1}^N \mu_o(u_i)} \quad (2.56)$$

N es el número de valores discretos u_i del universo de discurso de la variable de salida, $G_o(u)$. Si $G_o(u)$ es continuo, entonces las sumatorias se transforman en integrales, con lo que se obtiene que:

$$y = \frac{\int u \cdot G_o(u) du}{\int G_o(u) du} \quad (2.57)$$

Para utilizar la desdifusificación por centro de áreas se debe primero calcular los conjuntos difusos G_i , obtenidos al aplicar cada regla de la base de reglas, y después calcular el centro de gravedad del conjunto difuso de salida resultante, G_o .

Una variante de este método supone la desdifusificación de cada conjunto difuso G_i y la posterior combinación mediante una *suma-producto* de la forma:

$$y = \frac{\sum_{i=1}^M \mu_i \cdot G'_i}{\sum_{i=1}^M \mu_i} \quad (2.58)$$

G'_i es el valor defuzzificado del conjunto difuso G_i , activado al aplicar la i -ésima regla y μ_i es el grado de activación de la regla. G'_i es el área de G_i acotada por el resultado de la implicación de dicha i -ésima regla, pero también puede ser el máximo o el centro, dependiendo de la carga computacional que se desee. En todo caso, esta forma de desfuzzificación presenta una carga computacional menor que el método del centroide.

2.3.10.2 Desfuzzificación por Media de Centros

La variable de salida y se calcula como:

$$y = \frac{\sum_{i=1}^M \mu_{oi}(x_1, x_2, \dots, x_n) \cdot \overline{G}_i(x_1, x_2, \dots, x_n)}{\sum_{i=1}^M \mu_{oi}(x_1, x_2, \dots, x_n)} \quad (2.59)$$

\overline{G}_i es el centro del conjunto difuso G_i , que generalmente es el máximo valor, activado al aplicar la i -ésima regla.

La desfuzzificación por media de centros imponen mucha menos carga computacional que la desfuzzificación por centro de áreas, puesto que no requiere integración numérica ni la agregación de los conjuntos difusos de salida, pues los valores a utilizar en la desfuzzificación se calculan para cada conjunto difuso.

CAPÍTULO 3. DESARROLLO DEL PROGRAMA PARA EL CÁLCULO Y AJUSTE DE CONTROLADORES PID

El diagrama de flujo del proceso propuesto para el Cálculo y Ajuste de Controladores PID (CYRCPID) se muestra en la figura 3.1. Las secciones 3.1 a 3.4 describen cada una de los bloques funcionales en los que se divide el proceso.

En la sección 3.1 se describe el proceso de definición del sistema a controlar. Tal definición puede realizarse bien sea mediante una función de transferencia o mediante un archivo de datos con los resultados de una prueba escalón (curva de reacción).

El sistema definido debe ser posteriormente modelado, lo cual se describe en la sección de 3.2, *Modelado del Sistema*. En esta sección se calcula un modelo del sistema previamente definido. El método a utilizar para calcular el modelo se puede seleccionar de entre los varios métodos disponibles.

Una vez calculado el modelo, se calculan los parámetros del controlador PID utilizando los métodos clásicos mencionados en la sección 3.3, *Cálculo de Parámetros del Controlador*. Con el controlador calculado se valúa el desempeño estimado funcionando como regulador o como servomecanismo.

La sección 3.4, *Ajuste de Parámetros del Controlador*, describe el proceso de Ajuste de la constante proporcional del controlador calculado. Se describe el algoritmo y el sistema difuso implementado.

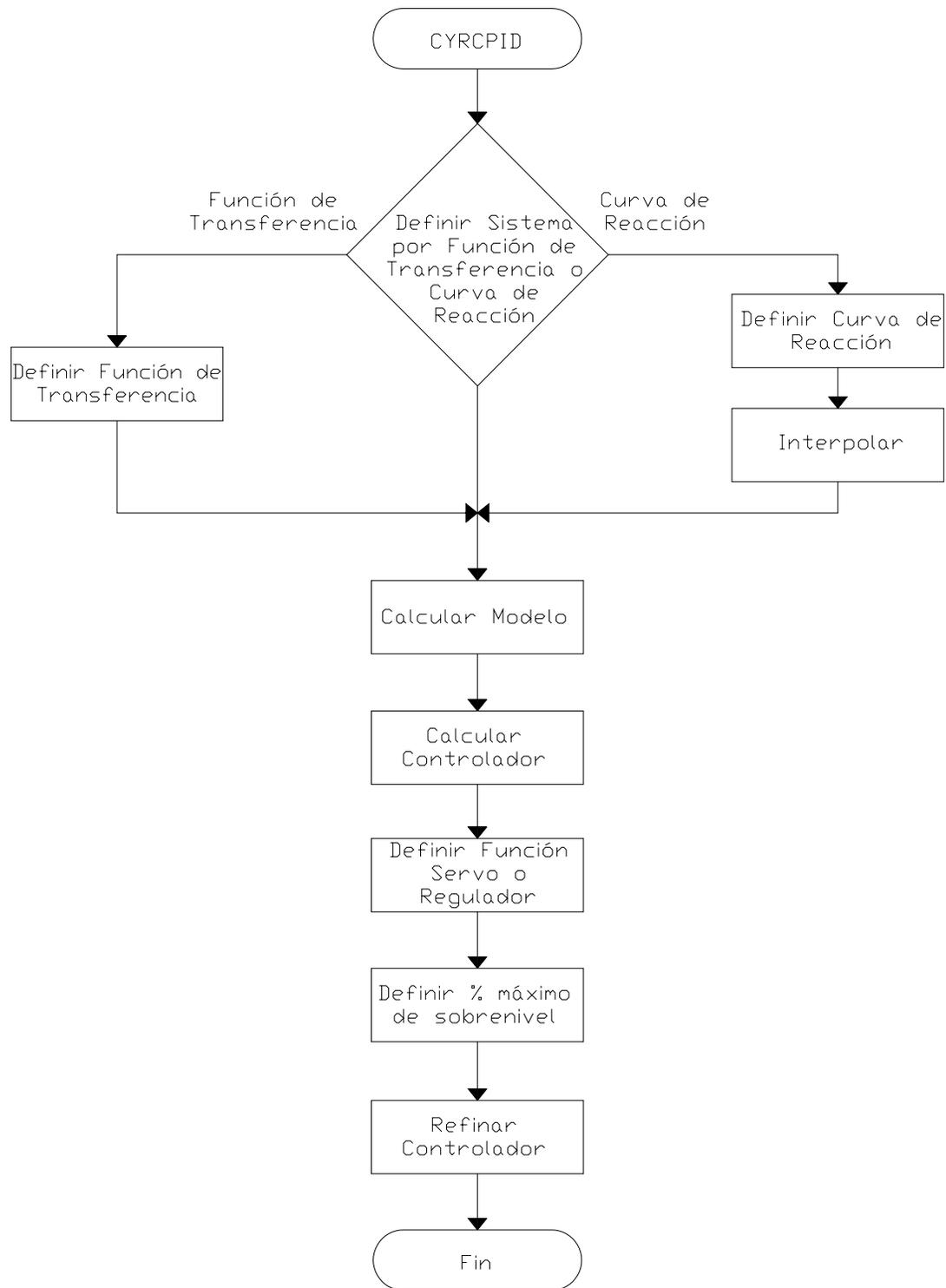


Figura 3.1. Diagrama de Flujo del Sistema Propuesto

3.1 DEFINICIÓN DEL SISTEMA

Como ya se mencionó, el sistema a controlar se puede definir mediante su función de transferencia o mediante los resultados obtenidos al realizar una prueba escalón. Cuando se define mediante una función de transferencia, posterior a la definición de ésta se realiza una prueba escalón cuyo objeto es obtener los valores de t y $f(t)$ necesarios para el posterior cálculo del modelo. Sin embargo, cuando el sistema se define mediante los resultados de una prueba escalón se realiza una interpolación cúbica (*cubic spline*) con el objeto de obtener más puntos y tratar de aumentar la precisión del modelo. El algoritmo utilizado es el siguiente:

En caso de que la definición del sistema sea mediante:

1. Función de transferencia:
 - Definir la función de transferencia.
 - Realizar prueba escalón para obtener t , $f(t)$, valor final, tiempo de estabilización y sobrenivel.
2. Resultados de prueba escalón:
 - Leer resultados desde archivo.
 - Realizar interpolación cúbica.
 - Obtener valor final, tiempo de estabilización y sobrenivel.

En el capítulo 4, sección 4.1, se muestra el código correspondiente al algoritmo anterior.

3.2 MODELADO DEL SISTEMA

Para el modelado del sistema se decidió seleccionar entre modelos de primer orden más tiempo muerto, de segundo orden sobreamortiguado más tiempo muerto y un modelo de segundo orden subamortiguado, sin tiempo muerto.

Para calcular los modelos de primer orden más tiempo muerto y segundo orden más tiempo muerto se seleccionaron los métodos de Ho *et al.* Los modelos calculados por estos métodos presentaron menores errores de predicción, de acuerdo con Alfaro en [5]. Para el modelo de segundo orden subamortiguado se utilizó el método de

Mikles y Fikar. Este método, utiliza el máximo sobrenivel y el mínimo valor siguiente para calcular los parámetros del modelo, por lo que previo a su utilización se requiere verificar que la respuesta del sistema presenta características de un sistema subamortiguado, es decir, si presenta sobreniveles. Dadas las limitaciones de los modelos de Ho *et al.*, descritas en las secciones 2.1.4 y 2.1.5, se incluyeron también el cálculo de un modelo POMTM según Smith y SOMTM según Jahanmiri y Fallahi.

También es posible dejar que el programa haga el cálculo del modelo por los distintos métodos y seleccione el que mejor se ajusta a los datos reales. La selección se realiza mediante la evaluación de la Integral del Valor Absoluto del Error de Predicción (IAEP). Si y_0 es el vector con los valores de la respuesta de la planta, y_1 el vector con los valores de la respuesta obtenidos del modelo que se está evaluando y t el tiempo de simulación, entonces la IAEP se calcula como:

$$IAEP = \int_0^{\infty} (y_1 - y_0) dt \quad (3.1)$$

Se calculan los valores predichos por cada modelo calculado y se evalúa la IAEP de cada uno, seleccionando aquél cuya IAEP sea menor.

El algoritmo completo utilizado es el siguiente:

En caso de que el modelado del sistema sea mediante:

1. Ho et al de primer orden más tiempo muerto:
Calcular modelo.
2. Ho et al de segundo orden más tiempo muerto:
Calcular modelo.
3. Mikles y Fikar:
Si existe sobrenivel, Calcular modelo
Caso contrario generar mensaje de error
4. Selección automática:

```

Calcular modelos de primer orden más tiempo muerto.
Calcular modelos de segundo orden más tiempo muerto
Si existe sobrenivel, Calcular modelo subamortiguado
    Caso contrario generar mensaje de error
Calcular respuesta escalón para cada modelo calculado.
Calcular IEAP
Comparar IEAP y seleccionar el menor.

```

Posterior al cálculo del modelo se genera una gráfica donde se muestran los valores predichos por el modelo seleccionado versus los datos originales del sistema, bien sea obtenidos de la función de transferencia o aquéllos que se usaron para definirlo mediante la curva de reacción. En el capítulo 4, sección 4.2, se muestra el código correspondiente al algoritmo anterior.

3.3 CÁLCULO DE PARÁMETROS DEL CONTROLADOR

Los parámetros controlador PID se pueden calcular mediante uno de los siguientes métodos:

1. Rovira *et al.* IAE (1969)
2. Sung *et al.* (1996)
3. Rivera *et al.* (1985)
4. Ziegler y Nichols
5. Cohen y Coon.

Una vez calculados los parámetros del controlador y su función de transferencia, se calcula la función de transferencia en lazo de cerrado del sistema, con el controlador operando como regulador y como servomecanismo. Posteriormente se simulan las respuestas a un escalón unitario y se grafican junto con la respuesta del sistema original. El algoritmo completo utilizado es el siguiente:

1. Seleccionar método de cálculo para los parámetros del controlador PID
2. Calcular parámetros del controlador
3. Calcular función de transferencia de lazo cerrado, en modos

regulador y servomecanismo.

4. Calcular respuestas a un escalón funcionando como regulador y como servomecanismo, y generar gráfica de respuestas del sistema controlado versus el sistema original.

El código correspondiente se muestra en el capítulo 4, sección 4.3.

3.3.1 Modos de Operación: Servo y Regulador

Cuando el sistema en lazo cerrado de la figura 2.1 opera en modo Servo, la salida debe responder a cambios en el punto de control o *Set Point*. En este caso, la función de transferencia en lazo cerrado es:

$$Gp_{LC}(s) = \frac{Gc(s) \cdot Gp(s)}{1 + Gc(s) \cdot Gp(s)} \quad (3.2)$$

Cuando se produce una perturbación interna en la planta, la salida de ésta debe mantenerse en el punto de control establecido. En este caso se habla de funcionamiento en modo Regulador, y la función de transferencia en lazo cerrado sería:

$$Gp_{LC}(s) = \frac{Gp(s)}{1 + Gc(s) \cdot Gp(s)} \quad (3.3)$$

3.4 AJUSTE DE PARÁMETROS DEL CONTROLADOR

3.4.1 Algoritmo de Ajuste

Para la Ajuste del controlador PID calculado en la sección 3.3, simplemente se calcula la diferencia relativa porcentual entre el máximo sobrenivel deseado por el usuario y el obtenido al simular la respuesta a un escalón unitario del sistema controlado. Esta diferencia se utiliza como entrada a un sistema difuso que genera un factor de corrección para aplicar a la ganancia proporcional del controlador y

modificarla de tal forma que el sobrenivel obtenido al simular la respuesta a un escalón del sistema controlado se “acerque” ala respuesta deseada.

Las operaciones anteriormente descritas se repiten hasta que el sobrenivel obtenido en la simulación sea igual o menor que el máximo sobrenivel deseado.

El algoritmo completo es el siguiente:

1. Calcular diferencia porcentual entre sobrenivel deseado y el obtenido.
2. Aplicar sistema difuso para calcular factor de corrección.
3. Aplicar factor de corrección a ganancia proporcional del controlador.
4. Calcular nueva función de transferencia del controlador.
5. Simular respuesta del sistema controlado a un escalón unitario.
6. Calcular la nueva diferencia porcentual entre sobrenivel deseado y el obtenido.
7. Si la nueva diferencia porcentual es mayor que la tolerancia, repetir desde el paso 1.
8. Calcular respuestas a un escalón unitario en modo regulador y servomecanismo, y generar gráfica de respuestas de sistema controlado versus sistema original.

Posteriormente se grafican las respuestas del sistema controlado a un cambio en escalón tanto en el punto de control como producto de una perturbación.

3.4.2 Sistema Difuso

Para el cálculo del factor de corrección a aplicar en la ganancia proporcional del controlador se utiliza un sistema difuso tipo Mamdani, con una estructura como la mostrada en la figura 3.1.

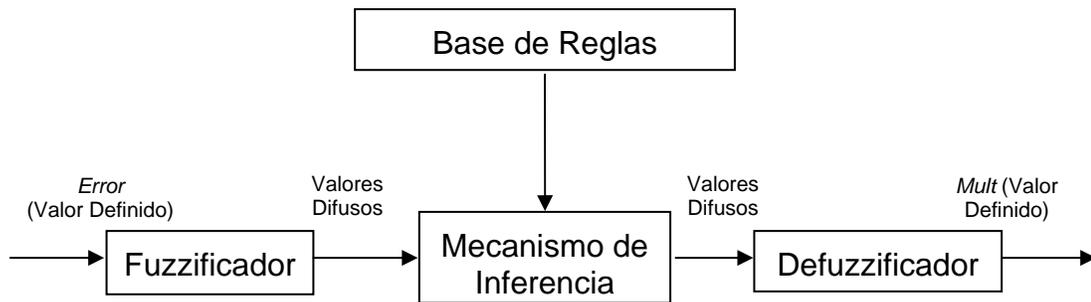


Figura 3.2. Estructura del Sistema Difuso Propuesto

Como método de difusificación se utiliza la difusificación Singleton y la desdifusificación se realiza mediante el método del Centro de Áreas. Para la inferencia difusa se utiliza la combinación de implicación por la regla del mínimo y la agregación por el máximo o *mecanismo de inferencia min-max o max-min*.

3.4.3 Variable Difusa de Entrada

La variable difusa de entrada utilizada en este sistema es la diferencia porcentual entre el sobrenivel deseado y el sobrenivel obtenido al simular la respuesta escalón del sistema con el controlador PID calculado en la sección 3.3. A esta variable se le asignó el nombre de *error*, y se definió para un *Universo de Discurso* en el rango $[-20\% ; 100\%]$, y las siguientes funciones de pertenencia:

$$\text{NG}(\text{error}) = \begin{cases} 0 & \text{si } (\text{error} \leq -20) \text{ ó } (\text{error} \geq -10) \\ 1 & \text{si } -20 < \text{error} < -15 \\ (-10 - \text{error}) / (-10 - (-15)) & \text{si } -15 < \text{error} < -10 \end{cases}$$

$$\text{NP}(\text{error}) = \begin{cases} 0 & \text{si } (\text{error} \leq -15) \text{ ó } (\text{error} \geq 0) \\ (\text{error} - (-15)) / (-10 - (-15)) & \text{si } -15 < \text{error} \leq -10 \\ 1 & \text{si } -10 < \text{error} < -5 \\ (-5 - \text{error}) / (-5 - 0) & \text{si } -5 < \text{error} < 0 \end{cases}$$

$$\begin{aligned}
 N(\text{error})= & \begin{cases} 0 & \text{si } \text{error} \leq -5 \\ (error - (-5)) / (0 - (-5)) & \text{si } -5 < \text{error} \leq 0 \\ (5 - \text{error}) / (5 - 0) & \text{si } 0 < \text{error} < 5 \\ 0 & \text{si } \text{error} > 5 \end{cases} \\
 PP(\text{error})= & \begin{cases} 0 & \text{si } (\text{error} \leq 0) \text{ ó } (\text{error} \geq 15) \\ (error - 0) / (5 - 0) & \text{si } 0 < \text{error} \leq 5 \\ 1 & \text{si } 5 < \text{error} < 10 \\ (15 - \text{error}) / (15 - 10) & \text{si } 10 < \text{error} < 15 \end{cases} \\
 PG(\text{error})= & \begin{cases} 0 & \text{si } (\text{error} \leq 10) \text{ ó } (\text{error} \geq 100) \\ (error - 10) / (15 - 10) & \text{si } 10 < \text{error} \leq 15 \\ 1 & \text{si } 15 < \text{error} < 100 \end{cases}
 \end{aligned}$$

La figura 3.3 muestra una representación gráfica de las funciones de pertenencia.

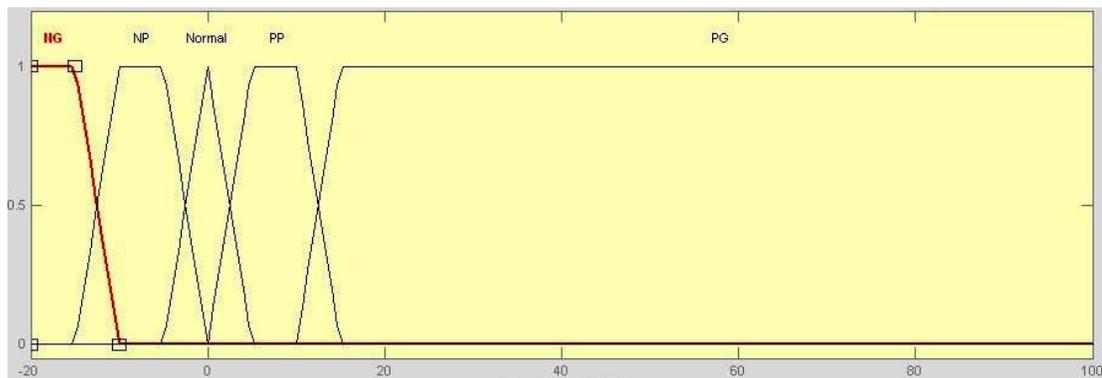


Figura 3.3. Funciones de Pertenencia para la Variable de Entrada

3.4.4 Variable Difusa de Salida

El sistema genera como salida un factor de corrección que se debe aplicar a la ganancia proporcional del controlador PID calculado, de manera de modificar el

sobrenivel presentado por la respuesta del sistema a un escalón unitario se “acerque” al sobrenivel deseado. A esta variable se le asignó el nombre de *mult*, se definió para un *Universo de Discurso* en el rango [0,5 ; 1,5], y las siguientes funciones de pertenencia:

$$\begin{aligned}
 MP(mult) &= \begin{cases} 0 & \text{si } mult \leq 0,5 \\ (0,65-mult)/(0,65-0,5) & \text{si } 0 < mult < 0,65 \\ 0 & \text{si } mult > 0,65 \end{cases} \\
 P(mult) &= \begin{cases} 0 & \text{si } (error \leq 0,5) \text{ ó } (error \geq 0,9) \\ (mult-0,5)/(0,65-0,5) & \text{si } 0,5 < mult \leq 0,65 \\ 1 & \text{si } 0,65 < mult < 0,75 \\ (0,9-mult)/(0,9-0,75) & \text{si } 0,75 < mult < 0,9 \end{cases} \\
 M(mult) &= \begin{cases} 0 & \text{si } (error \leq 0,75) \text{ ó } (error \geq 1,251) \\ (mult-0,75)/(0,9-0,75) & \text{si } 0,75 < mult \leq 0,9 \\ 1 & \text{si } 0,9 < mult < 1,1 \\ (1,1-mult)/(1,251-1,1) & \text{si } 1,1 < mult < 1,251 \end{cases} \\
 A(mult) &= \begin{cases} 0 & \text{si } (error \leq 1,1) \text{ ó } (error \geq 1,5) \\ (mult-1,1)/(1,251-1,1) & \text{si } 1,1 < mult \leq 1,251 \\ 1 & \text{si } 1,251 < mult < 1,35 \\ (1,35-mult)/(1,5-1,35) & \text{si } 1,35 < mult < 1,5 \end{cases}
 \end{aligned}$$

$$MA(mult)= \begin{cases} 0 & \text{si } mult \leq 1,35 \\ (mult-1,35)/(1,5-1,35) & \text{si } 1,35 < error < 1,5 \\ 0 & \text{si } mult > 1,5 \end{cases}$$

Las funciones de pertenencia definidas para el Factor de Corrección se muestran en la figura 3.4.

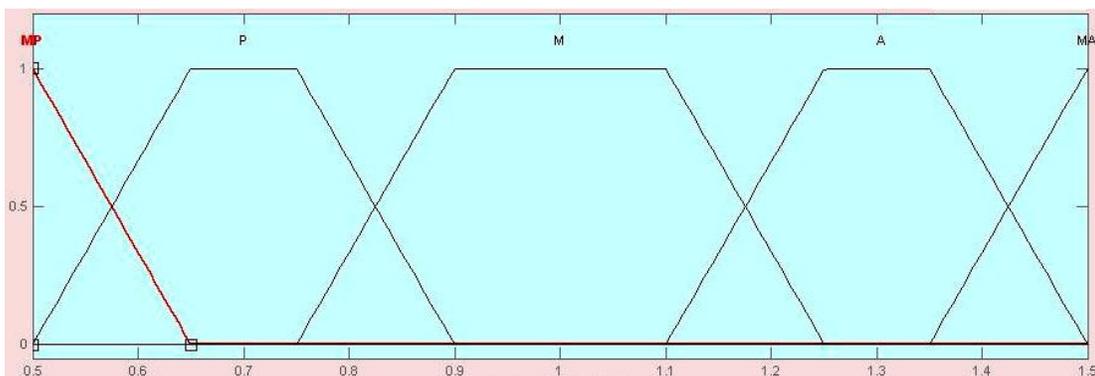


Figura 3.4. Funciones de Pertenencia para la Variable Salida

3.4.5 Base De Reglas Difusas

Para este sistema se definieron las siguientes reglas:

1. Si **Error** es NG, entonces **Multiplicador** es A
2. Si **Error** es NP, entonces **Multiplicador** es A
3. Si **Error** es N, entonces **Multiplicador** es M
4. Si **Error** es PP, entonces **Multiplicador** es P
5. Si **Error** es PG, entonces **Multiplicador** es MP

CAPÍTULO 4. IMPLEMENTACIÓN

El sistema se implementó de forma modular en Matlab[®] 7, donde cada módulo se corresponde con una de las etapas descritas en el capítulo anterior. Para las pruebas de funcionamiento se utilizaron las ecuaciones de dos plantas de prueba: la primera de ellas fue utilizada por Alfaro en [2] y [3], mientras que la segunda planta de prueba se tomó de página 359 de [14]. Cada módulo se probó de forma separada y posteriormente se realizó la prueba de todo el sistema completo. Al programa se le dio el nombre de CYRCPID, acrónimo de *Cálculo y Ajuste de Controladores PID*.

4.1 DEFINICIÓN DEL SISTEMA A CONTROLAR

La definición puede realizarse mediante una función de transferencia o desde un archivo con resultados de una prueba escalón.

Cuando se define mediante una función de transferencia, se solicita al usuario el numerador y el denominador de la misma. Éstos deben ser introducidos en forma de vectores entre corchetes.

Para la definición desde un archivo de datos, el usuario debe seleccionar el archivo desde donde se leerán los resultados de la prueba escalón. Dichos resultados deberán estar almacenados en formato de texto y distribuidos en tres columnas: la primera columna es el vector de tiempo, la segunda es el vector con los valores del escalón aplicado y la tercera columna contiene los valores de la respuesta de la planta al escalón aplicado.

El código desarrollado es el siguiente:

```
def = Opciones( 'DEFINICIÓN DEL SISTEMA A CONTROLAR',DEFINICIONES);
switch def
case 1
[num_Gps, den_Gps, Gps] = ObtenerTF;
definicion = 'funcion';
if length(num_Gps)>length(den_Gps)
disp('El sistema definido es impropio. No es posible continuar');
else
valorinicial = 0;
[y,t,valorfinal, sno, te]=PruebaEscalon(0,num_Gps,den_Gps,0,M,valorinicial);
SistemaDefinido = true;
disp('Sistema Definido');
figure(1);
plot(t,y,'-k');grid;
legend('Sistema Definido', 'location','SouthEast');
title('Respuesta Escalón Unitario del Sistema Original');
fprintf(salida,'Sistema definido mediante función de transferencia\n');
fprintf(salida,'Numerador (num_Gps) = %s\n',mat2str(num_Gps));
fprintf(salida,'Denominador (den_Gps) = %s\n',mat2str(den_Gps));
end % if length(num_Gps)>length(den_Gps)
case 2
definicion = 'archivo';
[Archivo,Path]=uigetfile('*.txt','Seleccione el archivo que contiene los datos de la prueba
escalón');
Archivo=[Path,Archivo];
S=load(Archivo);
t_orig=S(:,1);
u_orig=S(:,2);
y_orig=S(:,3);
valorinicial = y_orig(1);
figure(1);
plot(t_orig,y_orig,'.k');grid;
legend('Sistema Definido', 'location','SouthEast');
title('Respuesta Escalón Unitario del Sistema Original');
fprintf(salida,'Sistema definido desde el archivo \n%s\n',Archivo);

% Interpolación
tfinal = 1.1*t_orig(end);
tt = 0.1:0.1*(t_orig(end)-t_orig(1))/length(t_orig):tfinal;
yy = spline(t_orig,y_orig,tt);
t = tt';
y = yy';

% Calcula maximo
valorfinal = y(end);
if valorfinal>valorinicial
[Y,k]=max(y);
else
[Y,k]=min(y);
```

```

end;

% Cálculo del sobrenivel porcentual
if abs(valorfinal)<1e-4
    sno = 100*Y;
else
    sno = 100*(Y - valorfinal)/valorfinal;
end; % if valorfinal==0

% Calcula tiempo de estabilizacion
l = length(t);
while (abs((y(l)-valorfinal)/valorfinal)<0.02)&(l>1)
    l = l - 1;
end
te = t(l);

M = u_orig(1);
SistemaDefinido = true;
disp('Sistema Definido')
case 3
    disp('Sistema No Definido');
    disp('Definición cancelada por el usuario')
end; % switch def

```

4.2 MODELAJE DEL SISTEMA

El modelaje del sistema comienza con la selección del modelo a utilizar. Como se mencionó en la sección 3.2, se puede escoger entre 5 modelos distintos, de forma manual o automática. Las opciones disponibles son:

- Primer Orden más Tiempo Muerto de Smith
- Primer Orden más Tiempo Muerto de Ho *et al*
- Segundo Orden más Tiempo Muerto de Ho *et al* (Sobreamortiguado)
- Segundo Orden más Tiempo Muerto de Jahanmiri y Fallahi (Sobreamortiguado)
- Segundo Orden sin Tiempo Muerto de Mikles y Fikar (Subamortiguado)

El modelaje se realiza con el siguiente código:

```

modelocalculado = false;
if SistemaDefinido

```

```

ok = false;
while ~ok
m = Opciones('SELECCIÓN DEL MODELO DEL SISTEMA', MODELOS);
if m < 7 % m = 7 ==> Regreso al Menú Principal

disp('Calculando Modelo...');
if (sno == 0) & (m == 5) % Selección de Modelo Subamortiguado
    AplicaFikar = false;
    disp('No es posible calcular el modelo Mikles y Fikar');
    disp('El sistema no es subamortiguado');
else
    [n,d,K,Tau,tm,ym,t,vf1,sn,te,metodo] = Calcula_Modelo(y,t,m,M,valorinicial);
    titulo = {'Comparación entre el Sistema Original y el Modelo de ';char(MODELO(metodo))};
    yms = ym;
    figure(2);
    switch definicion
        case 'funcion'
            plot(t,y,'-k',t,yms,'-r');grid;
        case 'archivo'
            plot(t_orig,y_orig,'-k',t,yms,'r');grid;
    end; % switch definicion
    legend('Sistema Definido', 'Modelo Seleccionado','location','SouthEast');

% Si el sistema se definió por archivo de datos, se toma como
% función de transferencia el modelo calculado

    if def==2
        num_Gps = n;
        den_Gps = d;
    end;
    ok = true;
    title(titulo);
    fprintf(1,'\nModelo Calculado por el método de %s\n\n',char(MODELO(metodo)) );
    fprintf(1,'PARÁMETROS: \n');
    fprintf(1,' Ganancia (K) = %6.3g\n', K);
    fprintf(1,' Constante de Tiempo (Tau) = %6.3g\n', Tau);
    fprintf(1,' Tiempo Muerto (tm) = %6.3g\n', tm);

    fprintf(salida,'\nModelo Calculado por el método de %s\n\n',char(MODELO(metodo)) );
    fprintf(salida,'PARÁMETROS: \n');
    fprintf(salida,'Ganancia (K) = %6.3g ', K);
    fprintf(salida,'Constante de Tiempo (Tau) = %6.3g ', Tau);
    fprintf(salida,'Tiempo Muerto (tm) = %6.3g\n', tm);

    fprintf(salida,'Función de Transferencia del Modelo\n');
    fprintf(salida,'Numerador (n) = %s\n',mat2str(n));
    fprintf(salida,'Denominador (d) = %s\n\n',mat2str(d));
    fprintf(salida,'Desempeño del Modelo\n');
    fprintf(salida,'Valor Final %6.3g ',vf1);
    fprintf(salida,'Sobrenivel: %6.2f%% ', sn);
    fprintf(salida,'Tiempo de Estabilización (2%%): %6.3g\n\n',te);

```

```

end; % if (sno 0==0) & (m == 4)
else
    disp('Modelo No Calculado');
    disp('Cálculo cancelado por el usuario');
    ok = true;
end % if m < 7
end; % while ~ok
else
    disp('Se requiere definir primero el sistema para después modelarlo');
end; % if SistemaDefinido
ModeloCalculado = true;
disp('Presione una tecla para continuar...');
pause;

```

La función que realiza el cálculo del modelo es:

```

function [n,d,K,Tau,tm,ye,t,vf,sn,te,metodo,AF] = Calcula_Modelo(y,t,metodo,M,valorinicial);
% Cálculo del modelo del sistema.
% Parámetros de entrada:
% t : vector de tiempos de simulación
% y : vector de valores de la función
% metodo: Método de cálculo del modelo
% 1 = Smith de Primer Orden más Tiempo Muerto
% 2 = Ho et al. de Primer Orden más Tiempo Muerto
% 3 = Ho et al. de Segundo Orden más Tiempo Muerto
% 4 = Jahanmiri y Fallahi de Segundo Orden más Tiempo Muerto
% 5 = Mikles y Fikar (subamortiguado, sin tiempo muerto).
% 6 = Automático. Calcula los tres modelos y selecciona el más
% ajustado por el mínimo de la suma de los cuadrados del
% error de predicción (valor predicho - valor real).
% Parámetros de Salida:
% n : numerador de la función de transferencia del modelo.
% Típicamente, la ganancia de estado estacionario.
% d : denominador de la función de transferencia del modelo. Contiene
% la constante de tiempo (Tau), y el coeficiente de
% amortiguamiento (z).
% tm : tiempo muerto
% ye : vector de valores estimados (predichos).
% t : vector de tiempos de simulación
% AF : AplicaFikar, parámetros que es true si es posible aplicar el
% método de Mikles y Fikar
valorinicial = y(1);
valorfinal = y(end);
if valorfinal>valorinicial
    [Y,k]=max(y);
else
    [Y,k]=min(y);
end;
% Calcula sobrenivel porcentual

```

```

if abs(valorfinal)<1e-4
    sno = 100*Y;
else
    sno = 100*(Y - valorfinal)/valorfinal;
end; % if valorfinal==0
true = 1;
false = 0;
AF = false;
disp('Calculando modelo');
ok = false;
while ~ok
    switch metodo
        case 1
            disp('Smith ...');
            [n,d,tm] = Smith(y,t,M);
            ok = true;
            K = n;
            Tau = d(1);
            ok = true;
        case 2
            disp('Ho Orden 1 ...');
            [n,d,tm] = Ho_Orden_1(y,t,M);
            K = n
            Tau = d(1);
            ok = true;
        case 3
            disp('Ho Orden 2 ...');
            [n,d,tm] = Ho_Orden_2(y,t,M);
            K = n
            Tau = sqrt(d(1));
            ok = true;
        case 4
            disp('Jahanmiri y Fallahi ...');
            [n,d,tm] = Jahanmiri_Fallahi(y,t,M);
            K = n
            Tau = sqrt(d(1));
            ok = true;
        case 5
            disp('Mikles y Fikar ...');
            if (sno~=0) %(sobre nivel < 0)
                [n,d,tm] = Fikar(y,t,M);
                K = n
                Tau = sqrt(d(1));
                AF = true;
                ok = true;
            else
                disp('No es posible calcular el modelo Mikles y Fikar');
                disp('El sistema no es subamortiguado');
                AF = false;
            end;
        case 6

```

```

disp('Smith ...');
[na1,da1,tma(1)] = Smith(y,t,M);
[ym(:,1),t,vfa(1), sna(1), tea(1)]=PruebaEscalon(tma(1),na1,da1,t,M,valorinicial);
Chi2(1) = Chi_2(ym(:,1),y);
disp('Ho Orden 1 ...');
[na2,da2,tma(2),ok] = Ho_Orden_1(y,t,M);
if ok
    [ym(:,2),t,vfa(2), sna(2), tea(2)]=PruebaEscalon(tma(2),na2,da2,t,M,valorinicial);
    Chi2(2) = Chi_2(ym(:,2),y);
else
    Chi2(2) = realmax;
end;
disp('Ho Orden 2 ...');
[na3,da3,tma(3), ok] = Ho_Orden_2(y,t,M);
if ok
    [ym(:,3),t,vfa(3), sna(3), tea(3)]=PruebaEscalon(tma(3),na3,da3,t,M,valorinicial);
    Chi2(3) = Chi_2(ym(:,3),y);
else
    Chi2(3) = realmax;
end;
disp('Jahanmiri y Fallahi ...');
[na4,da4,tma(4), ok] = Jahanmiri_Fallahi(y,t,M);
if ok
    [ym(:,4),t,vfa(4), sna(4), tea(4)]=PruebaEscalon(tma(4),na4,da4,t,M,valorinicial);
    Chi2(4) = Chi_2(ym(:,4),y);
else
    Chi2(4) = realmax;
end;
disp('Mikles y Fikar ...');
if (sno~=0)  %(sobre nivel <> 0)
    [na5,da5,tma(5)] = Fikar(y,t,M);
    [ym(:,5),t,vfa(5),sna(5),tea(5)]=PruebaEscalon(tma(5),na5,da5,t,M,valorinicial);
    Chi2(5) = Chi_2(ym(:,5),y);
    AplicaFikar = true;
else
    disp('No es posible calcular el modelo Mikles y Fikar');
    disp('El sistema no es subamortiguado');
    Chi2(5) = realmax;
    AplicaFikar = false;
end;
disp('Modelos Calculados');
disp('Seleccionando mejor Modelo...');
% SELECCIÓN DEL MODELO A UTILIZAR MEDIANTE LA SUMA DE LOS
% MÍNIMOS CUADRADOS DE LAS DESVIACIONES ENTRE LOS VALORES
% ESTIMADOS Y LOS REALES
[Seleccionado, Seleccion] = min(Chi2);
ok = true;
end; % switch metodo
end; % while ~ok
if (metodo < 6)
    [ye,t,vf, sn, te]=PruebaEscalon(tm,n,d,t,M,valorinicial);

```

```

else
metodo = Seleccion;
ye = ym(:,Seleccion);
tm = tma(Seleccion);
vf = vfa(Seleccion);
sn = sna(Seleccion);
te = tea(Seleccion);
switch Seleccion
case 1
    n = na1;
    d = da1;
    K = n;
    Tau = d(1);
case 2
    n = na2;
    d = da1;
    K = n;
    Tau = d(1);
case 3
    n = na3;
    d = da3;
    K = n;
    Tau = sqrt(d(1));
case 4
    n = na4;
    d = da4;
    K = n;
    Tau = sqrt(d(1));
case 5
    n = na5;
    d = da5;
    K = n;
    Tau = sqrt(d(5));
end; % switch Seleccion
end; % if (metodo < 6)

```

4.3 CÁLCULO DEL CONTROLADOR PID

Las opciones disponibles para el cálculo del controlador PID consisten en seis métodos y una opción para regresar al menú principal. La estructura es como sigue:

1. Rovira et al (1969) IAE
2. Sung et al. (1996) para Servos
3. Sung et al. (1996) para Reguladores
4. Rivera et al (1985) sin tiempo muerto
5. Ziegler y Nichols

6. Cohen y Coon

7. Regresar al Menú Principal.

El cálculo del Controlador PID se realiza de acuerdo con el siguiente código:

```
c = Opciones('SELECCIÓN DEL MÉTODO DE CÁLCULO DEL CONTROLADOR', PIDS);
if c < 7
    disp('Calculando Controlador PID...');
    switch c
        case 1
            disp('Método de Rovira IAE');
            [Kc,Ti,Td] = PID_Rovira_IAE(K,Tau,tm);
        case 2
            disp('Sung para Servos');
            [Kc,Ti,Td, Error] = PID_Sung_Servo(K,Tau,1,tm);
        case 3
            disp('Sung para Reguladores');
            [Kc,Ti,Td, Error] = PID_Sung_Reg(K,Tau,1,tm);
        case 4
            z = d(2)/(2*sqrt(d(1)));
            disp('Rivera sin tiempo muerto');
            [Kc,Ti,Td] = PID_Rivera_0_tm_1(K,Tau,z);
        case 5
            disp('Método de Ziegler y Nichols');
            [Kc, Ti, Td] = PID_Ziegler_Nichols(K, Tau, tm);
        case 6
            disp('Método de Cohen y Coon');
            [Kc, Ti, Td] = PID_Cohen_Coon(K, Tau, tm);
    end; % switch(c)
else
    disp('Controlador No Calculado');
    disp('Cálculo cancelado por el usuario');
    disp('Presione una tecla para continuar...');
    pause;
end; % if c < 7
```

4.4 OPTIMIZACIÓN DE LA GANANCIA PROPORCIONAL DEL CONTROLADOR

La optimización de la ganancia proporcional (K_p) del controlador comienza con la selección del modo en el cual operará el mismo, sea como regulador o como servomecanismo. El algoritmo de optimización descrito en la sección 3.4.1 se

codificó de la siguiente forma:

```

d = Opciones('FUNCIÓN DEL CONTROLADOR', APLICACION);
if d < 3
    sobrenivel_deseado = input('¿Máximo Sobrenivel Deseado (%): ');
    switch d
        case 1
            disp('Optimizando controlador como servomecanismo');
            fprintf(salida, 'Optimizando controlador como servomecanismo\n');
            sobrenivel = sobrenivels;
        case 2
            disp('Optimizando controlador como regulador');
            fprintf(salida, 'Optimizando controlador como regulador\n');
            sobrenivel = sobrenivelr;
    end; %switch d
    fprintf(salida, 'para un sobrenivel máximo de %5.2f%%\n', sobrenivel_deseado);
    Error = round(100*(sobrenivel-sobrenivel_deseado)/sobrenivel_deseado)/100;
    fprintf(1, 'Error Kc Sobrenivel Tiempo_Estabilización Valor Final\n')
    MaxIter = 500;
    iter = 1;
    while (Error > 0) & (iter <= MaxIter)
        %Cálculo de la nueva Kc con el módulo Fuzzy
        a = readfis('Sobrepaso');
        Mult = evalfis(Error, a);
        Kc_anterior = Kc;
        if d == 1
            Kc = Kc * Mult; % Servo
        else
            Kc = Kc / Mult; % Regulador
        end;

        % Cálculo del nuevo controlador PID
        num_Gc = Kc * [Ti * Td Ti 1];
        den_Gc = [0 Ti 0];

        % Función de Transferencia en lazo cerrado de la planta más el controlador
        switch d
            case 1
                [num_lc, den_lc] = FT_Lazo_Cerrado(num_Gps, den_Gps, num_Gc, den_Gc, 'servo');
            case 2
                [num_lc, den_lc] =
FT_Lazo_Cerrado(num_Gps, den_Gps, num_Gc, den_Gc, 'regulador');
        end; % switch d

        % Verificación del resultado
        [yf, t, vf, sobrenivel, te] = PruebaEscalon(tm, num_lc, den_lc, t, M, valorinicial);
        fprintf(1, '%5.2f %7.4f %7.2f %5.2f %6.3f\n', Error, Kc, sobrenivel, te, vf);
        % Error = (sobrenivel-sobrenivel_deseado)/sobrenivel_deseado;
        Error = round(100*(sobrenivel-sobrenivel_deseado)/sobrenivel_deseado)/100;
        iter = iter + 1;
    end;

```

```

end; % while ((abs(Error) > 0.01) & (iter <= MaxIter))
if (iter > MaxIter)
    disp('EXCEDIDO EL NÚMERO MÁXIMO DE ITERACIONES');
    disp('ÚLTIMOS VALORES CALCULADOS:');
    fprintf(salida,'EXCEDIDO EL NÚMERO MÁXIMO DE ITERACIONES\n');
    fprintf(salida,'ÚLTIMOS VALORES CALCULADOS:\n\n');
else
    disp('Controlador Optimizado');
end
% Cálculo de las Nuevas Funciones de Transferencia en Lazo Cerrado
[num_lc_s,den_lc_s]=FT_Lazo_Cerrado(num_Gps,den_Gps,num_Gc,den_Gc,'servo');
[num_lc_r,den_lc_r]=FT_Lazo_Cerrado(num_Gps,den_Gps,num_Gc,den_Gc,'regulador');

% Cálculo de las Nuevas Respuestas a un Escalón Unitario
[yis,t,vfs,sobrenivels,tes]=PruebaEscalon(tm,num_lc_s,den_lc_s,t,M,valorinicial);
[yir,t,vfr,sobrenivelr,ter]=PruebaEscalon(tm,num_lc_r,den_lc_r,t,M,valorinicial);
fprintf(1,'PARÁMETROS DE AJUSTE \n Kc: %5.2f, Ti: %5.2f, Ts: %5.2f \n\n', Kc, Ti,
Td);
fprintf(1,'DESEMPEÑO ESPERADO COMO SERVO:\n');
fprintf(1,'Sobrenivel: %5.2f%% \n', sobrenivels);
fprintf(1,'Valor Final: %5.2f \n', vfs);
fprintf(1,'Tiempo de Estabilización: %6.2f \n\n', tes);
fprintf(1,'DESEMPEÑO ESPERADO COMO REGULADOR:\n');
fprintf(1,'Sobrenivel: %5.2f%% \n', sobrenivelr);
fprintf(1,'Valor Final: %5.2f \n', vfr);
fprintf(1,'Tiempo de Estabilización: %5.2f \n\n', ter);
fprintf(salida,'Parámetros de Ajuste \n Kc: %5.2f, Ti: %5.2f, Ts: %5.2f \n\n', Kc, Ti, Td);
fprintf(salida,'Desempeño Esperado como Servo:\n');
fprintf(salida,'Sobrenivel: %5.2f%% \n', sobrenivels);
fprintf(salida,'Valor Final: %5.2f \n', vfs);
fprintf(salida,'Tiempo de Estabilización: %6.2f \n\n', tes);
fprintf(salida,'Desempeño Esperado como Regulador:\n');
fprintf(salida,'Sobrenivel: %5.2f%% \n', sobrenivelr);
fprintf(salida,'Valor Final: %5.2f \n', vfr);
fprintf(salida,'Tiempo de Estabilización: %5.2f \n\n\n', ter);
else % if d<3
    disp('Controlador No Optimizado');
    disp('Cálculo cancelado por el usuario');
    disp('Presione una tecla para continuar...');
    pause;
end; % if d<3

```

4.5 INTERFAZ DE USUARIO

La interfaz de usuario desarrollada es sencilla y permite que el usuario tenga acceso a todas las opciones del programa. La figura 4.1 muestra la pantalla de inicio del programa. En esta pantalla se tienen a disposición todas las funciones del

programa:

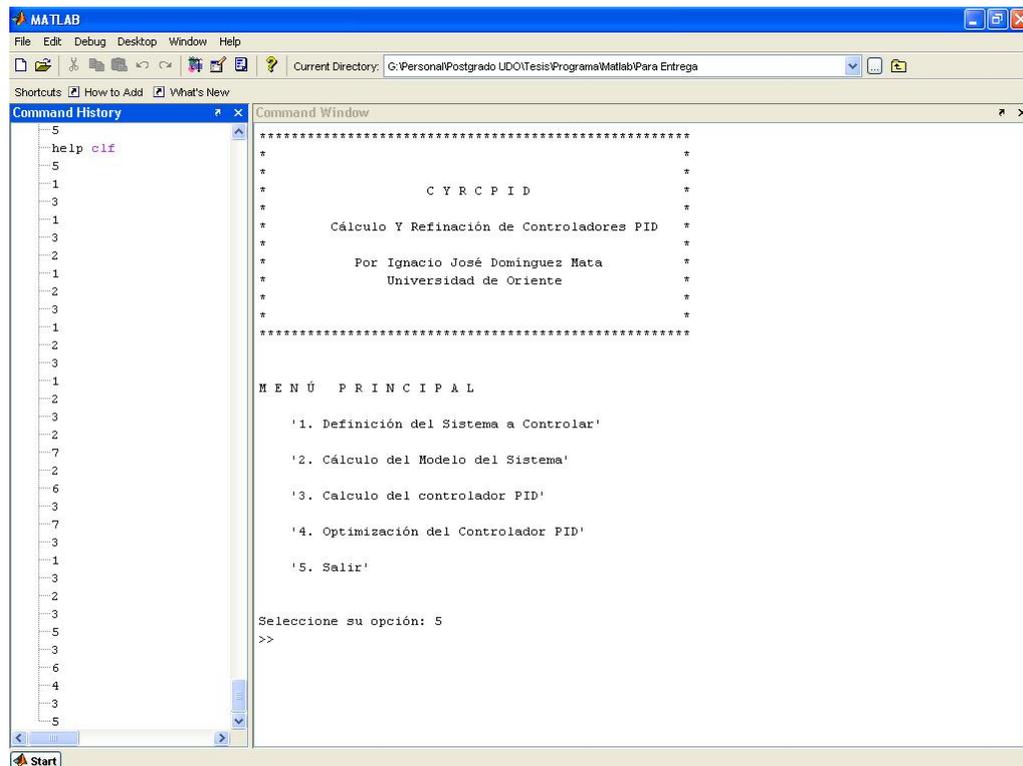


Figura 4.1. Pantalla Inicial del Programa CYRCPID

4.5.1 Definición del Sistema a Controlar

La opción 1 (*Definición del Sistema a Controlar*) permite, como su nombre lo indica, definir la planta para la cual se calculará y optimizará el controlador PID. La definición se puede realizar mediante una función de transferencia o desde un archivo con los datos de una curva de reacción. Cuando la definición se realiza mediante la función de transferencia de la planta, el programa solicita el numerador y el denominador de dicha función de transferencia, así como también el tiempo muerto. Esto se puede observar en la figura 4.2.

Si la definición de la planta se realiza mediante los datos de una curva de

reacción, se abre una ventana donde el usuario selecciona el archivo donde se encuentran dichos datos, como se muestra en la figura 4.3.

The screenshot shows the MATLAB Command Window with the following text:

```

*****
*
*
*           C Y R C P I D
*
*   Cálculo Y Refinación de Controladores PID
*
*   Por Ignacio José Domínguez Mata
*   Universidad de Oriente
*
*
*****

DEFINICIÓN DEL SISTEMA A CONTROLAR

'1. Mediante Función de transferencia'

'2. Desde Archivo de Datos'

'3. Regresar al Menú Principal'

Seleccione su opción: 1
Defina la Planta Gp(s)
  Indique el numerador: 1
  Indique el denominador: [0.125 0.875 1.75 1]
  Indique el tiempo muerto: 0.1

[0.125 0.875 1.75 1]

```

The Command History window on the left shows a list of commands and their execution status, with the current command being the definition of the plant parameters.

Figura 4.2. Definición de la Planta Mediante Función de Transferencia

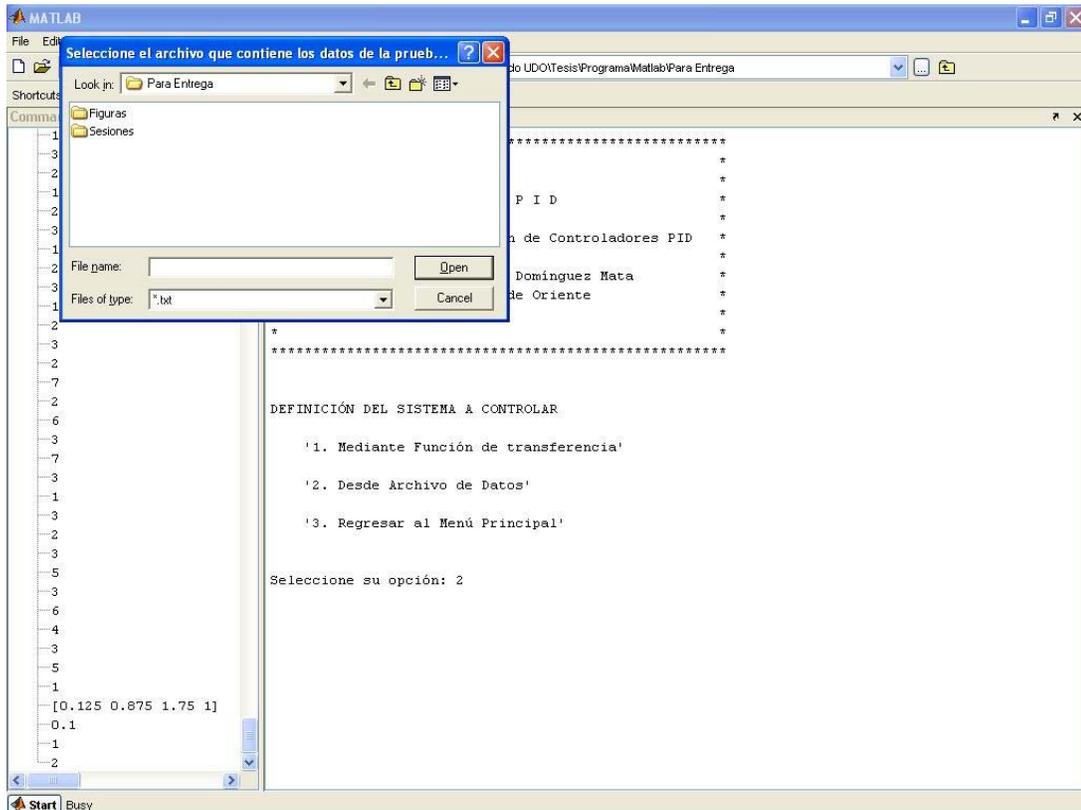


Figura 4.3. Definición de la Planta Mediante Curva de Reacción

4.5.2 Cálculo del Modelo del Sistema

La figura 4.4 muestra el submenú al cual se accede cuando se selecciona la opción 2 del menú principal (*Calculo del Modelo del Sistema*). Se puede seleccionar cualquiera de los modelos disponibles (opciones 1 a la 5) o se puede dejar que el programa calcule todos los modelos y seleccione automáticamente el que mejor representa a la planta (opción 6: *Selección Automática*). La selección se realiza de acuerdo con la Integral Absoluta del Error de Predicción definido en la sección 3.1.

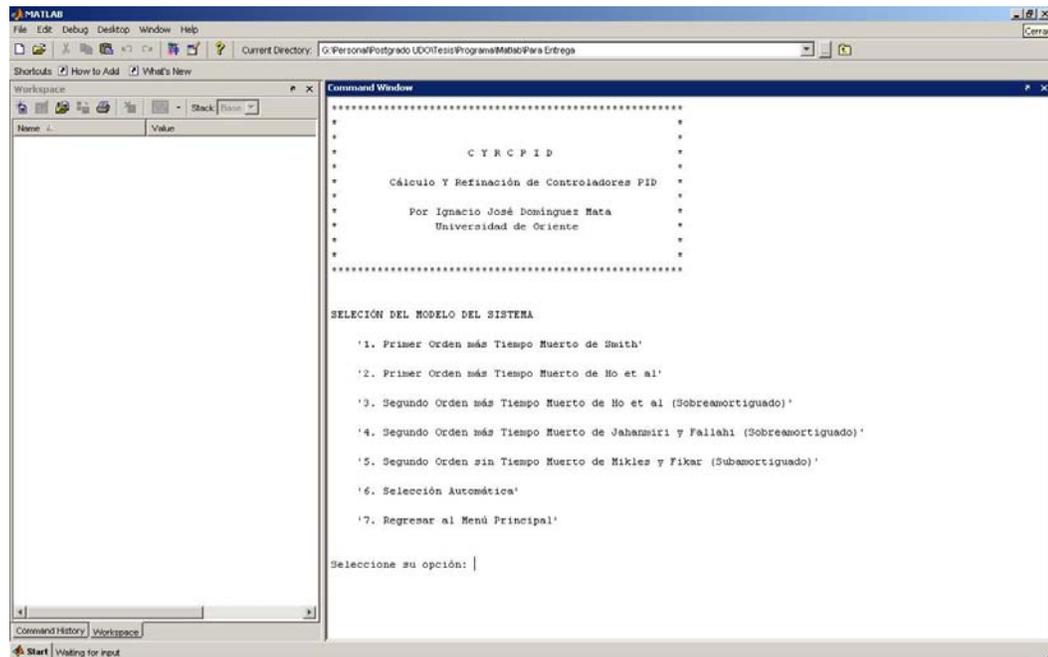


Figura 4.4. Opciones para el Cálculo del Modelo del Sistema

4.5.3 Cálculo del Controlador PID

La opción 3 (*Cálculo del Controlador PID*) permite acceder al submenú que se muestra en la figura 4.5. Cada opción de este submenú permite calcular los parámetros de un controlador PID Ideal por el método seleccionado y generar gráficas con la respuesta de la planta a una entrada en forma de escalón, tanto en modo servo como en modo regulador.

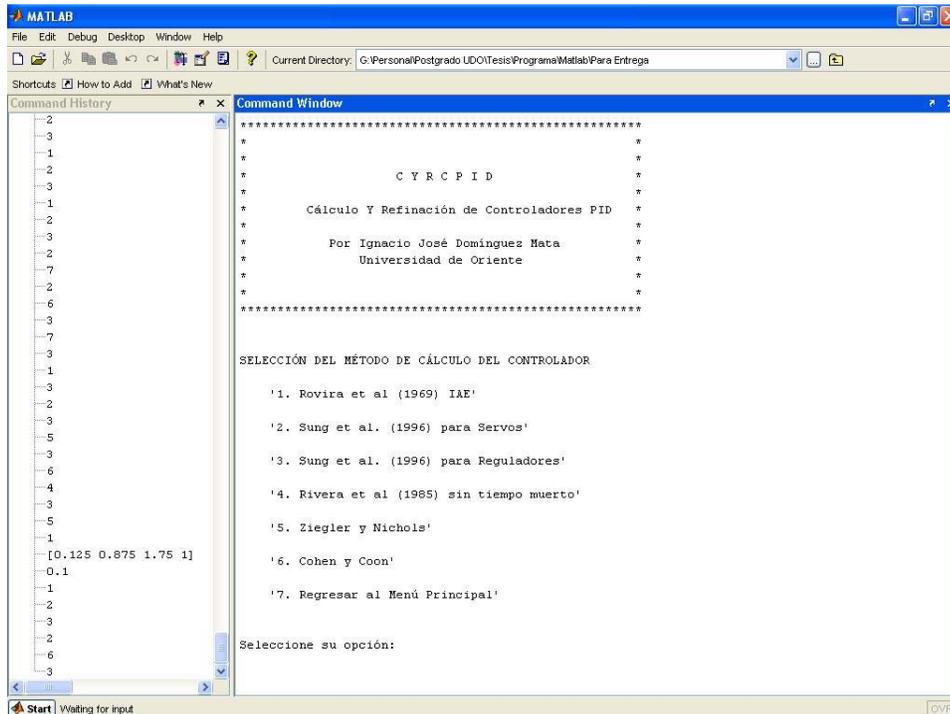


Figura 4.5. Opciones para el Cálculo del Controlador PID

4.5.4 Optimización del Controlador PID

La Ajuste de la constante proporcional del controlador PID calculado se realiza seleccionando la opción 4 (*Optimización del Controlador PID*). Al seleccionar esta opción se tiene acceso a un submenú que permite seleccionar el modo principal de funcionamiento del controlador, bien sea como servomecanismo o como regulador. Una vez seleccionado el modo de funcionamiento, el programa pregunta el máximo porcentaje de sobrenivel deseado, tal como se muestra en la figura 4.6.

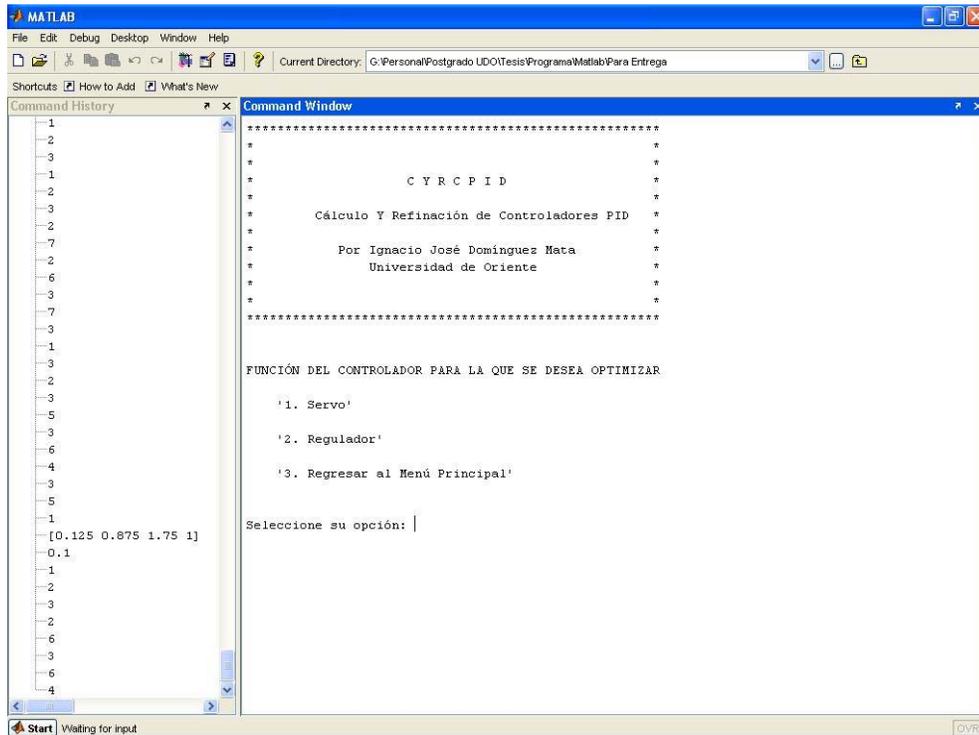


Figura 4.6. Opciones para la Optimización del Controlador PID

El programa almacena los resultados de cada sesión de trabajo en un archivo de texto ubicado en una carpeta llamada *Sesiones*. Esta carpeta se encuentra en el mismo directorio donde se encuentra instalado el programa. El nombre del archivo para cada sesión de trabajo se construye con la fecha y hora del momento, y su formato es *aaaammdd_hhmmss.txt*.

CAPÍTULO 5. PRUEBA

5.1 METODOLOGÍA DE PRUEBA

Las pruebas realizadas a objeto de este trabajo consistieron en calcular el mejor modelo para plantas de prueba definidas mediante sus funciones de transferencia y plantas definidas mediante los datos de la curva de reacción obtenida al aplicar una entrada en forma de escalón. Posteriormente se calcularon controladores PID por cada uno de los métodos disponibles en el programa y se compararon los desempeños del sistema con cada controlador calculado. Finalmente se optimizó el valor de la constante proporcional de los controladores calculados que no ocasionaron funcionamiento inestable de la planta de prueba.

Para comparar el desempeño del sistema con cada controlador calculado se utilizaron tres parámetros: el valor final, el sobrenivel y el tiempo de estabilización al 2% del valor final. Para el cálculo del sobrenivel, se tomaron en cuenta dos escenarios:

a.- Cuando el valor final es distinto de cero. En este caso, el sobrenivel se definió como la diferencia porcentual entre el máximo valor de la respuesta del sistema y el valor final, es decir:

$$Sobrenivel = 100 \cdot \frac{Máximo - ValorFinal}{ValorFinal} \quad (5.1)$$

b.- Cuando el valor final es igual a cero. En este caso, el sobrenivel se definió como:

$$Sobrenivel = 100 \cdot ValorFinal \quad (5.2)$$

El tiempo de estabilización al 2 % se definió también con base en dos escenarios:

a.- El sistema opera en modo servo. En este caso, el valor final de la respuesta es distinto a su valor inicial. En este caso, el tiempo de estabilización al 2 % se calcula como el tiempo en que el valor de la respuesta del sistema cae por debajo del 2 % de la diferencia entre el valor inicial y el valor final.

b.- El sistema opera en modo regulador. En este caso, como el valor final de la respuesta es igual al valor inicial, el tiempo de estabilización se define como el tiempo en el que la respuesta cae por debajo del 2 % del valor final.

5.2 PRUEBA CON FUNCIÓN DE TRANSFERENCIA

5.2.1 Planta de Prueba

Como planta de prueba se utilizó la misma planta utilizada por Alfaro en [2] y [3], y definida por:

$$Gp(s) = \frac{e^{-0,1s}}{(s + 1)(0,5s + 1)(0,25s + 1)} \quad (5.3)$$

De forma equivalente, la planta de prueba también se puede definir como:

$$Gp(s) = \frac{e^{-0,1s}}{0,125s^3 + 0,875s^2 + 1,75s + 1} \quad (5.4)$$

5.2.2 Identificación del Modelo de la Planta

Para la identificación del modelo de la planta, se utiliza la opción número 1, de la sección “Definición del Sistema a Controlar”. Esta opción permite definir la planta desde una función de transferencia.

Para definir la función de transferencia de la planta, se debe expresar en la forma de la ecuación 5.4. Una vez introducidos el numerador y el denominador de la función de transferencia, así como también el tiempo muerto, se genera la curva de reacción correspondiente. La figura 5.1 muestra la pantalla donde se define la función de transferencia.

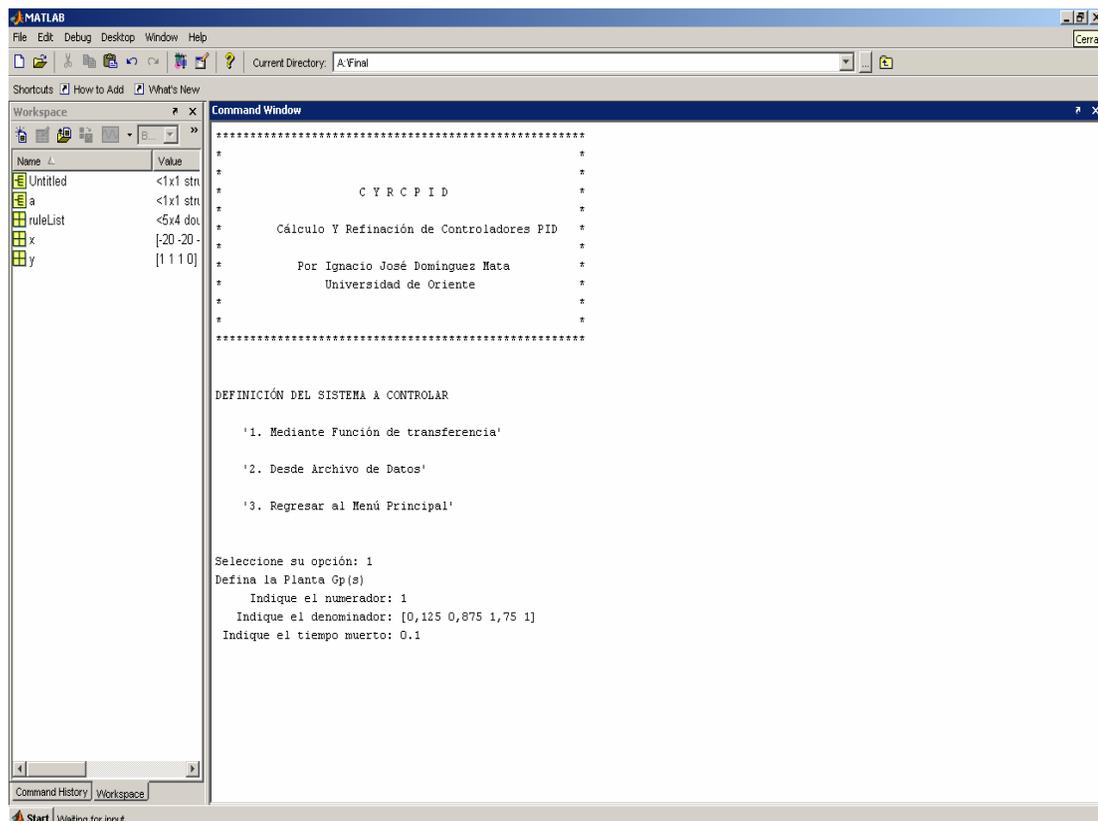


Figura 5.1. Definición de la Función de Transferencia de la Planta de Prueba

Para el cálculo del modelo se utiliza la opción “Cálculo del Modelo del Sistema”. Para fines de este trabajo, se calcularon todos los modelos disponibles y seleccionó el que presentó menor error de predicción. La figura 5.2 muestra la pantalla para el cálculo del modelo según el método de POMTM de Smith, mientras que las figuras 5.3 a 5.6 muestran la comparación entre la curva de reacción original y

la obtenida con cada uno de los modelos calculados. La tabla 5.1 muestra el error de predicción de cada modelo.

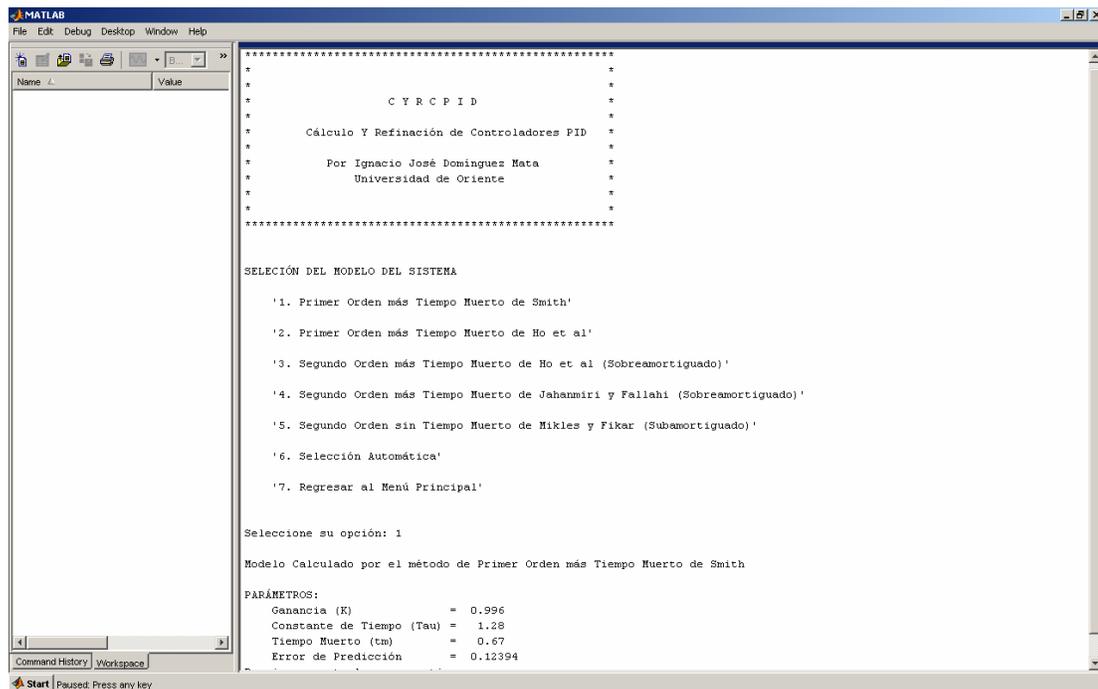


Figura 5.2. Cálculo del Modelo según Smith

Tabla 5.1. Parámetros de los Modelos Calculados para la Planta de Prueba mediante Función de Transferencia

| Parámetros | Modelos | | | |
|--------------------------------|---------|--------------------------|--------------------------|---------------------|
| | Smith | Ho <i>et al.</i> Orden 1 | Ho <i>et al.</i> Orden 2 | Jahanmiri y Fallahi |
| Ganancia (K) | 0,996 | 0,996 | 0,996 | 0,996 |
| Constante de Tiempo (τ) | 1,28 | 1,12 | 0,772 | 0,48 |
| Tiempo Muerto (t_m) | 0,67 | 0,77 | 0,284 | 0,39 |
| Error de Predicción | 0,1239 | 0,0977 | 0,0164 | 0,4674 |

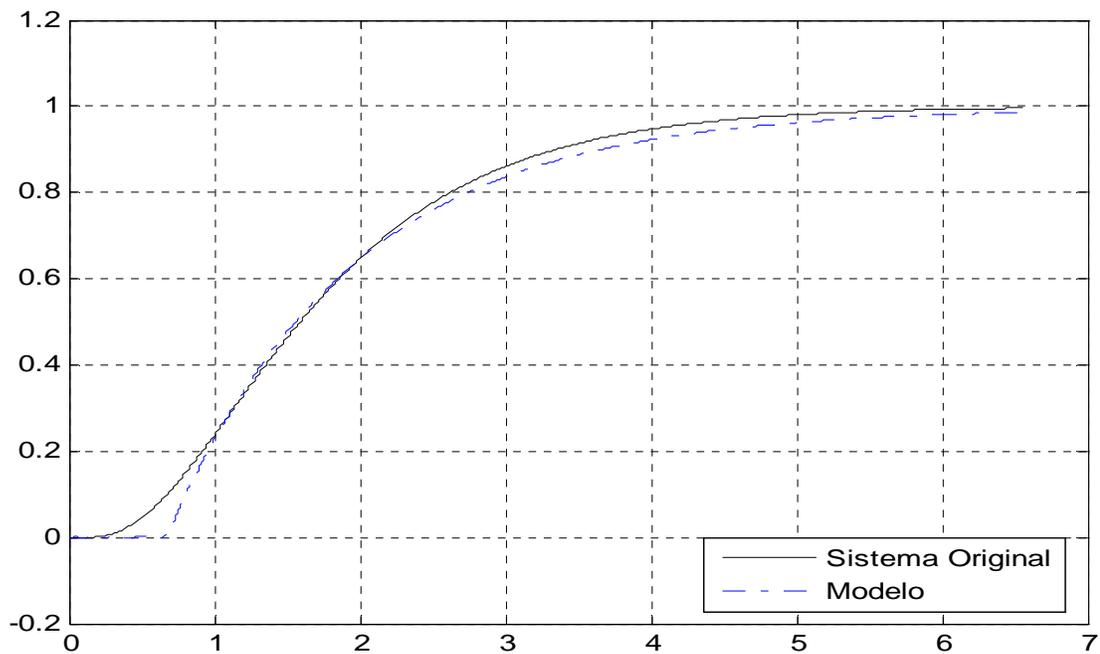


Figura 5.1. Comparación de la Curva de Reacción Original y la del Modelo según Smith

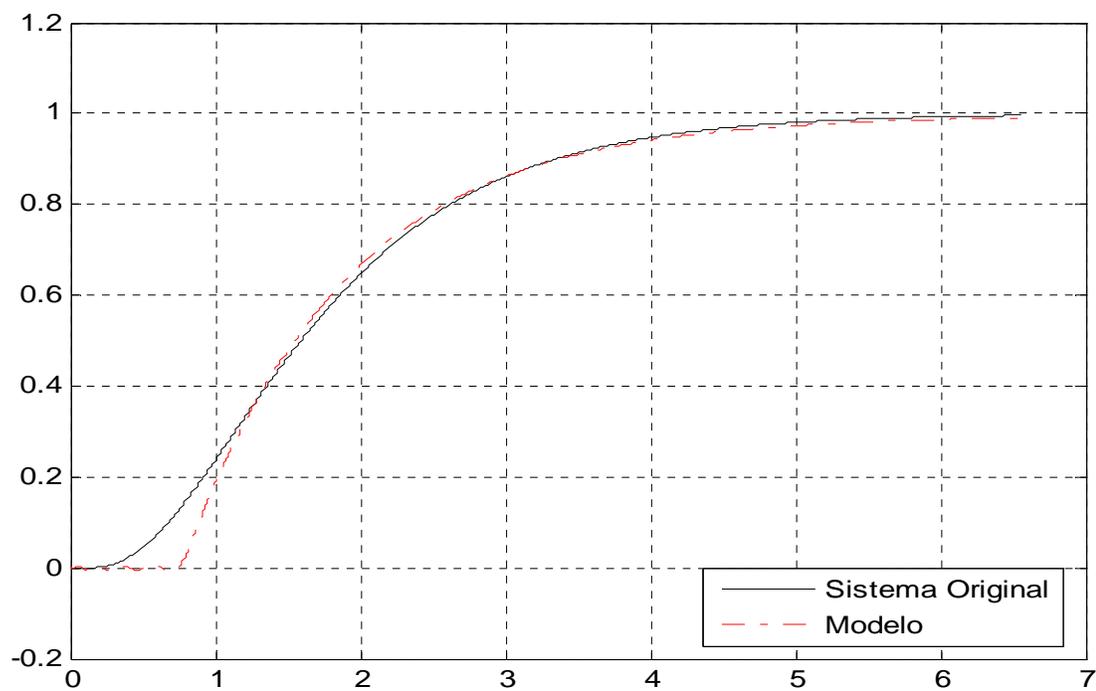


Figura 5.2. Comparación de la Curva de Reacción Original y la del Modelo según Ho *et al.*

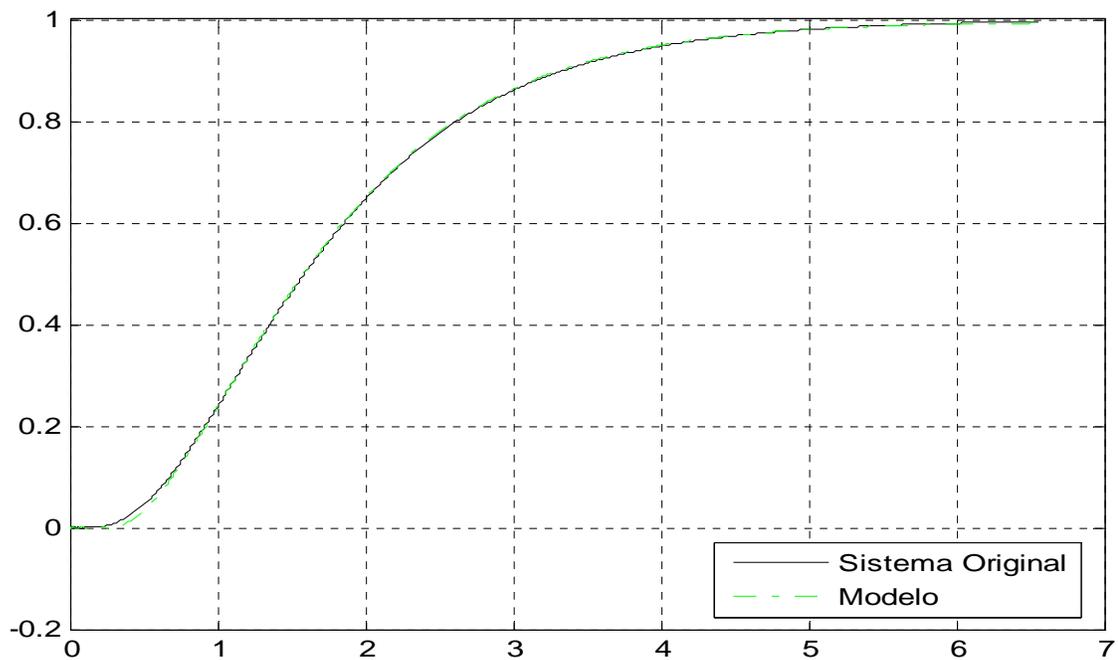


Figura 5.3. Comparación de la Curva de Reacción Original y la del Modelo según Ho et al.

SOMTM

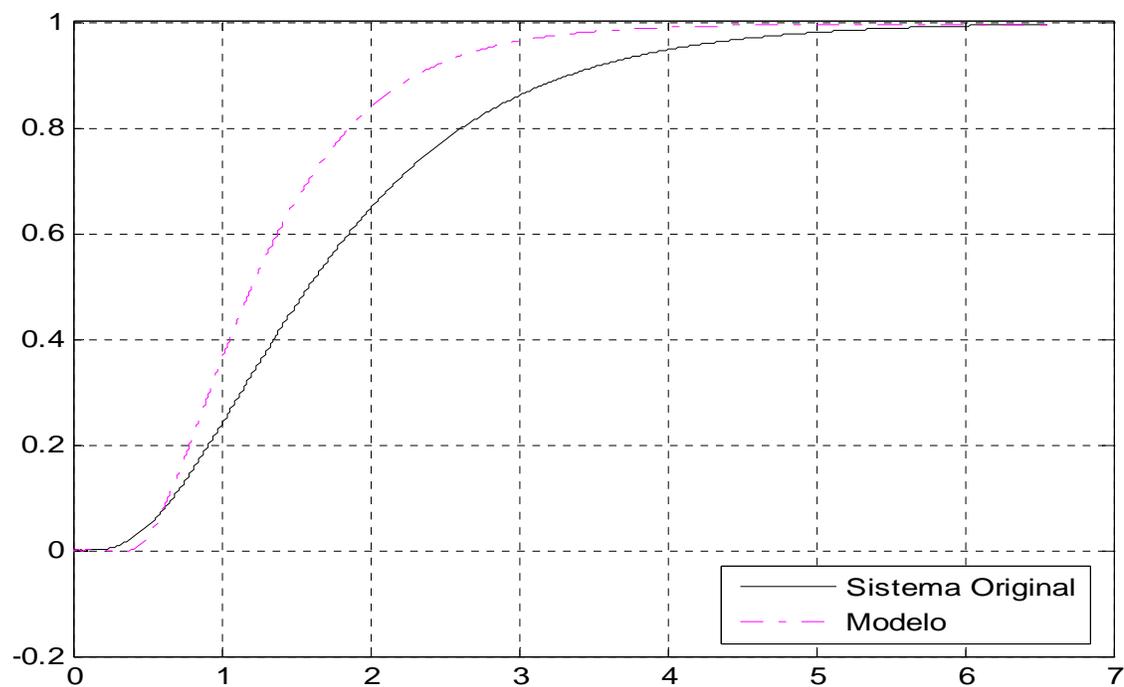


Figura 5.4. Comparación de la Curva de Reacción Original y la del Modelo según Jahanmiri y Fallahi

Se seleccionó el modelo de Segundo Orden Más Tiempo Muerto (SOMTM) de Ho *et al.*, por ser el que presentó menor error de predicción. La función de transferencia del modelo calculado es, entonces:

$$Gp(s) = \frac{e^{-0,284s}}{0,596s^2 + 1,544s + 1} \quad (5.2)$$

5.2.3 Cálculo del controlador PID

Para calcular el controlador PID se utilizaron los métodos disponibles, a objeto de mostrar las diferencias entre el desempeño de cada uno. La figura 5.5 muestra la pantalla para el cálculo del controlador por el método de Rovira *et al.* IAE, y la tabla 5.2 muestra los parámetros del controlador según los distintos métodos aplicados. El desempeño del sistema con cada controlador se muestra en las figuras 5.6 a 5.10.

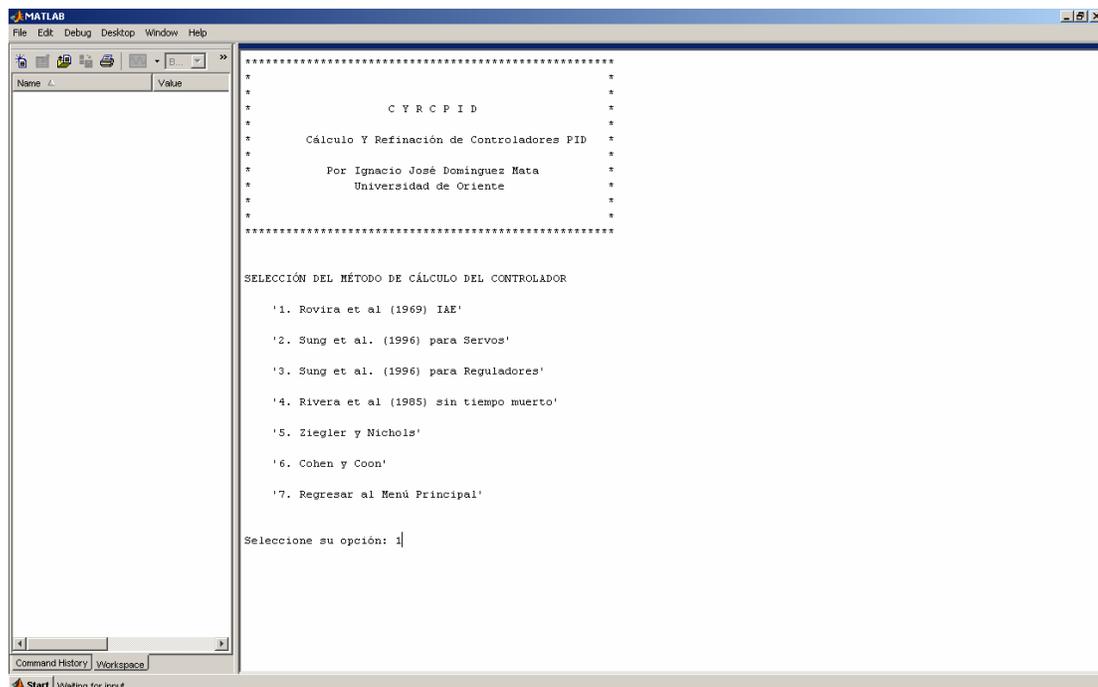


Figura 5.5. Cálculo del Controlador según Rovira *et al.* IAE

Tabla 5.2. Parámetros del Controlador PID Calculado según los Distintos Métodos Utilizados

| Parámetro | Método de Cálculo del Controlador PID | | | | | |
|-----------|---------------------------------------|------------------|-----------------------|--------|-------------------|--------------|
| | Rovira | Sung para Servos | Sung para Reguladores | Rivera | Ziegler y Nichols | Cohen y Coon |
| Kc | 2,60 | 18,41 | 21,92 | 20,08 | 3,28 | 3,90 |
| Ti | 1,12 | 1,59 | 0,06 | 1,54 | 0,57 | 0,25 |
| Td | 0,11 | 0,37 | 0,16 | 0,39 | 0,14 | 0,10 |

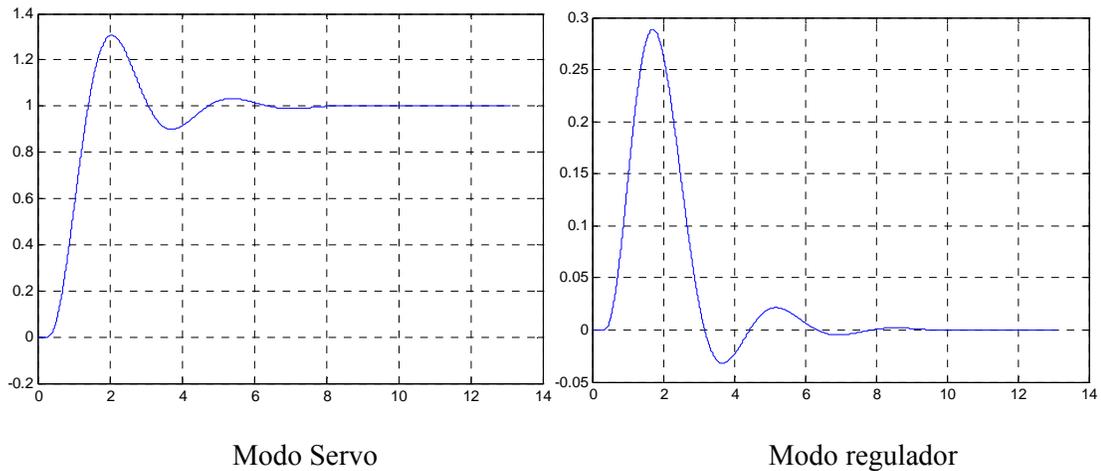


Figura 5.6. Desempeño del Sistema con el Controlador Calculado por el método de Rovira

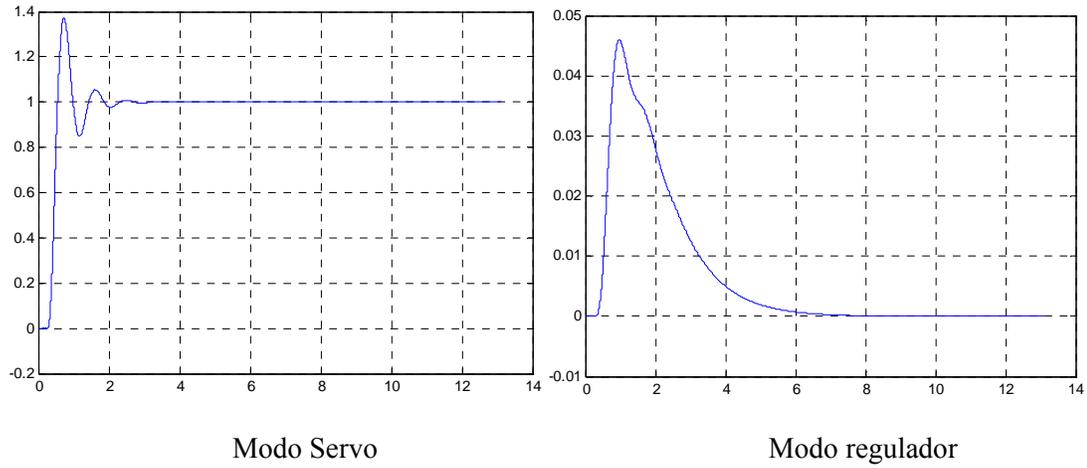


Figura 5.7. Desempeño del Sistema con el Controlador Calculado por el método de Sung para Servos

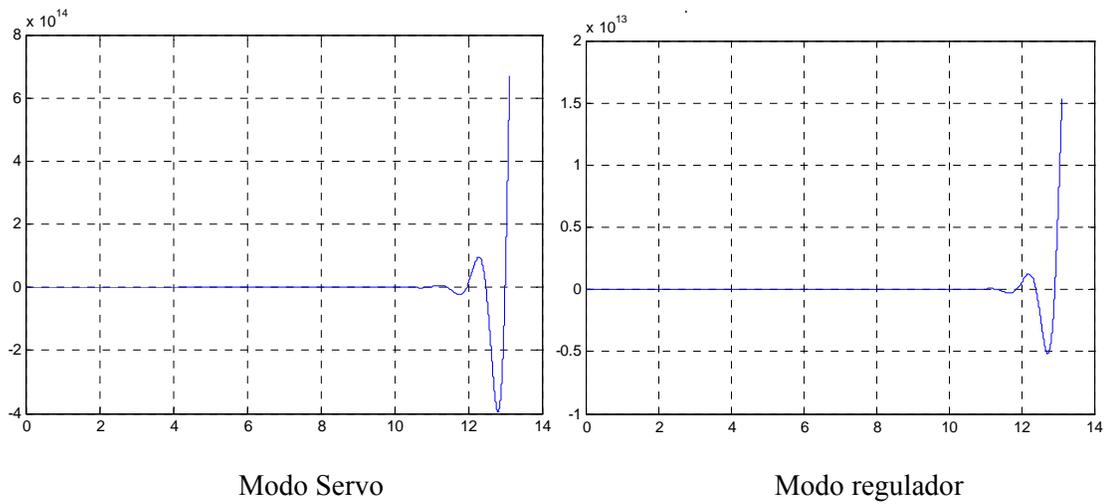


Figura 5.8. Desempeño del Sistema con el Controlador Calculado por el método de Sung para Reguladores

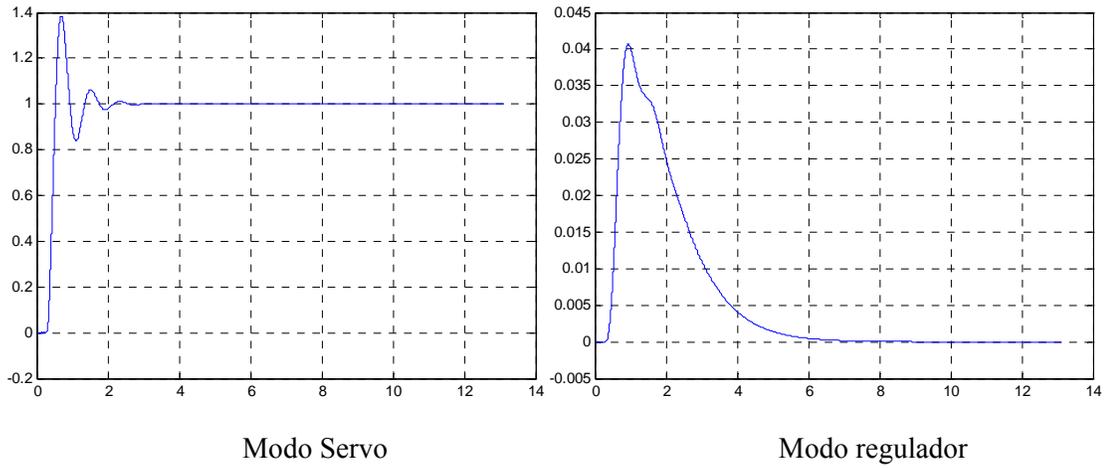


Figura 5.9. Desempeño del Sistema con el Controlador Calculado por el método de Rivera

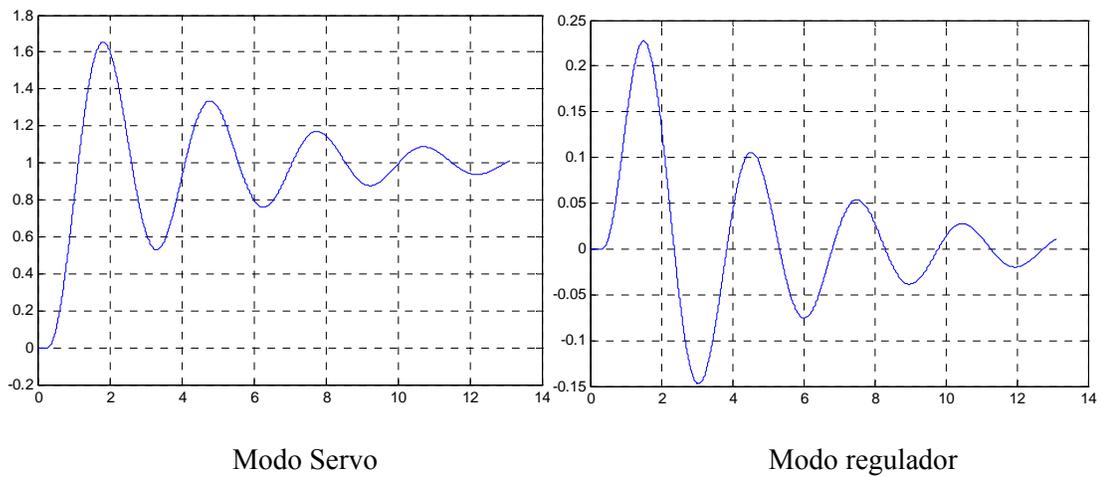


Figura 5.10. Desempeño del Sistema con el Controlador Calculado por el método de Ziegler y Nichols

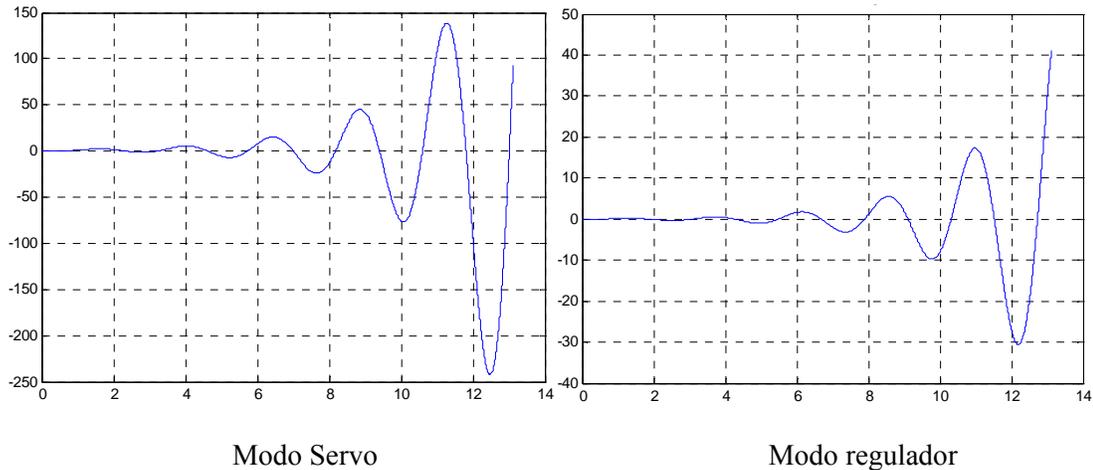


Figura 5.11. Desempeño del Sistema con el Controlador Calculado por el método de Cohen y Coon

Se puede observar que el sistema se comportó inestable cuando se utilizaron los controladores calculados de acuerdo con los métodos de Sung *et al.* para reguladores y Cohen y Coon, y que el desempeño es similar cuando se utilizan los controladores calculados por los métodos de Sung *et al.* para Servos y Rivera *et al.* La tablas 5.3 y 5.4 muestran el resumen del desempeño de los controladores calculados según los métodos de Rovira, Rivera, Sung para Servos, así como también el método de Ziegler y Nichols, tanto en modo servo como en modo regulador.

Tabla 5.3. Desempeño del Sistema Definido por Función de Transferencia, en Modo Servo

| Parámetro | Método de Cálculo del Controlador PID | | | |
|--------------------------|---------------------------------------|------------------|--------|-------------------|
| | Rovira | Sung para Servos | Rivera | Ziegler y Nichols |
| Valor Final | 1 | 1 | 1 | 1 |
| Sobrenivel (%) | 30,60 | 37,28 | 38,02 | 65,49 |
| Tiempo de Estabilización | 5,91 | 2,12 | 2,02 | 12,83 |

Tabla 5.4. Desempeño del Sistema Definido por Función de Transferencia, en Modo Regulador

| Parámetro | Método de Cálculo del Controlador PID | | | |
|--------------------------|---------------------------------------|------------------|--------|-------------------|
| | Rovira | Sung para Servos | Rivera | Ziegler y Nichols |
| Valor Final | 0 | 0 | 0 | 0 |
| Sobrenivel (%) | 28,87 | 4,60 | 4,07 | 22,75 |
| Tiempo de Estabilización | 5,40 | 2,43 | 2,31 | 10,84 |

5.2.4 Ajuste de la Constante Proporcional del controlador PID

Se decidió realizar la Ajuste de la constante proporcional de los controladores calculados según Rovira, Sung (servos), Rivera y Ziegler y Nichols, para que el sobrenivel del sistema no sobrepase el 15 % cuando actúa en modo servo. Para esto, en la sección “Optimización del Controlador PID” se seleccionó la opción “Servo” y se estableció el máximo sobrenivel deseado en 15 %, como se muestra en la figura 5.12.

Las nuevas constantes proporcionales obtenidas se muestra en las tablas 5.5, mientras que en las tablas 5.6 y 5.7 se muestran los indicadores de desempeño: valor final, sobrenivel y tiempo de estabilización.

Tabla 5.5. Parámetros de los Controladores PID Optimizados para Modo Servo, con un sobrenivel máximo del 15 %

| Parámetro | Método de Cálculo del Controlador PID | | | |
|-----------|---------------------------------------|---------------|--------|-------------------|
| | Rovira | Sung (Servos) | Rivera | Ziegler y Nichols |
| Kc | 1,46 | 6,56 | 6,76 | 0,41 |
| Ti | 1,12 | 1,59 | 1,54 | 0,57 |
| Td | 0,11 | 0,37 | 0,39 | 0,14 |

Los parámetros de desempeño del sistema con los controladores optimizados se puede observar en las tablas 5.6 y 5.7.

Tabla 5.6. Desempeño del Sistema Definido por Función de Transferencia, Optimizado para un sobrenivel máximo del 15 % como Servo, Funcionando como Servo

| Parámetro | Método de Cálculo del Controlador PID | | | |
|--------------------------|---------------------------------------|------------------|--------|-------------------|
| | Rovira | Sung para Servos | Rivera | Ziegler y Nichols |
| Valor Final | 1 | 1 | 1 | 1 |
| Sobrenivel (%) | 15,07 | 15,07 | 15,07 | 15,07 |
| Tiempo de Estabilización | 5,59 | 2,13 | 2,03 | 10,20 |

Tabla 5.7. Desempeño del Sistema Definido por Función de Transferencia, Optimizado para un sobrenivel máximo del 15 % como Servo, Funcionando como Regulador

| Parámetro | Método de Cálculo del Controlador PID | | | |
|--------------------------|---------------------------------------|------------------|--------|-------------------|
| | Rovira | Sung para Servos | Rivera | Ziegler y Nichols |
| Valor Final | 0 | 0 | 0 | 0 |
| Sobrenivel (%) | 38,40 | 12,46 | 11,89 | 55,79 |
| Tiempo de Estabilización | 4,09 | 3,79 | 3,67 | 5,43 |

Comparando los resultados mostrados en las tablas 5.3 y 5.6 se puede observar que en el caso de los controladores calculados por Sung *et al.* para Servos y Rivera *et al.* el tiempo de estabilización no sufrió variaciones significativas antes y después de la optimización. En el caso del controlador calculado según Rovira *et al.* el tiempo de estabilización disminuyó en un 5,41 %, mientras que en el caso del controlador calculado según Ziegler y Nichols la disminución fue de un 20,5 %.

En el caso de la función como regulador, los controladores calculados según Rovira *et al.* y Ziegler y Nichols mostraron disminuciones en el tiempo de

estabilización del 24,26 % y 49,91 % respectivamente. Los controladores calculados según Sung *et al.* para Servos y según Rivera *et al.* incrementaron el tiempo de estabilización del sistema en 55,97 % y 25,49 % respectivamente. Las figuras 5.12 a 5.15 muestran la comparación de desempeño del sistema con los distintos controladores calculados y optimizados, antes y después de la optimización.

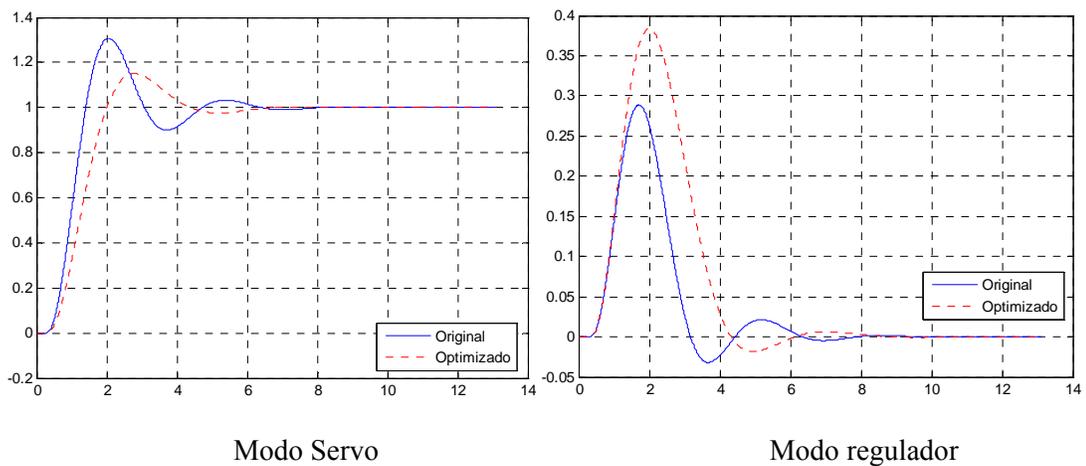


Figura 5.12. Sistema Definido por Función de Transferencia, Controlador Calculado según Rovira, Optimizado para Sobrenivel Máximo del 15 % como Servomecanismo

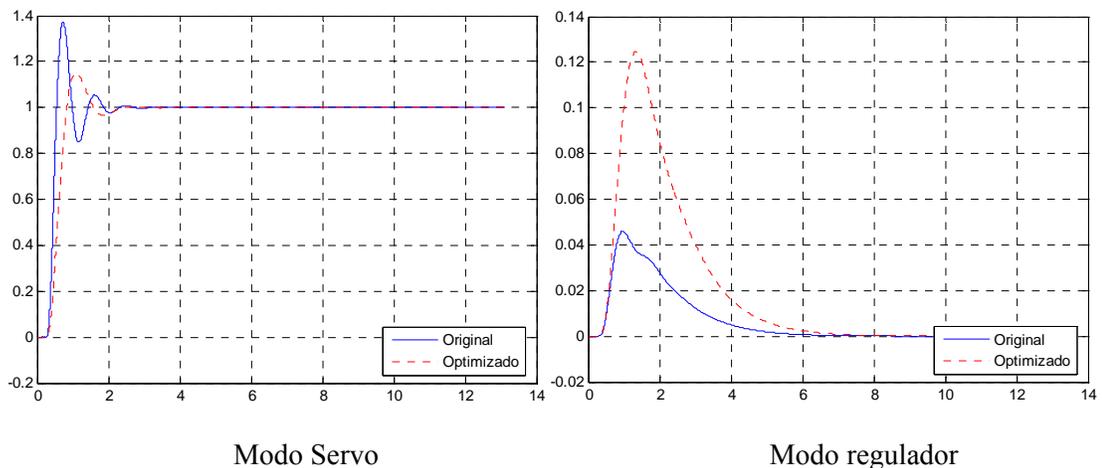


Figura 5.13. Sistema Definido por Función de Transferencia, Controlador Calculado según Sung (Servos), Optimizado para Sobrenivel Máximo del 15 % como Servomecanismo

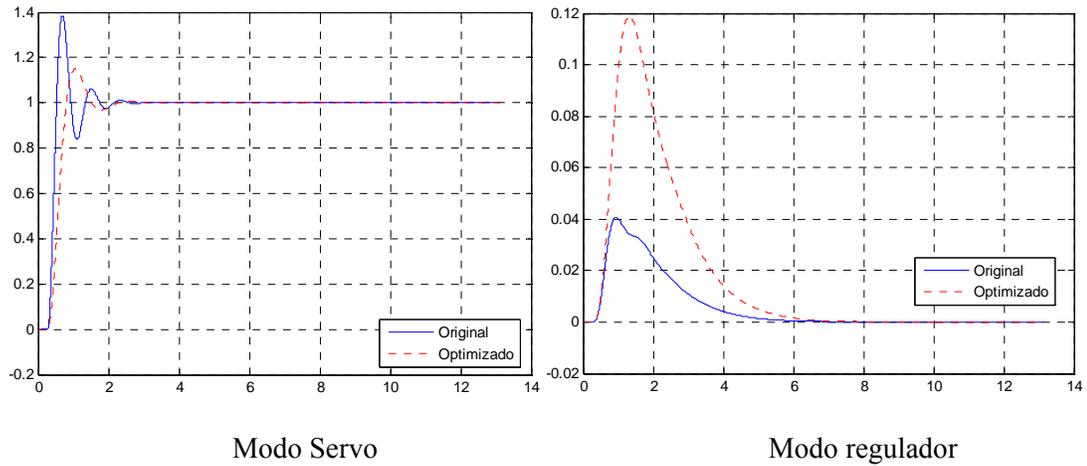


Figura 5.14. Sistema Definido por Función de Transferencia, Controlador Calculado según Rivera, Optimizado Sobrenivel Máximo del 15 % como Servomecanismo

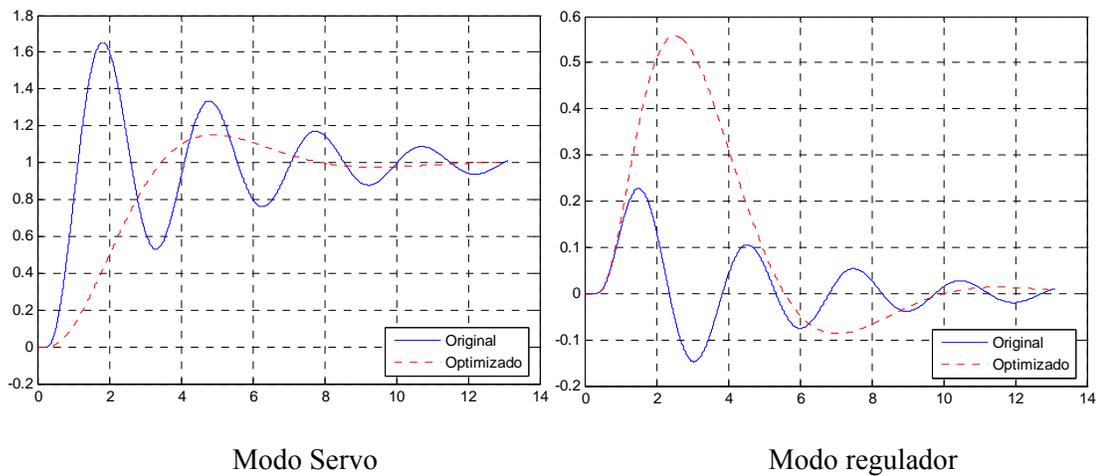


Figura 5.15. Sistema Definido por Función de Transferencia, Controlador Calculado según Ziegler y Nichols, Optimizado para Sobrenivel Máximo del 15 % como Servomecanismo

Se optimización nuevamente los controladores, ahora como reguladores y un sobrenivel máximo del 2 %. Para esto se seleccionó la función “Regulador” y se establece el máximo sobrenivel en 2 %. Las figuras 5.16 a 5.19 muestran el desempeño del sistema antes y después de su optimización.

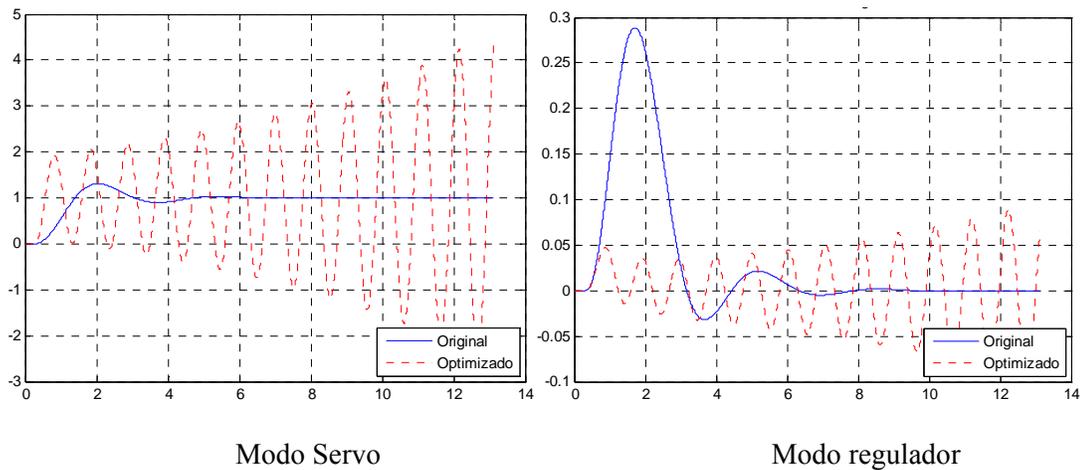


Figura 5.16. Sistema Definido por Función de Transferencia, Controlador según Rovira, Optimizado para Sobrenivel Máximo del 2 % como Regulador

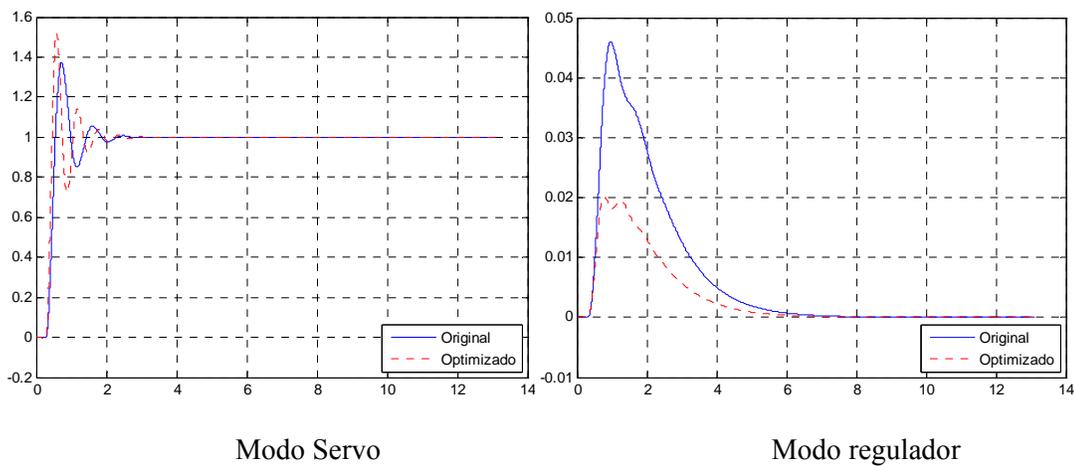


Figura 5.17. Sistema Definido por Función de Transferencia, Controlador Calculado según Sung (Servos), Optimizado para Sobrenivel Máximo del 2 % como Regulador

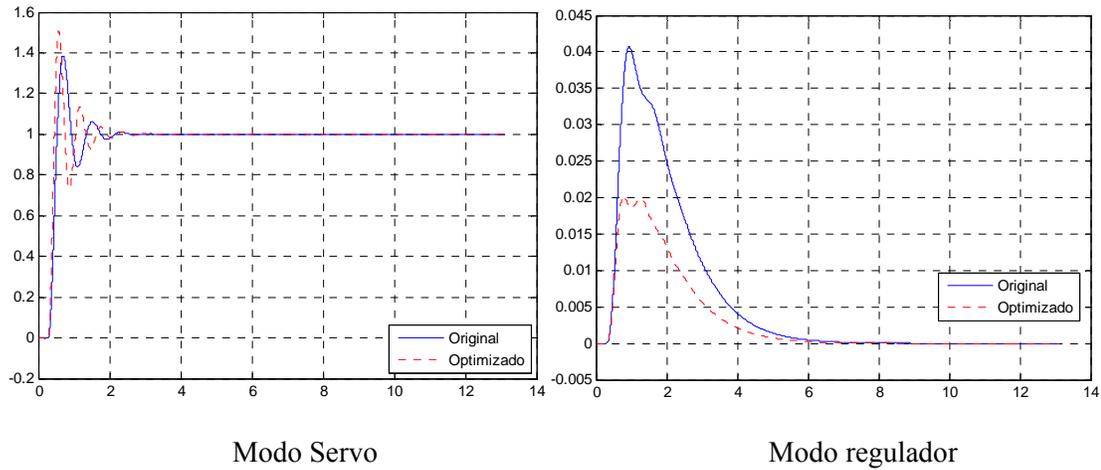


Figura 5.18. Sistema Definido por Función de Transferencia, Controlador Calculado según Rivera, Optimizado para Sobrenivel Máximo del 2 % como Regulador

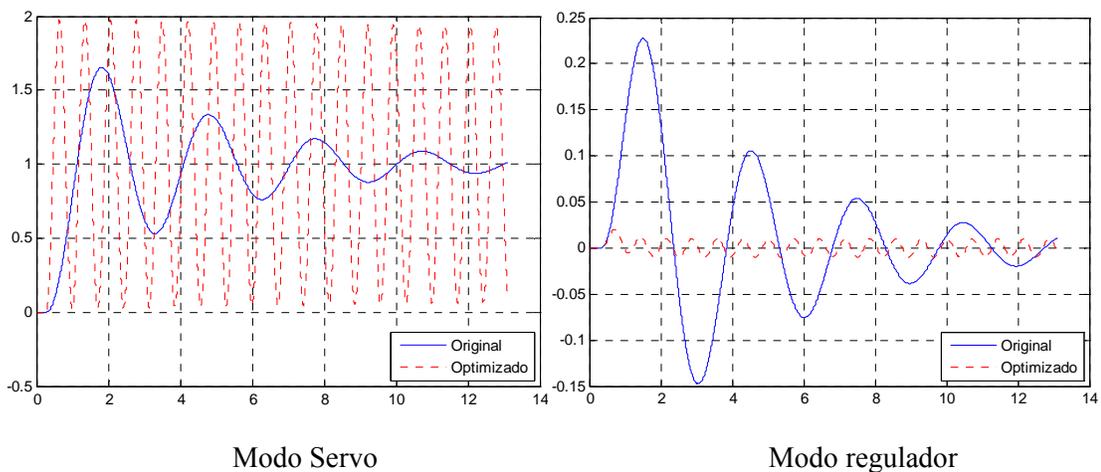


Figura 5.19. Sistema Definido por Función de Transferencia, Controlador Calculado según Ziegler y Nichols, Optimizado para Sobrenivel Máximo del 2 % como Regulador

En las figuras 5.16 a 5.19 muestran que el sistema cumple con el requerimiento de sobrenivel máximo del 2 % cuando se utilizan los controladores calculados por los métodos de Sung *et al.* (Servos) y Rovira *et al.* Cuando se utilizan los controladores calculados por los métodos de Rovira *et al.* y Ziegler y Nichols, el sistema se vuelve inestable en el caso del controlador calculado según Rovira *et al.*, mientras que con el

controlador calculado según Ziegler y Nichols el sistema presenta oscilaciones alrededor del punto de control durante todo el tiempo de control. La tabla 5.8 muestra los parámetros optimizados para los controladores calculados según los métodos de Sung *et al.* para Servos y Rivera *et al.*

Tabla 5.8. Parámetros de los Controladores PID Optimizados para Modo Regulador

| Parámetro | Método de Cálculo del Controlador | |
|-----------|-----------------------------------|--------|
| | Sung para Servos | Rivera |
| Kc | 36,85 | 37,46 |
| Ti | 1,59 | 1,54 |
| Td | 0,37 | 0,39 |

Por su parte, las tablas 5.9 y 5.10 muestran los parámetros de desempeño en modo servo y modo regulador.

Tabla 5.9. Desempeño del Sistema Definido por Función de Transferencia, Optimizado para Modo Regulador, Funcionando como Servo, con un sobrenivel máximo del 2 %

| Parámetro | Método de Cálculo del Controlador | |
|--------------------------|-----------------------------------|--------|
| | Sung para Servos | Rivera |
| Valor Final | 1 | 1 |
| Sobrenivel (%) | 51,83 | 50,90 |
| Tiempo de Estabilización | 2,11 | 1,87 |

Tabla 5.10. Desempeño del Sistema Definido por Función de Transferencia, Optimizado para Modo Regulador, Funcionando como Regulador, con un sobrenivel máximo del 2 %

| Parámetro | Método de Cálculo del Controlador | |
|--------------------------|-----------------------------------|--------|
| | Sung para Servos | Rivera |
| Valor Final | 0 | 0 |
| Sobrenivel (%) | 2,01 | 2,01 |
| Tiempo de Estabilización | 0,81 | 0,81 |

En el caso del controlador calculado según Rovira *et al.* sólo se pudo lograr un desempeño estable cuando se fijó el sobrenivel máximo en un 7 %, aunque el tiempo de estabilización fue mucho mayor que el sistema con el controlador sin optimizar.

Para mantener un tiempo de estabilización similar al del sistema con el controlador sin optimizar fue necesario aumentar el sobrenivel máximo al 15 %. En el caso del controlador calculado según Ziegler y Nichols, el sistema dejó de ser inestable cuando se optimizó para un sobrenivel máximo del 5 %, aunque el tiempo de estabilización fue mucho más largo que con el controlador sin optimizar. Se logró un tiempo de estabilización similar al del sistema sin optimizar cuando se fijó el sobrenivel máximo en un 20 %,

5.3 PRUEBA CON CURVA DE REACCIÓN

5.3.1 Planta de Prueba

Para la prueba del programa se utilizó la planta mostrada en la figura 5.20, tomada de la página 359, problema 7-17, de Smith y Corripio [14]. En este sistema, la temperatura del aire se regula mediante el flujo de combustible al calentador. Se realizó una variación del 5% en la salida del controlador de flujo de gas y se obtuvo una curva de reacción cuyos datos se muestran en la tabla 5.11. Se utilizará el

programa CYRCPID para calcular los ajustes del controlador de flujo de combustible.

Tabla 5.11. Datos de la Curva de Reacción de la Planta de Prueba

| Tiempo (minutos) | Temperatura °F | Tiempo (minutos) | Temperatura °F |
|------------------|----------------|------------------|----------------|
| 0,0 | 425,0 | 6,0 | 437,6 |
| 0,5 | 425,0 | 7,0 | 439,4 |
| 1,0 | 425,0 | 8,0 | 440,7 |
| 2,0 | 425,0 | 9,0 | 441,7 |
| 2,5 | 426,4 | 10,0 | 442,5 |
| 3,0 | 428,5 | 11,0 | 443,0 |
| 3,5 | 430,6 | 12,0 | 443,5 |
| 4,0 | 432,4 | 14,0 | 444,1 |
| 4,5 | 434,0 | 16,0 | 444,5 |
| 5,0 | 435,3 | 19,0 | 445,0 |
| 5,5 | 436,6 | | |

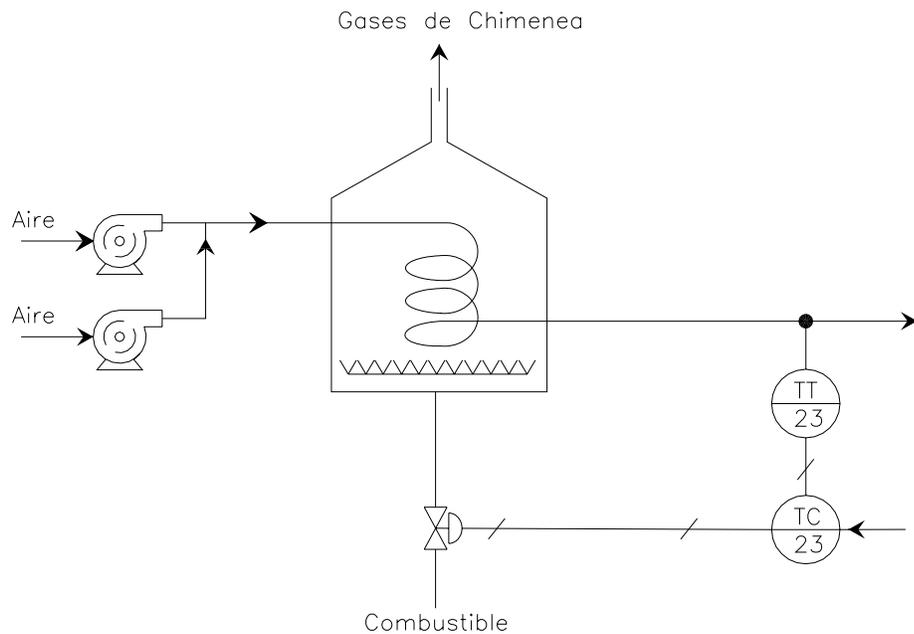


Figura 5.20. Planta de Prueba Definida por su Curva de Reacción

El proceso de cálculo y Ajuste se realiza de acuerdo con la siguiente secuencia:

- Identificar el mejor modelo para la planta

- Calcular los ajustes para un controlador PID.
- Verificar el comportamiento de la planta para los ajustes calculados.
- Ajustar la constante proporcional del controlador, de ser necesario, para que el comportamiento de la planta se ajuste al requerido.

5.3.2 Identificación del Modelo de la Planta

En la Sección “Definición de la Planta” se selecciona la opción “Desde Archivo de Datos”. Esto permite seleccionar el archivo donde se guardan los datos de la curva de reacción. Una vez leído el archivo, aparece el mensaje “Sistema Definido”. La curva de reacción del archivo se muestra en la figura 5.22, conjuntamente con la curva interpolada de acuerdo con lo establecido en la sección 3.1.

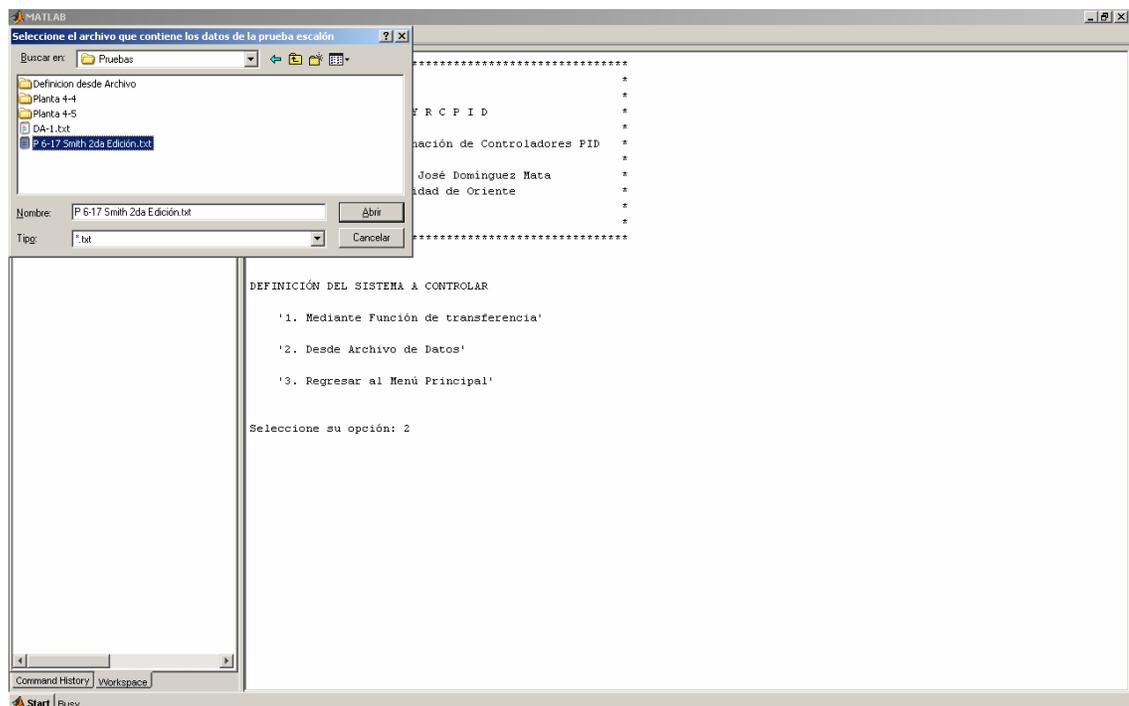


Figura 5.21. Definición de la Planta de Prueba desde Archivo de Datos de Curva de Reacción

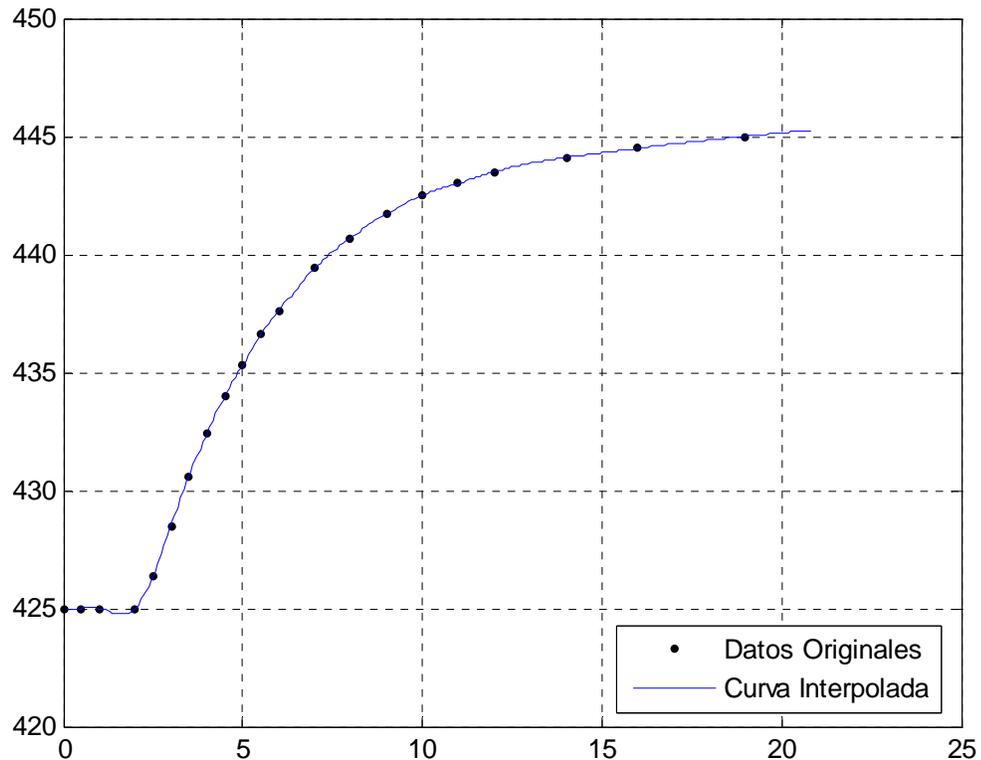


Figura 5.22. Curva de Reacción de la Planta de Prueba Definida desde un Archivo de Datos

Con la opción “Cálculo del Modelo” se selecciona el método a utilizar para modelar la planta. La opción “Selección Automática” permite, como se dijo en la sección 3.3, dejar que el programa calcule todos los modelos disponibles y seleccione el que presente menor Error de Predicción (EP).

Para la planta de prueba seleccionada, se calcularon todos los modelos, con excepción del modelo subamortiguado, obteniéndose los resultados mostrados en la tabla 5.10, y en las figuras 5.23 a 5.26.

El modelo de segundo orden más tiempo muerto de Jahanmiri y Fallahi presentó el menor error de predicción, por lo que seleccionó este modelo para representar la planta de prueba. Sin embargo los modelos de primer orden más tiempo

muerto de Smith y Ho *et al.* Presentaron porcentajes de error de predicción muy similares al modelo de Jahanmiri y Fallahi, por lo que también pudieran ser utilizados para representar la planta de prueba.

Tabla 5.10. Parámetros de los Modelos Calculados para la Planta de Prueba

| Parámetros | Modelos | | | |
|--------------------------------|---------|--------------------------|--------------------------|---------------------|
| | Smith | Ho <i>et al.</i> Orden 1 | Ho <i>et al.</i> Orden 2 | Jahanmiri y Fallahi |
| Ganancia (K) | 4,06 | 4,06 | 4,06 | 4,06 |
| Constante de Tiempo (τ) | 3,94 | 3,76 | 2,6 | 0,142 |
| Tiempo Muerto (t_m) | 2,14 | 2,31 | 0,68 | 2,18 |
| Error de Predicción | 2,192 | 2,999 | 7,360 | 2,003 |

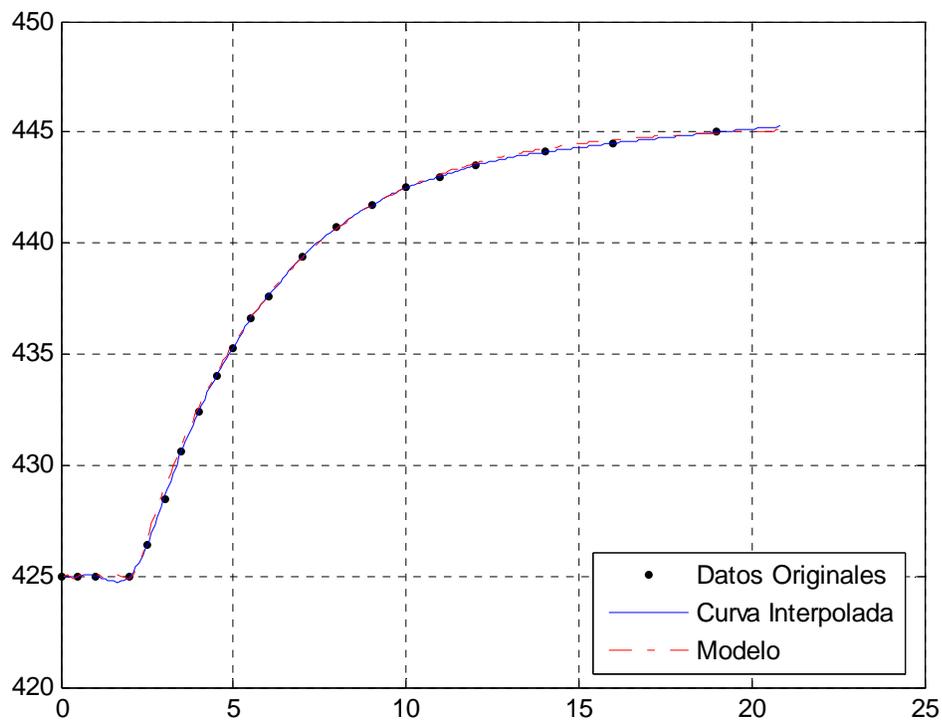


Figura 5.23. Comparación entre la Curva de Reacción de la Planta de Prueba y el Modelo de Primer Orden más Tiempo Muerto según Smith

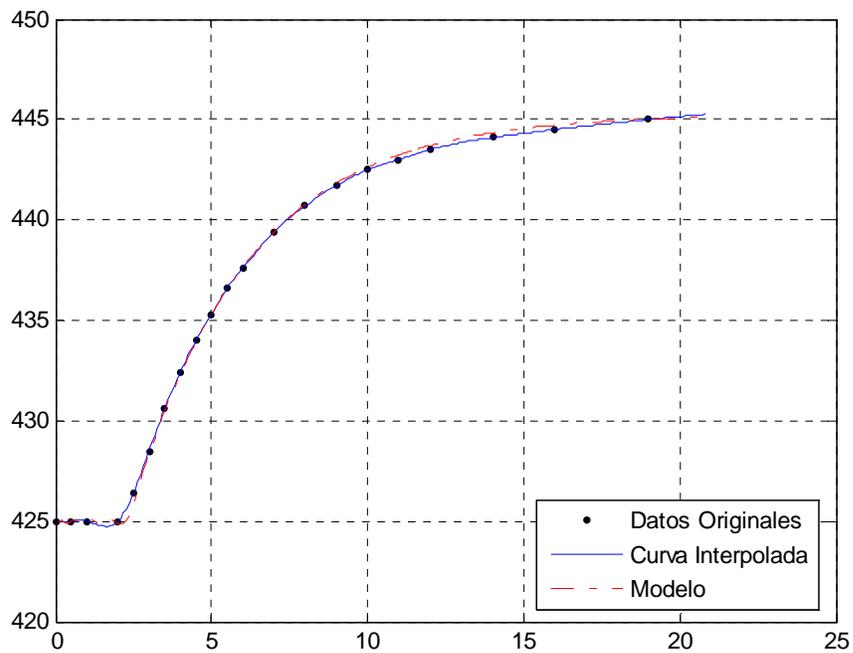


Figura 5.24. Comparación entre la Curva de Reacción de la Planta de Prueba y el Modelo de Primer Orden más Tiempo Muerto según Ho *et al.*

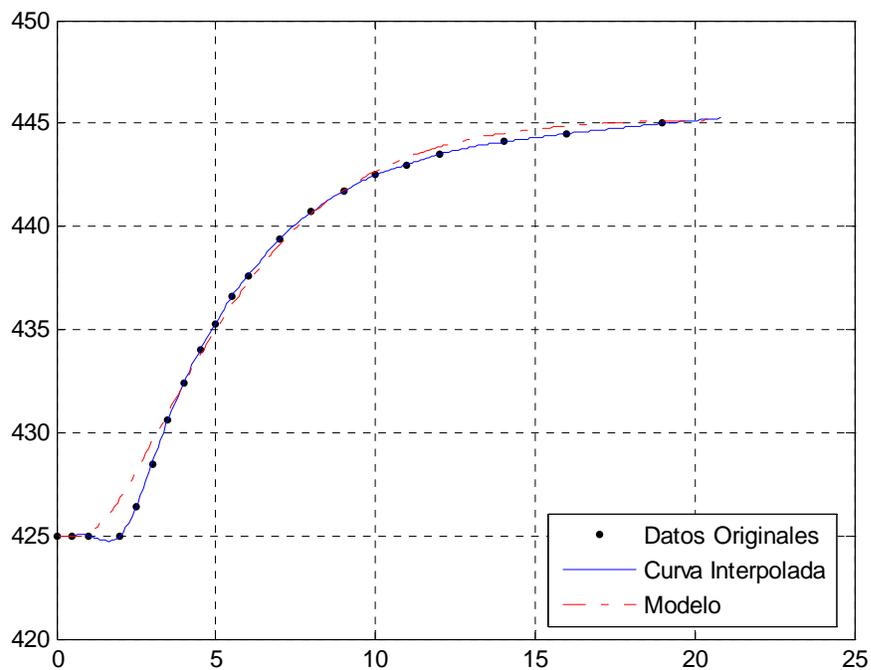


Figura 5.25. Comparación entre la Curva de Reacción de la Planta de Prueba y el Modelo de Segundo Orden más Tiempo Muerto según Ho *et al.*

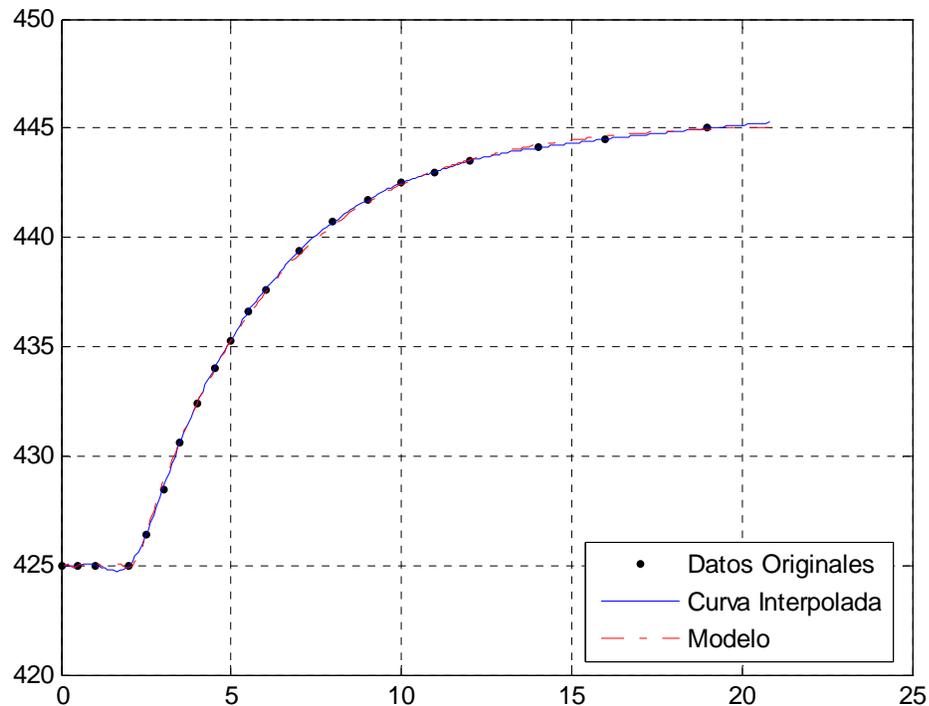


Figura 5.26. Comparación entre la Curva de Reacción de la Planta de Prueba y el Modelo de Segundo Orden más Tiempo Muerto según Jahanmiri y Fallahi

5.3.3 Cálculo del controlador PID

Al igual que en la sección 5.1.3, se calculó el controlador PID por todos los métodos disponibles. Las figuras 5.27 a 5.32 muestran el desempeño del sistema con cada calculador calculado, tanto en modo servo como en modo regulador. Se puede observar que el sistema se comportó de forma inestable cuando se utilizaron los controladores calculados por los métodos de Rovira, Sung para Reguladores y Ziegler y Nichols. La tabla 5.11 muestra los valores calculados para un controlador PID por los métodos de Sung *et al.* para Servos, Rivera *et al.* y Cohen y Coon, mientras que las tablas 5.12 y 5.13 muestran los parámetros de desempeño del sistema cuando se utilizaron dichos controladores.

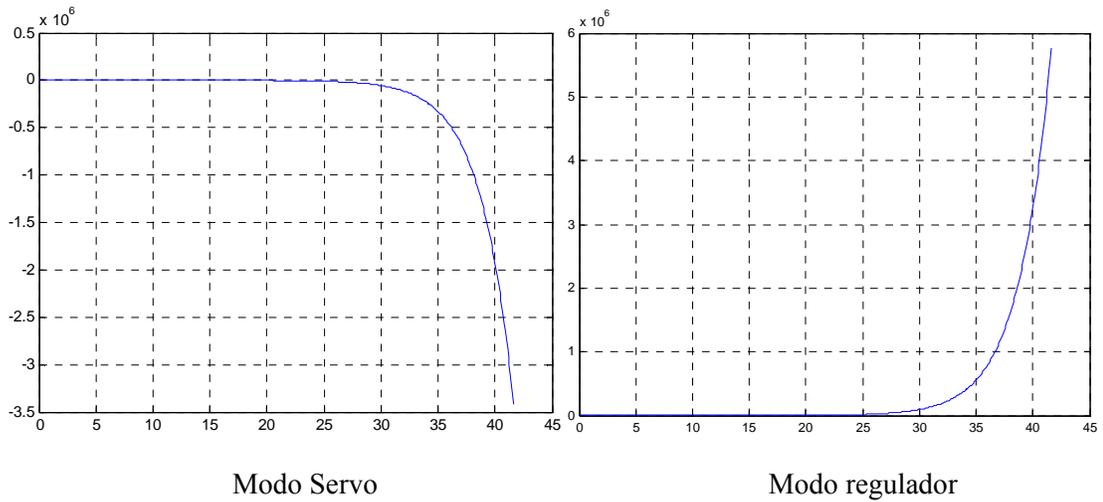


Figura 5.27. Desempeño del Sistema Definido por Curva de Reacción, Controlador
Calculado según Rovira

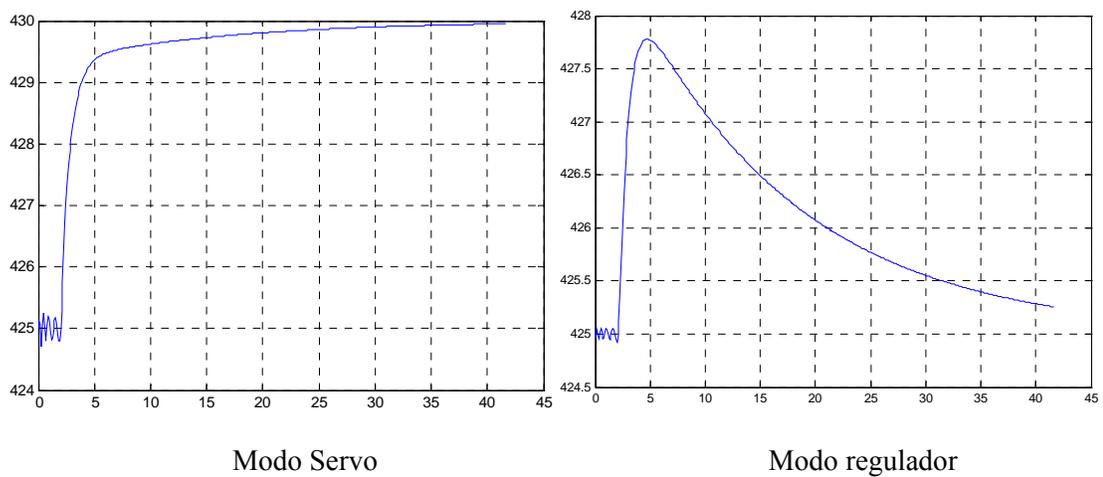


Figura 5.28. Desempeño del Sistema Definido por Curva de Reacción, Controlador
Calculado según Sung (Servos)

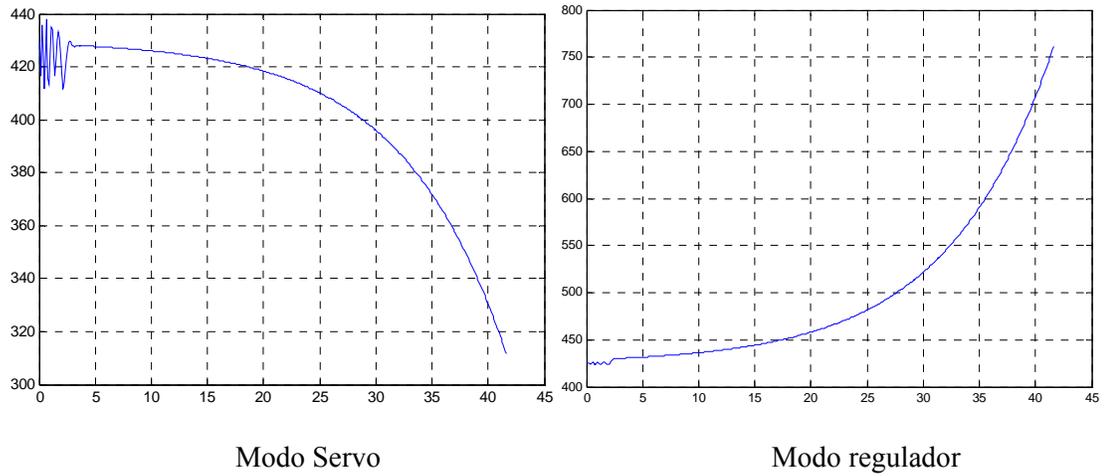


Figura 5.29. Desempeño del Sistema Definido por Curva de Reacción, Controlador
Calculado según Sung (Reguladores)

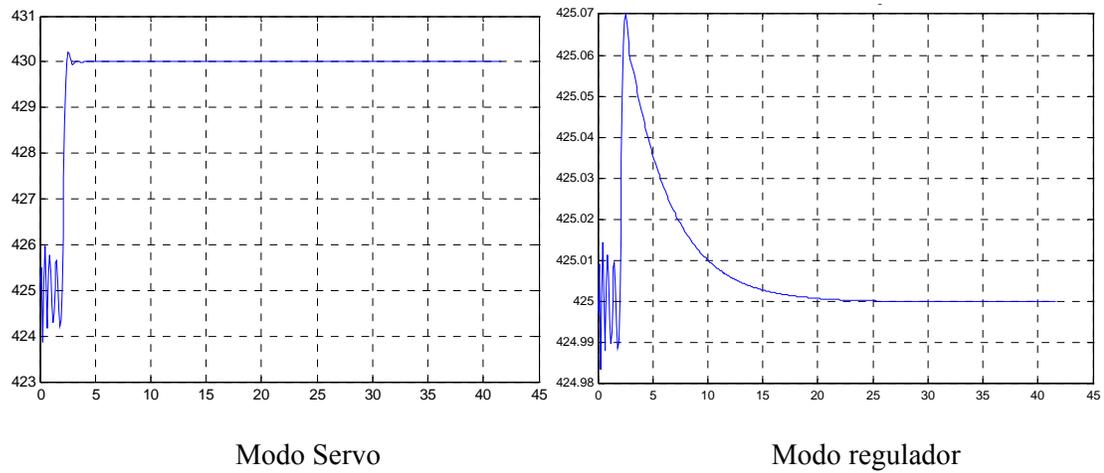


Figura 5.30. Desempeño del Sistema Definido por Curva de Reacción, Controlador
Calculado según Rivera

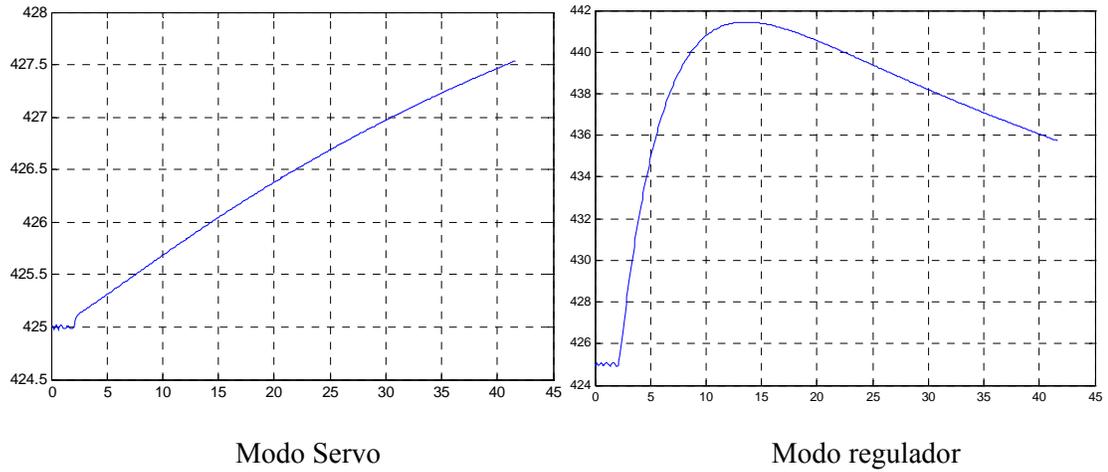


Figura 5.31. Desempeño del Sistema definido por Curva de Reacción, Controlador Calculado según Ziegler y Nichols

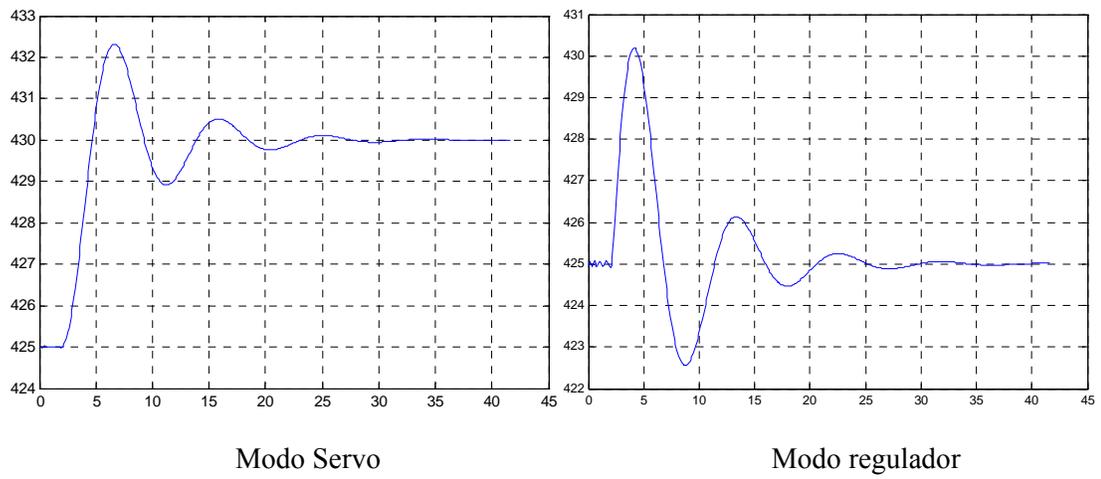


Figura 5.32. Desempeño del Sistema Definido por Curva de Reacción, Controlador Calculado según Cohen y Coon.

Tabla 5.11. Parámetros del Controlador PID Calculado según los Distintos Métodos Utilizados, para el Sistema Definido por Curva de Reacción

| Parámetro | Método de Cálculo del Controlador | | | | | |
|-----------|-----------------------------------|------------------|-----------------------|--------|-------------------|--------------|
| | Rovira | Sung para Servos | Sung para Reguladores | Rivera | Ziegler y Nichols | Cohen y Coon |
| Kc | 0,02 | 1.36 | 0,77 | 68,72 | 0,02 | 0,08 |
| Ti | -0,11 | 13,50 | -7,00 | 3,96 | 4,36 | 0,17 |
| Td | 0,60 | 0,22 | -1,20 | 0,01 | 0,09 | 0,21 |

Tabla 5.12. Desempeño del Sistema Definido por Curva de Reacción, en Modo Servo

| Parámetro | Método de Cálculo del Controlador | | |
|--------------------------|-----------------------------------|--------|--------------|
| | Sung para Servos | Rivera | Cohen y Coon |
| Valor Final | 430 | 430 | 430 |
| Sobrenivel (%) | 0 | 0,05 | 0,54 |
| Tiempo de Estabilización | 30,04 | 2,62 | 25,70 |

Tabla 5.13. Desempeño del Sistema Definido por Curva de Reacción, en Modo Regulador

| Parámetro | Método de Cálculo del Controlador | | |
|--------------------------|-----------------------------------|--------|--------------|
| | Sung para Servos | Rivera | Cohen y Coon |
| Valor Final | 425 | 425 | 425 |
| Sobrenivel (%) | 0,65 | 0,02 | 1,22 |
| Tiempo de Estabilización | 0 | 0 | 0 |

5.3.4 Ajuste de la Constante Proporcional del controlador PID

Se decidió ajustar la constante proporcional para que el sistema tuviera un sobrenivel del 0,3% ante una entrada escalón, actuando como regulador. Para esto, y siguiendo el mismo procedimiento de la sección 5.2.4, se seleccionó como función “Regulador” y se estableció el máximo porcentaje de sobrenivel en 0,3%. La Ajuste se realizó para los controladores calculados según Sung *et al.* y según Cohen y Coon. Los parámetros de los controladores optimizados se muestran en la tabla 5.14, mientras que los desempeños de la planta con los controladores optimizados se muestran en las tablas 5.15 y 5.16, y en las figuras 5.33 y 5.34.

Tabla 5.14. Parámetros de los Controladores PID para la Planta Definida por Curva de Reacción, Optimizados para un Sobrenivel Máximo del 0,3 % como Regulador

| Parámetro | Método de Cálculo del Controlador | |
|-----------|-----------------------------------|--------------|
| | Sung para Servos | Cohen y Coon |
| Kc | 3,33 | 1,13 |
| Ti | 13,50 | 0,17 |
| Td | 0,22 | 0,21 |

Tabla 5.15. Desempeño del Sistema Optimizado para un Sobrenivel Máximo del 0,3 % como Regulador, Funcionando como Servomecanismo

| Parámetro | Método de Cálculo del Controlador | |
|--------------------------|-----------------------------------|--------------|
| | Sung para Servos | Cohen y Coon |
| Sobrenivel | 0,18% | 0,50% |
| Valor Final | 430,00 | 425,00 |
| Tiempo de Estabilización | 41,62 | 41,62 |

Tabla 5.16. Desempeño del Sistema Optimizado para un Sobrenivel Máximo del 0,3 % como Regulador, Funcionando como Regulador

| Parámetro | Método de Cálculo del Controlador | |
|--------------------------|-----------------------------------|----------------|
| | Sung para Servos | Cohen y Coon r |
| Sobrenivel | 0,57% | 0,50% |
| Valor Final | 430,00 | 425,00 |
| Tiempo de Estabilización | 41,62 | 41,62 |

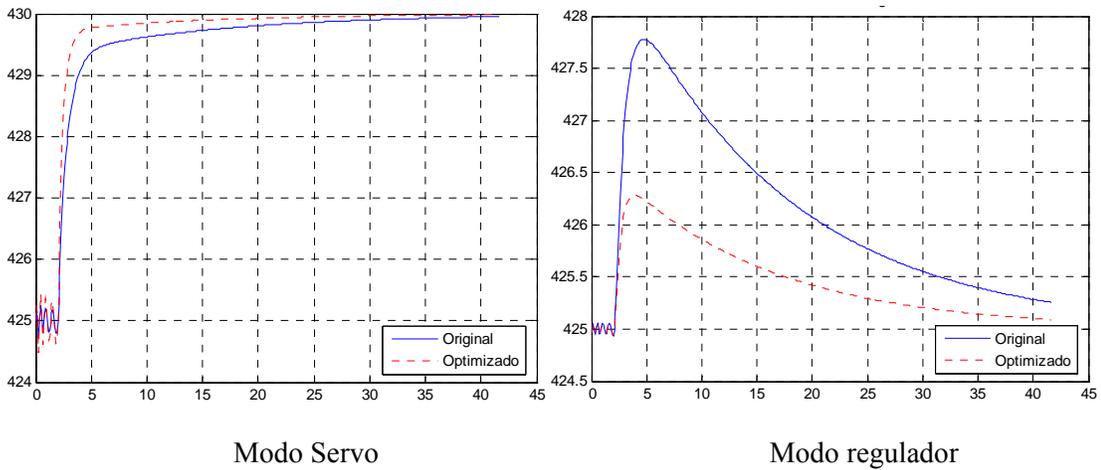


Figura 5.33. Sistema Definido por Curva de Reacción, Controlador Calculado según Sung (Servos), Optimizado para Sobrenivel Máximo del 0,3 % como Regulador

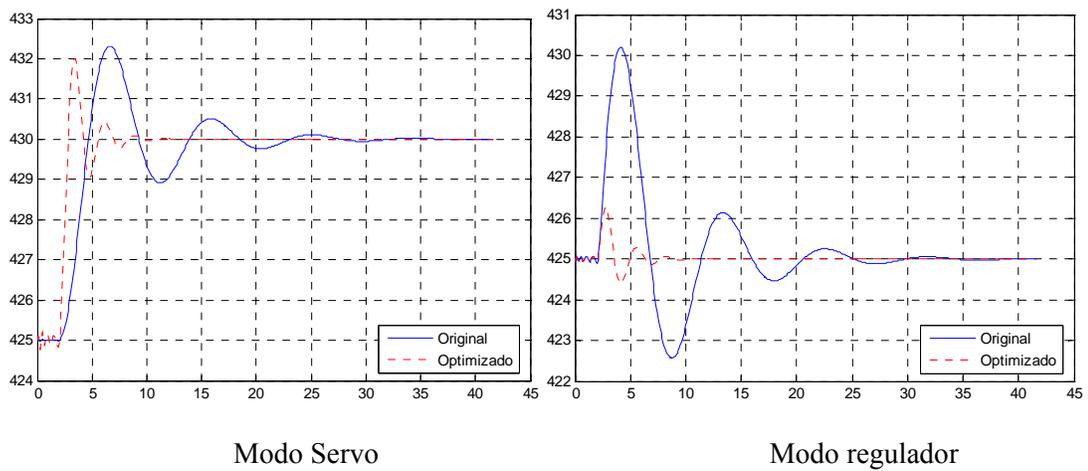


Figura 5.34. Definido por Curva de Reacción, Controlador Calculado según Cohen y Coon, Optimizado para Sobrenivel Máximo del 0,3 % como Regulador

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

El programa desarrollado permite el cálculo del mejor modelo de primer o de segundo orden para plantas, ya sean éstas definidas mediante su función de transferencia o mediante la curva de reacción generada al aplicar una entrada en forma de escalón.

El uso de la interpolación cúbica, cuando se define la planta mediante los datos de una curva de reacción generada al aplicar una entrada en forma de escalón unitario, permite generar una muy buena aproximación a la respuesta de la planta. Esto a la vez hace que se pueda calcular un mejor modelo del sistema a controlar.

Los modelos de primer orden más tiempo muerto y segundo orden más tiempo muerto de Ho *et al.* tienen limitaciones relacionadas con el tiempo de respuesta de la planta a la cual se intenten aplicar. Para que puedan ser aplicados a una planta en particular, debe cumplirse que $t_{35} \geq 0,223t_{85}$ para el caso del modelo de primer orden más tiempo muerto, y $t_{35} \geq 0,365t_{85}$ para el modelo de segundo orden más tiempo muerto.

Los controladores calculados por los métodos de Ziegler y Nichols, así como también los calculados por el método de Cohen y Coon, tienden a presentar oscilaciones más pronunciadas y prolongadas que los calculados por los otros métodos disponibles en el programa.

Los métodos para el cálculo de controladores PID desarrollados por Rovira *et al.* presentan limitaciones en su aplicabilidad, dadas por la relación entre el tiempo

muerto y la constante de tiempo del modelo. Para el caso del método basado en el criterio IAE, debe cumplirse que $\frac{t_m}{\tau} < 5,6923$, mientras que si desea utilizar el método de cálculo basado en el criterio ITAE se debe cumplir que $\frac{t_m}{\tau} < 5,249$.

De acuerdo con los resultados obtenidos en las simulaciones realizadas, no siempre es posible ajustar la constante proporcional de un controlador PID para que el sistema a controlar cumpla con el máximo sobrenivel deseado. En estos casos se requiere un compromiso entre el desempeño deseado y la estabilidad del sistema.

RECOMENDACIONES

Dado que este programa se limitó solamente a modificar la constante proporcional de un controlador PID, se recomienda desarrollar módulos que permitan el ajuste de los tiempos de integración y derivación.

El programa fue desarrollado en MatLab®, el cual es un software costoso. En aras de hacerlo más accesible se recomienda desarrollarlo en un ambiente de software libre, tal como el SciLab, por ejemplo.

BIBLIOGRAFÍA

- [1] J.G. ZIEGLER and N. B. NICHOLS (1942). *OPTIMUM SETTINGS FOR AUTOMATIC CONTROLLERS*. Transactions of the A.S.M.E., November, 1942, pp. 759-765.

- [2] Víctor M. Alfaro Ruiz (2002). *MÉTODOS DE SINTONIZACIÓN DE CONTROLADORES PID QUE OPERAN COMO REGULADORES*. Revista Ingeniería, volumen 12, San José, Costa Rica, pp. 21-36.

- [3] Víctor M. Alfaro Ruiz (2003). *MÉTODOS DE SINTONIZACIÓN DE CONTROLADORES PID QUE OPERAN COMO SERVOMECANISMOS*. Revista Ingeniería, volumen 13, San José, Costa Rica, pp. 13-29.

- [4] J.C. Basilio y S. R. Matos (2002). *DESIGN OF PI AND PID CONTROLLERS WITH TRANSIENT PERFORMANCE SPECIFICATION*. IEEE Transactions on Education, Volumen 45, Número 4, Noviembre de 2002, pp 364-370.

- [5] Víctor M. Alfaro Ruiz (2001). *IDENTIFICACIÓN DE PROCESOS SOBREAMORTIGUADOS UTILIZANDO TÉCNICAS DE LAZO ABIERTO*. Revista Ingeniería, volumen 11, San José, Costa Rica, pp. 11-25.

- [6] Ján Mikleš y Miroslav Fikar. *PROCESS MODELLING, IDENTIFICATION, AND CONTROL*. Berlín, Springer, 2002, pp. 225-230.

- [7] Víctor M. Alfaro Ruiz (2005). *ACTUALIZACIÓN DEL MÉTODO DE SINTONIZACIÓN DE CONTROLADORES DE ZIEGLER Y NICHOLS*. Revista Ingeniería, volumen 15, San José, Costa Rica, pp. 39-52.
- [8] B. Martín del Brío y A. Sanz Molina. *REDES NEURONALES Y SISTEMAS DIFUSOS*. 2da edición. Madrid, RA-MA Editorial, 2002
- [9] The Mathworks. *FUZZY LOGIC TOOLBOX FOR USE WITH MATLAB*, version 2, 1998.
- [12] Jerry M. Mendel (1995). *FUZZY LOGIC SYSTEMS FOR ENGINEERING: A TUTORIAL*. Proceedings of the IEEE, Vol. 83, No. 3, pp 345-377.
- [13] F. Martin McNeill, Ellen Thro. *FUZZY LOGIC: A PRACTICAL APPROACH*. Chestnut Hill, Academic Press, Inc., 1994.
- [14] C. Smith, A. Corripio. *PRINCIPLES AND PRACTICE OF AUTOMATIC PROCESS CONTROL*. 2° Ed., Wiley, 1997. pp 359-360.
- [15] Y. Li, K. Ang, G. Chong (2006). *PID CONTROL SYSTEMS ANALYSIS AND DESIGN*. IEEE Control Systems Magazine, Vol. 26, No. 1. pp.32-41
- [16] M. Heraoui (2006). *LABORATORIO VIRTUAL DE SISTEMAS DE CONTROL REALIMENTADOS*. Universidad de Oriente, Barcelona.

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

| | |
|------------------|--|
| TÍTULO | OPTIMIZACIÓN DE UN CONTROLADOR PID POR MÉTODOS CLÁSICOS Y MEDIANTE LÓGICA DIFUSA |
| SUBTÍTULO | |

AUTOR (ES):

| APELLIDOS Y NOMBRES | CÓDIGO CULAC / E MAIL |
|------------------------------|---|
| Domínguez Mata, Ignacio José | CVLAC: 9.275.709 E MAIL: ignaciojdominguez@gmail.com |
| | CVLAC: E MAIL: |
| | CVLAC: E MAIL: |
| | CVLAC: E MAIL: |

PALABRAS O FRASES CLAVES:

Controladores PID

Identificación de Sistemas

Modelos de Primer Orden

Modelos de Segundo Orden

Lógica Difusa

Optimización de Controladores

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

| ÁREA | SUBÁREA |
|---------------------------------|----------------|
| Ingeniería y Ciencias Aplicadas | Automatización |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

RESUMEN (ABSTRACT):

En este trabajo se desarrolla un programa que identifica una planta mediante varios modelos de primer y/o segundo orden y selecciona el que mejor la represente. Los Modelos utilizados son los de Smith, Ho *et al.*, Mikles y Fikar, y Jahanmiri y Fallahi, y se selecciona aquél que presente la menor integral absoluta del error de predicción. La planta puede definirse mediante su función de transferencia o mediante su curva de reacción. También se calculan controladores PID por varios métodos clásicos (Ziegler y Nichols, Cohen y Coon, Rivera *et al.*, Rovira *et al.* y Sung *et al.*), y se ajusta su constante proporcional mediante un algoritmo en lógica difusa para que la respuesta del sistema en lazo cerrado no exceda de un valor establecido. Se concluye que no siempre es posible lograr el máximo sobrenivel deseado. Esto depende del sistema y del método de cálculo del controlador PID.

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

CONTRIBUIDORES:

| APELLIDOS Y NOMBRES | ROL / CÓDIGO CVLAC / E_MAIL | | | | |
|----------------------------|------------------------------------|----------------------------------|-------------|-----------|-------------|
| Heraoui, Margarita | ROL | CA | AS X | TU | JU |
| | CVLAC: | 9.453.103 | | | |
| | E_MAIL | mheraoui@gmail.com | | | |
| | E_MAIL | | | | |
| García, Félix | ROL | CA | AS | TU | JU X |
| | CVLAC: | 3.672.075 | | | |
| | E_MAIL | fpadilla200@gmail.com | | | |
| | E_MAIL | | | | |
| Ramírez, Miguel | ROL | CA | AS | TU | JU X |
| | CVLAC: | 11.150.835 | | | |
| | E_MAIL | ramirezmb@pdvsa.com | | | |
| | E_MAIL | miguel_ramirez02@yahoo.es | | | |
| | ROL | CA | AS | TU | JU |
| | CVLAC: | | | | |
| | E_MAIL | | | | |
| | E_MAIL | | | | |

FECHA DE DISCUSIÓN Y APROBACIÓN:

| | | |
|-------------|------------|------------|
| 2009 | 06 | 27 |
| AÑO | MES | DÍA |

LENGUAJE. SPA

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

ARCHIVO (S):

| NOMBRE DE ARCHIVO | TIPO MIME |
|-----------------------------------|------------------------|
| Optimizacion de Controladores PID | Application/msword |
| Listado Programa | Application/mfcwordpad |
| | |

CARACTERES EN LOS NOMBRES DE LOS ARCHIVOS: A B C D E F G H I
 J K L M N O P Q R S T U V W X Y Z . a b c d e f g h i j k l m n o p q r s t u v w x y z . 0 1 2
 3 4 5 6 7 8 9 .

ALCANCE

ESPACIAL: _____ (OPCIONAL)

TEMPORAL: _____ (OPCIONAL)

TÍTULO O GRADO ASOCIADO CON EL TRABAJO:

_Especialista en Automatización e Informática Industrial _____

NIVEL ASOCIADO CON EL TRABAJO:

_Postgrado _____

ÁREA DE ESTUDIO:

_Ingeniería Eléctrica _____

INSTITUCIÓN:

_Universidad de Oriente. Núcleo de Anzoátegui _____

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

DERECHOS

___ De acuerdo al artículo 44 del reglamento de trabajos de grado

“Los Trabajos de grado son exclusiva propiedad de la Universidad de Oriente y solo podrán ser utilizadas a otros fines con el consentimiento del consejo de núcleo respectivo, quien lo participara al Consejo Universitario”

AUTOR

AUTOR

AUTOR

TUTOR

JURADO

JURADO

POR LA SUBCOMISIÓN DE TESIS