



Universidad
Nacional
de Loja

Universidad Nacional de Loja

Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables

Carrera de Ingeniería Electromecánica

Diseño de un controlador avanzado para el control de nivel en el sistema de tanques interconectados del Laboratorio de Automatización de la CIEM

Trabajo de Titulación previo, a la
obtención del título de Ingeniero
Electromecánico

AUTOR:

Roger Javier Romero Jara

DIRECTOR:

Ing. Julio César Cuenca Tinitana, Mg. Sc.

Loja – Ecuador

2023

Certificación

Loja, 8 de septiembre de 2023

Ing. Julio César Cuenca Tinitana, Mg. Sc.

DIRECTOR DEL TRABAJO DE TITULACIÓN

CERTIFICO:

Que he revisado y orientado todo el proceso de elaboración del Trabajo de Titulación denominado: **Diseño de un controlador avanzado para el control de nivel en el sistema de tanques interconectados del Laboratorio de Automatización de la CIEM**, previo a la obtención del título de **Ingeniero Electromecánico**, de la autoría del estudiante **Roger Javier Romero Jara** con **cédula de identidad Nro. 1104124571**, una vez que el trabajo cumple con todos los requisitos exigidos por la Universidad Nacional de Loja, para el efecto, autorizo la presentación del mismo para su respectiva sustentación y defensa.

Ing. Julio César Cuenca Tinitana, Mg. Sc.

DIRECTOR DEL TRABAJO DE TITULACIÓN

Autoría

Yo, **Roger Javier Romero Jara**, declaro ser el autor del presente Trabajo de Titulación y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos y acciones legales por el contenido del mismo. Adicionalmente, acepto y autorizo a la Universidad Nacional de Loja la publicación de mi Trabajo de Titulación en el Repositorio Institucional – Biblioteca Virtual.

Firma:



Cédula de identidad: 1104124571

Fecha: 10 de noviembre de 2023

Correo electrónico: roger.javi06@gmail.com

Teléfono: 0939009777

Carta de autorización por parte del autor, para consulta, reproducción parcial o total y/o publicación electrónica del texto completo, del Trabajo de Titulación.

Yo, **Roger Javier Romero Jara**, declaro ser autor del Trabajo de Titulación denominado: **Diseño de un controlador avanzado para el control de nivel en el sistema de tanques interconectados del Laboratorio de Automatización de la CIEM**, como requisito para optar por el título de **Ingeniero Electromecánico**; autorizo al sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos muestre la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Institucional.

Los usuarios pueden consultar el contenido de este trabajo en el Repositorio Institucional, en las redes de información del país y del exterior con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia del Trabajo de Titulación que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los diez días del mes de noviembre del dos mil veintitrés.

Firma:



Autor: Roger Javier Romero Jara

Cédula de identidad: 1104124571

Dirección: Catamayo

Correo electrónico: roger.javi06@gmail.com

Teléfono: 0939009777

DATOS COMPLEMENTARIOS:

Director del Trabajo de Titulación: Ing. Julio César Cuenca Tinitana, Mg. Sc.

Dedicatoria

A Dios por haberme permitido llegar hasta aquí con salud y por darme las fuerzas para seguir adelante y no desmayar ante las adversidades de la vida.

A mi madre Katy Jara por sus consejos, su amor, su apoyo incondicional y por haberme forjado como una persona de bien con principios y valores. A mi querida abuela materna por todo el cariño brindado y por sus palabras de ánimo y fortaleza en aquellos momentos difíciles.

Roger Javier Romero Jara

Agradecimiento

A la Universidad Nacional de Loja, carrera de Ingeniería Electromecánica y a toda su planta docente por la formación académica brindada en el ámbito ingenieril, para poder desempeñarme con ética y responsabilidad en la vida profesional.

De manera especial, mi agradecimiento al Ing. Julio Cuenca Tinitana por la guía y el asesoramiento brindado en el desarrollo de este proyecto.

Roger Javier Romero Jara

Índice de contenidos

Portada	i
Certificación	ii
Autoría	iii
Carta de autorización	iv
Dedicatoria	v
Agradecimiento	vi
Índice de contenidos	vii
Índice de tablas	xi
Índice de figuras.....	xii
Índice de anexos.....	xvii
Índice de símbolos y abreviaturas.....	xviii
1. Título	1
2. Resumen	2
Abstract	3
3. Introducción	4
4. Marco teórico	6
4.1 Capítulo 1: Aspectos generales del control de nivel	6
4.1.1 Control de nivel y su importancia en la industria	6
4.1.1.1 Sistema de tanques acoplados.	6
4.1.2 Componentes básicos de un sistema de control.....	9
4.1.2.1 Variables.....	10
4.1.2.2 Elementos.....	10
4.1.2.3 Señales.....	10
4.1.3 Clasificación de los sistemas de control	11
4.1.3.1 Control de lazo abierto.	11
4.1.3.2 Control de lazo cerrado	11
4.1.4 Instrumentación en el control de un sistema hidráulico de tanques acoplados..	12
4.1.4.1 Medidores de nivel.....	12
4.2 Capítulo 2: Modelado matemático e identificación de sistemas	14
4.2.1 Identificación de sistemas	14
4.2.1.1 Diseño experimental.....	15
4.2.1.1.1 Variables de diseño.....	15
4.2.1.1.2 Señal de excitación.	15
4.2.1.1.3 Cantidad de datos.....	15

4.2.1.2	Metodología de la identificación.....	15
4.2.2	System Identification Toolbox de MATLAB®	16
4.2.3	Función de transferencia	17
4.2.3.1	Polos y ceros de una función de transferencia.	18
4.2.3.2	Estabilidad de un sistema.	19
4.2.4	Sistemas de primer orden.....	20
4.2.4.1	Función de transferencia de primer orden.....	21
4.2.5	Sistemas de segundo orden	21
4.2.5.1	Función de transferencia de segundo orden.	22
4.2.6	Retardo o tiempo muerto de un sistema.....	22
4.3	Capítulo 3: Diseño del controlador clásico PID y avanzado MPC	24
4.3.1	Control PID.....	24
4.3.1.1	Acción de control proporcional.....	25
4.3.1.2	Acción de control integral.....	26
4.3.1.3	Acción de control derivativo.....	26
4.3.2	Control Predictivo basado en Modelo (MPC)	27
4.3.2.1	Principio de funcionamiento del control predictivo.....	29
4.3.2.2	Elementos básicos del control predictivo.....	31
4.3.2.2.1	Modelo del proceso.	31
4.3.2.2.2	Modelo de las perturbaciones.....	32
4.3.2.2.3	Función objetivo.....	33
4.3.2.2.4	Obtención de la ley de control.....	34
4.3.3	MPC Toolbox - MATLAB®	35
4.3.3.1	Parámetros del MPC.....	35
4.3.3.1.1	Estructura del MPC y modelo interno de la planta.....	36
4.3.3.1.2	Tiempo de muestreo, horizonte de predicción y horizonte de control.	36
4.3.3.1.3	Restricciones.....	37
4.3.3.1.4	Pesos.....	38
4.3.3.1.5	Robustez y agresividad.....	38
4.3.4	Sintonización y optimización de controladores mediante algoritmos evolutivos.....	39
4.3.4.1	Algoritmo por multidinámicas para optimización global (MAGO).....	40
5.	Metodología.....	42
5.1	Materiales	44
5.2	Descripción del sistema prototipo de dos tanques interconectados	45

5.3	Experimento de identificación y recolección de datos.....	47
5.3.1	Planificación del experimento de identificación.....	47
5.3.1.1	Caso 1: Llenado del tanque 1 con la bomba 1.....	48
5.3.1.2	Caso 2: Llenado de los tanques interconectados 1 y 2 con la bomba 1	48
5.3.1.3	Caso 3: Llenado del tanque 2 con la bomba 2.....	48
5.3.1.4	Caso 4: Llenado de los tanques interconectados 1 y 2 con la bomba 2	48
5.3.2	Recolección de datos.....	48
5.4	Modelado matemático del sistema de tanques interconectados	51
5.5	Diseño del controlador avanzado MPC.....	53
5.5.1	Sintonización del controlador MPC.....	53
5.5.1.1	Algoritmo para la función objetivo.	54
5.5.1.2	Programación para evaluar el MPC.	56
5.5.1.3	Algoritmo por Multidinámicas para Optimización Global (MAGO).	58
5.5.2	Implementación del controlador MPC en Simulink	60
5.6	Diseño e implementación del controlador clásico PID en Simulink.....	63
5.7	Diseño de la interfaz gráfica de usuario (GUI)	66
6.	Resultados.....	71
6.1	Datos obtenidos	71
6.2	Modelos matemáticos del sistema de tanques interconectados.....	72
6.2.1	Caso 1: Llenado del tanque 1 con la bomba 1	72
6.2.2	Caso 2: Llenado de los tanques interconectados 1 y 2 con la bomba 1	74
6.2.3	Caso 3: Llenado del tanque 2 con la bomba 2	76
6.2.4	Caso 4: Llenado de los tanques interconectados 1 y 2 con la bomba 2.....	77
6.3	Diseño del controlador avanzado MPC.....	79
6.3.1	Controlador MPC para el caso 1	80
6.3.1.1	Parámetros obtenidos	80
6.3.1.2	Implementación del controlador MPC en Simulink.....	80
6.3.2	Controlador MPC para el caso 2.....	82
6.3.2.1	Parámetros obtenidos	82
6.3.2.2	Implementación del controlador MPC en Simulink.....	83
6.3.3	Controlador MPC para el caso 3.....	85
6.3.3.1	Parámetros obtenidos	85
6.3.3.2	Implementación del controlador MPC en Simulink.....	85
6.3.4	Controlador MPC para el caso 4.....	87
6.3.4.1	Parámetros obtenidos	87

6.3.4.2	Implementación del controlador MPC en Simulink.....	88
6.4	Diseño del controlador clásico PID.....	90
6.4.1	Controlador PID para el caso 1	90
6.4.1.1	Implementación del controlador PID en Simulink.....	90
6.4.1.2	Parámetros obtenidos	90
6.4.2	Controlador PID para el caso 2.....	92
6.4.2.1	Implementación del controlador PID en Simulink.....	92
6.4.2.2	Parámetros obtenidos	93
6.4.3	Controlador PID para el caso 3.....	94
6.4.3.1	Implementación del controlador PID en Simulink.....	94
6.4.3.2	Parámetros obtenidos	94
6.4.4	Controlador PID para el caso 4.....	96
6.4.4.1	Implementación del controlador PID en Simulink.....	96
6.4.4.2	Parámetros obtenidos	96
6.5	Evaluación de desempeño y comparación del controlador avanzado MPC con el controlador clásico PID.....	98
6.5.1	Comparación del controlador PID con el MPC para el caso 1	98
6.5.2	Comparación del controlador PID con el MPC para el caso 2	100
6.5.3	Comparación del controlador PID con el MPC para el caso 3	102
6.5.4	Comparación del controlador PID con el MPC para el caso 4	104
6.6	Interfaz gráfica de usuario (GUI).....	106
7.	Discusión.....	109
8.	Conclusiones.....	111
9.	Recomendaciones.....	112
10.	Bibliografía.....	113
11.	Anexos.....	116

Índice de tablas:

Tabla 1. Efectos ocasionados en la salida del sistema al incrementar cada parámetro del PID de forma independiente.....	27
Tabla 2. Valores de ECR para suavizar las restricciones.....	37
Tabla 3. Prioridad de seguimiento de las entradas y salidas del proceso, según el valor de los pesos.....	38
Tabla 4. Seudocódigo del algoritmo MAGO.....	40
Tabla 5. Consideraciones y recomendaciones a tener en cuenta respecto a los parámetros del MPC.....	54
Tabla 6. Parámetros del MPC para el tanque 1 (caso 1).....	80
Tabla 7. Parámetros del MPC para el tanque 2 (caso 2).....	82
Tabla 8. Parámetros del MPC para el tanque 2 (caso 3).....	85
Tabla 9. Parámetros del MPC para el tanque 1 (caso 4).....	87
Tabla 10. Parámetros del controlador PID para el tanque 1 (caso 1).....	91
Tabla 11. Parámetros del controlador PID para el tanque 2 (caso 2).....	93
Tabla 12. Parámetros del controlador PID para el tanque 2 (caso 3).....	95
Tabla 13. Parámetros del controlador PID para el tanque 1 (caso 4).....	97
Tabla 14. Características de desempeño del controlador MPC y PID para el tanque 1 (caso 1).....	100
Tabla 15. Características de desempeño del controlador MPC y PID para el tanque 2 (caso 2).....	101
Tabla 16. Características de desempeño del controlador MPC y PID para el tanque 2 (caso 3).....	103
Tabla 17. Características de desempeño del controlador MPC y PID para el tanque 1 (caso 4).....	106

Índice de figuras:

Figura 1. Sistema SISO (a) y sistema MIMO (b), conectados en serie.	7
Figura 2. Control de nivel de líquidos en la industria láctea.	7
Figura 3. Banco de celdas de flotación.	8
Figura 4. Etapas de la elaboración del azúcar.	9
Figura 5. Relación básica de un sistema en el control de procesos.	10
Figura 6. Diagrama esquemático de un sistema de control en lazo abierto.	11
Figura 7. Diagrama esquemático del sistema de control en lazo cerrado de una caldera.	12
Figura 8. Medición de nivel por ultrasonido.	13
Figura 9. Proceso para el modelado de un sistema.	14
Figura 10. Ventana principal de la System Identification de MATLAB®.	16
Figura 11. Opciones de importación de datos en la System Identification de MATLAB®.	17
Figura 12. Estado dinámico y estacionario de la salida de un sistema.	18
Figura 13. Comportamiento de un sistema dependiendo de la localización de los polos y ceros en el plano complejo s.	19
Figura 14. Estabilidad de un sistema de acuerdo a la ubicación de los polos en el plano complejo s.	20
Figura 15. Sistema críticamente estable e inestable.	20
Figura 16. Retardo de transporte y retardo aparente en un sistema.	23
Figura 17. Diagrama de bloques de un controlador PID.	25
Figura 18. Control predictivo en tanques acoplados.	28
Figura 19. Estrategia del control predictivo basado en modelo.	29
Figura 20. Analogía del MPC con la conducción de un automóvil.	31
Figura 21. Horizonte deslizante en la ley de control.	34
Figura 22. Cuadro de diálogo para definir la estructura del MPC y linealizar el modelo.	36
Figura 23. Barra deslizante para ajustar la robustez o agresividad del controlador.	39
Figura 24. Esquema general de los algoritmos evolutivos.	39
Figura 25. Optimización de la función de Rastrigin con MAGO.	41
Figura 26. Metodología para el cumplimiento de los objetivos específicos.	42
Figura 27. Sistema prototipo de tanques interconectados del laboratorio de automatización de la CIEM.	46
Figura 28. Esquema del sistema de tanques interconectados.	47
Figura 29. Instalación de la librería LINX para la comunicación de LabVIEW con Arduino.	49
Figura 30. Instalación del firmware en la placa Arduino.	49

Figura 31. Panel frontal del instrumento virtual diseñado en LabVIEW para la recolección de datos.	50
Figura 32. Ventana principal de la Identificación de Sistemas de MATLAB® para la estimación del modelo.	51
Figura 33. Importación de los datos de entrada y salida para la Identificación de Sistemas..	52
Figura 34. Cuadro de diálogo para la estimación del modelo de la planta.	53
Figura 35. Definición de la función de transferencia y de los parámetros del controlador MPC.	56
Figura 36. Definición de la función de transferencia para un sistema con retardo.....	56
Figura 37. Definición del conjunto de posibles soluciones, número de individuos y número de generaciones.....	57
Figura 38. Ejecución del algoritmo desarrollado para evaluar el MPC y encontrar sus parámetros.....	58
Figura 39. Diagrama de flujo del algoritmo MAGO.	59
Figura 40. Bloque MPC Controller para implementación de controladores predictivos en Simulink.....	60
Figura 41. Ventana principal del bloque del controlador MPC.....	61
Figura 42. Definición de la estructura del MPC.	61
Figura 43. Importación del modelo obtenido al controlador MPC.....	62
Figura 44. Selección de la función de transferencia importada.	62
Figura 45. Asignación de los parámetros del MPC.	63
Figura 46. Bloque PID Controller para la implementación del controlador clásico PID en Simulink.	64
Figura 47. Interfaz de la aplicación PID Tuner para la sintonización del controlador PID....	64
Figura 48. Método anti-windup implementado al controlador PID en Simulink.	65
Figura 49. Ventana principal para el diseño de la GUI.	66
Figura 50. Inspector de propiedades de los elementos de la GUI.....	67
Figura 51. Ingreso a la función asociada al elemento para su programación.	67
Figura 52. Control PID en Simulink para la GUI.	69
Figura 53. Interfaz gráfica de usuario (GUI).	70
Figura 54. Archivo de Excel generado por LabVIEW el cual contiene el registro de datos obtenidos en el experimento de identificación.	71
Figura 55. Esquema de llenado del tanque 1 con la bomba 1.....	72
Figura 56. Validación del modelo obtenido para el tanque 1 (caso 1).	73
Figura 57. Respuesta del sistema a la señal escalón y diagrama de polos y ceros (caso 1). ..	73
Figura 58. Esquema de llenado de los tanques interconectados 1 y 2 con la bomba 1.	74
Figura 59. Validación del modelo obtenido para el tanque 2 (caso 2).	75

Figura 60. Respuesta del sistema a la señal escalón y diagrama de polos y ceros para el tanque 2 (caso 2).	75
Figura 61. Esquema de llenado del tanque 2 con la bomba 2.	76
Figura 62. Validación del modelo obtenido para el tanque 2 (caso 3).	77
Figura 63. Respuesta del sistema a la señal escalón y diagrama de polos y ceros (caso 3).	77
Figura 64. Esquema de llenado de los tanques interconectados 1 y 2 con la bomba 2.	78
Figura 65. Validación del modelo obtenido para el tanque 1 (caso 4).	79
Figura 66. Respuesta del sistema a la señal escalón y diagrama de polos y ceros para el tanque 1 (caso 4).	79
Figura 67. Diagrama de control del MPC en Simulink para el tanque 1 (caso 1).	80
Figura 68. Gráfica comparativa de los resultados de sintonización obtenidos para el tanque 1 (caso 1) con el algoritmo diseñado.	81
Figura 69. Comportamiento del controlador MPC para el tanque 1 (caso 1) con los diferentes errores de sintonización determinados por el algoritmo desarrollado.	82
Figura 70. Diagrama de control del MPC en Simulink para el tanque 2 (caso 2).	83
Figura 71. Gráfica comparativa de los resultados de sintonización obtenidos para el tanque 2 (caso 2) con el algoritmo diseñado.	84
Figura 72. Comportamiento del controlador MPC para el tanque 2 (caso 2) con los diferentes errores de sintonización determinados por el algoritmo desarrollado.	84
Figura 73. Diagrama de control del MPC en Simulink para el tanque 2 (caso 3).	85
Figura 74. Gráfica comparativa de los resultados de sintonización obtenidos para el tanque 2 (caso 3) con el algoritmo diseñado.	86
Figura 75. Comportamiento del controlador MPC para el tanque 2 (caso 3) con los diferentes errores de sintonización determinados por el algoritmo desarrollado.	87
Figura 76. Diagrama de control del MPC en Simulink para el tanque 1 (caso 4).	88
Figura 77. Gráfica comparativa de los resultados de sintonización obtenidos para el tanque 1 (caso 4) con el algoritmo diseñado.	89
Figura 78. Comportamiento del controlador MPC para el tanque 1 (caso 4) con los diferentes errores de sintonización determinados por el algoritmo desarrollado.	89
Figura 79. Diagrama de control PID en Simulink para el tanque 1 (caso 1).	90
Figura 80. Sintonización del controlador PID para el tanque 1 (caso 1).	90
Figura 81. Respuesta del controlador PID diseñado para el tanque 1 (caso 1).	91
Figura 82. Control PID del tanque 1 sin implementar el método anti-windup.	92
Figura 83. Control PID del tanque 1 implementando el método anti-windup.	92
Figura 84. Diagrama de control PID en Simulink para el tanque 2 (caso 2).	93
Figura 85. Sintonización del controlador PID para el tanque 2 (caso 2).	93
Figura 86. Respuesta del controlador PID diseñado para el tanque 2 (caso 2).	94

Figura 87. Diagrama de control PID en Simulink para el tanque 2 (caso 3).....	94
Figura 88. Sintonización del controlador PID para el tanque 2 (caso 3).....	95
Figura 89. Respuesta del controlador PID diseñado para el tanque 2 (caso 3).....	95
Figura 90. Diagrama de control PID en Simulink para el tanque 1 (caso 4).....	96
Figura 91. Sintonización del controlador PID para el tanque 1 (caso 4).....	96
Figura 92. Análisis de la respuesta del controlador PID para el tanque 1 (caso 4), minimizando el tiempo en que el sistema alcanza el estado estable.....	97
Figura 93. Respuesta del controlador PID diseñado para el tanque 1 (caso 4).....	97
Figura 94. Comparación de los controladores diseñados para el tanque 1 (caso 1) a una referencia de 5 litros.....	98
Figura 95. Comparación de los controladores diseñados para el tanque 1 (caso 1) a una referencia de 12 litros.....	99
Figura 96. Análisis de la respuesta del controlador MPC para un nivel de referencia de 12 litros.....	99
Figura 97. Análisis de rechazo de perturbaciones en el control de nivel del tanque 1 (caso 1).	100
Figura 98. Comparación de los controladores diseñados para el tanque 2 (caso 2) a una referencia de 4 litros.....	101
Figura 99. Comparación de los controladores diseñados para el tanque 2 (caso 2) a una referencia de 8 litros.....	101
Figura 100. Análisis de rechazo de perturbaciones en el control de nivel del tanque 2 (caso 2).	102
Figura 101. Comparación de los controladores diseñados para el tanque 2 (caso 3) a una referencia de 5 litros.....	102
Figura 102. Comparación de los controladores diseñados para el tanque 2 (caso 3) a una referencia de 12 litros.....	103
Figura 103. Análisis de rechazo de perturbaciones en el control de nivel del tanque 2 (caso 3).	104
Figura 104. Comparación de los controladores diseñados para el tanque 1 (caso 4) a una referencia de 2 litros.....	104
Figura 105. Comparación de los controladores diseñados para el tanque 1 (caso 4) a una referencia de 3,5 litros.....	105
Figura 106. Comparación de los controladores diseñados para el tanque 1 (caso 4) a una referencia de 5 litros.....	105
Figura 107. Análisis de rechazo de perturbaciones en el control de nivel del tanque 1 (caso 4).	106
Figura 108. Simulación de la respuesta del controlador PID y MPC para el tanque 2 (caso 3) a un setpoint de 1 empleando la GUI.....	107

Figura 109. Comparación de la respuesta del controlador PID con el MPC para el tanque 2 (caso 3) a un setpoint de 3 empleando la GUI.	107
Figura 110. Comparación de las curvas del controlador MPC para el tanque 1 (caso 1), obtenidas de la sintonización manual realizada con la GUI.	108
Figura 111. Programación en el diagrama de bloques para el instrumento virtual (VI) diseñado.	116
Figura 112. Comportamiento dinámico de llenado del tanque 1 (caso 1).	127
Figura 113. Comportamiento dinámico de llenado del tanque 1 (caso 2).	127
Figura 114. Comportamiento dinámico de llenado del tanque 2 (caso 2).	128
Figura 115. Comportamiento dinámico de llenado del tanque 2 (caso 3).	128
Figura 116. Comportamiento dinámico de llenado del tanque 1 (caso 4).	129
Figura 117. Comportamiento dinámico de llenado del tanque 2 (caso 4).	129
Figura 118. Función de transferencia del tanque 1 (caso 2).	130
Figura 119. Función de transferencia del tanque 2 (caso 4).	131

Índice de anexos:

Anexo 1.	Programación en diagrama de bloques para el VI diseñado en LabVIEW 2019.	116
Anexo 2.	Algoritmo para la función objetivo	117
Anexo 3.	Programación para evaluar el MPC	118
Anexo 4.	Algoritmo por Multidínamicas para Optimización Global (MAGO)	119
Anexo 5.	Código para cargar las funciones de transferencia al Workspace de MATLAB®	123
Anexo 6.	Programación para abrir el Simulink y cargar las imágenes a la GUI.....	123
Anexo 7.	Programación para verificar el caso seleccionado en la GUI y cargar los parámetros al controlador	124
Anexo 8.	Función enviar datos para la GUI	124
Anexo 9.	Función extraer datos para la GUI	125
Anexo 10.	Programación para el controlador clásico PID de la GUI.....	125
Anexo 11.	Programación para el controlador avanzado MPC de la GUI.....	126
Anexo 12.	Gráficas de la dinámica de nivel del sistema de tanques interconectados con los datos obtenidos en el experimento de identificación	127
Anexo 13.	Función de transferencia del tanque 1 (caso 2).....	130
Anexo 14.	Función de transferencia del tanque 2 (caso 4).....	131
Anexo 15.	Interfaz gráfica de usuario (GUI).....	131
Anexo 16.	Instrumento virtual diseñado en LabVIEW para la recolección de datos.....	131
Anexo 17.	Algoritmo desarrollado en MATLAB® para la sintonización del MPC	131
Anexo 18.	Certificado de traducción del resumen.....	132

Índice de símbolos y abreviaturas:

Símbolos

K_d	Constante derivativa
K_i	Constante integral
K_p	Constante proporcional
$u_f(t)$	Entradas anteriores al instante actual t
ζ	Factor de amortiguamiento
ω_n	Frecuencia natural no amortiguada en [rad/s]
J	Función de costo o función objetivo
K	Ganancia estática del sistema
N_u	Horizonte de control
N	Horizonte de predicción
Δu	Incremento de la variable manipulada
N_2	Límite máximo de predicción
N_1	Límite mínimo de predicción
$\lambda(j)$	Peso de importancia que deseo darle al esfuerzo de control
$\delta(j)$	Peso de minimización del error entre el setpoint y la salida del proceso
$\hat{y}(t + k t)$	Predicciones de la variable controlada
θ	Retardo de tiempo del sistema en [s]
$y(t)$	Salida del sistema (variable controlada)
D_{out}	Salida derivativa
I_{out}	Salida integral
P_{out}	Salida proporcional
$u(t)$	Señal de control (variable manipulada)
$e(t)$	Señal de error
$u_c(t)$	Señales de control en los instantes futuros
t_s	Tiempo de establecimiento en [s]
T_s	Tiempo de muestreo en [s]
$t + k$	Tiempo futuro
$w(t + k)$	Trayectoria de referencia futura
$r(t)$	Valor de referencia o setpoint
$u(t + k t)$	Valores futuros de la variable manipulada

Abreviaturas

MAGO	Algoritmo por multidinámicas para optimización global
AE	Algoritmos evolutivos
ARIMA	Autorregresivo integrado de media móvil
DMC	Control por matriz dinámica
MPC	Control predictivo basado en modelo
NMPC	Control predictivo basado en modelo no lineal
PFC	Control predictivo funcional
GPC	Control predictivo generalizado
PID	Control proporcional, integral y derivativo
PLC	Controlador lógico programable
VI	Instrumento virtual
GUI	Interfaz gráfica de usuario
HMI	Interfaz hombre-máquina
LTI	Lineal e invariante en el tiempo
OPC	OLE para control de procesos
Peso ECR	Peso de la variable de holgura
DSC	Registro de datos y control de supervisión
ECR	Relajación de restricciones por igual
MIMO	Sistema de múltiples entradas y múltiples salidas
SISO	Sistema de una entrada y una salida
OLE	Vinculación e incrustación de objetos

1. Título

Diseño de un controlador avanzado para el control de nivel en el sistema de tanques interconectados del Laboratorio de Automatización de la CIEM.

2. Resumen

El presente proyecto de titulación aborda el diseño y evaluación mediante simulación de un Controlador Predictivo Basado en Modelo (MPC) para el control de nivel en el sistema de tanques interconectados del Laboratorio de Automatización de la carrera de Ingeniería Electromecánica, con el fin de comparar las características que presenta un sistema de control de nivel avanzado respecto al control convencional PID y cuantificar su desempeño, puesto que este último resulta ser ineficiente cuando el proceso es complejo y presenta grandes retardos de tiempo. Se partió obteniendo el modelo matemático que describe el comportamiento dinámico de la planta de estudio, mediante el método de identificación de sistemas, usando la herramienta que proporciona MATLAB® para este proceso. Luego, se sintonizó el controlador predictivo empleando un algoritmo evolutivo con una función objetivo programada para este caso, a fin de determinar qué parámetros me permite obtener un control óptimo; posteriormente se implementó en Simulink para su diseño, simulación y análisis, utilizando el MPC Toolbox de MATLAB®. A su vez también se diseñó un control PID para el sistema de tanques interconectados, considerando las limitaciones físicas del actuador a través de una estrategia conocida como anti-windup, mejorando así la respuesta de control. Implementado ambos controladores en Simulink se evaluó su desempeño mediante simulación y se cuantificó sus resultados, donde se evidencia que el control predictivo presenta un mejor rendimiento a comparación del controlador clásico PID. Además, se desarrolló una aplicación mediante una plataforma GUI que permite al usuario seleccionar cada uno de los casos planteados y ejecutar la simulación de la planta con los controladores MPC y PID diseñados, pudiendo observar mediante gráficas su comportamiento y rendimiento; adicionalmente, se tiene la opción de modificar cada uno de los parámetros preestablecidos del MPC, por lo cual se puede realizar una sintonización manual y ejecutar su simulación para comparar resultados.

***Palabras claves:** control de nivel, MPC, controlador PID, anti-windup, retardos de tiempo, identificación de sistemas.*

Abstract

The present degree project approaches the design and evaluation through the simulation of a predictive controller based on model (MPC) to control the level in the tank system interconnected from Automation Laboratory of Electromechanics Engineering career, with the purpose to compare the characteristics that presents an advanced level controller system regarding PID conventional control and quantify its development, because this last results being inefficient when the process is complex and presents large time delays. It started by obtaining the mathematical model that describe the dynamic behavior of study plant, through the system identification method, using the tool that provides MATLAB® for this process. Then, it was tuning the predictive controller using an evolutive algorithm with an objective function set for this case, in order to determine which parameters allows to get an optimal controller; using the MPC Toolbox of MATLAB®. At the same time, also it was designed a PID control for the interconnected systems tanks, considering the physics limitations of the actuator through a strategy known as anti-windup, improving the control answer. Setting up both controllers in Simulink their performance was evaluated through simulation and it was quantifier their outcomes, where it is evinced that the predictive controller presents a better performance compared to classic PID controller. In addition, it was developed an application through a GUI platform which allows to the user select each one of brought case and run the simulation of the plant with the MPC and PID controllers designed, being able to observe through graphics their behavior and performance; therefore, it has the option of modify each one of the parameters preestablish MPC, thus it can make a manual tuning and run their simulation to compare results.

Key words: *level control, MPC, PID controller, anti-windup, time delays, systems identification.*

3. Introducción

La amplia competencia en el sector industrial ha llevado a la optimización y mejoramiento de los sistemas de control, en los que se requiere una mayor eficiencia, mejores estándares de calidad, aumento de la productividad y un uso eficiente de la energía. Con el fin de resolver los distintos problemas que surgen en los procesos industriales como aquellos que presentan dinámicas complejas, con retardos de tiempo, restricciones, perturbaciones y múltiples variables, se opta por implementar sistemas de control avanzado, debido a que estos problemas son tratados de forma ineficiente por las técnicas de control clásico como el PID.

El control predictivo basado en modelo (MPC) es una técnica de control avanzado muy exitosa dentro de la industria, esto debido a que puede ser implementado en una gran variedad de procesos y al rendimiento que ha mostrado en distintas aplicaciones con respecto a otras técnicas de control. Este controlador es uno de los más populares a nivel industrial cuando se desea regular procesos muy complejos, pues involucra en su acción de control la formulación del problema, el uso de un modelo para predecir el comportamiento futuro del sistema objeto de estudio, las limitaciones físicas de la planta y la seguridad del proceso, permitiendo así optimizar la energía consumida y reducir los costos de operación.

Los sistemas de control de nivel y flujo en tanques, son esencialmente importantes para muchas de las industrias, entre ellas se encuentra la industria química, petroquímica, farmacéutica, plantas de tratamiento de agua, industria alimentaria, entre otras; donde se requiere administrar materias primas líquidas, productos químicos y subproductos que son almacenados y transportados de un tanque a otro. Inicialmente el controlador MPC fue desarrollado para dar solución al control de plantas químicas y refinerías de petróleo, pero actualmente puede encontrarse en varias industrias (Gutarra, 2016).

Un sistema bien conocido y ampliamente utilizado en el sector industrial es el de tanques interconectados, donde es indispensable llevar un control de nivel de líquidos. Estos líquidos deben bombearse de un tanque a otro y almacenarse para luego ser llevados a otros procesos, por lo que el nivel de llenado o vaciado del tanque siempre debe verificarse y el flujo entre ellos debe regularse; además, estos tanques al estar interconectados actúan en conjunto y el vaciado o llenado de un tanque depende del otro, interactuando los niveles entre sí, por lo que es necesario controlar cada una de estas variables con total exactitud.

El laboratorio de automatización de la carrera de ingeniería electromecánica (CIEM) de la Universidad Nacional de Loja (UNL) cuenta con un sistema prototipo de tanques interconectados para la medición y control de nivel de líquidos, con el fin de que los estudiantes de esta carrera puedan realizar prácticas en el área del control automático. Debido a que este

sistema posee únicamente un control on-off y no cuenta con un control de nivel netamente automatizado, se desea optimizar este banco de pruebas diseñando un sistema de control de nivel avanzado MPC, considerando la dinámica compleja del sistema, el cual puede ser implementado en un futuro proyecto si así se desea.

El objetivo general que ha guiado al presente proyecto es “Diseñar un sistema de control de nivel avanzado para el sistema prototipo de tanques interconectados”.

Para cumplir con el objetivo general se han planteado los siguientes objetivos específicos:

- Determinar un modelo matemático que represente la dinámica del sistema prototipo de tanques interconectados.
- Diseñar un controlador avanzado para el control de nivel del sistema prototipo de tanques interconectados.
- Comparar mediante simulación las características del controlador avanzado diseñado con controladores convencionales aplicados al prototipo de tanques.

Con la ejecución de este proyecto se pretende analizar las ventajas que presenta el controlador de nivel avanzado MPC comparado con controladores convencionales PID. Además, se expone la metodología empleada para la obtención del modelo matemático del sistema de tanques interconectados y el procedimiento para el diseño de un controlador MPC. Permitiendo al lector entender sobre el funcionamiento de los sistemas de control y cómo resolver problemas relacionados con el control de nivel de líquidos, que requieren un controlador de nivel avanzado debido a la complejidad en la dinámica del proceso.

El control de nivel y su flujo en tanques constituye parte esencial de cualquier industria y por consiguiente de cualquier sistema económico basado en la producción, donde el control predictivo MPC genera la más alta expectativa. Por tal razón, el estudio y diseño de este sistema de control es de gran importancia para el sector industrial.

4. Marco teórico

4.1 Capítulo 1: Aspectos generales del control de nivel

El control de nivel de líquidos y su flujo en tanques, constituye un problema básico de toda industria basada en la producción; dichos líquidos deben bombearse de un tanque a otro y almacenarse para luego ser llevados a otros procesos, por lo que el nivel de llenado o vaciado del líquido siempre debe verificarse y el flujo entre ellos debe regularse, además, estos tanques por lo general están interconectados, es decir, actúan en conjunto y el vaciado o llenado de un tanque depende del otro, interactuando los niveles entre sí, siendo necesario controlar cada una de estas variables con total exactitud, con el fin de mejorar la eficiencia de los procesos productivos y generar un ahorro económico al minimizar las acciones de control.

4.1.1 Control de nivel y su importancia en la industria

Un sistema de control de nivel es un conjunto de instrumentos o elementos con parte lógica, que actúan simultáneamente sobre una estructura hidráulica cuyo fin es conseguir un dominio de las variables de salida, mediante la manipulación de las variables de control, garantizando así el nivel o flujo del líquido en un valor de operación preestablecido.

Los sistemas de control de nivel y flujo en tanques, son de gran importancia para muchas de las industrias, entre ellas se encuentran la industria química, petroquímica, farmacéutica y sistemas de tratamiento de aguas; donde se requiere administrar materias primas líquidas, productos químicos y subproductos que son almacenados y transportados de un tanque a otro. Como ejemplos, se tiene el control de nivel máximo de agua en tanques de almacenamiento para evitar su derrame y garantizar la seguridad de dicha estructura, control de nivel de agua de alimentación en calderas para asegurar un funcionamiento fiable y un vapor de alta calidad, control de nivel mínimo de agua para garantizar el buen funcionamiento del sistema y evitar la cavitación en bombas, entre otros.

4.1.1.1 Sistema de tanques acoplados

Un sistema ampliamente utilizado en el sector industrial, es el de tanques acoplados, basado en la ley de vasos comunicantes, la cual afirma que si se vierte un líquido homogéneo a un conjunto de recipientes que están unidos en su parte inferior por un tubo, dicho fluido alcanzará el mismo nivel en todos los recipientes, sin importar su forma o volumen; esto se debe a que la presión atmosférica en su superficie es la misma.

A nivel académico se han diseñado y construido distintas configuraciones para el estudio y desarrollo de sistemas de control lineales y no lineales. Entre ellos se encuentra el sistema de tanque interconectados en serie, el cual consiste en tanques unidos uno tras otro de forma directa, es decir la salida de un tanque es la entrada del siguiente. Se puede encontrar

una variedad de sistemas conectados en serie, con una entrada y una salida (SISO) y con múltiples entradas y múltiples salidas (MIMO), tal como se muestra en la **Figura 1**, donde las entradas en este sistema son el caudal que entrega cada una de las bombas y las salidas el nivel en el tanque 1 y el tanque 2 (Tlachi, 2018).

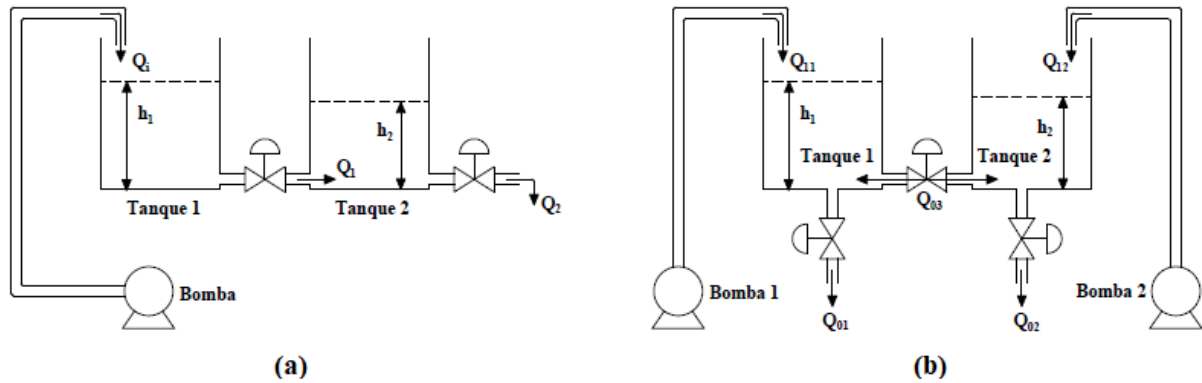


Figura 1. Sistema SISO (a) y sistema MIMO (b), conectados en serie.

En la **Figura 2** se puede apreciar un proceso de control de nivel en la industria láctea, en el cual se observa cómo se mezclan una variedad de ingredientes en un tanque mezclador centrífugo y recirculan a otro tanque para una mezcla más homogénea. Una vez obtenida una mezcla de calidad, es llevada a otro depósito para su almacenamiento y posterior bombeo a otros procesos.

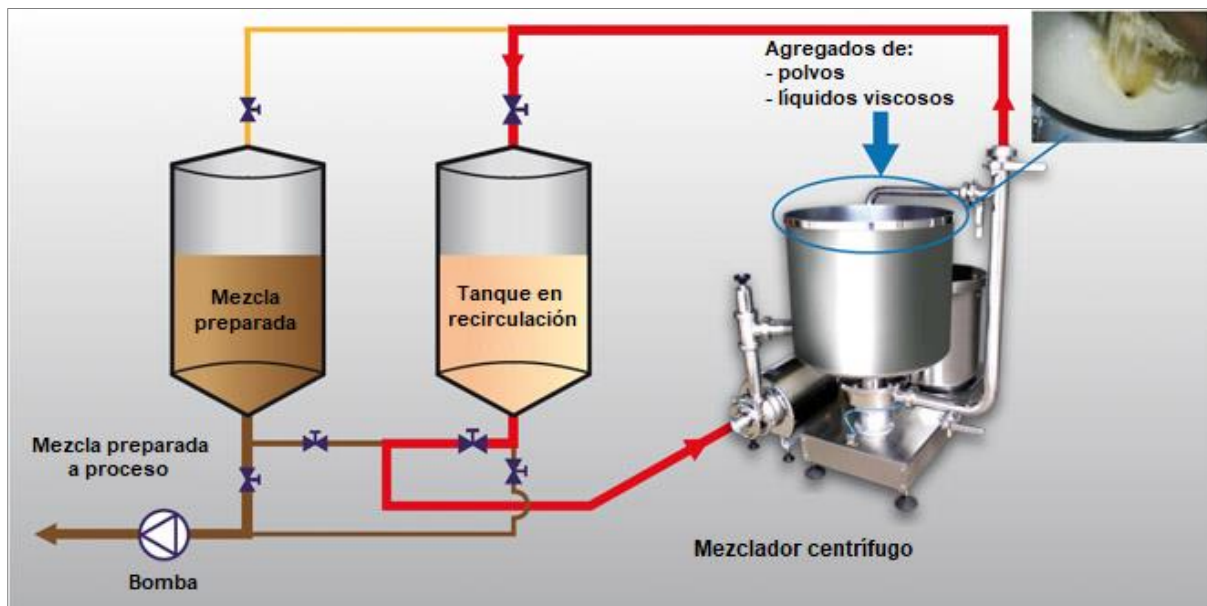


Figura 2. Control de nivel de líquidos en la industria láctea.

Fuente: (Gutarra, 2016).

Otro ejemplo típico de la aplicación de estos sistemas se da en el procesamiento de minerales, específicamente en el control de nivel en celdas de flotación, las cuales necesitan de un control riguroso y eficiente, debido a que el control de nivel en estas plantas es complejo

por las fuertes perturbaciones en los flujos de entrada a las celdas; además, una acción de control ejecutada en cualquier punto del circuito de celdas, tiende a ser transmitida a las demás unidades del sistema aguas arriba y aguas abajo (Sánchez, 2018).

Las celdas de flotación mostradas en la **Figura 3** son utilizadas para el proceso de extracción de minerales, como el oro, el cobre y otros metales; estos equipos mecánicos ponen en contacto el mineral con el aire, el agua y los reactivos químicos, cuya función es separar de forma eficiente las partículas de minerales útiles de otros compuestos desde la pulpa, la cual es una mezcla de mineral molido o pulverizado con agua o una solución acuosa (Fimbres, 2018).



Figura 3. Banco de celdas de flotación.

Fuente: (Mundo Minero, 2021).

En el proceso de producción de azúcar, concretamente en la etapa de evaporación, la medición de nivel en el evaporador es esencial para garantizar la calidad del producto, dado que un nivel de jugo clarificado bajo puede dañar todo el lote y causar una parada imprevista. En este proceso se emplea un sistema de evaporación de múltiple efecto como el que se observa en la **Figura 4**, el cual consta de varios evaporadores conectados entre sí, de tal manera que la evaporación producida en el primero sirve de vapor de calefacción al segundo evaporador y así sucesivamente. Esto tiene como objetivo eliminar el agua del jugo claro que procede del clarificador (85 % de agua y 15 % de sólidos) para alcanzar una meladura cuya composición contenga alrededor del 58 % al 62 % de sólidos (Córdova, 2017).

En la etapa de evaporación del azúcar se controla el nivel del jugo clarificado en cada uno de los evaporadores y la presión del vapor; además de la manipulación de los flujos de entrada.

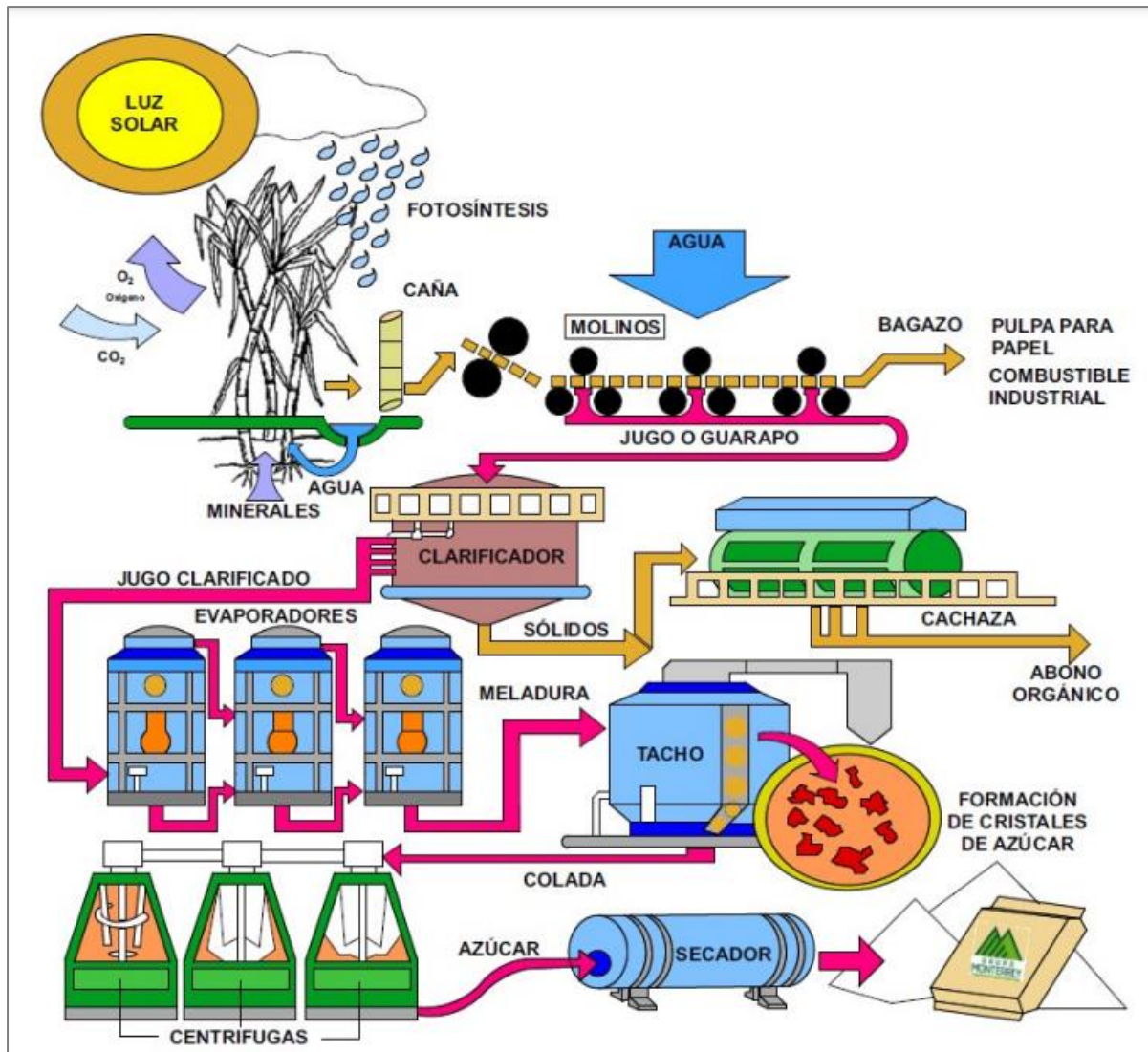


Figura 4. Etapas de la elaboración del azúcar.
Fuente: (Córdova, 2017).

4.1.2 Componentes básicos de un sistema de control

Dentro de la rama de ingeniería un sistema está definido por una o más variables de entrada o de actuación (u), las cuales manipulan física, química, eléctrica o mecánicamente un determinado proceso, y las variables de salida controladas (y). La relación entre estos tres componentes se expone en la **Figura 5**, donde se tiene a modo de ejemplo el sistema de un invernadero, en el que las variables de salida son la humedad relativa y la temperatura interior, controladas por las variables de entrada que son el sistema de pulverización de agua y de calefacción, respectivamente.

El objetivo de todo sistema de control es lograr que las salidas de un proceso se mantengan en un valor o consigna deseado, mediante la manipulación de las variables de entrada.

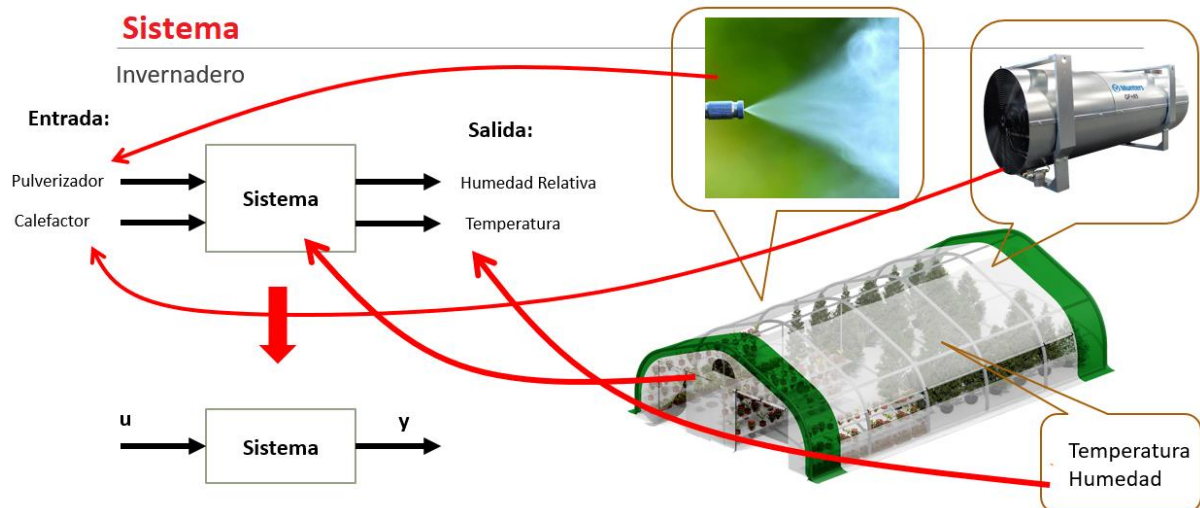


Figura 5. Relación básica de un sistema en el control de procesos.
Fuente: (Castaño Giraldo, 2020).

Un sistema de control está conformado por variables tanto de entrada como de salida, elementos de control como actuadores, sensores y modelado del proceso. Además de señales que son la referencia y las perturbaciones.

4.1.2.1 Variables

Las variables que intervienen en todo sistema de control son:

- **Variable manipulada:** es el parámetro que el elemento controlador modifica para afectar el valor de la variable controlada o de salida, estabilizando el sistema.
- **Variable controlada:** es el parámetro más importante del proceso, denotado como la salida del sistema, el cual es monitoreado y medido a través de un sensor, debiéndose mantener en un valor constante.

4.1.2.2 Elementos

El conjunto de elementos que interconectados entre sí permiten el control automático de un sistema son: el actuador, la planta, el proceso, el sensor y el controlador.

El actuador es el dispositivo que recibe como entrada la acción de control que permite modificar la salida del sistema. La planta es el sistema físico que se va a controlar; por otra parte, el proceso es el conjunto de operaciones a automatizar, en donde se estabiliza la variable de control. El sensor es un dispositivo de medición que informa el estado de la señal de salida hasta un comparador y el controlador es el cerebro del sistema de control, programado por un operador (Amador Soto, 2019).

4.1.2.3 Señales

Existen dos señales en un sistema de control, la referencia y las perturbaciones.

- **Referencia:** es el valor deseado en la variable de salida del proceso o variable controlada.
- **Perturbación:** es una señal que tiende a afectar el valor de la variable de salida del sistema. Cuando la perturbación se genera dentro del sistema se denomina interna; por el contrario, si se produce fuera del sistema, se denomina perturbación externa y es una entrada no manipulada (Amador Soto, 2019).

4.1.3 Clasificación de los sistemas de control

Los sistemas de control se clasifican en sistemas de lazo abierto y lazo cerrado. En el lazo abierto nunca se sabe cómo se comporta la variable de salida, mientras que en el lazo cerrado se conoce la evolución de dicha variable en todo momento.

4.1.3.1 Control de lazo abierto

Es aquel sistema de control en el cual la acción de control es independiente de la salida del sistema; se caracteriza por que la información o variables que controlan el proceso circulan en una sola dirección, **Figura 6**.



Figura 6. Diagrama esquemático de un sistema de control en lazo abierto.

Fuente: (Torres & Chanoluisa, 2015).

Los sistemas de control en lazo abierto, no presentan problemas de inestabilidad a comparación de los sistemas de lazo cerrado. Sin embargo, en presencia de perturbaciones responden de forma ineficiente.

4.1.3.2 Control de lazo cerrado

El control de lazo cerrado, también llamado sistema de control por retroalimentación, es aquel en el que la acción de control es en cierta manera dependiente de la salida del sistema. Se caracteriza porque permite comparar la salida medida por el sensor con la entrada del sistema, a través de una realimentación, permitiendo establecer una acción de control adecuada como consecuencia de la diferencia entre ambas (Castillo, 2014).

En la **Figura 7**, se muestra el ejemplo de un sistema de control en lazo cerrado de una caldera, en el cual se puede observar que el valor medido por el sensor de temperatura es comparado con el setpoint, dando un error de 1 °C al restar estos dos valores, en base a esto el controlador sabrá si abrir o cerrar la válvula con el objetivo de volver cero dicho error (Castaño Giraldo, 2020).

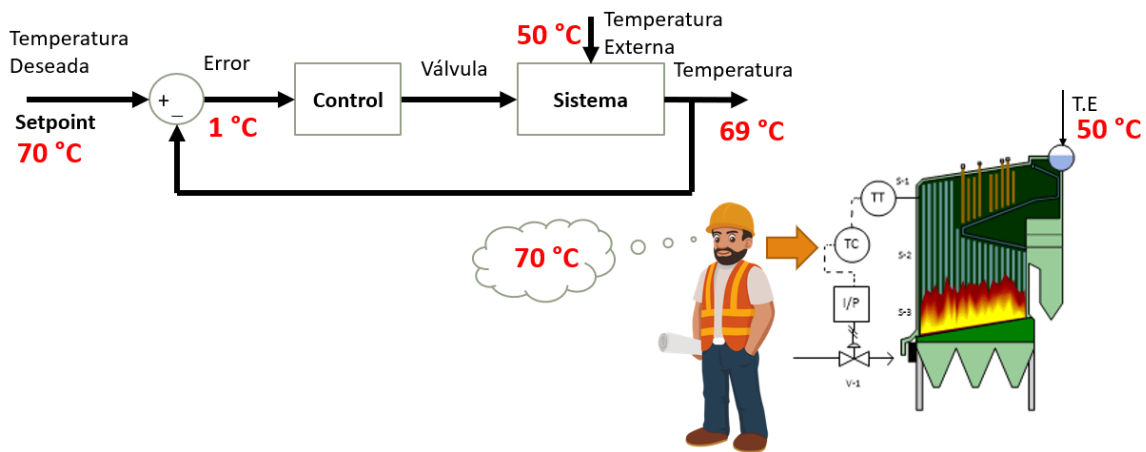


Figura 7. Diagrama esquemático del sistema de control en lazo cerrado de una caldera.

Fuente: (Castaño Giraldo, 2020).

4.1.4 Instrumentación en el control de un sistema hidráulico de tanques acoplados

La instrumentación y el control de nivel en tanques acoplados consta de múltiples elementos, como bombas hidráulicas, válvulas de control de caudal y retención, depósitos de almacenamiento e instrumentos para la medición del nivel de líquido, sumado a la parte de los componentes para el acondicionamiento de la señal y el software de control (Tlachi, 2018).

Los elementos que componen la parte del control y la medición de las variables son: las válvulas, la bomba centrífuga y el medidor de nivel.

- **Válvulas:** permiten ajustar el flujo del fluido para regular la variable del proceso. Puede controlarse manualmente o por medio de un actuador eléctrico, hidráulico o neumático (Tlachi, 2018).
- **Bomba centrífuga:** realiza el transporte del líquido y puede operar a distintas velocidades. El caudal se puede regular por arranque/parada, estrangulamiento de la tubería que conduce el fluido usando válvulas manuales o automáticas, y por variación de la velocidad del impulsor de la bomba mediante el uso de un variador de frecuencia; este último método es el más eficiente para el control de nivel, ya que la capacidad de bombeo se ajusta exactamente a los requisitos del proceso.

4.1.4.1 Medidores de nivel

La medición de nivel de un fluido es de las variables más importantes a considerar en los procesos productivos industriales. Basados en la experiencia técnica, al momento de seleccionar el tipo de medidor se prefieren los medidores estáticos y aquellos que no requieren de un contacto con el fluido, y que además se pueden ubicar en lugares exteriores al recipiente (Bohórquez et al., 2017).

Los instrumentos de medición y control de nivel pueden basarse en diferentes tecnologías: mecánica, inductiva, por presión, ultrasonidos, óptica y radar, entre otros.

Existen medidores de nivel con y sin contacto. Los medidores de nivel de contacto, como su nombre lo indica, necesitan entrar en contacto con el material ya sea este sólido o líquido, para activar la señal/alarma e indicar el nivel del mismo.

Por otra parte, los medidores de nivel de no contacto, a menudo son colocados encima del material sólido o líquido a medir y no necesita que entren en contacto para determinar el nivel en el recipiente. Ejemplos de estos tipos de instrumentos son:

- **Medidores con láser:** son colocados fuera del recipiente por lo que nunca llegan a tener contacto con el material. Estos sistemas láser emiten un haz estrecho hacia el material y miden el remanente del haz reflejado, midiendo el tiempo que le toma al haz ir y retornar desde el material. Los medidores de nivel con láser son utilizados en industrias de la minería y construcción, petroquímica y otras (Altec, 2021).
- **Medidores ultrasónicos:** en este tipo de sensores la altura de nivel se calcula midiendo el tiempo que tardan los impulsos ultrasónicos en viajar desde el sensor hasta la superficie reflectante y su tiempo de vuelta. Los sensores ultrasónicos (ver **Figura 8**) pueden medir sin problemas, medios agresivos, abrasivos y viscosos.

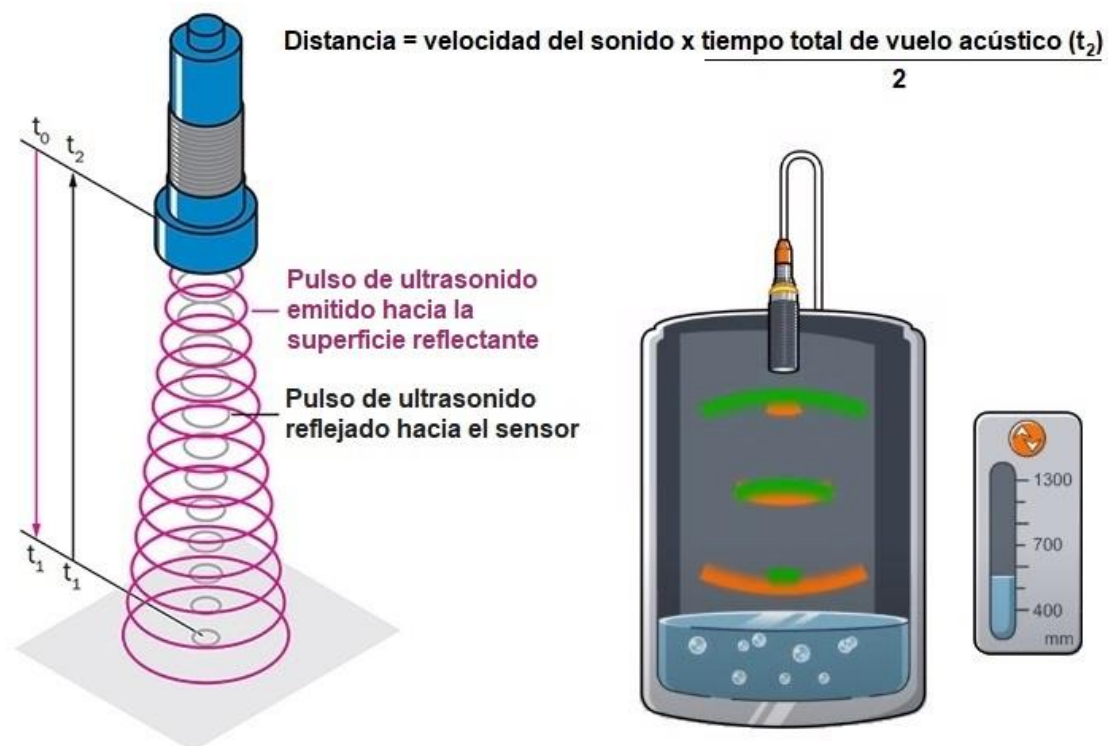


Figura 8. Medición de nivel por ultrasonido.

4.2 Capítulo 2: Modelado matemático e identificación de sistemas

Un modelo matemático es una representación simplificada, a través de ecuaciones o fórmulas matemáticas de algún sistema o fenómeno de la vida real, representando la relación que existe entre las distintas variables, parámetros y restricciones.

4.2.1 Identificación de sistemas

El objetivo del modelado de sistemas es determinar un modelo matemático que describa la dinámica del sistema de estudio. Si este modelo involucra datos experimentales, el proceso se conoce como identificación de sistemas (Ortiz et al., 2011).

La identificación de sistemas es un método basado en la experimentación, mediante el cual se puede obtener su modelo matemático, siendo este método el más utilizado para el control de procesos reales. Para obtener un modelo matemático mediante identificación de sistemas, se debe registrar una serie de datos que contengan los valores de las entradas y salidas del sistema objeto de estudio, sin importar las leyes que gobiernen la dinámica del mismo.

Hay que tener en cuenta que el sistema real no es igual al modelo matemático que se usa para describirlo. Por lo cual, se deben comparar para determinar qué tan preciso es el modelo obtenido. El procedimiento para el modelado se resume en la **Figura 9**.

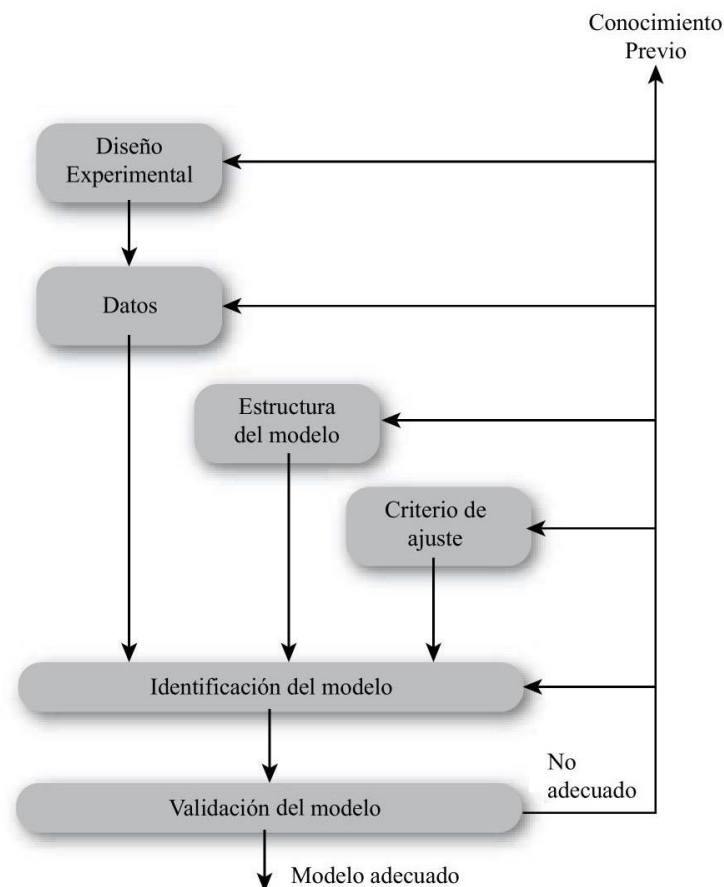


Figura 9. Proceso para el modelado de un sistema.

Fuente: (Ortiz et al., 2011).

4.2.1.1 Diseño experimental

El diseño de un experimento de identificación incluye varias decisiones: cuáles señales medir y cuándo medirlas, qué señales manipular y cómo manipularlas. Lo cual requiere de una planificación antes de comenzar el proceso de modelado.

4.2.1.1.1 Variables de diseño

Para obtener datos que sean lo suficientemente informativos, las variables en el experimento de identificación deben elegirse adecuadamente. Para esto, es necesario decidir cuáles señales se considerarán como salidas y cuáles como entradas, debido a que esto define donde estarán ubicados los sensores y que señales serán manipuladas para excitar el sistema durante la experimentación. Si existe alguna variable que afecta a la planta y no puede manipularse, se considerará como perturbación (Ortiz et al., 2011).

4.2.1.1.2 Señal de excitación

Es la señal con que se excita la entrada del sistema. Lo ideal es hacer que el experimento asemeje la situación en la cual se empleará el modelo; sin embargo, esto puede resultar difícil, ya que el objetivo de la identificación es, en muchos casos, encontrar las condiciones adecuadas de operación.

El grado de libertad que se tiene para manipular las entradas puede variar de acuerdo al sistema o proceso a analizar; en el caso de las industrias es posible que no se pueda manipular la planta cuando está en una producción continua. Según Ortiz et al. (2011) “la señal que se elija depende principalmente del método que se va a emplear en la etapa de identificación. Por ejemplo, si se usa un método de análisis transitorio, se debe utilizar una señal escalón, mientras que para el método de análisis de frecuencia, se debe utilizar una señal sinusoidal” (p. 27).

4.2.1.1.3 Cantidad de datos

La cantidad de datos que deben tomarse para la identificación de sistemas depende del método de identificación usado; no obstante, un criterio clave a la hora de decidir el número de datos a recopilar, es que la respuesta del sistema logre superar el estado transitorio y llegue al estado estable, siempre que esto sea posible.

4.2.1.2 Metodología de la identificación

Los métodos de identificación adquieren información del proceso a través de los datos registrados de las entradas y salidas del sistema objeto de estudio; además, la confiabilidad del modelo matemático obtenido depende de la dinámica con que se haya variado las entradas y cómo esos cambios se ven reflejados en las salidas. La ventaja de este método es que se puede obtener el modelo dinámico del sistema de una manera sencilla y rápida, sin necesidad de conocer a profundidad la funcionalidad del proceso.

Según Gutarra (2016) los pasos a seguir para el desarrollo de la identificación son los siguientes:

1. Planificar el experimento estableciendo datos aleatorios para las entradas y luego recoger los datos de las salidas después de que hayan sido procesados por el sistema.
2. Inspeccionar y filtrar los datos obtenidos, removiendo tendencias y sobresaltos.
3. Elegir un tipo de modelo que se ajuste a los datos experimentales y presente un mínimo error.
4. Validar el modelo obtenido, evaluando su respuesta ante saltos escalón e impulsos, con el fin de determinar posibles defectos y diagnosticar sus causas.

Una vez obtenido los datos por experimentación, se recomienda hacer un filtrado de los mismos, ya que esto permite remover algunos datos que son generados por ruidos de alta frecuencia y que no se desea que intervengan en el modelo conseguido por identificación.

4.2.2 System Identification Toolbox de MATLAB®

System Identification Toolbox es una herramienta de MATLAB® para la identificación de sistemas, la cual permite crear modelos matemáticos de sistemas dinámicos lineales y no lineales a partir de datos de entrada y salida, obtenidos mediante experimentación.

El ingreso a esta aplicación se realiza mediante la instrucción “*systemIdentification*” escrita en la ventana de comandos de MATLAB®. La ventana principal de la herramienta de identificación de sistemas (ver **Figura 10**) se divide en dos secciones: la primera es para el conjunto de datos de entrada para el análisis, y la segunda para los modelos creados como resultado de las operaciones realizadas con los datos importados.

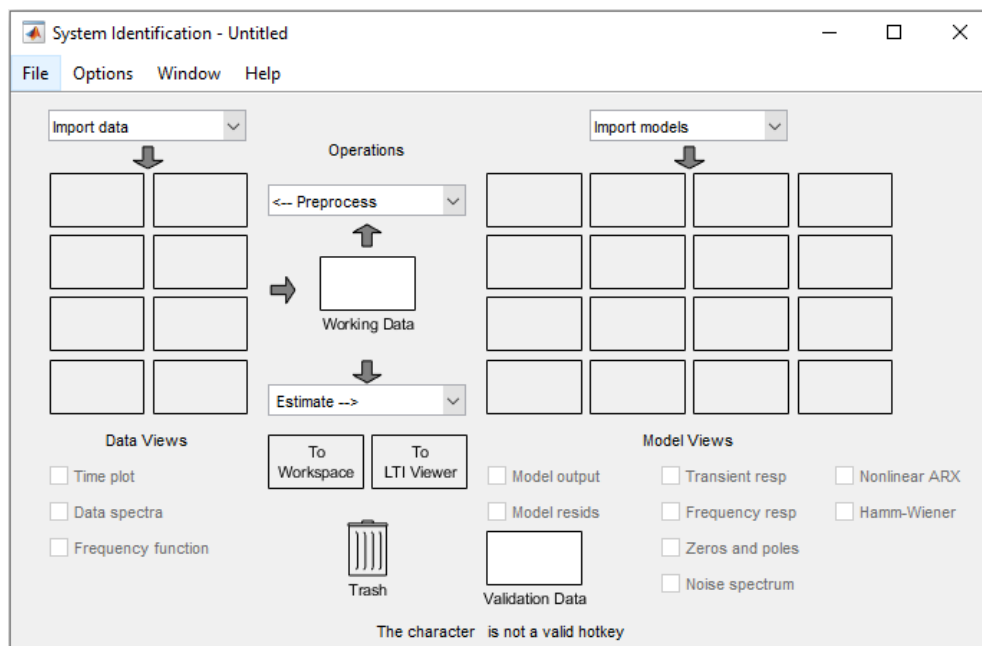


Figura 10. Ventana principal de la System Identification de MATLAB®.

System Identification presenta tres opciones de importación, entre ellas, la importación de datos en el dominio del tiempo, en el dominio de la frecuencia y la importación de objetos; estas tres opciones se muestran en la **Figura 11**.

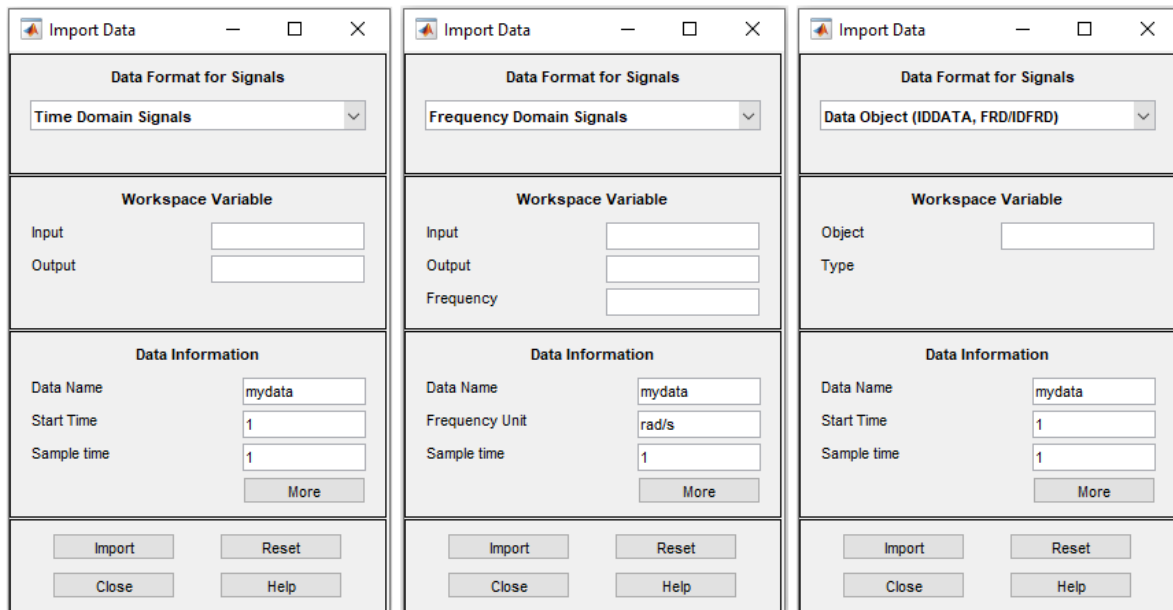


Figura 11. Opciones de importación de datos en la System Identification de MATLAB®.

En el dominio del tiempo los datos requeridos son los vectores de entrada (input) y salida (output) con un número n de filas y una columna, el *Data Name* hace referencia al nombre del conjunto de datos, el cuadro de *Start Time* corresponde al tiempo de inicio con el que fueron obtenidos los vectores de entrada y salida, mientras que el *Sample Time* se refiere al tiempo de muestreo de estos vectores.

En el dominio de la frecuencia los datos requeridos son los mismos que en el dominio del tiempo, adicionando únicamente el valor de la frecuencia. Por último, en la opción de importar objetos, dentro del cuadro de *Object Type* se coloca el nombre del objeto a importar desde el espacio de trabajo (workspace) de MATLAB®, el resto de datos son iguales a los ya descritos anteriormente (Jara & Maldonado, 2015).

4.2.3 Función de transferencia

Una función de transferencia $G(s)$ es una función matemática lineal que permite representar el comportamiento dinámico y estacionario de cualquier sistema. Se define como el cociente entre la transformada de Laplace de la salida de un sistema LTI (lineal e invariante en el tiempo) y la transformada de Laplace de la entrada a dicho sistema evaluado con condiciones iniciales iguales a cero, **Ecuación 1**.

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_0 \cdot s^m + b_1 \cdot s^{(m-1)} + \dots + b_{(m-1)} \cdot s + b_m}{a_0 \cdot s^n + a_1 \cdot s^{(n-1)} + \dots + a_{(n-1)} \cdot s + a_n} \quad (1)$$

Además de representar la dinámica de un sistema LTI que originalmente estaba en el dominio del tiempo como una ecuación diferencial a ecuaciones algebraicas en función de la variable compleja “s”, la función de transferencia también nos permite estudiar el comportamiento y la estabilidad del sistema.

El comportamiento de un proceso en el tiempo se lo puede representar de dos formas, tal como vemos en la **Figura 12**. En la zona dinámica la variable de salida del sistema va variando con el tiempo; por el contrario, en la zona estacionaria, el sistema ya no depende más del tiempo, por lo que la variable de salida se mantiene en el mismo valor, sin importar si el tiempo sigue creciendo.

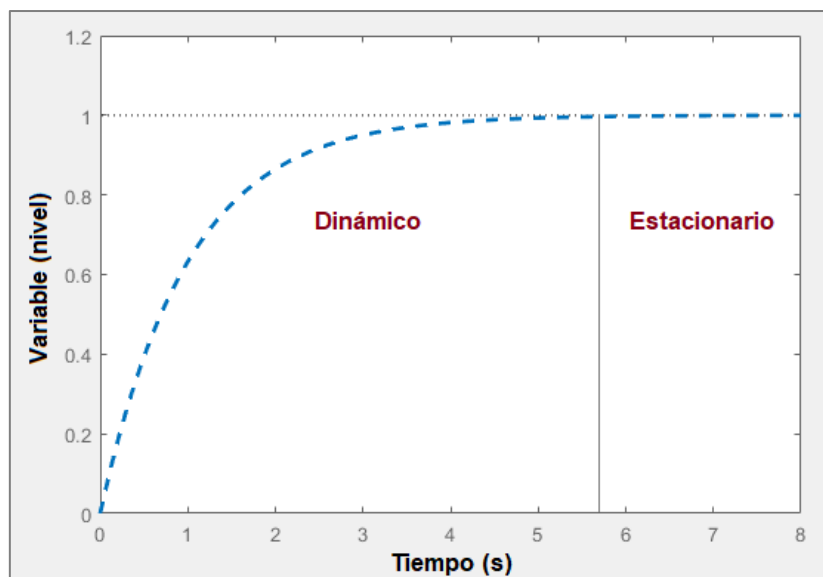


Figura 12. Estado dinámico y estacionario de la salida de un sistema.

Existen dos procedimientos para obtener una función de transferencia, el primero de ellos implica linealizar una ecuación diferencial y luego aplicar la transformada de Laplace, y el segundo, recopilar los datos del proceso a través de los sensores para aplicar posteriormente técnicas de identificación de sistemas.

4.2.3.1 Polos y ceros de una función de transferencia

Los polos (denotados con una x) y ceros (denotados con un círculo) de una función de transferencia caracterizan la forma y el comportamiento que tendrá un determinado sistema ante una eventual entrada de excitación.

En la **Figura 13**, se observa algunos casos de ejemplo sobre el comportamiento que tendrá el sistema dependiendo de donde se localizan los polos y ceros en el plano complejo “s”. Como se puede observar en dicha gráfica, si se tiene una serie de polos y un cero en la parte real negativa se obtiene una respuesta exponencial que va a converger en algún punto, en cambio, si se tiene un polo en la parte real positiva se va a tener una respuesta que también es

exponencial, pero va a ser divergente, y si por el contrario se tiene un cero en esta parte, se va a obtener una respuesta inversa que se estabiliza en algún punto.

También se tiene el caso en donde tenemos polos que no solo tienen parte real sino también imaginaria, obteniendo por ende una respuesta oscilatoria que se va amortiguando en el tiempo, llegando finalmente al estado estacionario. Por otro lado, tenemos una respuesta oscilatoria cuando los polos están justo en el eje imaginario y no se tiene parte real. Además, se tiene otro tipo de respuesta en la que el sistema es oscilatorio divergente y estas oscilaciones comienzan a crecer indefinidamente, debido a que se tiene polos con parte imaginaria y parte real positiva.

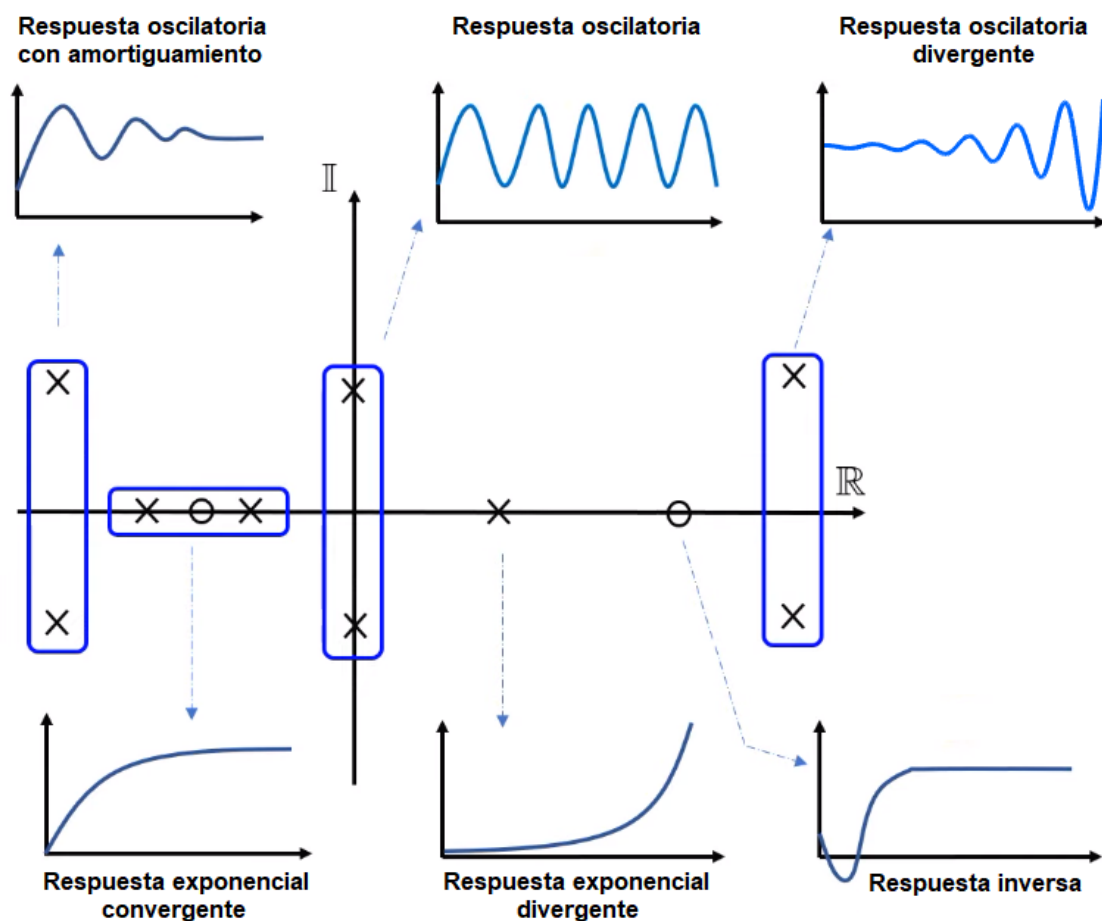


Figura 13. Comportamiento de un sistema dependiendo de la localización de los polos y ceros en el plano complejo s .

4.2.3.2 Estabilidad de un sistema

Al analizar una función de transferencia, se observa que ésta se compone de un polinomio tanto en el numerador como en el denominador. Cuando se iguala el polinomio del numerador a cero, se obtiene unas raíces que se conoce como los “ceros del sistema” y al hacer lo mismo con el polinomio del denominador, se obtiene sus raíces llamadas “polos del sistema”. Los ceros y polos pueden ser graficados en el plano complejo “ s ” y aquí podemos

determinar si una función de transferencia es estable o inestable, simplemente con observar donde se encuentran ubicados sus polos (Castaño Giraldo, 2021).

Si algún polo del sistema se encuentra ubicado en el semiplano derecho del plano “s”, el sistema es inestable, y si encontramos algún cero en esta zona, nuestro sistema no será inestable. Por lo tanto, la estabilidad dependerá de la parte real de los polos del sistema y todos los polos deben estar en la parte real negativa (semiplano izquierdo) para que sea estable, tal como se puede observar en la **Figura 14**.

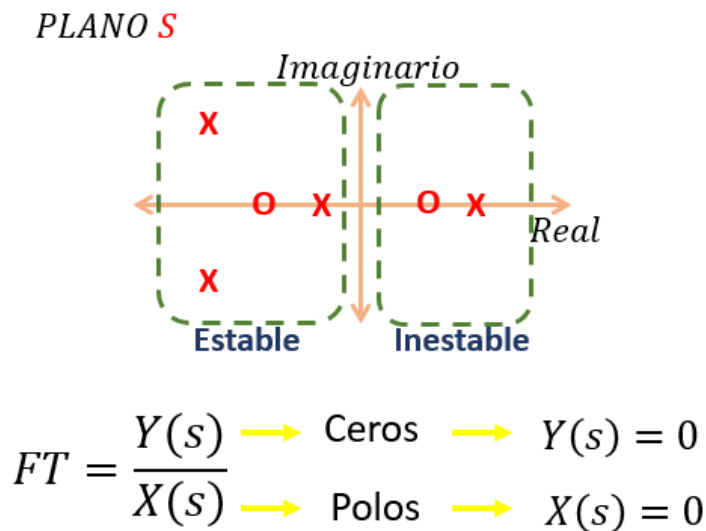


Figura 14. Estabilidad de un sistema de acuerdo a la ubicación de los polos en el plano complejo s.
Fuente: (Castaño Giraldo, 2021).

También existen unos casos especiales, por ejemplo, un sistema es limitadamente o críticamente estable si hay un polo en el origen y los demás polos en el semiplano negativo. En cambio, si existe más de un polo en el origen el sistema se vuelve inestable, **Figura 15**.

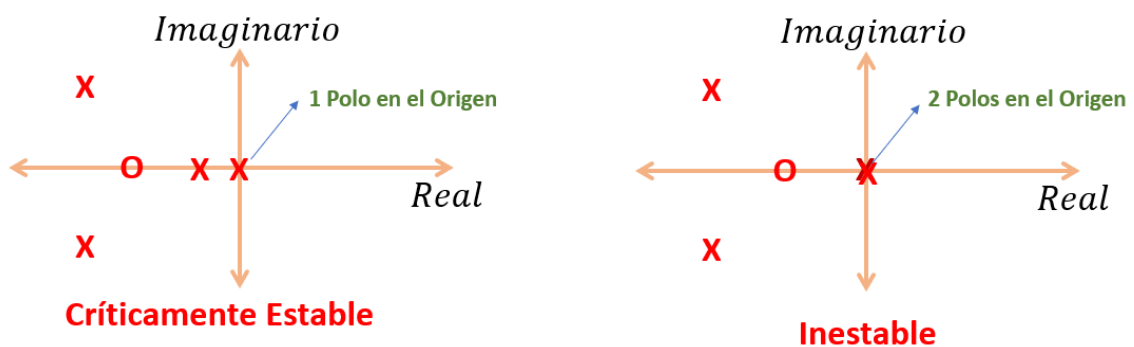


Figura 15. Sistema críticamente estable e inestable.
Fuente: (Castaño Giraldo, 2021).

4.2.4 Sistemas de primer orden

Para la sintonización y el diseño de un sistema de control, es necesario contar con información del comportamiento dinámico del proceso a controlar, normalmente representado en forma de un modelo matemático de orden reducido (primer o segundo orden).

Los sistemas de primer orden son aquellos que tienen un solo polo y están representados por ecuaciones diferenciales ordinarias de primer orden, es decir, el máximo orden de la derivada es de orden uno. Considerando el caso de las ecuaciones diferenciales lineales de primer orden, con coeficientes (a_1, a_0, b_0) constantes y condición inicial cero, tenemos la **Ecuación 2** para un sistema dinámico lineal de primer orden (Castaño Giraldo, 2020).

$$a_1 \frac{dy(t)}{dt} + a_0 y(t) = b_0 u(t), \quad \text{con: } y(0) = 0 \quad (2)$$

Un sistema de primer orden sirve para poder expresar en forma matemática y simple el comportamiento de un proceso o sistema real a lo largo del tiempo cuando se aplica alguna señal de excitación en sus entradas.

4.2.4.1 Función de transferencia de primer orden

La **Ecuación 3** representa el modelo de una función de transferencia de primer orden, más retardo o tiempo muerto:

$$G(s) = \frac{K}{\tau s + 1} e^{-\theta s} \quad (3)$$

Donde:

- K : ganancia estática del sistema
- τ : constante de tiempo del sistema en [s]
- θ : retardo de tiempo del sistema en [s]

Para identificar un sistema de primer orden representado por una función de transferencia, basta con observar el denominador, donde el máximo exponente de la variable compleja “s” debe ser igual a 1. El tiempo que le toma al sistema en llegar al estado estable, denominado tiempo de establecimiento (t_s) se calcula empleando la **Ecuación 4**.

$$t_s = 4\tau \quad (4)$$

4.2.5 Sistemas de segundo orden

Los sistemas de segundo orden son aquellos que tienen dos polos y están representados por ecuaciones diferenciales ordinarias de segundo orden, es decir, el máximo orden de la derivada es de orden dos. Considerando el caso de las ecuaciones diferenciales lineales de segundo orden, con coeficientes (a_2, a_1, a_0, b_0) constantes y condición inicial cero, tenemos la **Ecuación 5** para un sistema dinámico lineal de segundo orden (Castaño Giraldo, 2020).

$$a_2 \frac{d^2 y(t)}{dt^2} + a_1 \frac{dy(t)}{dt} + a_0 y(t) = b_0 u(t), \quad \text{con: } y(0) = 0 \quad (5)$$

4.2.5.1 Función de transferencia de segundo orden

La **Ecuación 6** representa el modelo de una función de transferencia de segundo orden:

$$G(s) = K \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (6)$$

Donde:

- K : ganancia estática del sistema
- ω_n : frecuencia natural no amortiguada expresada en [rad/s]
- ζ : factor de amortiguamiento

Dependiendo del valor que tome el factor de amortiguamiento (ζ) el sistema tendrá diversos comportamientos como los que se describe a continuación:

- Si $\zeta = 0$ el sistema será oscilatorio.
- Si $0 < \zeta < 1$ el sistema será subamortiguado.
- Si $\zeta = 1$ el sistema será críticamente amortiguado.
- Si $\zeta > 1$ el sistema será sobreamortiguado.
- Si $\zeta < 0$ el sistema será inestable.

4.2.6 Retardo o tiempo muerto de un sistema

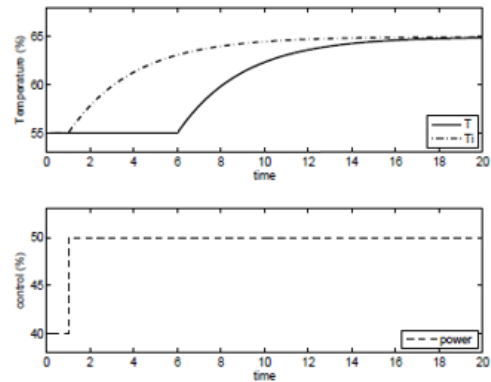
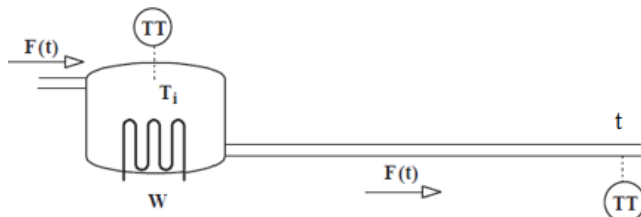
Los retardos de tiempo son fenómenos comunes en procesos industriales y sistemas térmicos. En el control de procesos, el retardo o tiempo muerto se refiere al tiempo que transcurre desde que estimulamos al sistema con alguna señal de entrada hasta que percibimos un cambio en la variable de salida.

En la **Figura 16**, se observa dos ejemplos de retardo. En el retardo aparente, se tiene un sistema de un tanque con una resistencia calefactora la cual va a calentar el fluido al interior del mismo; por lo tanto, si se tuviera un sensor de temperatura ubicado en el tanque, al momento de encender la resistencia inmediatamente se observaría el cambio de temperatura, como el que se muestra entre líneas entrecortadas. Sin embargo, como existe un sensor ubicado al final de la tubería, al aplicar el estímulo hay que esperar que toda esa energía de calentamiento vaya calentando el fluido que va circulando a través de la tubería hasta finalmente llegar al sensor, detectando así la variación de temperatura solo un tiempo después de haber aplicado dicho estímulo en la entrada, esto se conoce como retardo del sistema, el cual se ve representado en la gráfica con una línea continua.

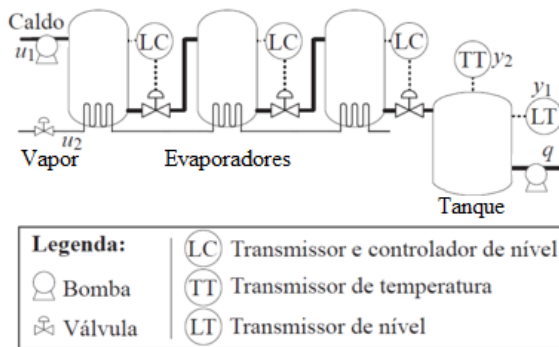
También, se tiene como ejemplo un sistema de evaporadores conectados en serie (ver **Figura 16**), el cual presenta un retardo aparente, dado por la acumulación de varios subsistemas. Si analizamos el primer evaporador, vemos que la respuesta en la salida de este

subsistema es casi inmediata; sin embargo, si analizamos el último evaporador observamos que éste presenta un retardo en su salida mucho mayor a los anteriores.

Retardo de Transporte



Retardo Aparente



Legenda:	LC	Transmissor e controlador de nivel
Bomba	TT	Transmissor de temperatura
Válvula	LT	Transmissor de nivel

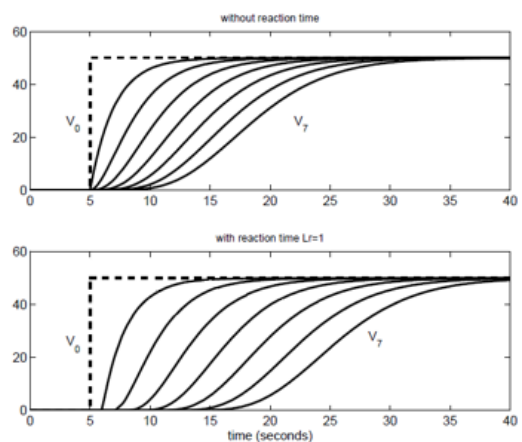


Figura 16. Retardo de transporte y retardo aparente en un sistema.

Fuente: (Castaño Giraldo, 2021).

En sistemas lineales continuos e invariantes en el tiempo, el retardo de un sistema viene representado por $e^{-\theta s}$, donde θ es el retardo de tiempo. De forma frecuencial podemos hacer una representación polar de este sistema, donde podemos notar de la **Ecuación 7**, que al pasar al dominio de la frecuencia sustituyendo s por $j\omega$, observaremos que frecuencialmente la magnitud del retardo siempre será unitaria y únicamente lo que hace es modificar la fase del sistema.

$$e^{-\theta s} = e^{-j\omega\theta} = 1 \angle -\omega\theta \quad (7)$$

En los sistemas de control, el retardo es crítico ya que afecta significativamente a la estabilidad del sistema; por lo cual, si se desea diseñar un controlador para un sistema con retardo, se debe crear estrategias de control ideales para tratar este problema, como por ejemplo controladores predictivos basados en modelo.

4.3 Capítulo 3: Diseño del controlador clásico PID y avanzado MPC

Conforme la humanidad ha ido evolucionando a lo largo del tiempo, se ha visto en la necesidad de implementar nuevas tecnologías que permitan resolver los problemas que surgen a diario. En la industria, antiguamente la obtención de un producto se lo realizaba de forma manual, requiriendo de personal que operara la maquinaria de producción, en el cual el operador humano era el encargado de observar las variables del proceso y de corregir los errores producidos cuando estas variables se desviaban del valor deseado; esto resultaba poco eficiente, por lo que nace la idea de automatizar dichos procesos con el fin de mejorar la producción, generar un ahorro económico y ser más competitivos.

En varios de los procesos industriales el objetivo es mantener una variable en su setpoint o valor de referencia, mediante la manipulación de las variables de entrada (elementos actuadores) de la planta; para lo cual los controladores proporcionales, integrales y derivativos (PID) solucionan bien la mayoría de estos problemas de control monovariable (caudal, presión, nivel, temperatura, concentración, etc.). Sin embargo, algunos procesos industriales presentan dinámicas complejas, con retardos, restricciones, perturbaciones y múltiples variables, en los que una entrada afecta a más de una salida, y recíprocamente una salida es afectada por más de una entrada, como es el caso del sistema de tanques acoplados o interconectados, los cuales necesitan de una estrategia o algoritmo de control avanzado, debido a que en ocasiones estos problemas son tratados de forma ineficiente por las técnicas de control clásico como el PID (Gutarra, 2016).

4.3.1 Control PID

El control Proporcional Integral Derivativo (PID) es una estrategia de control ampliamente utilizado en las industrias, debido a la simplicidad de su ajuste en los parámetros para obtener un buen desempeño en el lazo de control. Tiene como objetivo calcular el error del sistema entre el valor deseado y el valor medido por el sensor, para aplicar una acción correctiva que ajuste al proceso a través de un actuador.

La **Ecuación 8** corresponde a la ecuación en tiempo continuo del controlador PID.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (8)$$

Donde:

- $u(t)$: señal de control que va al actuador o variable manipulada del proceso
- K_p : constante proporcional
- $e(t)$: señal de error

- K_i : constante integral
- τ : variable de integración
- K_d : constante derivativa
- t : tiempo actual

Los parámetros fundamentales del control PID son tres: proporcional (P), integral (I) y derivativo (D). La estructura de este controlador se muestra en la **Figura 17**.

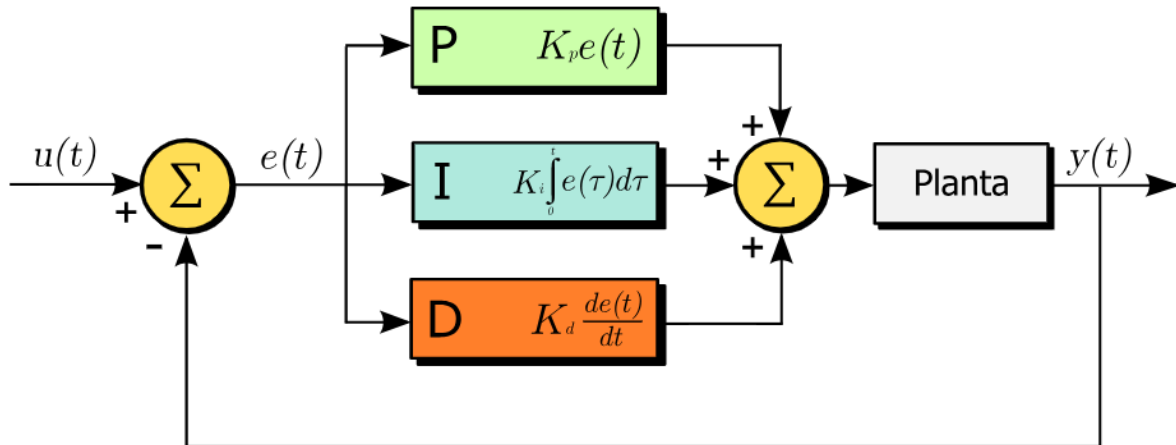


Figura 17. Diagrama de bloques de un controlador PID.

Fuente: (Sánchez Ortiz, 2022).

4.3.1.1 Acción de control proporcional

La acción proporcional (P) genera una salida que es proporcional al error actuante, incrementando la variable de control cuando el error (producto de la diferencia entre el valor deseado y el valor medido) es grande. La respuesta se obtiene multiplicando el error por una constante K_p , de acuerdo a la **Ecuación 9**.

$$P_{out} = K_p e(t) = K_p (r(t) - y(t)) \quad (9)$$

Donde:

- P_{out} : salida proporcional
- K_p : constante proporcional
- $e(t)$: señal de error
- $r(t)$: valor de referencia o setpoint
- $y(t)$: salida del sistema (valor medido)

Es importante considerar a la hora de sintonizar el controlador que una constante de proporcionalidad muy grande puede ocasionar una salida no deseada e incluso volver al sistema inestable.

4.3.1.2 Acción de control integral

La acción integral (I) elimina los errores de estado estable y permite un completo rechazo a las perturbaciones constantes, actuando como un filtro de ruido. La respuesta corresponde a la suma del error instantáneo sobre el tiempo, considerando el error anterior, es decir, toma los datos del error pasado y los va integrando en el tiempo; esta acción de control viene dada por la **Ecuación 10**.

$$I_{out} = K_i \int_0^t e(\tau) d\tau \quad (10)$$

Donde:

- I_{out} : salida integral
- K_i : constante integral
- τ : variable de integración

El error integral acelera el proceso de respuesta de la planta, pero al acumular errores puede causar sobreimpulsos.

4.3.1.3 Acción de control derivativo

La acción derivativa (D) anticipa hacia dónde va el proceso mediante la observación de la rapidez en el cambio del error. Esta predicción es hecha simplemente calculando la derivada del error en el instante de tiempo actual; por lo tanto, la salida derivativa del sistema viene expresada por la **Ecuación 11**.

$$D_{out} = K_d \frac{de(t)}{dt} \quad (11)$$

Donde:

- D_{out} : salida derivativa
- K_d : constante derivativa
- $\frac{de(t)}{dt}$: derivada del error en función del tiempo

Con la acción derivativa activa se puede aumentar la velocidad de respuesta del controlador, al incrementar la ganancia proporcional y disminuir la constante de tiempo integral. Sin embargo, al acelerar demasiado el sistema, se pueden generar grandes perturbaciones.

En la **Tabla 1** se muestra los efectos que se tiene en la respuesta de un sistema al incrementar o disminuir cada uno de los parámetros del controlador PID.

Tabla 1. Efectos ocasionados en la salida del sistema al incrementar cada parámetro del PID de forma independiente.

Parámetro	Tiempo de Levantamiento	Sobreimpulso	Tiempo de Estabilización	Error de Estado Estacionario	Estabilidad
K_p	Disminuye	Aumenta	Cambio menor	Disminuye	Disminuye
K_i	Disminuye	Aumenta	Aumenta	Se elimina	Disminuye
K_d	Cambio menor	Disminuye	Disminuye	Sin efecto	Mejora si K_d es pequeño

Fuente: (Sánchez Ortiz, 2022).

4.3.2 Control Predictivo basado en Modelo (MPC)

El control predictivo basado en modelo (Model Predictive Control, MPC) es una técnica de control avanzado muy popular en la industria debido a su adaptabilidad a una amplia gama de procesos, a su rendimiento en comparación con otras técnicas de control y apertura que le permite cumplir con los requisitos del diseño. Este controlador es uno de los preferidos a nivel industrial cuando se desea regular procesos complejos, debido a que incorpora la formulación del problema, el uso de un modelo para predecir el comportamiento futuro de la planta bajo estudio, las limitaciones físicas de la planta y la seguridad del proceso en su acción de control; permitiendo así optimizar la energía consumida y reducir los costos de operación.

Las principales ventajas que presenta el MPC frente al control clásico son las siguientes:

- Permite resolver problemas de control con múltiples entradas y múltiples salidas (MIMO), así como sistemas con una entrada y una salida (SISO). También es capaz de manejar sistemas con mayor número de salidas que de entradas.
- Puede ser aplicado a procesos lineales y no lineales donde la estabilidad y robustez en muchas ocasiones son difíciles de obtener.
- Su formulación toma en cuenta los valores máximos y mínimos de operación de los actuadores, así como la variación o incremento que pueden tener las variables controladas y manipuladas.
- Puede operar en el límite de las restricciones, lo que lleva a tener condiciones más óptimas, permitiendo utilizar las variables del sistema en todo su rango de operación e incluso saturarlas cuando es necesario (Oviden, 2016).
- Presenta una buena respuesta en sistemas con dinámicas complejas tales como: sistemas con retardo, con respuesta inversa y sistemas inestables (Oviden, 2016).
- Minimiza costos y maximiza el rendimiento.

Según Oviden (2016) esta técnica de control también presenta algunas desventajas, mencionadas a continuación:

- Necesita de un modelo dinámico con un alto índice de confiabilidad.
- Conlleva un costo computacional elevado. Sin embargo, si no hay restricciones o el sistema presenta una dinámica relativamente lenta, el costo computacional disminuye notablemente.
- Requiere un algoritmo de optimización online, lo que conlleva a tener altos costos en la implementación del controlador.
- Se suele aplicar en sistemas de dinámica lenta por el costo computacional. Sin embargo, hoy en día también se aplica a sistemas con dinámica rápida debido a los avances tecnológicos y a la evolución de la técnica de control (Guzmán, 2018).

Inicialmente el controlador MPC fue desarrollado para solucionar problemas de control en plantas químicas y refinerías de petróleo, pero actualmente puede encontrarse en la mayoría de industrias, entre ellas: metalúrgica, alimentaria, automotriz, aeroespacial, robótica y otras.

El control predictivo (ver **Figura 18**) es una técnica de control que utiliza un modelo matemático del proceso (modelo de predicción) con el fin de predecir el valor que tendrá las variables de salida a lo largo de un intervalo de tiempo. Por lo tanto, el controlador actúa en el instante actual, basándose en la predicción del comportamiento futuro del proceso (Moromenacho & Rosero, 2010).

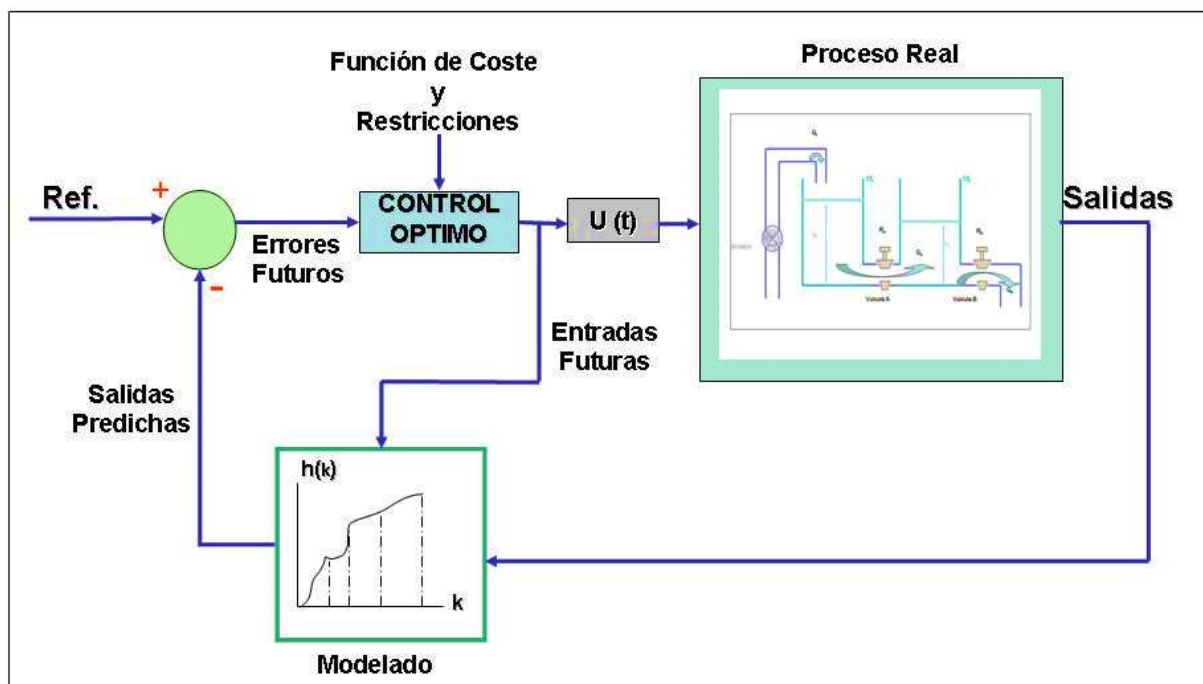


Figura 18. Control predictivo en tanques acoplados.

Fuente: (Moromenacho & Rosero, 2010).

4.3.2.1 Principio de funcionamiento del control predictivo

De acuerdo a Sánchez (2018) “el control predictivo no es una estrategia de control específica, sino una amplia gama de métodos de control que hacen uso explícito de un modelo para obtener una señal de control que permita optimizar una función objetivo”. En la **Figura 19**, se muestra la estrategia de diseño de toda la familia de controladores (algoritmos de control) pertenecientes al MPC.

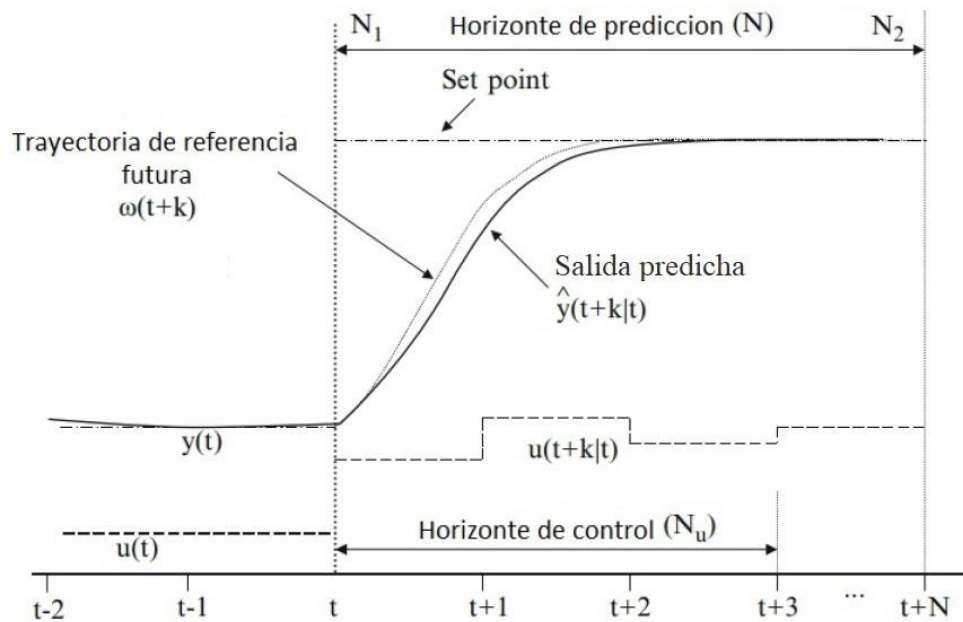


Figura 19. Estrategia del control predictivo basado en modelo.

Fuente: (Tlachi, 2018).

A continuación, se describe la notación matemática que se usará para explicar el principio básico de funcionamiento del controlador predictivo, el cual se caracteriza por la siguiente estrategia, resumida en tres pasos:

- $u(t)$: entrada del proceso (variable manipulada)
- $y(t)$: salida del proceso (variable controlada)
- t : tiempo actual
- $t + k$: tiempo futuro
- $u(t + k|t)$: valores futuros de la variable manipulada
- $\hat{y}(t + k|t)$: predicciones de la variable controlada
- $w(t + k)$: trayectoria de referencia futura
- N : horizonte de predicción

1. Utilizando el modelo del proceso, en cada instante de tiempo t , se predicen las salidas futuras del sistema para un horizonte determinado N , llamado horizonte de predicción. La predicción de estas salidas dada por $\hat{y}(t + k|t)$ para $k = 1, 2, \dots, N$ depende de los valores conocidos de las entradas y salidas pasadas hasta el instante t , y de las señales de control futuras $u(t + k|t)$ para $k = 0, \dots, N - 1$ que han de ser calculadas y enviadas al sistema.
2. Las secuencias de control a aplicar en el futuro se calculan optimizando un criterio determinado (función de costo o función objetivo), para mantener el proceso lo más cerca posible a la trayectoria de referencia $w(t + k)$, la cual describe cómo se quiere guiar la salida del proceso desde su valor actual $y(t)$ hasta el punto de operación o setpoint. El criterio mencionado, regularmente es una función cuadrática de los errores entre la salida predicha y la trayectoria de referencia futura; en algunos casos, el esfuerzo de control realizado también se incluye en la función objetivo. Si el modelo es lineal, el criterio es cuadrático y no existen restricciones, se obtiene una solución de control explícita, es decir, se obtiene una fórmula para calcular el valor de las entradas a futuro que me permite minimizar la función de costo; caso contrario, es necesario aplicar algoritmos numéricos para la optimización, lo cual incrementa la carga computacional (Tlachi, 2018).
3. En el instante de tiempo actual, se aplica al proceso real únicamente el primer comando de la secuencia de control óptimo obtenida $u(t|t)$, despreciando el resto de las señales de control calculadas, ya que en el siguiente instante de muestreo todas las secuencias de tiempo se desplazan y se obtiene una nueva medición de la salida $y(t + 1)$, repitiendo el procedimiento nuevamente. Esto lleva a una nueva entrada de control $u(t + 1|t + 1)$, la cual es generalmente diferente de la calculada previamente $u(t + 1|t)$, aplicando así lo que se conoce como horizonte deslizante.

La analogía entre la técnica de control predictivo y la estrategia empleada al conducir un vehículo (ver **Figura 20**) puede ayudar a explicar de mejor manera cómo funciona el MPC. Al pilotar el vehículo, el conductor conoce su trayectoria planificada para un horizonte de control finito N , en base a esto determinará las secuencias de control a realizar (acelerar, frenar, cambiar de marcha y maniobrar) para seguir la trayectoria deseada a lo largo del recorrido teniendo en cuenta las características del vehículo y el entorno (modelo mental). Sólo la primera acción de control de la secuencia calculada mentalmente es ejecutada por el conductor

en cada instante de tiempo actual t , y el procedimiento se repite en los sucesivos instantes $t + 1, \dots, t + N$ utilizando el concepto de horizonte deslizante (Sendoya, 2013).

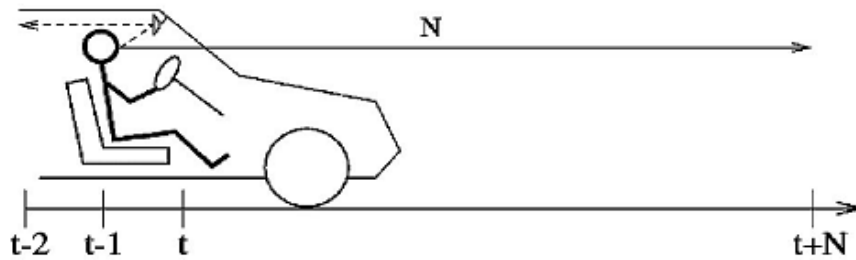


Figura 20. Analogía del MPC con la conducción de un automóvil.

Fuente: (Sendoya, 2013).

4.3.2.2 Elementos básicos del control predictivo

El control predictivo basado en modelo posee principalmente cuatro elementos, los cuales se detallan a continuación:

- Modelo del proceso
- Modelo de las perturbaciones
- Función objetivo
- Obtención de la ley de control

4.3.2.2.1 Modelo del proceso

El modelo del proceso es aquel a partir del cual se calcula las predicciones $\hat{y}(t + k|t)$, y debe representar la dinámica del sistema a controlar lo más preciso posible, para garantizar el buen desempeño y la fiabilidad del MPC.

Las diferentes estrategias de control predictivo están basadas en diferentes modelos del proceso. Algunas de ellas se mencionan a continuación:

- **Respuesta al escalón:** Utilizado por el DMC y sus variantes, este modelo se basa en la respuesta al escalón del proceso, cuya salida está dada por:

$$y(t) = y_0 + \sum_{i=1}^N g_i \Delta u(t - i) \quad (12)$$

Donde:

- y_0 : valor actual de la salida del proceso
- g_i : muestreos de los valores de la salida a consecuencia de la entrada escalón $\Delta u = u(t) - u(t - 1)$
- N : horizonte de predicción

Tomando a los valores de $y_0 = 0$, entonces la función de predicción viene dada por:

$$\hat{y}(t+k|t) = \sum_{i=1}^N g_i \Delta u(t+k-i|t) \quad (13)$$

- **Función de transferencia:** Utilizado por el GPC y otros, su concepto se basa en la función de transferencia $G = B/A$ cuya salida es:

$$A(z^{-1})y(t) = B(z^{-1})u(t) \quad (14)$$

Donde:

- A y B : son polinomios de n_a y n_b respectivamente

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{n_a} z^{-n_a}$$

$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + \dots + b_{n_b} z^{-n_b}$$

La función de predicción viene dada por:

$$\hat{y}(t+k|t) = \frac{B(z^{-1})}{A(z^{-1})} u(t+k-i|t) \quad (15)$$

- **Espacio de estados:** Utilizado por el PFC, se basa en un modelo en espacio de estados, de la forma:

$$x(t) = Ax(t-1) + Bu(t-1) \quad (16)$$

$$y(t) = Cx(t) \quad (17)$$

Donde:

- x : vector de estados
- A : matriz de estados
- B : matriz de control
- C : matriz de salida

El modelo de predicción viene dado por:

$$\hat{y}(t+k|t) = C\hat{x}(t+k|t) = C \left[A^k x(t) + \sum_{i=1}^k A^{i-1} B u(t+k-i|t) \right] \quad (18)$$

4.3.2.2 Modelo de las perturbaciones

Dado que es importante tener en cuenta la aparición de ruidos, entradas no medibles y errores en el modelado del proceso, el estudio del modelo de las perturbaciones es crucial. Según Moromenacho & Rosero (2010), el ARIMA (Autorregresivo Integrado de Media Móvil) es uno de los modelos más populares para el estudio de las perturbaciones, el cual tiene en cuenta dos tipos diferentes de perturbaciones: el movimiento browniano (en procesos con

equilibrio energético) y los cambios aleatorios que se producen en instantes aleatorios (cambio en la calidad del material).

La diferencia entre la salida medida y la calculada por el modelo es:

$$n(t) = \frac{C(z^{-1})e(t)}{D(z^{-1})} \quad (19)$$

Para una perturbación constante, la expresión está dada por:

$$n(t) = \frac{e(t)}{1 - z^{-1}} \quad (20)$$

Donde:

- $D(z^{-1})$: incluye al integrador $\Delta = 1 - z^{-1}$ con el fin de tener un error nulo en estado estable.
- $e(t)$: ruido blanco (ruido entorno a una media de cero).
- $C(z^{-1})$: igual a 1.

La predicción está dada por:

$$\hat{n} = (t + k|t) = n(t) \quad (21)$$

4.3.2.2.3 Función objetivo

La función objetivo o función de costo expresa el criterio a optimizar, cuyo objetivo es que la salida futura del proceso \hat{y} siga a una trayectoria de referencia deseada w , dentro de un horizonte establecido, al mismo tiempo que penaliza al esfuerzo de control Δu necesario para conseguirlo, con el fin de gastar menos energía. Esta función está definida por la **Ecuación 22**, donde en el primer sumatorio se busca minimizar el error cuadrático provocado por la diferencia entre el setpoint y la salida del proceso, mientras que el segundo sumatorio busca minimizar el esfuerzo de control aplicado sobre el sistema.

$$J = \sum_{j=N_1}^{N_2} \delta(j) [\hat{y}(t + j|t) - w(t + j)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta u(t + j - 1)]^2 \quad (22)$$

Donde:

- N_1 : límite mínimo de predicción
- N_2 : límite máximo de predicción
- N_u : horizonte de control
- $\delta(j), \lambda(j)$: pesos, δ pondera la importancia que deseo darle a la minimización del error y el coeficiente λ pondera la importancia del esfuerzo de control.
- $w(t + j)$: aproximación de la trayectoria de referencia futura.

Una de las potencialidades que tiene el control MPC con relación al control clásico, es el poder involucrar las restricciones del proceso en el cálculo de la ley de control. Esto debido a que la operación de los procesos industriales casi siempre están sujetas a restricciones físicas, operacionales o de seguridad (González & González, 2023).

El MPC generalmente toma en cuenta restricciones en la señal de control (u), en el incremento de la acción de control (Δu) y en la salida (y).

4.3.2.2.4 Obtención de la ley de control

La ley de control se define por $u(t + k|t)$ y se obtiene de:

- Optimizar la función de costo J , para ello es necesario calcular $\hat{y}(t + k|t)$ en función de las señales de control futuras, y de las entradas y salidas pasadas, empleando el modelo del proceso.
- Cuando el modelo es lineal y no existen restricciones, la solución es analítica, y en otros casos la solución será mediante métodos iterativos.

Se recomienda diseñar una ley de control que reduzca el número de grados de libertad en el horizonte de predicción, ya que esto mejora la robustez y el comportamiento del sistema, para evitar la obtención de señales de control de alta frecuencia no deseadas que provocarían la inestabilidad del sistema. Debido al uso de la estrategia de horizonte deslizante, la señal de control $u(t + k|t)$ no permanece constante, ya que ésta debe calcularse en cada intervalo de muestreo, como se muestra en la **Figura 21**.

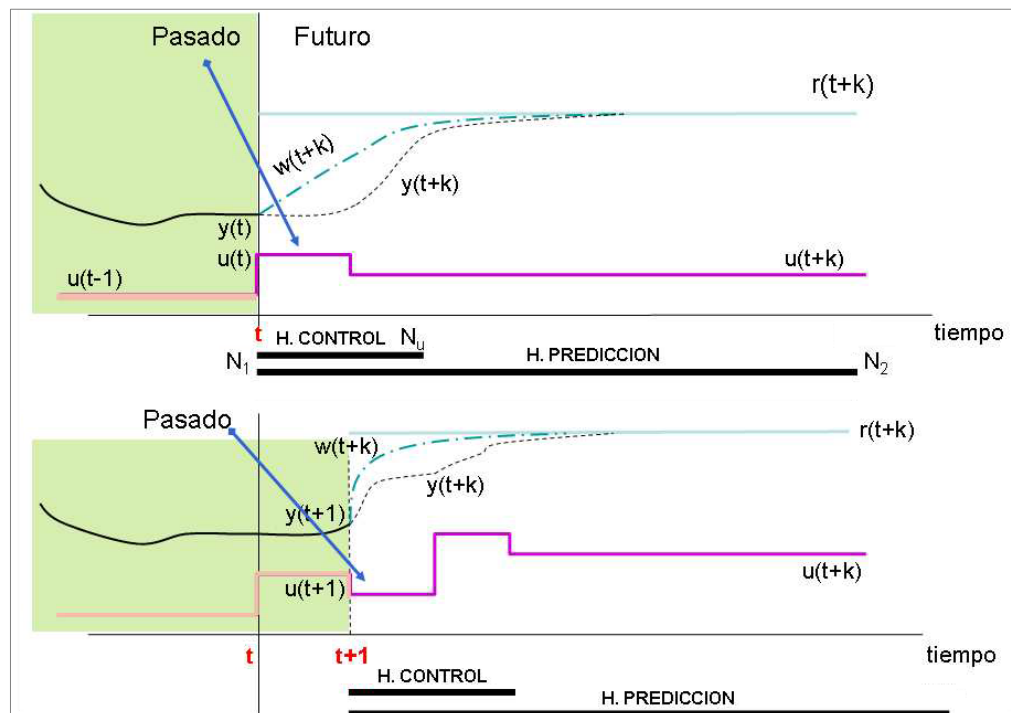


Figura 21. Horizonte deslizante en la ley de control.
Fuente: (Moromenacho & Rosero, 2010).

4.3.3 MPC Toolbox - MATLAB®

Model Predictive Control Toolbox es una herramienta de MATLAB® que proporciona funciones, una interfaz gráfica de usuario y bloques de Simulink para el diseño y la simulación de controladores predictivos basados en modelos.

Además, la aplicación MPC Designer permite diseñar y simular controladores predictivos basados en modelos, tanto en MATLAB® como en Simulink, mediante la cual se podrá:

- Diseñar de forma interactiva controladores predictivos basados en modelos y validar su rendimiento utilizando escenarios de simulación.
- Obtener modelos de planta lineales, mediante la linealización de modelos de Simulink.
- Revisar el diseño de los controladores para detectar posibles problemas numéricos o de estabilidad, en tiempo de ejecución.
- Comparar los gráficos de respuesta para múltiples controladores predictivos.
- Generar modelos en Simulink con un controlador MPC y un modelo de la planta.
- Generar scripts en MATLAB® para automatizar el diseño del controlador MPC y las tareas de simulación.

El MPC Toolbox requiere que el modelo del proceso sea lineal e invariante en el tiempo (LTI), el cual puede ser definido como: función de transferencia, espacio de estados, polos y ceros, modelo obtenido mediante la herramienta identificación de sistemas de MATLAB® o linealizando un modelo en Simulink (Moromenacho & Rosero, 2010).

4.3.3.1 Parámetros del MPC

Para diseñar correctamente el controlador MPC se necesitan varios parámetros:

- Estructura del MPC (número de entradas, salidas, perturbaciones medidas y no medidas).
- Modelo interno de la planta.
- Tiempo de muestreo.
- Horizonte de predicción.
- Horizonte de control.
- Restricciones.
- Pesos.
- Robustez o agresividad del controlador.

Mediante la aplicación MPC Designer que incorpora el Model Predictive Control Toolbox, se puede configurar todos estos parámetros.

4.3.3.1.1 Estructura del MPC y modelo interno de la planta

En primer lugar, para definir la estructura del MPC es necesario seleccionar cuáles y cuántas son las entradas y salidas del sistema; así como las perturbaciones medidas y no medidas, en caso de existir.

Las salidas medidas (MO, measured outputs) son las señales que deben mantenerse en el punto de consigna deseado, las variables manipuladas (MV, manipulated variables) son las señales que el MPC ajusta para alcanzar sus objetivos de control y las perturbaciones medidas (MD, measured disturbances) son señales que el controlador no puede ajustar, pero las utiliza para la compensación feedforward (control anticipativo), el cual va a intentar atenuar o eliminar de la mejor manera posible el efecto que esta perturbación está causando a nuestro proceso. El número de estas entradas, salidas y perturbaciones se asignan dando clic en el botón “Change I/O Sizes” de la **Figura 22**.

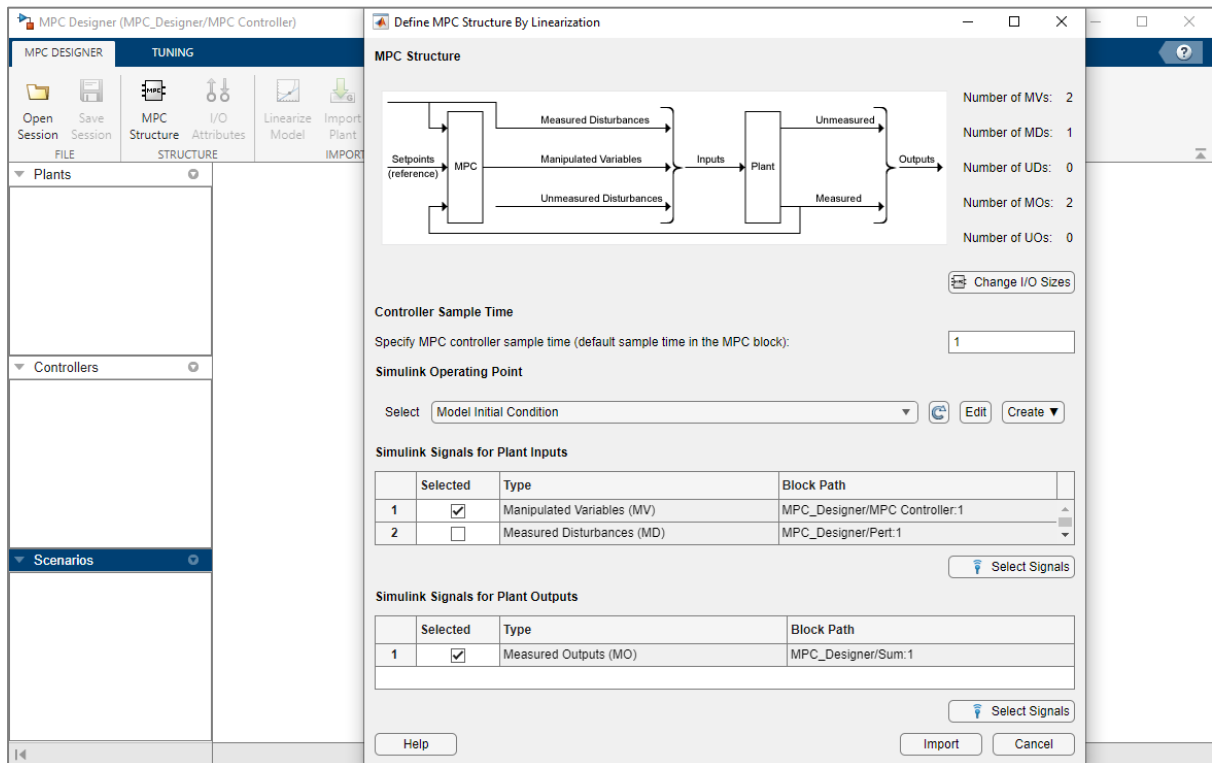


Figura 22. Cuadro de diálogo para definir la estructura del MPC y linealizar el modelo.

4.3.3.1.2 Tiempo de muestreo, horizonte de predicción y horizonte de control

Los tres parámetros principales de los controladores MPC son el tiempo de muestreo, el horizonte de predicción y el horizonte de control. De ellos dependen el esfuerzo computacional y la eficiencia del controlador.

Cualitativamente, a medida que disminuye el tiempo de muestreo, el rechazo de perturbaciones desconocidas suele mejorar; sin embargo, a medida que el tiempo de muestreo

se reduce, el esfuerzo computacional aumenta drásticamente; por lo tanto, la elección óptima es un equilibrio entre rendimiento y esfuerzo de cálculo. Para el control de procesos, normalmente el tiempo de muestreo es mayor o igual a 1 s; otras aplicaciones, como la automoción o la industria aeroespacial, pueden requerir un tiempo de muestreo inferior a 1 s (Capuano, 2021).

El horizonte de predicción debe ser mayor al horizonte de control. Este parámetro se puede estimar en base al tiempo de estabilización del sistema ante una entrada escalón unitario, para sistemas que tienen una respuesta lenta. Por otra parte, los horizontes de predicción superiores a 50 rara vez son necesarios, a menos que el tiempo de muestreo sea pequeño.

El horizonte de control está comprendido entre 1 y el horizonte de predicción. Las directrices sugieren fijar el horizonte de control entre el 10 % y el 20 % del horizonte de predicción. Si el horizonte de control es demasiado grande, algunos movimientos en las variables medidas podrían no afectar a ninguna de las salidas de la planta antes del final del horizonte de predicción (Capuano, 2021).

4.3.3.1.3 Restricciones

Una de las ventajas más importantes del control predictivo basado en modelo es la optimización de la secuencia de control teniendo en cuenta las restricciones de las variables manipuladas (actuadores) y las salidas medidas del proceso. Se trata de límites físicos o límites impuestos por el diseñador del controlador para una aplicación específica.

Las restricciones pueden clasificarse en restricciones duras o blandas: las duras no pueden violarse, mientras que las blandas pueden violarse dentro de cierto intervalo tolerable. Según Capuano (2021), para suavizar una restricción, es posible establecer el valor de relajación de restricciones por igual (ECR) en un valor positivo (cero implica una restricción dura). Las directrices recomiendan los siguientes valores para suavizar las restricciones:

Tabla 2. Valores de ECR para suavizar las restricciones.

Relajación de Restricciones por Igual (ECR, Equal Constraint Relaxation)	
0	No se permite ninguna violación (restricción dura)
0,05	Se permite una infracción muy pequeña (restricción casi dura)
0,2	Infracción pequeña permitida (restricción bastante dura)
1	Suavidad media
5	Se permite una infracción superior a la media (restricción bastante suave)
20	Infracción grande permitida (restricción muy suave)

Fuente: (Capuano, 2021).

A veces, las perturbaciones y los errores de predicción pueden provocar violaciones inesperadas de las restricciones en un sistema real. Intentar evitar estas violaciones endureciendo las restricciones suele degradar el rendimiento del controlador (Capuano, 2021).

4.3.3.1.4 Pesos

El control predictivo basado en modelo resuelve un problema de optimización en cada intervalo de tiempo. Para lograr la optimización, se minimiza una función de costo o también llamada función objetivo en la que cada variable de entrada o salida tiene su propio peso. Cuanto mayor sea el peso de la variable, más tenderá el controlador a mantenerla cerca de su objetivo para minimizar la función de costo.

Los pesos de las salidas medidas deben ser mayores que cero, si se desea considerar todas esas salidas en la función de costo. Por el contrario, los pesos de las variables manipuladas deben ser distintos de cero, sólo si éstas tienen objetivos. Se sugiere dar prioridad al seguimiento de las variables medidas o controladas (Capuano, 2021). Las directrices para la prioridad de los pesos son las siguientes:

Tabla 3. Prioridad de seguimiento de las entradas y salidas del proceso, según el valor de los pesos.

Prioridad de los Pesos	
0,05	Prioridad baja: Se acepta un gran error de seguimiento
0,2	Prioridad por debajo del promedio
1	Prioridad media (por defecto)
5	Prioridad superior a la media
20	Prioridad alta: Se desea un pequeño error de seguimiento

Fuente: (Capuano, 2021).

También es importante considerar la tasa de cambio o incremento de las variables manipuladas (Δu); la simulación proporciona un rendimiento más robusto del controlador, pero un seguimiento de referencia más pobre cuanto mayor sea el valor de este peso. Por último, se debe considerar el peso de la variable de holgura (ECR Weight), ya que esta variable representa la violación de las restricciones y cuanto mayor sea con respecto a los pesos de entrada y salida, más se penaliza la violación de las restricciones. Un peso ECR excesivamente grande distorsiona la optimización de las variables manipuladas (Capuano, 2021).

4.3.3.1.5 Robustez y agresividad

Una vez elegidos los pesos, es posible hacer que el controlador sea más agresivo o más robusto, es decir que tenga una respuesta de estado estable más rápida o más lenta, moviendo la barra deslizante de la **Figura 23**.

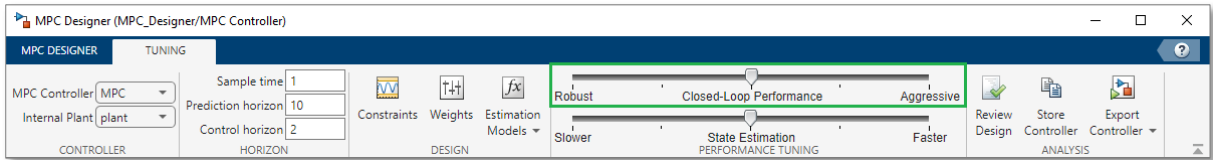


Figura 23. Barra deslizante para ajustar la robustez o agresividad del controlador.

El cambio a un control más robusto reduce las ponderaciones de las salidas y las variables medidas y aumenta la tasa de cambio de esta última, teniendo movimientos de control más conservadores. El control más agresivo, en cambio, conduce a un mejor seguimiento de las referencias con mayores variaciones de las variables manipuladas (Capuano, 2021).

4.3.4 Sintonización y optimización de controladores mediante algoritmos evolutivos

En el control de procesos, muchos de los problemas que aparecen en la industria, pueden ser vistos como problemas de optimización, dado que el objetivo final es identificar una combinación de parámetros que satisfagan las restricciones propias del problema, ya que siempre se busca maximizar o minimizar una función objetivo para garantizar índices de rendimiento eficaces. Los Algoritmos Evolutivos (AE) son métodos de búsqueda y optimización que emulan los mecanismos de selección natural y herencia genética según la teoría neo-darwinista de la evolución biológica (Hernández & Carmona, 2014).

En la **Figura 24**, se muestra el esquema general de funcionamiento de los AE.

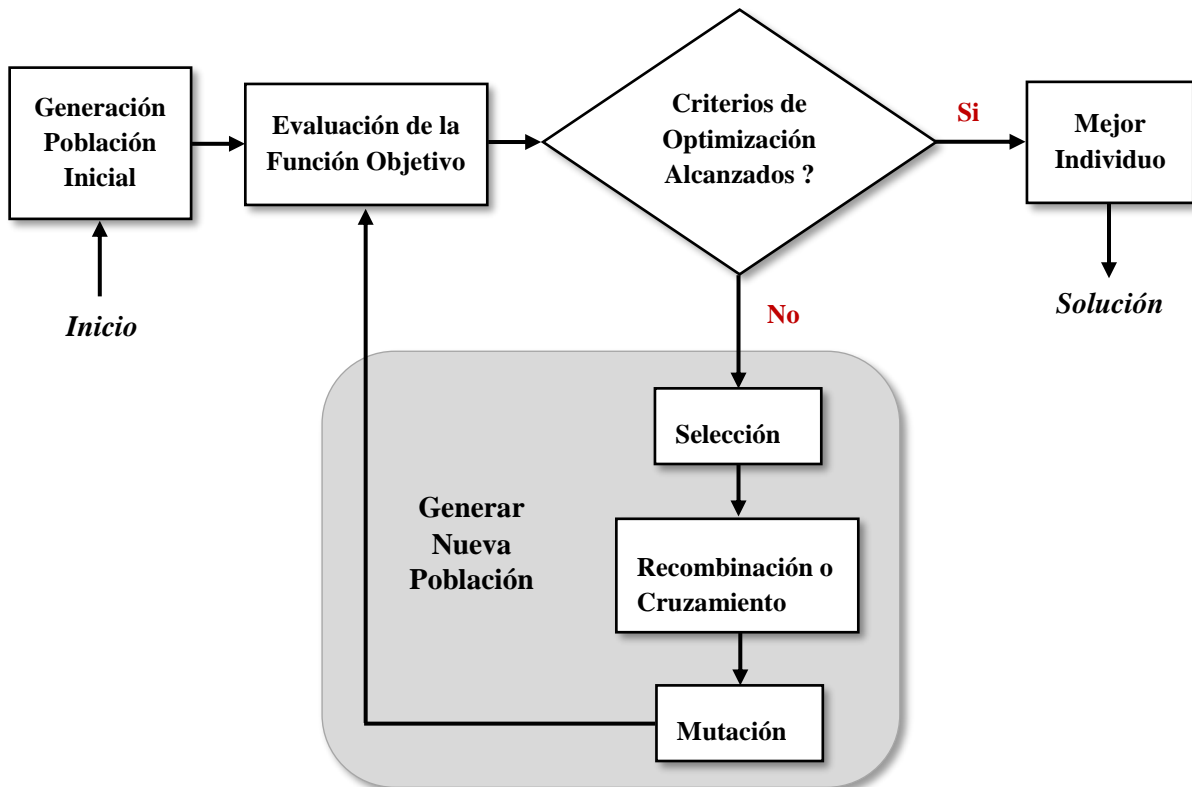


Figura 24. Esquema general de los algoritmos evolutivos.

Los algoritmos evolutivos trabajan con una población de individuos, que representan soluciones candidatas a un determinado problema. Esta población se somete a ciertas transformaciones y después a un proceso de selección que favorece a los mejores, formando una nueva generación; a partir de estos se repite el proceso de forma cíclica, hasta que después de un cierto número de generaciones el mejor individuo de la población esté cerca de la solución buscada.

4.3.4.1 Algoritmo por multidinámicas para optimización global (MAGO)

El Algoritmo por Multidinámicas para Optimización Global (MAGO, por acrónimo en inglés), fue inicialmente presentado y desarrollado por Hernández & Ospina (2010). El MAGO hace uso de tres dinámicas de población diferentes: una distribución uniforme cambiante para explorar el espacio de búsqueda, un mecanismo de propagación de diferencias relacionado con los mejores individuos y una estrategia de preservación de la diversidad. El usuario final sólo debe proporcionar el tamaño de la población y el número de generaciones (iteraciones).

A continuación, en la **Tabla 4** se presenta el pseudocódigo del algoritmo MAGO.

Tabla 4. Seudocódigo del algoritmo MAGO.

Seudocódigo MAGO	
1:	$j = 0$, Generación de la población inicial, con una distribución uniforme aleatoria en el espacio de búsqueda.
2:	Repetir.
3:	Evaluar cada individuo por medio de la función objetivo.
4:	Calcular la matriz de covarianza de la población y la primera, segunda y tercera dispersión.
5:	Calcular las cardinalidades $N1$, $N2$ y $N3$ de los grupos $G1$, $G2$ y $G3$.
6:	Seleccionar $N1$ de los mejores individuos, modificarlos de acuerdo a la función objetivo y hacerlos competir. Pasar los ganadores a la próxima generación $j + 1$.
7:	Realizar un muestreo desde la distribución uniforme en el hiper rectángulo $[LB(j), UB(j)]$ con $N2$ individuos, y pasar a la siguiente generación $j + 1$.
8:	Realizar un muestreo desde la distribución uniforme en todo el espacio de búsqueda con $N3$ individuos y pasar a la siguiente generación $j + 1$.
9:	$j = j + 1$.
10:	Hasta que se satisfaga algún criterio de terminación.

Fuente: (Hernández & Carmona, 2014).

De forma similar a los algoritmos evolutivos, el MAGO comienza con una población de soluciones potenciales que se dispersa aleatoriamente por el espacio de búsqueda y, a continuación, se divide de forma autónoma en tres subgrupos (G1, G2 y G3), cada uno de los cuales tiene su propia evolución. Para mostrar cómo se determina la cardinalidad de cada uno de los grupos: dinámica emergente (G1), dinámica de multitudes (G2) y dinámica accidental (G3); se muestra en la **Figura 25**, la optimización bidimensional de la función de Rastrigin con MAGO, la cual es una función no convexa que regularmente es empleada como test de desempeño en diversos algoritmos de optimización.

En esta gráfica (ver **Figura 25**), se muestran las cardinalidades de los grupos G1, G2 y G3 en diferentes generaciones, en la cual se puede observar, que conforme aumenta la generación, el recuadro del espacio de búsqueda de cada uno de los grupos que contiene un determinado número de individuos, se va reduciendo. La cardinalidad del grupo G1, representa el número de individuos que se encuentran dentro del recuadro interior. La cardinalidad del grupo G2, representa el número de individuos que se encuentran dentro del recuadro central, pero no dentro del interior. La cardinalidad del grupo G3, representa el número de individuos que se encuentran fuera del recuadro central, pero dentro del recuadro exterior que contiene el espacio de búsqueda.

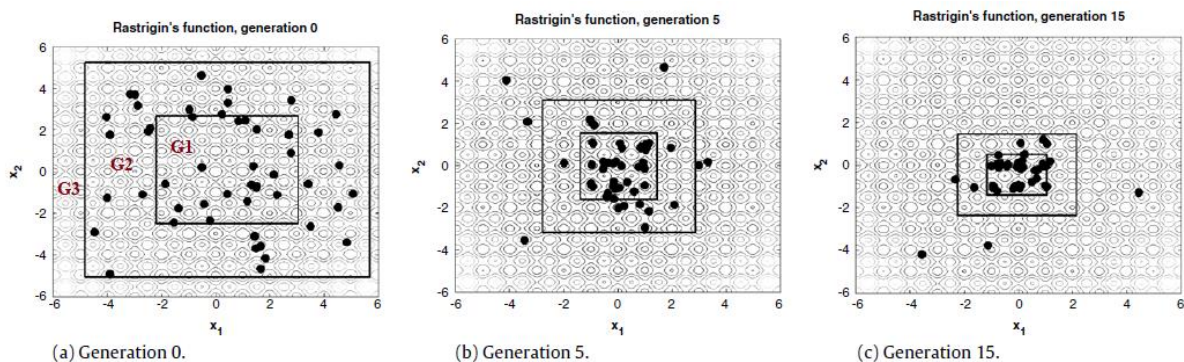


Figura 25. Optimización de la función de Rastrigin con MAGO.

Fuente: (Hernández & Ospina, 2010).

5. Metodología

El trabajo propuesto fue desarrollado empleando el método científico con un enfoque cuantitativo e investigativo aplicado, partiendo de los conocimientos adquiridos de la investigación teórica sobre la automatización y control de procesos, para el diseño y evaluación mediante simulación del controlador predictivo basado en modelo (MPC), logrando obtener los resultados deseados.

La metodología aplicada para el cumplimiento de cada uno de los objetivos, contempla tres etapas o fases, las cuales se detallan en la **Figura 26**, mediante un diagrama de flujo.

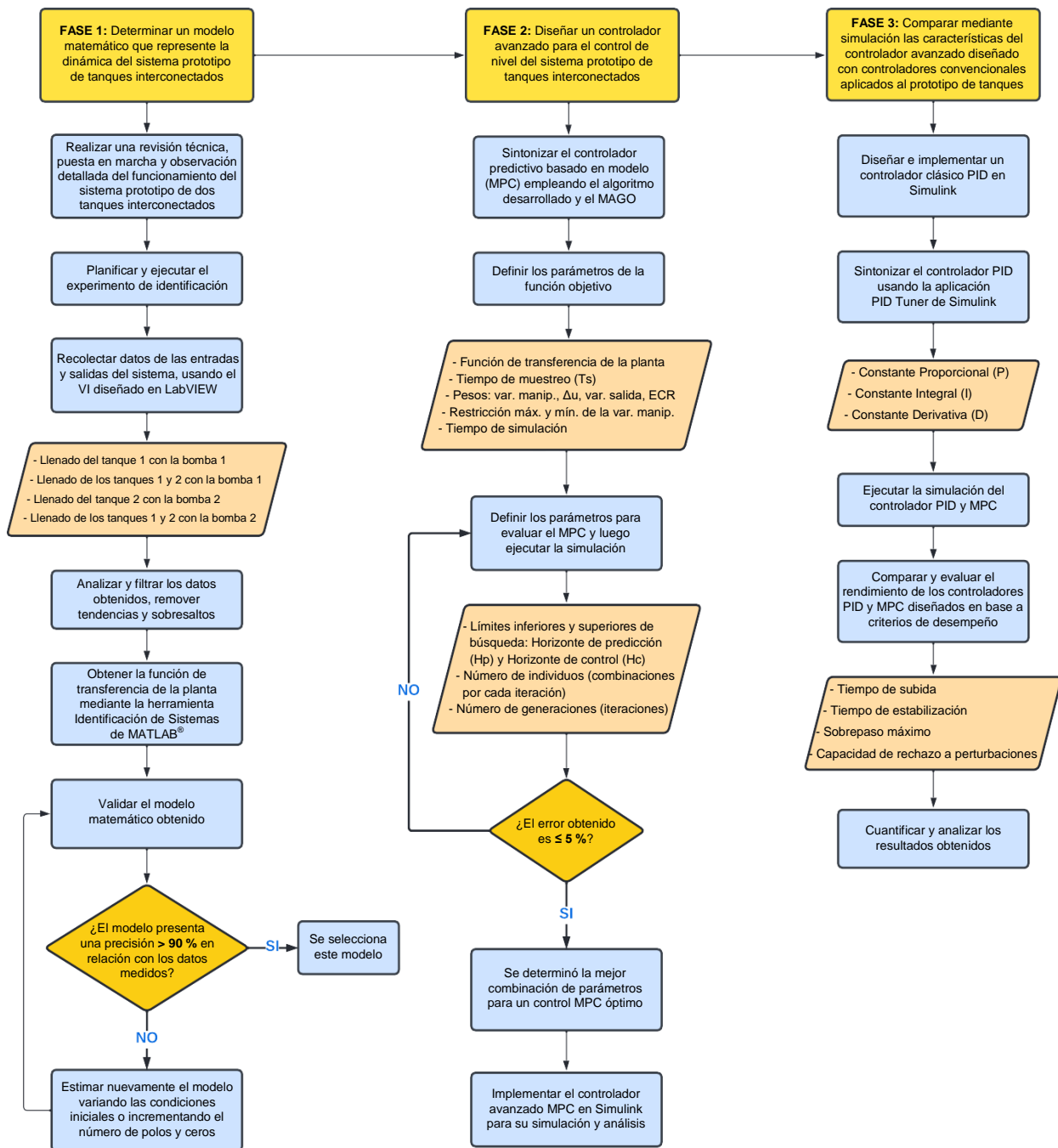


Figura 26. Metodología para el cumplimiento de los objetivos específicos.

En la primera fase, se realiza una revisión teórica y metodológica sobre identificación de sistemas, diseño de controladores convencionales y controladores predictivos basados en modelos. Seguido, se efectuó una revisión técnica, puesta en marcha y observación detallada del funcionamiento del banco de dos tanques interconectados, para luego proceder a planificar el experimento de identificación, donde se decidió llevar a cabo cuatro experimentos en diferentes configuraciones de llenado, detallados a continuación:

- Caso 1: Llenado del tanque 1 con la bomba 1, con el fin de controlar el nivel en este tanque.
- Caso 2: Llenado de los tanques interconectados 1 y 2 con la bomba 1, donde el objetivo es controlar el nivel en el segundo tanque, ya que este último presenta una dinámica compleja debido a que cada tanque recibe un caudal diferente. El primero producido por el caudal que entrega la bomba y el segundo por el acoplamiento entre ellos, presentando un retardo de tiempo.
- Caso 3: Llenado del tanque 2 con la bomba 2, con el fin de controlar el nivel en este tanque.
- Caso 4: Llenado de los tanques interconectados 1 y 2 con la bomba 2, donde el objetivo al igual que en el caso 2, es controlar el nivel en el último tanque, es decir, el tanque 1, el cual presenta un retardo.

Planeado el experimento de identificación se procedió a su ejecución, obteniendo y registrando los datos de las entradas y salidas del proceso, haciendo uso de un instrumento virtual (VI) desarrollado en LabVIEW 2019. Una vez obtenido estos datos, se realizó un análisis y filtrado de los mismos, removiendo tendencias y sobresaltos, luego se procedió a determinar la función de transferencia de la planta mediante la herramienta Identificación de Sistemas (System Identification) de MATLAB[®]. Por último, se validó el modelo y se seleccionó aquel que representa lo más preciso posible la dinámica del sistema, para cada caso planteado. Con este procedimiento se cumplió con el primer objetivo “Determinar un modelo matemático que represente la dinámica del sistema prototipo de tanques interconectados”.

En la segunda fase, se desarrolla un algoritmo en MATLAB[®], para la sintonización del controlador predictivo basado en modelo (MPC), el cual consta de tres partes o archivos de programación que trabajan en conjunto, permitiendo determinar de esta manera la mejor combinación de parámetros para un control de nivel óptimo en cada uno de los casos planteados.

El primer algoritmo corresponde a la función objetivo, donde se establece la conexión del controlador predictivo con el sistema (modelo de la planta); en este código se define la

función de transferencia obtenida en la fase 1, los parámetros del MPC que serán considerados como constantes a lo largo del proceso iterativo (tiempo de muestreo, restricciones y pesos) y el tiempo de simulación. La segunda programación permite evaluar el MPC y en ésta se estable los límites de búsqueda para los parámetros del horizonte de predicción y horizonte de control, el tamaño de la población (número de combinaciones a estimar por cada iteración) y el número de generaciones (iteraciones). La tercera programación corresponde al MAGO (Algoritmo por Multidimensionales para Optimización Global), el cual realiza el proceso de iteración o búsqueda de la mejor combinación de parámetros del MPC, de acuerdo a la función objetivo programada.

El algoritmo diseñado para la sintonización del MPC se ejecutó varias veces hasta obtener como resultado un porcentaje de error mínimo, menor o igual al 5 %, lo cual indica que la combinación de parámetros del MPC obtenida, da como resultado un control óptimo. Determinado los parámetros del controlador predictivo, se procedió a su implementación en Simulink para su simulación y análisis. Cumpliendo de esta forma con el segundo objetivo “Diseñar un controlador avanzado para el control de nivel del sistema prototipo de tanques interconectados”.

En la tercera fase, se diseña un control PID usando la aplicación PID Tuner de Simulink, para luego realizar pruebas de desempeño y una comparación del controlador avanzado MPC con el controlador clásico PID, cumpliendo así con el tercer objetivo “Comparar mediante simulación las características del controlador avanzado diseñado con controladores convencionales aplicados al prototipo de tanques”.

Además, se desarrolló una interfaz gráfica que permite al usuario seleccionar cada uno de los casos planteados y ejecutar la simulación de la planta con los controladores MPC y PID diseñados, pudiendo observar mediante gráficas su comportamiento y rendimiento.

5.1 Materiales

Los materiales, equipos y recursos empleados en el presente proyecto de titulación son los siguientes:

➤ Materiales

- Multímetro
- Sensores ultrasónicos HC-SR04
- Destornilladores
- Llave de tubos
- Cinta de teflón
- Cables jumper

- Limpiador dieléctrico de contactos
- **Recursos tecnológicos**
 - Laptop marca Lenovo Z40-70 Core i7
 - Software de ingeniería
 - LabVIEW 2019
 - MATLAB® R2022a y Simulink
 - AutoCAD 2020
 - Word
 - Excel
- **Recursos Humanos**
 - Director de tesis
 - Autor del proyecto
- **Equipo de estudio**
 - Banco de tanques interconectados para el control de nivel de líquidos del laboratorio de automatización de la CIEM.

5.2 Descripción del sistema prototipo de dos tanques interconectados

El laboratorio de automatización de la carrera de ingeniería electromecánica (CIEM) de la Universidad Nacional de Loja (UNL) cuenta con un sistema prototipo de tanques interconectados para la medición y control de nivel de líquidos, con el fin de que los estudiantes de esta carrera puedan realizar prácticas en el área del control automático.

El banco de pruebas mostrado en la **Figura 27**, permite la conexión del sistema de tanques acoplados a un computador por medio de una Tarjeta Arduino Mega 2560 y módulos XBee Pro S1 que establecen la comunicación inalámbrica. La estructura del banco está construida en dos módulos separables para facilitar su transporte, con placas de madera contrachapada de 18 mm y cada tanque está hecho en vidrio de 6 mm de espesor con medidas de 20 cm x 20 cm (base) x 40 cm (altura). Las bombas centrífugas están ubicadas en la base del banco junto con el reservorio de agua para los tanques; además, en la salida de cada bomba se encuentra instalado un bypass de regreso al tanque para evitar sobrepresión en las electroválvulas. La Tarjeta Arduino Mega 2560 está acoplada en una caja plástica de protección junto con los circuitos de los relés y una fuente de poder de 12 V y 5 V. El nivel de líquido en cada tanque es medido por un sensor ultrasónico HC-SR04.

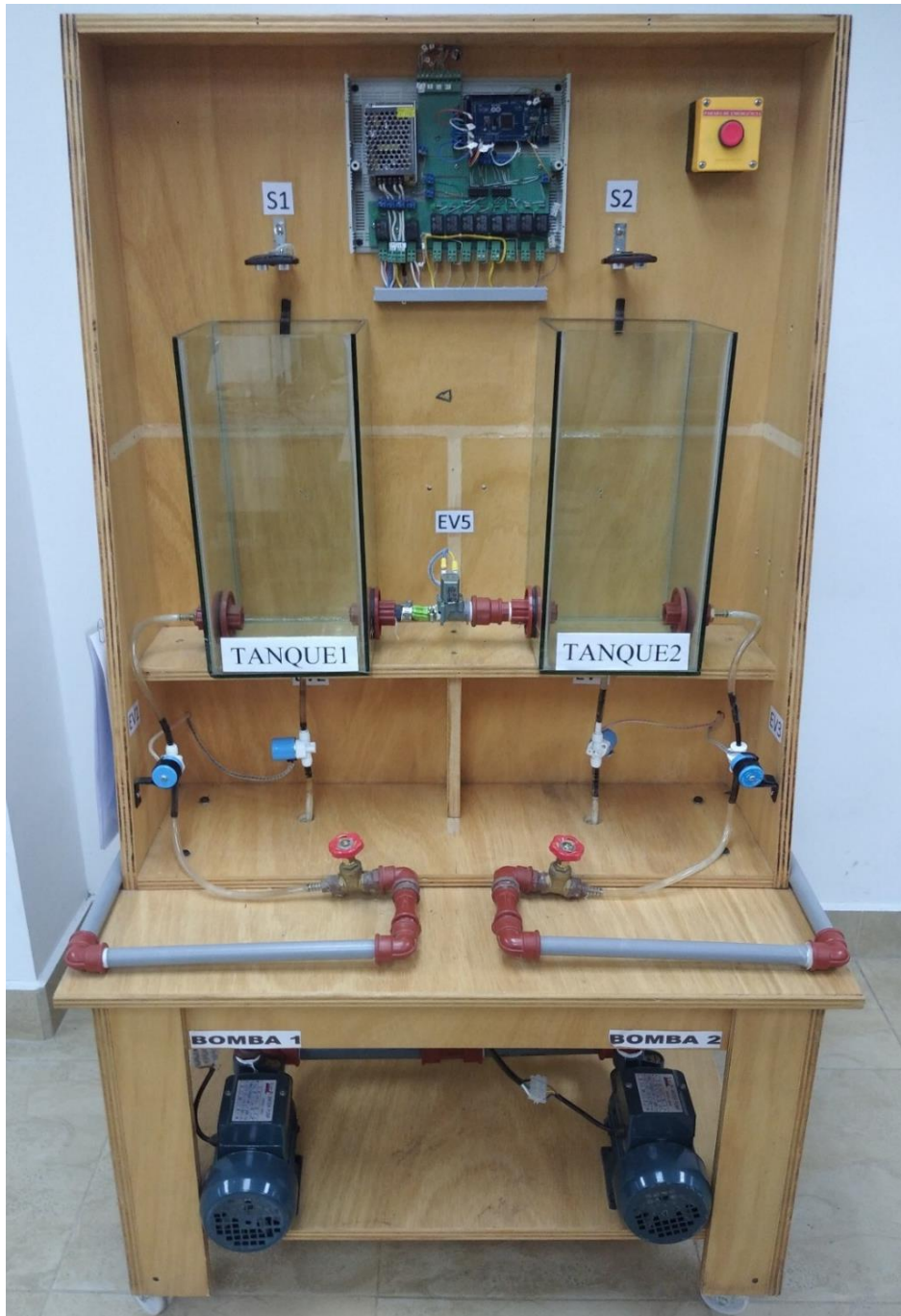


Figura 27. Sistema prototipo de tanques interconectados del laboratorio de automatización de la CIEM.

Los dos tanques del banco de pruebas para la medición y control de nivel de líquidos, están comunicados entre sí por medio de una tubería, misma que lleva una electroválvula para cortar o dar paso al suministro de agua; cada tanque dispone de un sensor ultrasónico en la parte superior, al igual que el depósito de alimentación de agua, y el suministro de agua a los tanques es controlado mediante electroválvulas tanto en la alimentación como en el drenado. Además, este banco de pruebas presenta dos configuraciones de llenado de acuerdo a su programación realizada en LabVIEW; una configuración que permite llenar simultáneamente

tanto el tanque 1 como el tanque 2 con una sola bomba, de acuerdo al valor de nivel que se le asigna a cada tanque (si el nivel del tanque 1 es mayor al del tanque 2, entonces funciona la bomba 1, caso contrario funcionará la bomba 2), y una configuración que permite llenar independientemente el tanque 1 con la bomba 1 y el tanque 2 con la bomba 2.

Cabe destacar que este banco posee únicamente un control on-off, es decir, una vez que el nivel del tanque ha llegado al valor designado, automáticamente se apagan las bombas y se activan las electroválvulas que permiten su vaciado, drenando completamente el agua.

5.3 Experimento de identificación y recolección de datos

5.3.1 Planificación del experimento de identificación

En la planeación del experimento de identificación para obtener el modelo matemático del sistema, se consideró cuatro configuraciones de llenado. En la **Figura 28**, se muestra el esquema general del sistema de tanques interconectados, el cual consta de 3 tanques con un sensor de nivel ultrasónico ubicado en la parte superior de cada uno de ellos, 5 electroválvulas, 2 bombas de agua y 2 válvulas de globo. El control de nivel se realiza sobre el tanque 1 y el tanque 2, mientras que el otro tanque sirve como reservorio de agua.

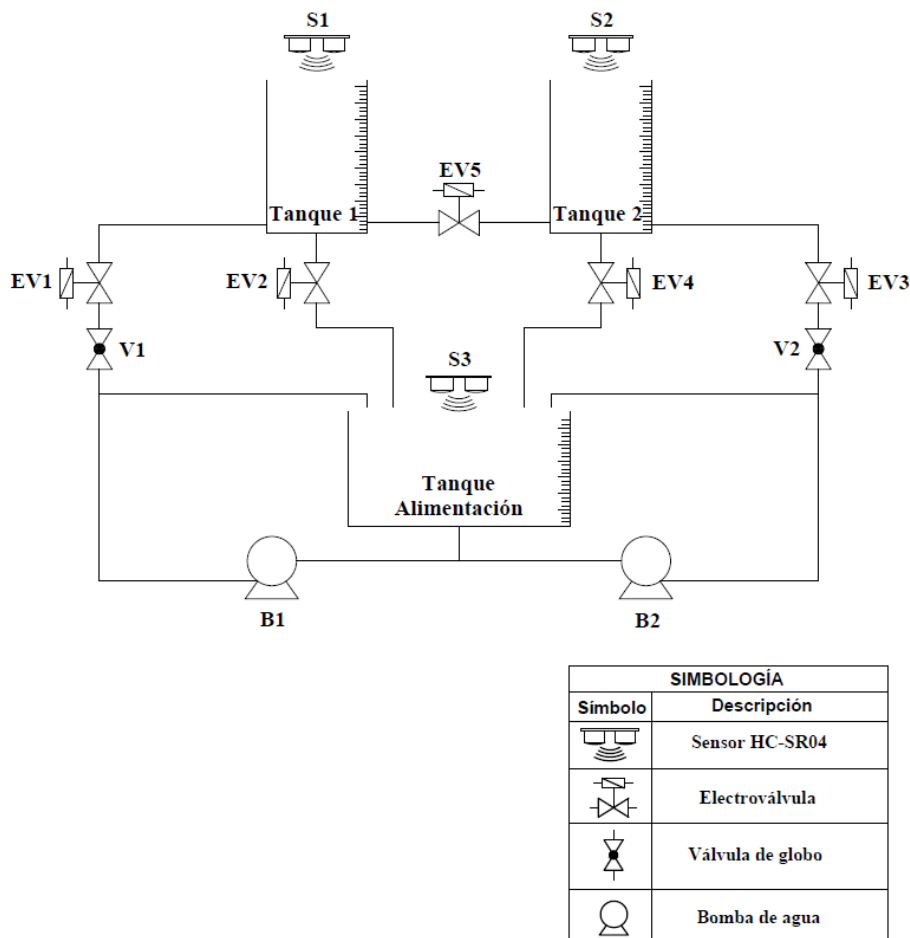


Figura 28. Esquema del sistema de tanques interconectados.

Las cuatro configuraciones de llenado o casos considerados para la recolección de datos se detallan a continuación.

5.3.1.1 Caso 1: Llenado del tanque 1 con la bomba 1

Para este caso se activa la bomba 1, la electroválvula 1 (EV1) y la electroválvula 2 (EV2), las demás electroválvulas permanecen inactivas (cerradas), llenándose únicamente el tanque 1.

El objetivo en esta configuración es controlar el nivel en este tanque.

5.3.1.2 Caso 2: Llenado de los tanques interconectados 1 y 2 con la bomba 1

Para este caso se activa la bomba 1, electroválvula 1 (EV1), la electroválvula de paso (EV5) y la electroválvula 4 (EV4), las demás electroválvulas permanecen inactivas (cerradas).

El objetivo en esta configuración es controlar el nivel en el último tanque, es decir, el tanque 2, ya que éste presenta una dinámica compleja debido a que cada tanque recibe un caudal diferente, el primero producido por el caudal que entrega la bomba y el segundo por el acoplamiento entre ellos, presentando un retardo de tiempo.

5.3.1.3 Caso 3: Llenado del tanque 2 con la bomba 2

Para este caso se activa la bomba 2, electroválvula 3 (EV3) y la electroválvula 4 (EV4), las demás electroválvulas permanecen inactivas (cerradas), llenándose únicamente el tanque 2.

El objetivo en esta configuración es controlar el nivel en este tanque.

5.3.1.4 Caso 4: Llenado de los tanques interconectados 1 y 2 con la bomba 2

Para este caso se activa la bomba 2, electroválvula 3 (EV3), la electroválvula de paso (EV5) y la electroválvula 2 (EV2), las demás electroválvulas permanecen inactivas (cerradas).

El objetivo al igual que en el caso 2, es controlar el nivel en el último tanque, es decir, el tanque 1, el cual presenta un retardo.

5.3.2 Recolección de datos

Para la recolección y registro de datos de las entradas y salidas del proceso mediante experimentación, fue necesario diseñar y desarrollar un instrumento virtual (VI) en LabVIEW por medio del cual se controló de forma manual la activación de las bombas y electroválvulas. Esto con el fin de obtener datos del proceso de llenado de los tanques en las cuatro configuraciones planificadas antes de llevar a cabo el experimento de identificación, para estimar el modelo.

La comunicación de LabVIEW con la tarjeta Arduino Mega 2560 para el control del banco de tanques interactuantes, se enlazó a través de LINX (MakerHub). Esta librería está diseñada para facilitar el desarrollo de aplicaciones embebidas usando LabVIEW; además,

proporciona VIs fáciles de usar para interactuar con Arduino, a través de un firmware que debemos cargar a nuestra placa.

La instalación de LINX (ver **Figura 29**) se realizó por medio del software JKI VI Package Manager, el cual permite encontrar e instalar complementos de LabVIEW desde la red de herramientas de National Instruments (NI). Además, fue necesario tener previamente instalado el complemento NI-VISA para la comunicación por puerto serial.

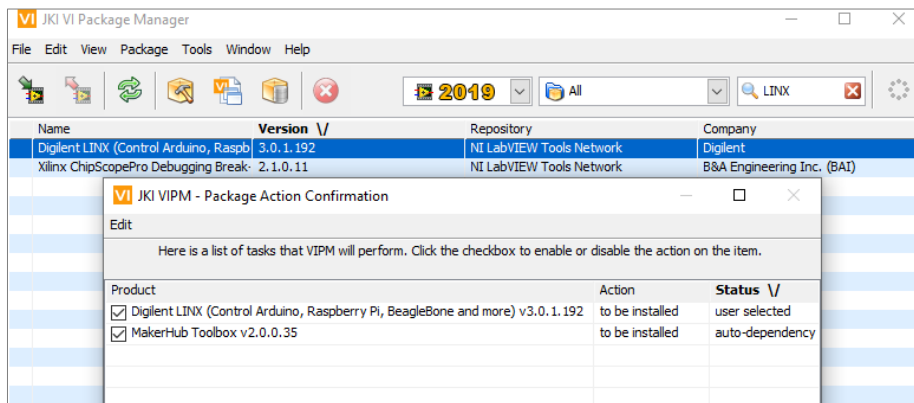


Figura 29. Instalación de la librería LINX para la comunicación de LabVIEW con Arduino.

Una vez instalado LINX se procedió a cargar el firmware a la placa Arduino Mega 2560 desde la barra de menús de LabVIEW (Tools >> MakerHub >> LINX >> LINX Firmware Wizard). Para ello, se debe seleccionar la familia y el tipo de placa al que se va a cargar el firmware, tal como se muestra en la **Figura 30**. A continuación, se elige el puerto COM donde está conectada nuestra placa Arduino y se finaliza la instalación.

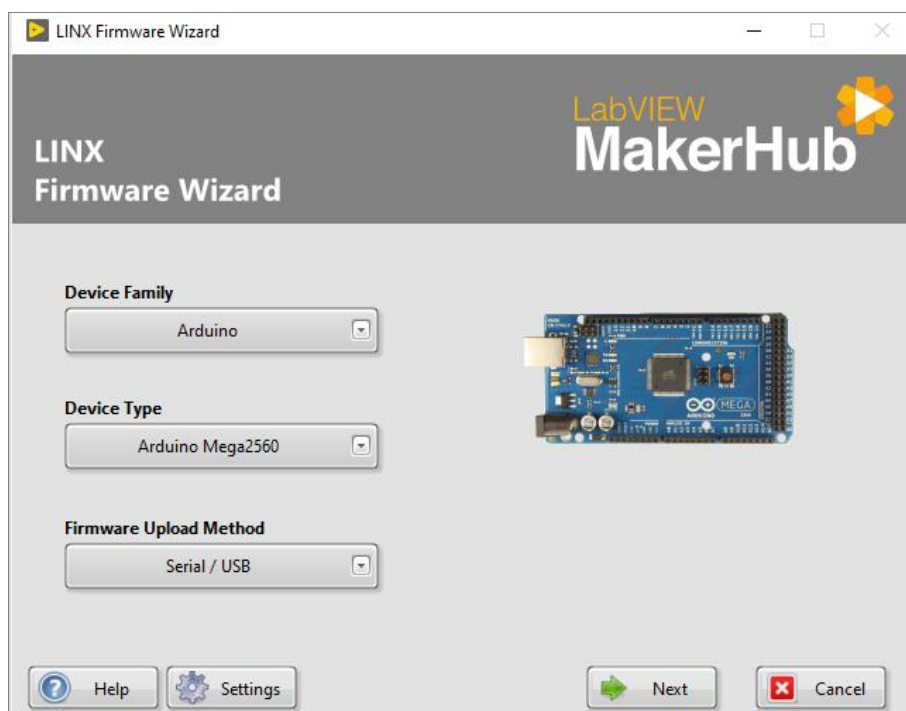


Figura 30. Instalación del firmware en la placa Arduino.

Terminada la instalación de LINX y cargado el firmware, se procedió con el experimento de identificación para la adquisición de datos.

En la **Figura 31**, se muestra el panel frontal del instrumento virtual (VI) diseñado en LabVIEW 2019. Para su ejecución es necesario seleccionar previamente el puerto COM que conecta el Arduino a nuestro computador y la dirección del archivo de Excel sobre el cual se guardarán los datos.

En el **Anexo 1**, se presenta la programación en el diagrama de bloques del VI diseñado.

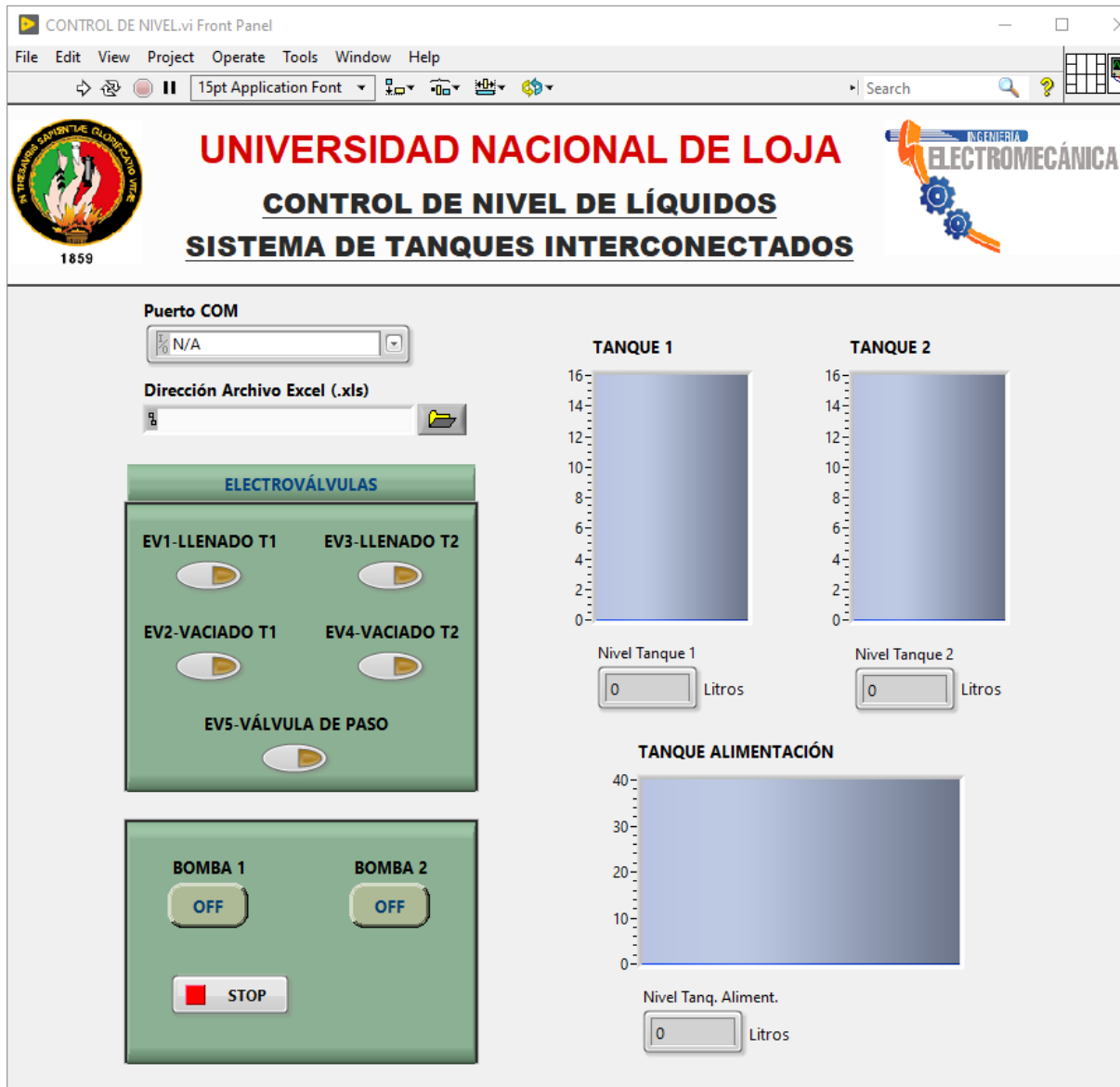


Figura 31. Panel frontal del instrumento virtual diseñado en LabVIEW para la recolección de datos.

Se recopiló un total de 1024 datos para cada configuración de llenado con un intervalo de tiempo de un segundo entre cada dato, mismos que fueron almacenados automáticamente por el software en una hoja de Excel conformada por cinco columnas, en la cual se registró respectivamente, el tiempo, la activación de la bomba 1 y bomba 2, y el nivel de llenado medido

en litros del tanque 1 y tanque 2. Se realizaron varias pruebas por cada configuración de llenado y se promedió cada uno de los datos obtenidos, consiguiendo así las curvas de nivel que representan la dinámica del sistema.

Luego, se analizaron y filtraron los datos obtenidos, removiendo tendencias y sobresaltos que pudieron haberse generado por ruidos de alta frecuencia, los cuales no se desea que intervengan en el proceso del modelado por identificación.

5.4 Modelado matemático del sistema de tanques interconectados

La función de transferencia que representa la dinámica del sistema de tanques interconectados, se obtuvo a través de la herramienta Identificación de Sistemas (System Identification Toolbox) de MATLAB®. El ingreso a esta aplicación se realiza mediante la instrucción “*systemIdentification*” escrita en la ventana de comandos.

La ventana principal de la herramienta de Identificación de Sistemas, la cual se expone en la **Figura 32** está dividida en dos secciones: la primera referida al conjunto de datos importados para el análisis, y una segunda sección en donde se localizan los modelos creados en base a las operaciones efectuadas sobre los datos importados.

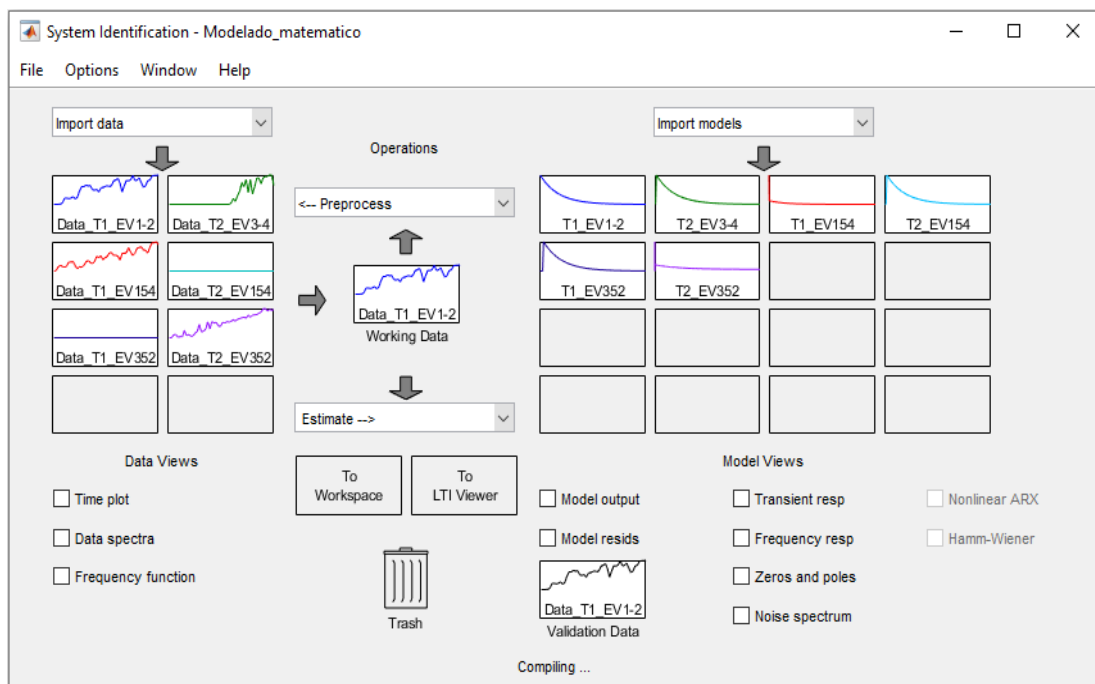


Figura 32. Ventana principal de la Identificación de Sistemas de MATLAB® para la estimación del modelo.

Antes de estimar el modelo, se procede a importar al workspace de MATLAB® los datos de entrada (activación de la bomba) y salida (nivel de llenado del tanque) que fueron obtenidos en el proceso de experimentación, definiendo el tipo de variable de salida a importar

como matriz numérica. Luego, estos se importaron a la herramienta de identificación para su análisis como datos en el dominio del tiempo, tal como se muestra en la **Figura 33**.

En este cuadro de diálogo (ver **Figura 33**) se ingresa el nombre del conjunto de datos de entrada (input) y salida (output) a importar que previamente fueron cargados al workspace y se le asigna un nombre (data name) que permita identificar cada una de las configuraciones de llenado. En el recuadro de start time y sample time, correspondiente al tiempo de inicio con el que fueron obtenidos los datos y al tiempo de muestreo, respectivamente, se asigna un valor de 0 y 1, ya que el intervalo de tiempo entre cada uno de los datos obtenidos en el experimento de identificación fue de un segundo.

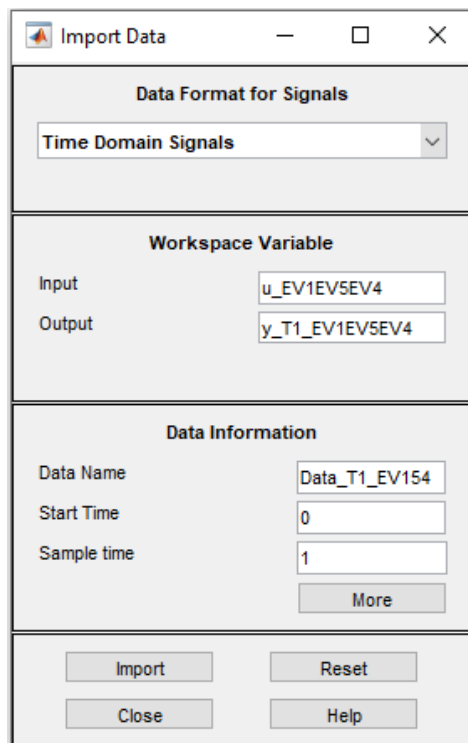


Figura 33. Importación de los datos de entrada y salida para la Identificación de Sistemas.

La estimación del modelo se realiza mediante la opción Process Models, mostrada en la **Figura 34**, la cual es recomendada para obtener modelos matemáticos de procesos con fines de control, como es el caso del presente proyecto. Para obtener la función de transferencia es necesario elegir el orden de la misma y si conlleva o no un retardo de tiempo.

Cada uno de los modelos obtenidos fueron analizados y se seleccionaron solo aquellos que representan la dinámica del sistema con un porcentaje de exactitud mayor al 90 %. Los modelos que presentaron un porcentaje menor, fueron eliminados y estimados nuevamente, ya sea variando las condiciones iniciales o incrementando el número de polos y ceros.

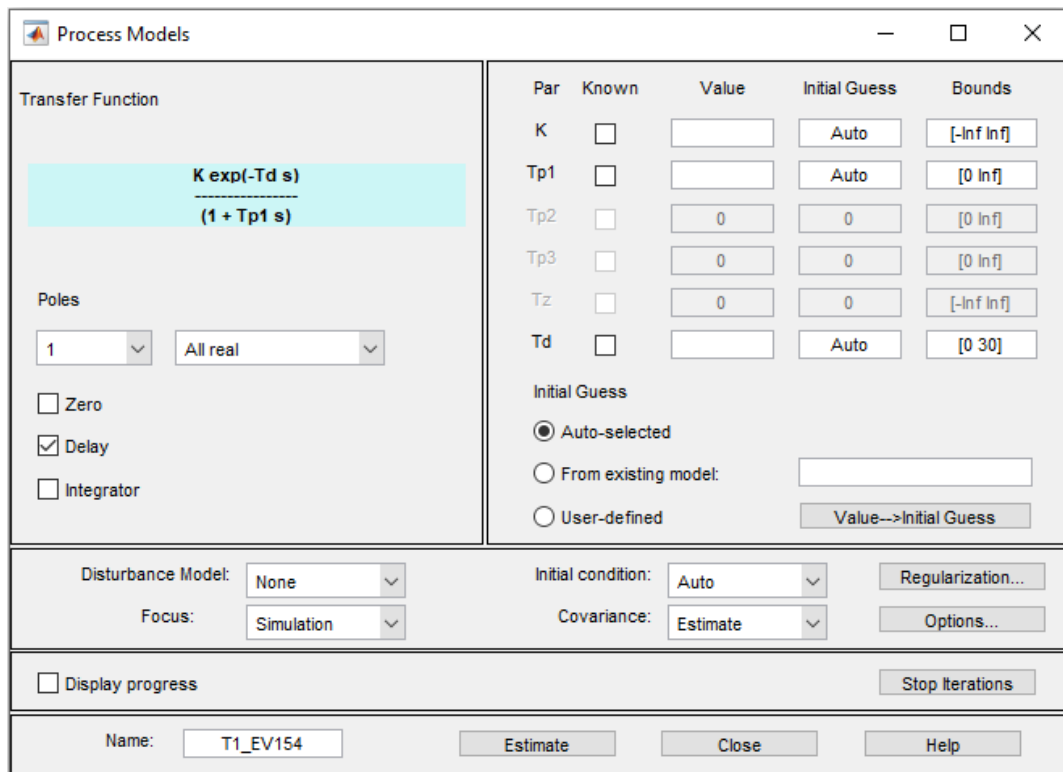


Figura 34. Cuadro de diálogo para la estimación del modelo de la planta.

Con esto se cumple con el primer objetivo “Determinar un modelo matemático que represente la dinámica del sistema prototipo de tanques interconectados”.

5.5 Diseño del controlador avanzado MPC

El diseño del controlador predictivo basado en modelo (MPC) para el sistema de tanques interconectados fue desarrollado en dos etapas: la primera referida a la sintonización y una segunda etapa donde se implementa el controlador diseñado en Simulink, con el fin de evaluar su rendimiento mediante simulación, usando la aplicación MPC Toolbox de MATLAB® a través de un sistema de control a lazo cerrado.

5.5.1 Sintonización del controlador MPC

Para la sintonización del MPC se diseña un algoritmo de control en MATLAB® que permite la estimación de los parámetros más óptimos del controlador a través de un algoritmo evolutivo (MAGO).

El algoritmo diseñado consta de tres partes: una programación para la función objetivo, una segunda programación para evaluar el MPC en la que se estable los límites de búsqueda para los parámetros, el tamaño de la población y el número de generaciones o iteraciones, y una tercera programación que corresponde al MAGO (Algoritmo por Multidínámicas para Optimización Global), el cual realiza la búsqueda de los mejores parámetros. A continuación, se detalla cada uno de ellos.

5.5.1.1 Algoritmo para la función objetivo.

En esta programación se define la función objetivo y se establece la conexión del controlador MPC con el sistema (modelo de la planta).

El modelo obtenido representa la dinámica del sistema con la bomba operando a su máxima capacidad; partiendo de ello, la función objetivo lo que hace es realizar mediante simulación una comparación entre la curva que representa la dinámica del sistema y la curva de control del MPC sintonizado, determinando el porcentaje de error y considerando el seguimiento del setpoint, es decir, esta función objetivo hará que el controlador opere la bomba (actuador) a su máxima capacidad, minimizando el tiempo que tarda la salida del sistema (variable controlada) en llegar al setpoint, pero considerando a su vez que la curva de control obtenida no sobrepase el límite máximo de operación del actuador. Lo mencionado anteriormente, se puede entender mejor observando la **Figura 68** y **69** generada para este análisis y que es parte de los resultados obtenidos.

Los parámetros considerados para la sintonización del controlador MPC fueron los siguientes: tiempo de muestreo (T_s), horizonte de predicción (H_p), horizonte de control (H_c), peso de la variable manipulada, peso al incremento de la variable manipulada, peso de la variable de salida, peso de la variable de holgura (ECR), restricción mínima y máxima de la variable manipulada. Los valores de cada uno de estos parámetros se asignaron en base a las recomendaciones establecidas para este sistema de control por Capuano (2021), descritas en el capítulo 3 del marco teórico, las cuales se resumen en la **Tabla 5**. Para el valor de los pesos se consideró lo expuesto en la **Tabla 3**, referida a la asignación de pesos según la prioridad de seguimiento de las entradas y salidas del proceso.

Tabla 5. Consideraciones y recomendaciones a tener en cuenta respecto a los parámetros del MPC.

Parámetro MPC	Consideraciones y recomendaciones
Tiempo de muestreo	<ul style="list-style-type: none">- Para el control de procesos, se recomienda un tiempo de muestreo mayor o igual a 1 s.- Cualitativamente, a medida que disminuye el tiempo de muestreo, el rechazo de perturbaciones desconocidas generalmente mejora, pero a su vez el esfuerzo computacional aumenta.
Horizonte de predicción	<ul style="list-style-type: none">- El horizonte de predicción debe ser mayor al horizonte de control.- En el caso de sistemas con retardo de tiempo, el horizonte de predicción debe ser mayor a este valor.
Horizonte de control	<ul style="list-style-type: none">- Las pautas sugieren establecer el horizonte de control entre el 10% y el 20 % del horizonte de predicción.

Parámetro MPC	Consideraciones y recomendaciones
Peso de la variable manipulada (u) y de salida (y)	<ul style="list-style-type: none"> - Se sugiere priorizar el seguimiento de la variable de salida. - Cuanto mayor sea el peso de la variable, más tenderá el controlador a mantenerla cerca de su objetivo para minimizar la función de costo o función objetivo.
Peso al incremento de la variable manipulada (Δu)	<ul style="list-style-type: none"> - En el caso de la tasa de cambio o incremento de las variables manipuladas, la simulación proporciona un rendimiento del controlador más robusto, pero un seguimiento de referencia más pobre cuanto mayores son los valores de este peso.
Peso de la variable de holgura (ECR)	<ul style="list-style-type: none"> - Esta variable representa la violación de las restricciones y cuanto mayor sea, con respecto a los pesos de entrada y salida, más se penaliza la violación de las restricciones. - Un peso ECR demasiado grande distorsiona la optimización de las variables manipuladas.
Restricciones	<ul style="list-style-type: none"> - Establece los límites para las variables de entrada (manipuladas) y las variables de salida (medidas) que resultan de límites físicos o límites impuestos para la aplicación específica por el diseñador del controlador.

Fuente: (Capuano, 2021).

En la **Figura 35**, se presenta el código implementado en MATLAB[®] para la definición de la función de transferencia y los parámetros del controlador MPC, estos datos son definidos por el usuario. En la parte de la función de transferencia se asigna el modelo obtenido en el proceso de identificación de sistemas; por otro lado, el valor de los parámetros del MPC se pueden variar para mejorar los resultados obtenidos en la sintonización con el algoritmo, en este caso se varió únicamente el peso a la tasa de cambio de la variable manipulada y el peso de la variable de salida. En base a esto, el algoritmo determina que combinación de parámetros me permite optimizar el controlador, minimizando la función objetivo.

Es importante mencionar que para el proceso de iteración y búsqueda de la mejor combinación, se consideró dos variables, el horizonte de predicción y el horizonte de control, estos valores los determina automáticamente el algoritmo, mientras que los demás parámetros como el tiempo de muestreo, los pesos y las restricciones se consideraron constantes a lo largo del proceso iterativo; por lo tanto, estos valores son asignados por el usuario y pueden modificarse en caso de que el resultado obtenido no sea el deseado.

```

funcion_obj_mpc.m
1 function [error, y, y1] = funcion_obj_mpc(Valor)
2
3 % Definir la función de transferencia del sistema
4 num = [27.735]; % Numerador de la función de transferencia
5 den = [1995.3 1]; % Denominador de la función de transferencia
6 G = tf(num, den);
7
8 % Definir los parámetros del controlador MPC
9 Ts = 1; % Tiempo de muestreo
10 Hp = fix(Valor(1)); % Horizonte de predicción
11 Hc = fix(Valor(2)); % Horizonte de control
12 weights = struct('ManipulatedVariables', 1, 'ManipulatedVariablesRate', 0.1, 'OutputVariables', 5, 'ECR', 100);
13 MV = struct('Min', 0, 'Max', 1); % Límites de la variable manipulada
14
15 % Tiempo final de simulación (definir parámetro)
16 Tfinal = 100;

```

Figura 35. Definición de la función de transferencia y de los parámetros del controlador MPC.

Luego de definir la función de transferencia y los parámetros de control, se asigna el tiempo de simulación en la línea de código 16 presentada en la **Figura 35**. El tiempo de simulación se define en base al tiempo que tarda la planta sin control en llegar a un valor de uno y debe ser asignado por el usuario.

Para el caso de los modelos con retardo de tiempo, este se añade a la salida del sistema donde se definió la función de transferencia, mediante la instrucción “*OutputDelay*”, tal como se muestra en la **Figura 36**.

```

% Definir la función de transferencia del sistema
num = [9.3496]; % Numerador de la función de transferencia
den = [1086.9 1]; % Denominador de la función de transferencia
G = tf(num, den, 'OutputDelay', 86.838); % Retardo de 86.838 segundos

```

Figura 36. Definición de la función de transferencia para un sistema con retardo.

El resto de la programación desarrollada permite crear el objeto de control del MPC, cargando los parámetros asignados a dicho controlador, simular el MPC y el modelo de la planta ante una entrada escalón, y determinar el porcentaje de error de la función objetivo comparando la curva del modelo de la planta (y) con la curva del controlador MPC (y_1), tal como se describió al inicio. En el **Anexo 2**, consta el código de programación desarrollado para la función objetivo.

5.5.1.2 Programación para evaluar el MPC.

En esta programación se establece el conjunto o matriz de posibles soluciones que pueden minimizar la función objetivo y se indica el tamaño de la población (número de individuos) y el número de generaciones (iteraciones).

Primeramente, se crea la variable de salida donde se van a almacenar los resultados obtenidos. Luego, en la parte del código mostrado en la **Figura 37**, se define los límites de evaluación para el horizonte de predicción (1 a 40) y el horizonte de control (4 a 10), estos

límites son establecidos por el usuario y se asignan considerando las recomendaciones dadas por Capuano (2021) para los parámetros del MPC (ver **Tabla 5**), el cual sugiere establecer el horizonte de control entre el 10 % y 20 % del horizonte de predicción. A continuación, se define el número de individuos (n) y número de generaciones (ng), asignando para este caso 40 individuos y 25 generaciones, ejecutando en total 1000 combinaciones para la estimación de los mejores parámetros. Estos datos son asignados por el usuario, los cuales se fueron variando en cada ejecución del algoritmo, hasta obtener un error en la sintonización del MPC menor o igual al 5 %.

```

11 %Definir los límites de evaluación para los parámetros del MPC
12 lb =[1 4]; %Límites inferiores
13 ub =[40 10]; %Límites superiores
14
15 n = 40;
16 ng = 25;
17 %salida2 = [Horizonte_predicción Horizonte_control Error], [0 0 0]

```

Figura 37. Definición del conjunto de posibles soluciones, número de individuos y número de generaciones.

En base a la programación mostrada en la **Figura 37** el algoritmo determinará del conjunto de valores asignados (matriz de posibles soluciones) para el horizonte de predicción y horizonte de control, la combinación de parámetros más óptimos que permiten minimizar la función objetivo, considerando en el proceso iterativo los demás parámetros del MPC (tiempo de muestreo, pesos y restricciones) como valores constantes, mismos que fueron establecidos en la programación de la función objetivo.

El resto de programación permite cargar la función objetivo y llamar al algoritmo MAGO para que se ejecute y realice el proceso de búsqueda en base a la función objetivo, la matriz de soluciones, el número de individuos y el número de generaciones. Seguido, el algoritmo evalúa la solución encontrada, determina el error de la misma y almacena los resultados obtenidos en la variable de salida creada al inicio de la programación, ejecutando consecutivamente el algoritmo hasta obtener un resultado aceptable o alcanzar el número de iteraciones establecido.

Una vez finalizado el proceso de búsqueda se genera la gráfica de la respuesta del controlador MPC con los parámetros determinados y se presenta únicamente la solución que obtuvo el mínimo error, tal como se observa en la **Figura 38**. Cada vez que se ejecuta el algoritmo se almacena el resultado obtenido en la variable de salida, los números de la primera columna corresponden al horizonte de predicción, la segunda columna al horizonte de control y la tercera columna al error obtenido, los demás parámetros del MPC son los que se definieron

en la función objetivo, los cuales se consideraron constantes a lo largo del proceso iterativo. Los valores obtenidos del horizonte de control y horizonte de predicción son redondeados a un número entero, ya que el MPC Toolbox de Simulink solo admite valores enteros para estos dos parámetros.

En el **Anexo 3**, se muestra el código de programación desarrollado para evaluar el MPC.

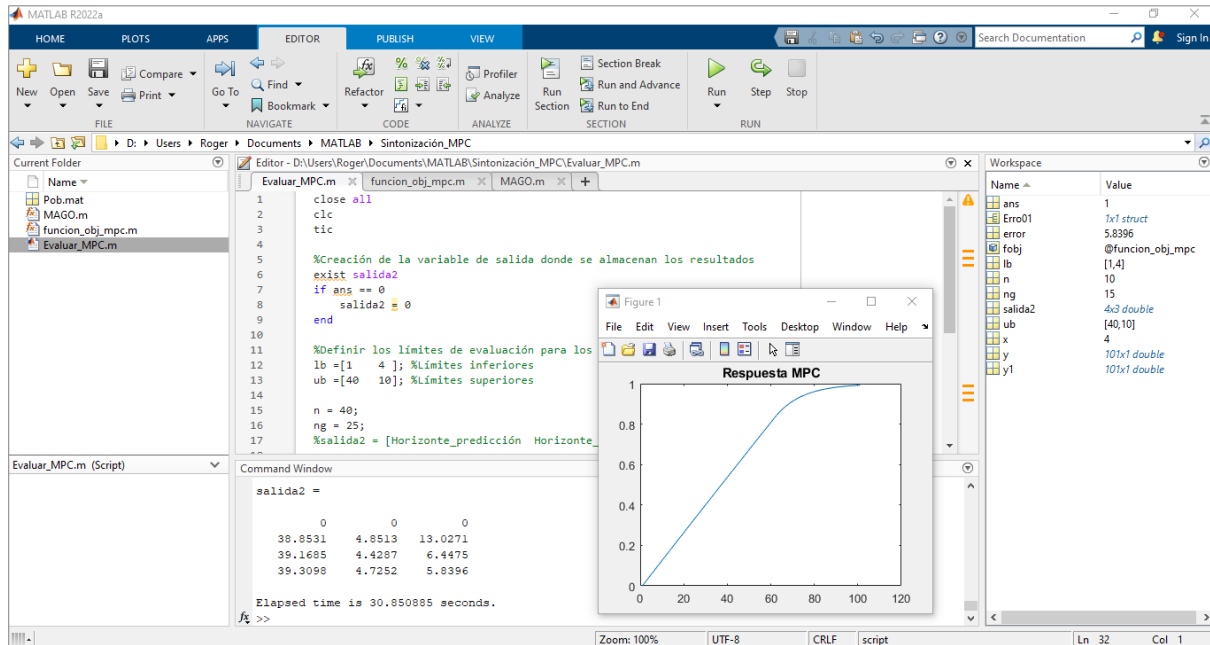


Figura 38. Ejecución del algoritmo desarrollado para evaluar el MPC y encontrar sus parámetros.

Es importante mencionar que para el proceso de iteración y búsqueda de la mejor combinación, además del horizonte de predicción y el horizonte de control se puede incluir sin ningún problema todos los parámetros del MPC; sin embargo, para este caso se decidió considerar sólo estos dos, debido a que son los más importantes en este sistema de control, manteniendo el resto de parámetros en un valor fijo, asignado a criterio propio en base a las recomendaciones establecidas para su sintonización, obteniendo los resultados deseados.

5.5.1.3 Algoritmo por Multidínamicas para Optimización Global (MAGO).

La iteración y búsqueda de la mejor combinación de parámetros para un control MPC óptimo, se realizó mediante un algoritmo evolutivo denominado MAGO. Este algoritmo fue formalmente presentado por Hernández & Ospina (2010) y ha sido ampliamente estudiado como aplicado en varios proyectos de investigación o tesis relacionadas al control de procesos para la sintonización y optimización de controladores PID y MPC.

Basado en los proyectos de investigación relacionados a este tema, se optó por aplicar este algoritmo para la sintonización del controlador MPC. El MAGO necesita de cuatro parámetros para su funcionamiento y ejecución: una función objetivo, un límite superior e

inferior (conjunto o matriz de posibles soluciones), el tamaño de la población (número de individuos) y el número de generaciones (iteraciones).

A continuación, se muestra el diagrama de flujo del algoritmo MAGO (ver **Figura 39**) y se detalla la estrategia que emplea para la búsqueda de la mejor solución.

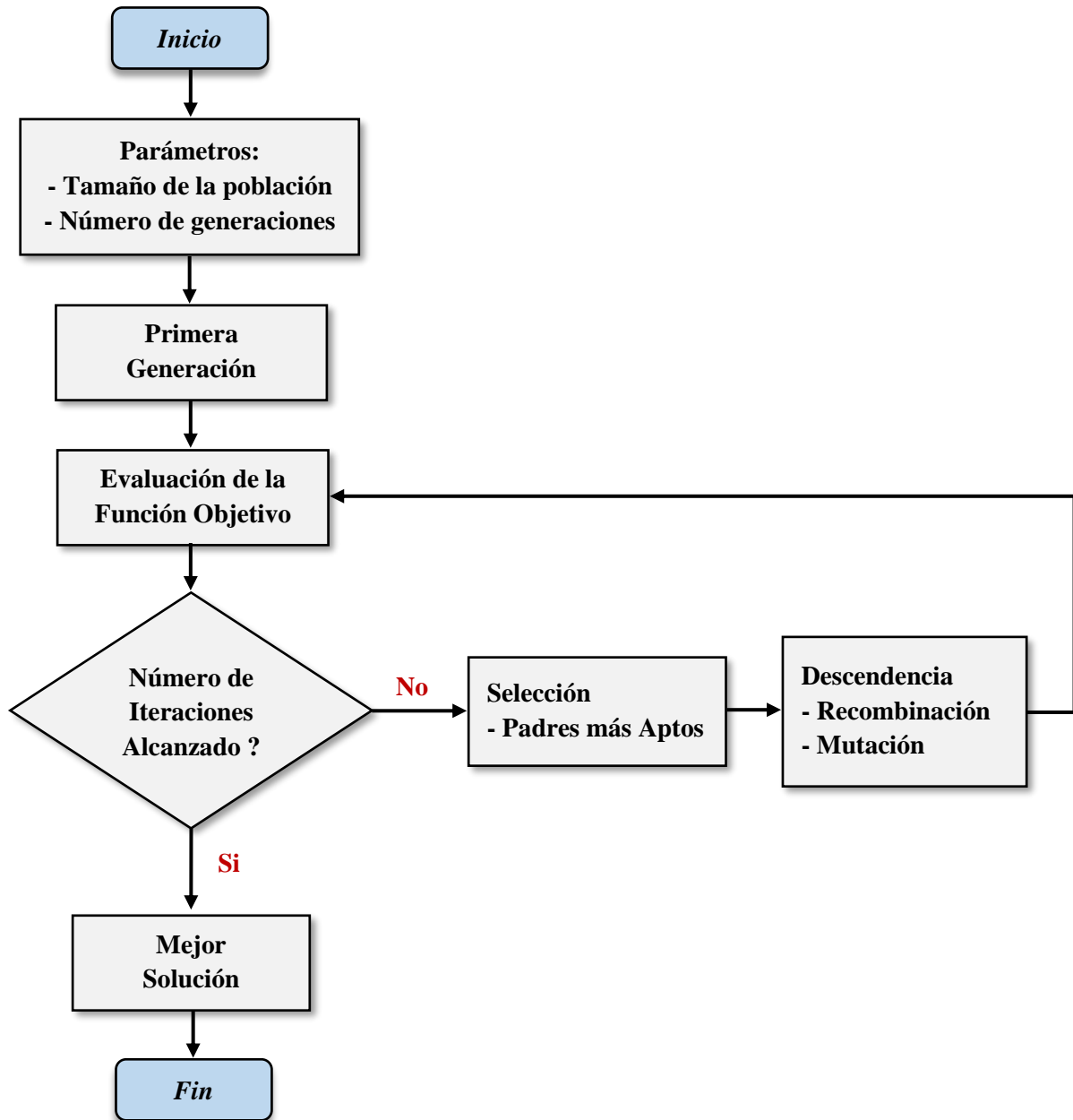


Figura 39. Diagrama de flujo del algoritmo MAGO.

Partiendo del número de individuos (tamaño de la población) y el número de generaciones, el algoritmo realiza un total de n combinaciones de acuerdo al tamaño de la población asignada, cuyos valores están dentro de la matriz de posibles soluciones, evaluando cada combinación por medio de la función objetivo y obteniendo como resultado la primera generación. Luego, para la siguiente generación se realiza un proceso de selección, cruzamiento

y mutación; para lo cual se crean tres grupos, en el primer grupo (G1) se crean nuevos individuos alrededor de la combinación que presentó el mínimo error en la generación anterior, en el segundo grupo (G2) se crean otras combinaciones alrededor de la media poblacional actual y en el tercer grupo (G3) se crea un nuevo número de combinaciones pero al azar, reduciendo así el espacio de búsqueda y repitiendo el mismo procedimiento para las siguientes generaciones hasta alcanzar el número de iteraciones establecido.

Cabe destacar que el total de combinaciones generadas entre el grupo G1, G2 y G3 en cada generación, son en base al número de individuos designado por el usuario, es decir, sumada las combinaciones generadas entre los tres grupos, se tiene un total de 40 combinaciones por cada generación, para este caso. Por otro lado, si el resultado no satisface el criterio de desempeño del usuario, se puede minimizar este error incrementando el tamaño de la población y el número de generaciones.

En el **Anexo 4**, se proporciona el algoritmo MAGO, para la optimización y sintonización del controlador MPC.

5.5.2 Implementación del controlador MPC en Simulink

Una vez sintonizado el MPC para cada uno de los casos planteados, se procedió a su diseño e implementación en un software de simulación (Simulink) para su respectivo análisis. Para ello, primero se importó los modelos obtenidos al Workspace de MATLAB®, mediante una instrucción ejecutada en el Command Window (ver **Anexo 5**).

El controlador MPC se implementó a través un sistema de control de lazo cerrado usando el bloque MPC Controller de Simulink, presentado en la **Figura 40**, ubicando en este diagrama de control la función de transferencia y el retardo de tiempo en aquellos procesos que presentaron este atraso en su salida.

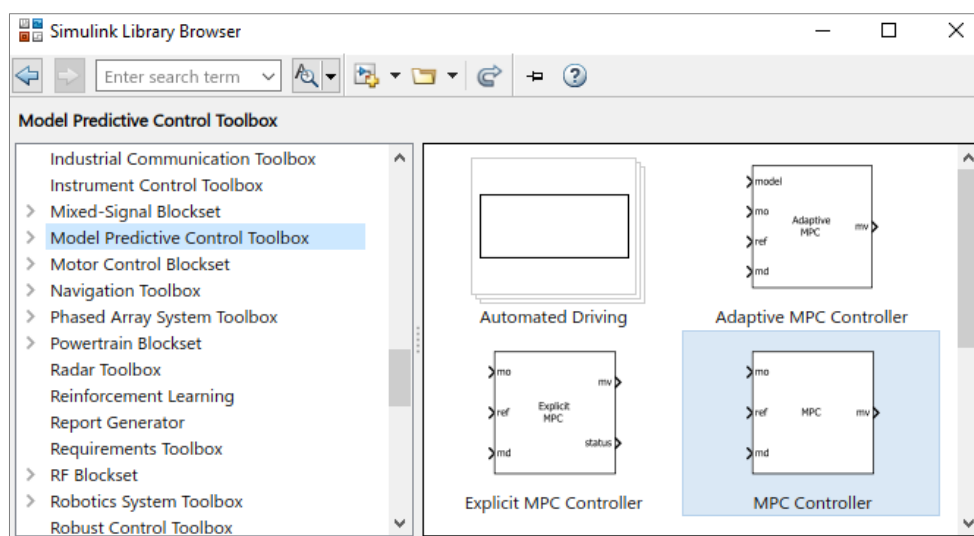


Figura 40. Bloque MPC Controller para implementación de controladores predictivos en Simulink.

Armado el diagrama de control, se procede al diseño y configuración del MPC mediante la aplicación MPC Designer que incluye el bloque MPC Controller de Simulink, a la cual se accede dando clic en el botón “*Design*” de la **Figura 41**. El primer paso a seguir es definir la estructura del MPC, para lo cual se debe seleccionar cuáles y cuántas son las entradas y salidas del sistema, tal como se muestra en la **Figura 42**. Automáticamente la aplicación detecta el número de entradas y salidas del sistema de acuerdo al diagrama implementado en Simulink, de no ser así se puede configurarlas dando clic en el botón “*Change I/O Sizes*”.

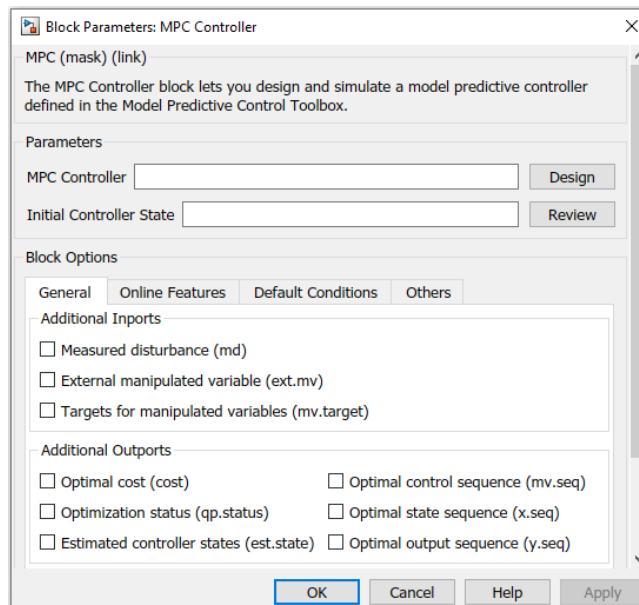


Figura 41. Ventana principal del bloque del controlador MPC.

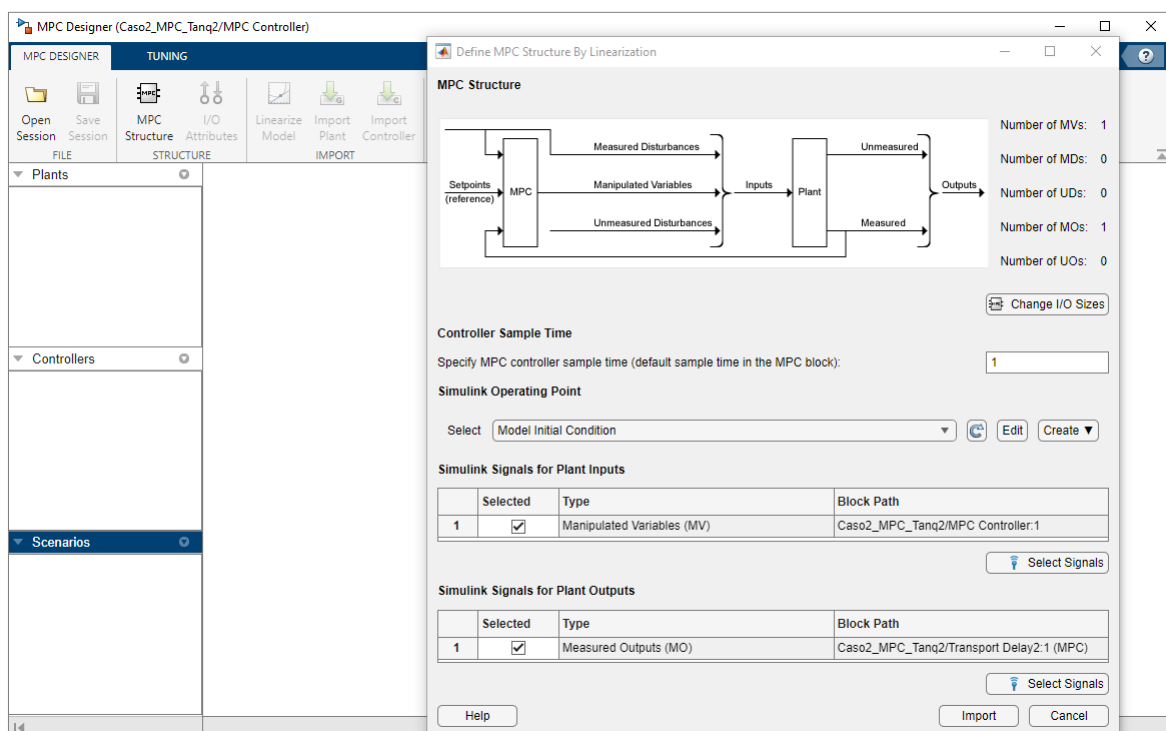


Figura 42. Definición de la estructura del MPC.

Definida la estructura del MPC, se procede a importar la función de transferencia desde el Workspace de MATLAB®, tal como se indica en la **Figura 43**. Luego, se selecciona el modelo importado en la pestaña Tuning (ver **Figura 44**), para que se cargue a la aplicación.

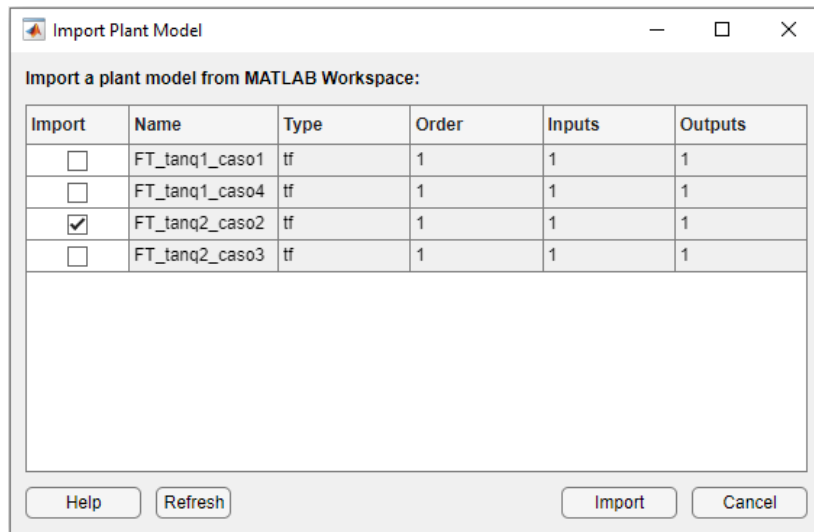


Figura 43. Importación del modelo obtenido al controlador MPC.

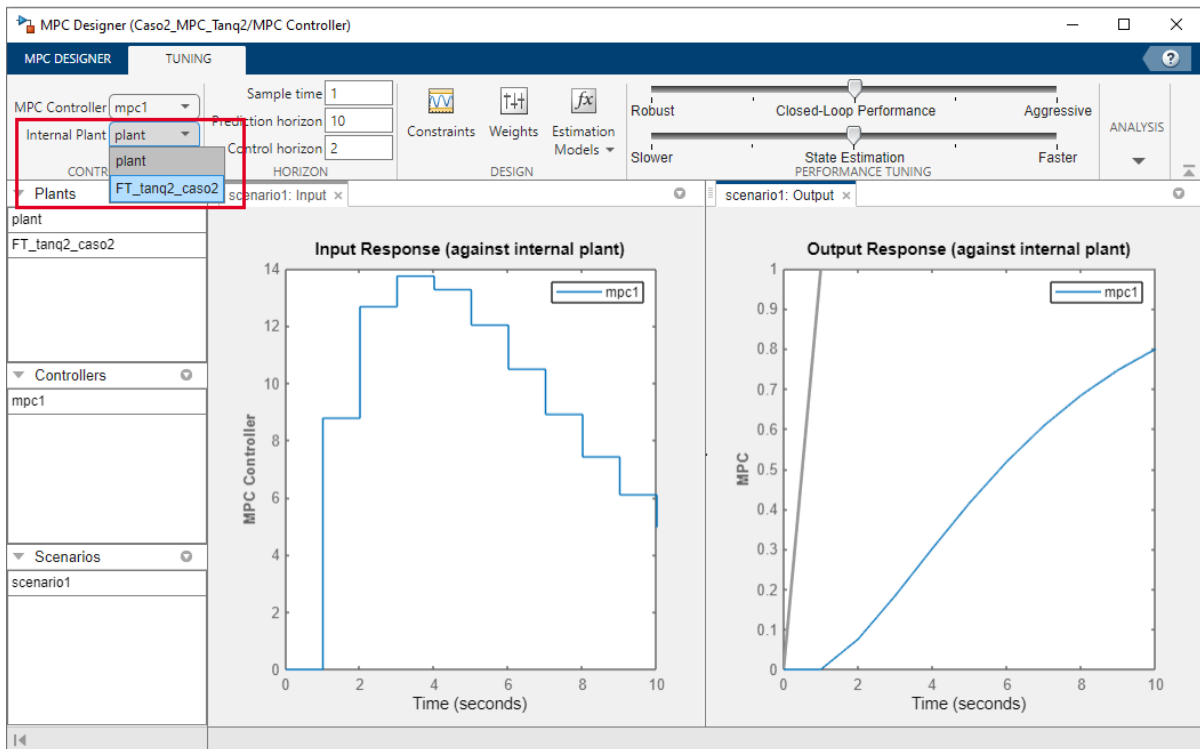


Figura 44. Selección de la función de transferencia importada.

Importado el modelo de la planta y cargado a la aplicación MPC Designer, se procedió a ingresar los parámetros obtenidos en la sintonización con el algoritmo MAGO, tal como se expone en la **Figura 45**. Estos parámetros son el tiempo de muestreo, horizonte de predicción, horizonte de control, restricción mínima y máxima de la variable manipulada, y los pesos

(variable manipulada, incremento de la variable manipulada, variable de salida y ECR). Finalmente, se exporta el diseño del controlador al bloque del MPC para poder realizar su simulación.

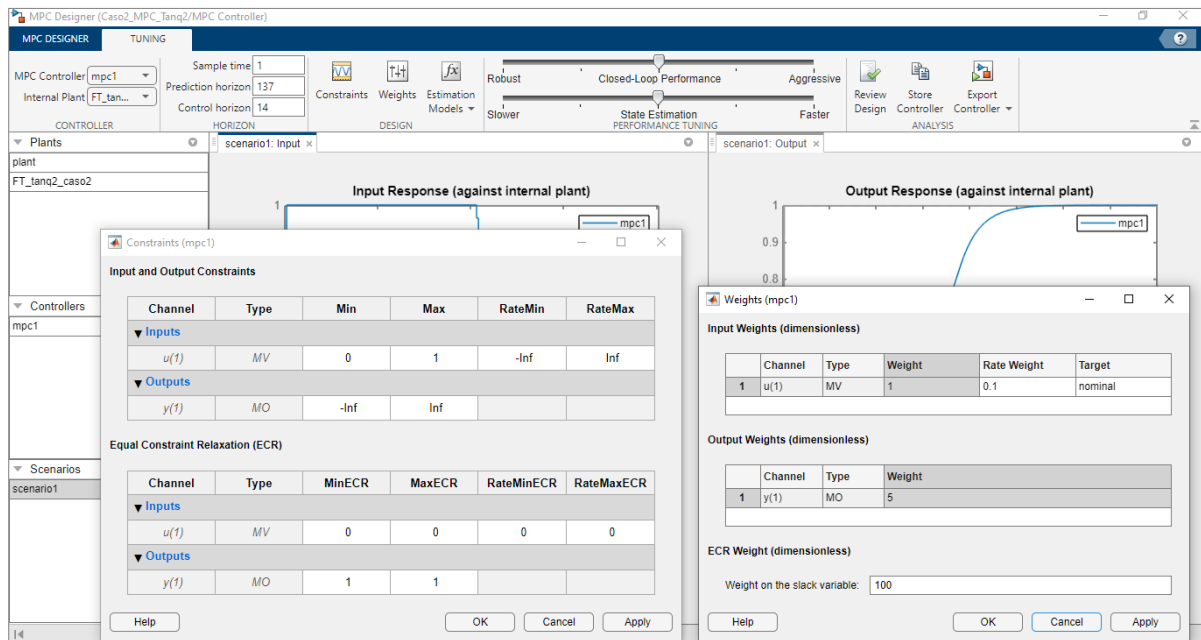


Figura 45. Asignación de los parámetros del MPC.

La restricción mínima y máxima de la variable manipulada para el MPC es de 0 a 1, esto debido a que el experimento de identificación fue realizado aplicando una entrada escalón unitario como señal de excitación para conocer la respuesta del sistema, la cual se usó como dato de entrada en la identificación de sistemas para determinar el modelo matemático de la planta; por lo tanto, este valor de 1 indica que la bomba está operando a su máxima capacidad, es decir, al 100 %.

Sintonizado e implementado el MPC en Simulink se cumple con el segundo objetivo “Diseñar un controlador avanzado para el control de nivel del sistema prototipo de tanques interconectados”.

5.6 Diseño e implementación del controlador clásico PID en Simulink

Con las funciones de transferencia obtenidas para cada una de las configuraciones de llenado, se procedió al diseño e implementación del controlador clásico PID en un software de simulación, para su posterior análisis y comparación con el controlador avanzado MPC.

El controlador PID se implementó a través un sistema de control de lazo cerrado usando el bloque PID Controller de Simulink, mostrado en la **Figura 46**, ubicando en este diagrama de control la función de transferencia y el retardo de tiempo en aquellos procesos que presentaron este atraso en su salida.

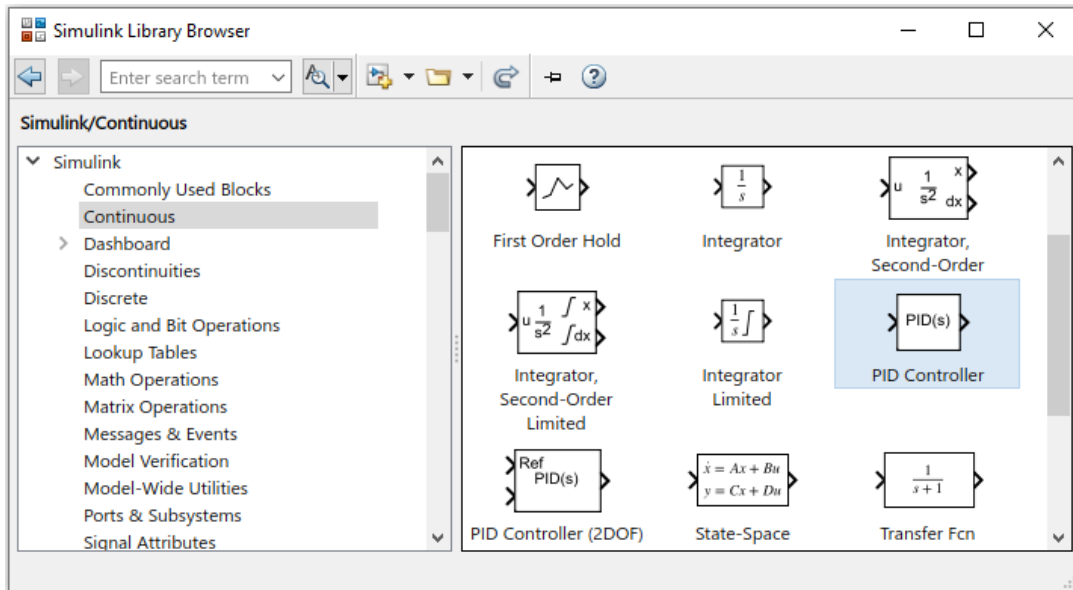


Figura 46. Bloque PID Controller para la implementación del controlador clásico PID en Simulink.

Armado el diagrama de control PID se procede a su sintonización mediante la aplicación PID Tuner que incluye el bloque PID Controller de Simulink, presentado en la **Figura 47**. Esta herramienta efectúa una primera sintonización ajustando automáticamente las ganancias del controlador con respecto al modelo de la planta, misma que puede ser modificada por el usuario, permitiendo refinar su diseño y en consecuencia el comportamiento de la curva de control y la estabilidad del sistema.

La aplicación PID Tuner, además permite analizar el rendimiento del controlador en base a parámetros como el tiempo de subida, el tiempo de estabilización, el porcentaje que la señal excede su setpoint, entre otros.

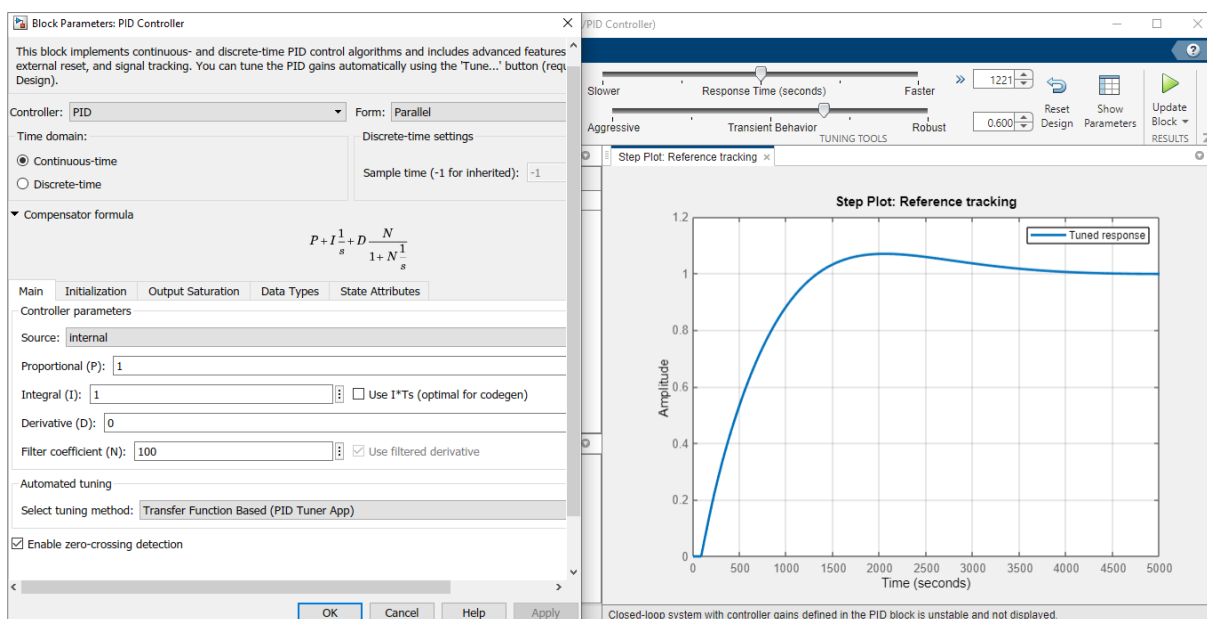


Figura 47. Interfaz de la aplicación PID Tuner para la sintonización del controlador PID.

En cada uno de los procesos (modelos obtenidos) se ajustó las ganancias del controlador hasta obtener un control PID óptimo. Además, se consideró las limitaciones físicas del actuador empleando una estrategia conocida como anti-windup, ya que de no ser así en el proceso real (implementación física del controlador) podemos llegar a tener un comportamiento no deseado de la variable controlada.

El anti-windup es un mecanismo de control que evita que la acción integral del controlador PID continúe creciendo cuando el sistema se encuentra en los límites de saturación del actuador (Castaño Giraldo, 2020). El bloque PID Controller de Simulink incorpora dos métodos anti-windup integrados, back-calculation y clamping, empleando este último método en el diseño del controlador, tal como se evidencia en la **Figura 48**.

Al igual que en el diseño del controlador MPC, los límites de saturación del actuador considerados para la planta son de 0 a 1, esto debido a que el experimento de identificación fue realizado aplicando una entrada escalón unitario como señal de excitación para conocer la respuesta del sistema, la cual se usó como dato de entrada en la identificación de sistemas para determinar el modelo matemático de la planta; por lo tanto, este valor de 1 indica que la bomba está operando a su máxima capacidad.

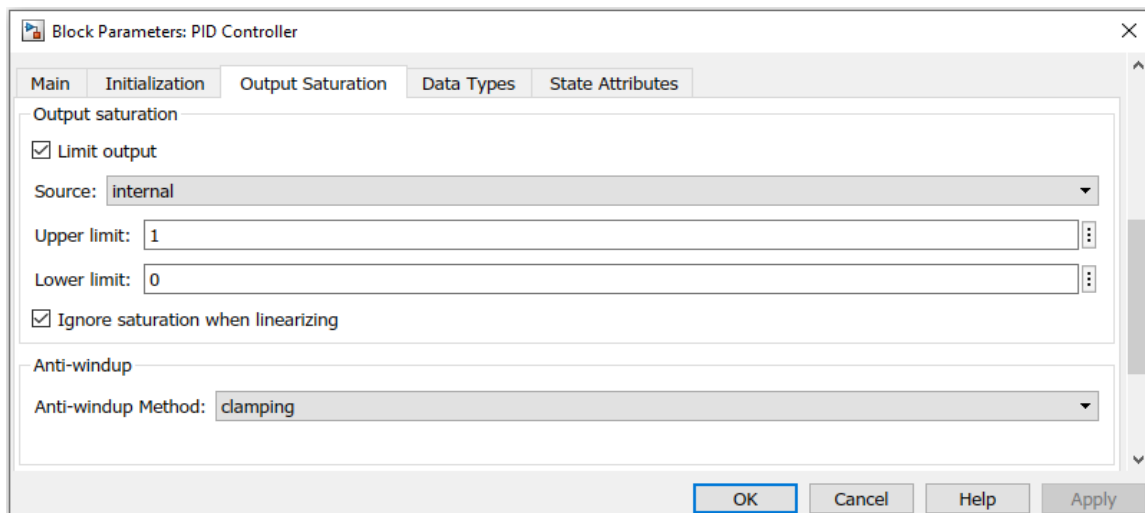


Figura 48. Método anti-windup implementado al controlador PID en Simulink.

Sintonizado e implementado el controlador PID en Simulink, se procedió a ejecutar su simulación junto con el controlador MPC, para comparar y evaluar el rendimiento de ambos controladores en base a criterios de desempeño, cumpliendo así con el tercer objetivo “Comparar mediante simulación las características del controlador avanzado diseñado con controladores convencionales aplicados al prototipo de tanques”.

5.7 Diseño de la interfaz gráfica de usuario (GUI)

Con el fin de que el usuario pueda interactuar con el sistema de tanques interconectados y los controladores diseñados, se desarrolló una interfaz gráfica en MATLAB® que permite al usuario seleccionar cada uno de los casos planteados y ejecutar la simulación de la planta con los controladores PID y MPC, pudiendo observar mediante gráficas su comportamiento y rendimiento. Además, se puede modificar cada uno de los parámetros preestablecidos del MPC y ejecutar una nueva simulación, para realizar comparaciones y observar si al cambiar dichos parámetros el controlador mejora o se vuelve ineficiente.

La interfaz gráfica de usuario se diseñó mediante la aplicación GUIDE (entorno de desarrollo de GUI) de MATLAB®, mostrada en la **Figura 49**. Para acceder a esta aplicación se ejecuta la instrucción “*guide*” en la ventana de comandos.

Los elementos usados en el diseño de la interfaz gráfica son: botones, cuadros de texto, axes (ejes) para la gráfica y las imágenes, contenedores (paneles) para agrupar elementos y menús desplegables (listbox). Al dar doble clic sobre cualquier elemento aparece un cuadro de diálogo para editar sus propiedades (ver **Figura 50**). El Tag indica el nombre de elemento con el cual puede ser identificado y se usa para acceder al mismo o generar una acción a través de programación. En el caso de los cuadros de texto, en la parte de string se ingresa el texto que se quiere mostrar al usuario, mientras que, en los cuadros de texto editables, en la parte mencionada se asigna un valor. Para definir un cuadro de texto como editable se cambia el tipo de estilo del elemento, pasando de texto a editable (`style >> edit`), este se usa para el setpoint y los parámetros del controlador MPC, los cuales pueden ser modificados por el usuario.

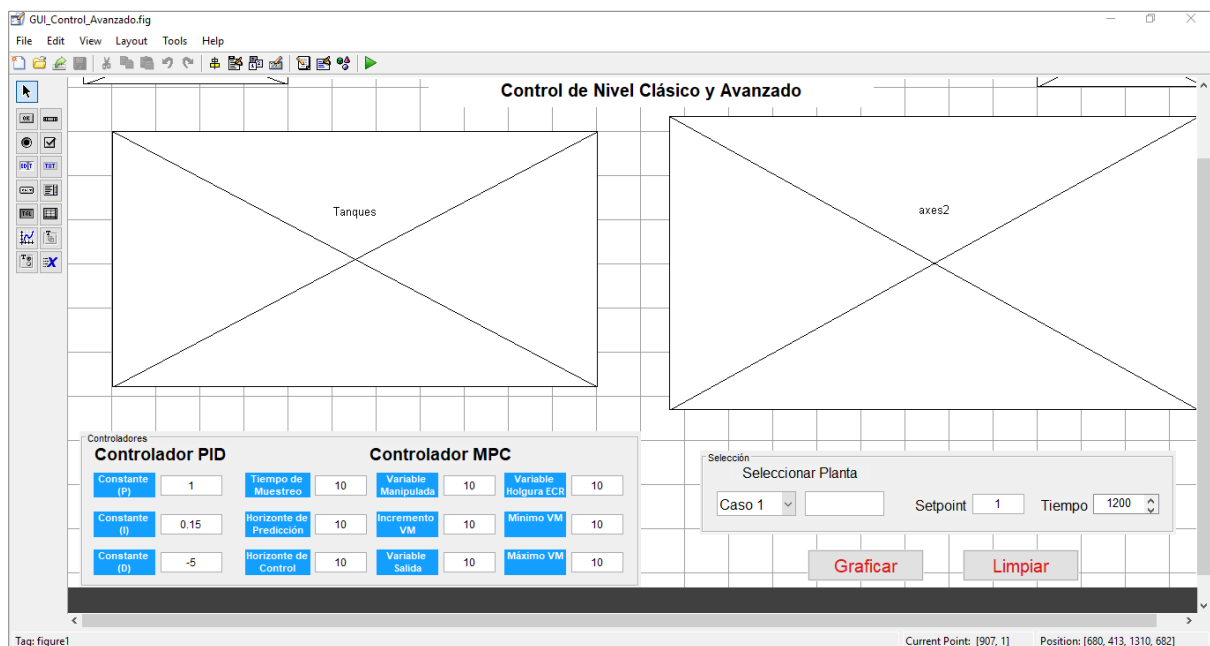


Figura 49. Ventana principal para el diseño de la GUI.

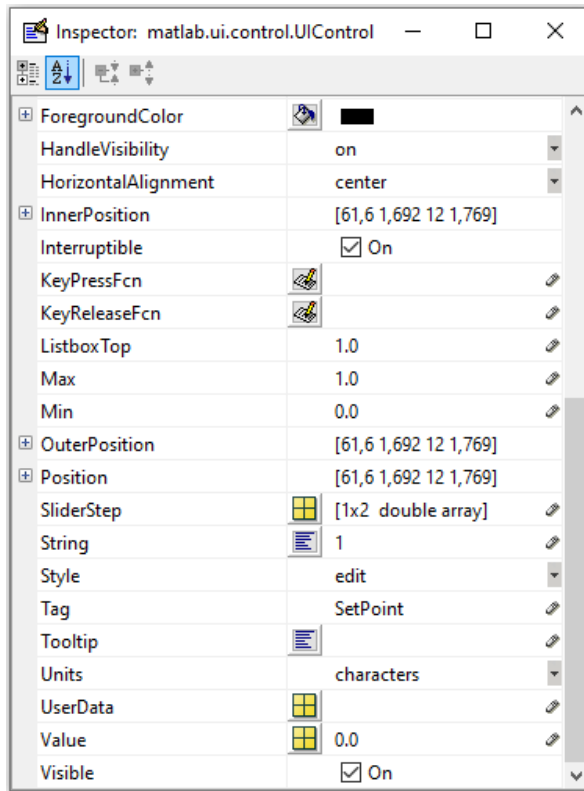


Figura 50. Inspector de propiedades de los elementos de la GUI.

Una vez ubicado todos elementos necesarios para la interfaz, se procede a su programación. Para ello, se selecciona el elemento sobre el que se desea trabajar y se ingresa a la función que está asociada a dicho elemento, dando clic en “*Callback*”, tal como se muestra en la **Figura 51**.

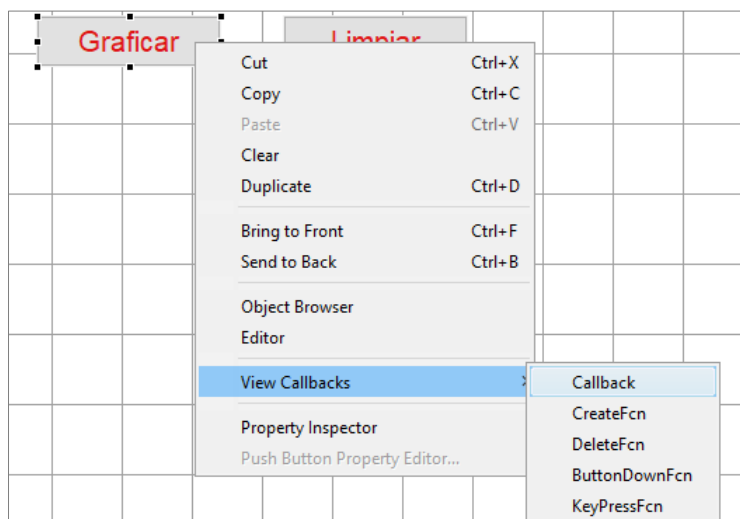


Figura 51. Ingreso a la función asociada al elemento para su programación.

La aplicación GUIDE genera una codificación por defecto, la cual no puede ser editada ya que afectaría al funcionamiento de la interfaz. En la primera función (ver **Anexo 6**), mediante código se da la instrucción al programa para que abra el archivo de Simulink donde

se encuentra el controlador PID y cargue todas las imágenes que deseo presentar al usuario, estas imágenes son el logo de la universidad, el logo de la carrera y el esquema del sistema de tanques interconectados. Todo lo que está dentro de esta función, se ejecuta antes que la interfaz gráfica se haga visible.

Seguido, se procede a programar el cuadro de lista donde se selecciona cada uno de los casos. Lo que se realiza en esta programación es que, al momento de ejecutar la interfaz y seleccionar uno de los casos, automáticamente se carga los parámetros del controlador PID y MPC, dándole la instrucción al programa para que ejecute una determinada acción, cada vez que se escoja una opción dentro de esta lista.

Para ingresar a la función que está asociada al caso uno del cuadro de lista, se ejecuta la acción mostrada en la **Figura 51**. En esta función, para validar que efectivamente se está dentro del elemento que se quiere programar, se debe guiar por el nombre asignado (`listbox1`). Luego, con el código `“indiceSeleccionado = get(hObject, 'Value')”` se verifica lo que el usuario ha seleccionado y se ordena al programa que extraiga ese dato `“opciones = get(hObject, 'String')”`; por lo tanto, si el usuario ha escogido el caso 1, se activa el botón graficar y se procede a llenar los parámetros del PID y del MPC. Esta programación se muestra en el **Anexo 7** y se aplica la misma codificación para todos los casos.

Para cargar y leer los parámetros del controlador MPC en la interfaz, se utiliza dos funciones, una función para enviar los datos (ver **Anexo 8**) y cargarlos al controlador, y otra función que permite extraer los datos ingresados por el usuario (ver **Anexo 9**) para realizar una nueva simulación con estos parámetros.

Finalmente, se procede a programar el botón graficar, ingresando a su función tal como se ha indicado anteriormente. En esta parte se extrae toda la información y se carga a los controladores, dependiendo del caso seleccionado por el usuario.

Mediante la programación mostrada en el **Anexo 10**, se procede al diseño del controlador PID utilizando el archivo de Simulink previamente creado, el cual está listo para ser ejecutado y recibir los datos asignados. Para el diseño del controlador PID, primeramente, se define la función de transferencia del sistema, luego mediante una instrucción se ordena al programa que lea el setpoint y el tiempo de simulación ingresados en el cuadro de texto editable. A continuación, se procede a hacer una lectura de los cuadros de texto editables que contienen los parámetros del PID, cuyos valores fueron asignados y cargados en la codificación anterior. Estos datos deben ser trabajados como datos del tipo string (cadena de caracteres) para que los pueda leer el programa. El comando `“num2str”` sirve para convertir de número a string.

Seguido, con el comando “`set_param('PID_tipo/Setpoint','After',num2str(setpoint))`”, se da la instrucción al programa para que ejecute y abra el archivo de Simulink llamado “PID_tipo”, luego se dirige al bloque nombrado “Setpoint” y dentro de dicho bloque se dirige al cuadro llamado “After”, para setear este parámetro y cargar el valor del setpoint establecido o ingresado por el usuario, tal como se observa en la **Figura 52**. El mismo procedimiento se aplica para los demás parámetros, como el tiempo de muestreo o simulación, el retardo y las constantes, proporcional (P), integral (I) y derivativa (D).

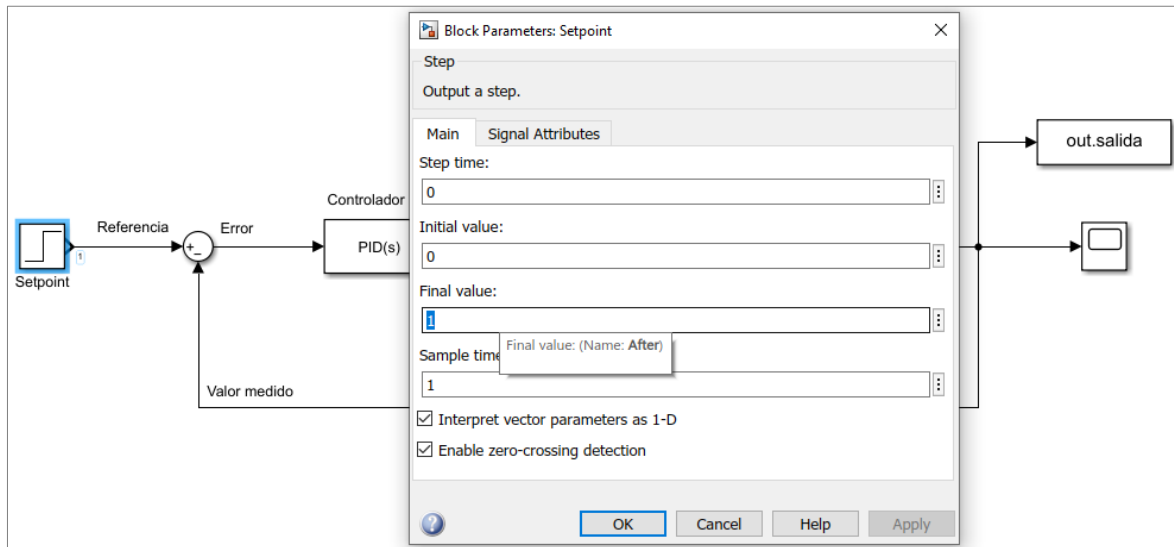


Figura 52. Control PID en Simulink para la GUI.

Para el caso de la configuración y diseño del MPC, la programación realizada solo hace uso de MATLAB®, sin emplear Simulink como lo fue en el caso del PID. La programación para el MPC se muestra en el **Anexo 11**, en el cual, primero se define la función de transferencia de la planta y luego se carga los datos al controlador mediante la función extraer datos, esta función permite extraer los datos asignados en la codificación interna y los nuevos parámetros ingresados por el usuario para la simulación. Seguido, se crea el objeto del controlador MPC el cual almacena cada uno de estos parámetros, luego se define la señal de entrada escalón y se simula la respuesta del controlador.

La codificación descrita para el PID y MPC, se aplica igual para todos los casos, modificando ciertas partes como la función de transferencia y añadiendo el retardo de tiempo en los procesos que presentan un atraso en la salida del sistema.

La ejecución de la interfaz gráfica se realiza desde el entorno de desarrollo de la GUI (ver **Figura 49**) o ejecutando el archivo “GUI_Control_Avanzado.m” desde la ventana principal de MATLAB®. En la **Figura 53**, se presenta la interfaz gráfica desarrollada, el botón graficar se activa al momento de seleccionar el caso que se desea simular.

En el **Anexo 15**, se proporciona el archivo de la GUI desarrollada.

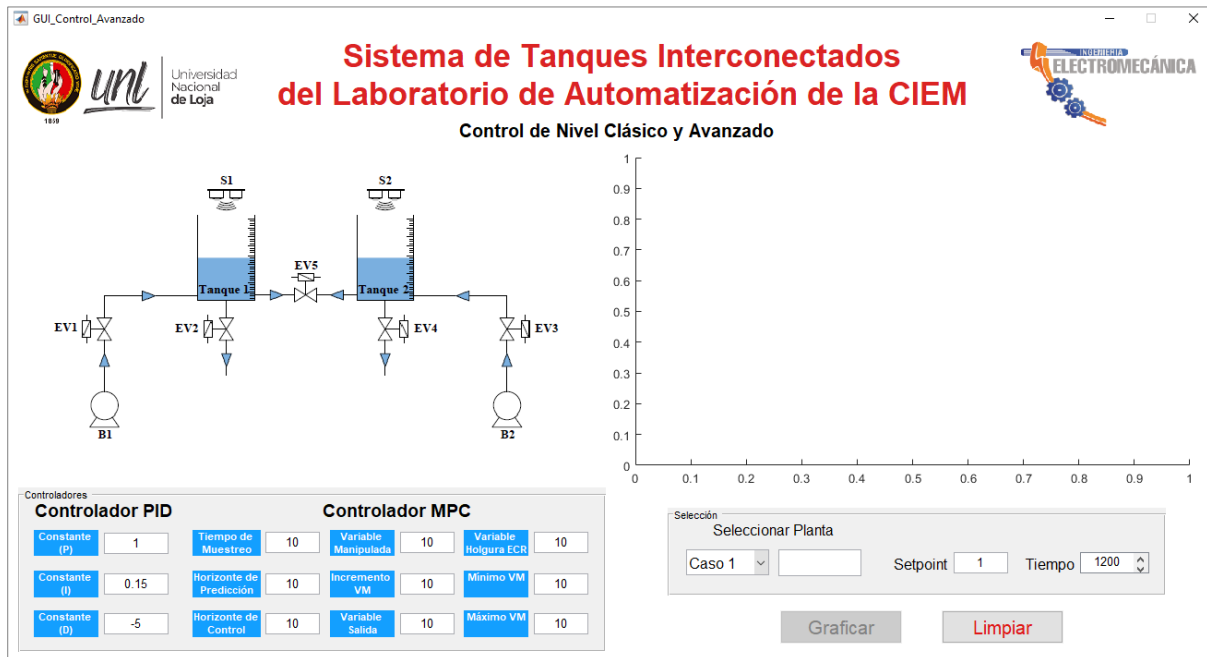


Figura 53. Interfaz gráfica de usuario (GUI).

6. Resultados

6.1 Datos obtenidos

En la **Figura 54**, se evidencia el archivo de Excel generado por el instrumento virtual (VI) diseñado en LabVIEW, una vez que se ha finalizado el experimento de identificación para el primer caso planteado; en el cual se registra el tiempo, la activación de las bombas y el nivel de llenado de los tanques, medido en litros por los sensores ultrasónicos. En cada una de las configuraciones de llenado, se realizó varias pruebas y se promedió cada uno de los datos obtenidos, para determinar las curvas de nivel que representan la dinámica del sistema de tanques interconectados.

Es importante mencionar que los sensores presentan una precisión o error de medición de $\pm 3 \text{ mm}$ por lo que la curva de nivel obtenida no es exactamente lineal, teniendo pequeñas variaciones, lo cual se puede observar en la gráfica de nivel vs. tiempo, generada para este análisis. Estas variaciones también pueden deberse a pequeñas perturbaciones; sin embargo, no afectan al proceso para la obtención del modelo matemático, mediante identificación de sistemas.

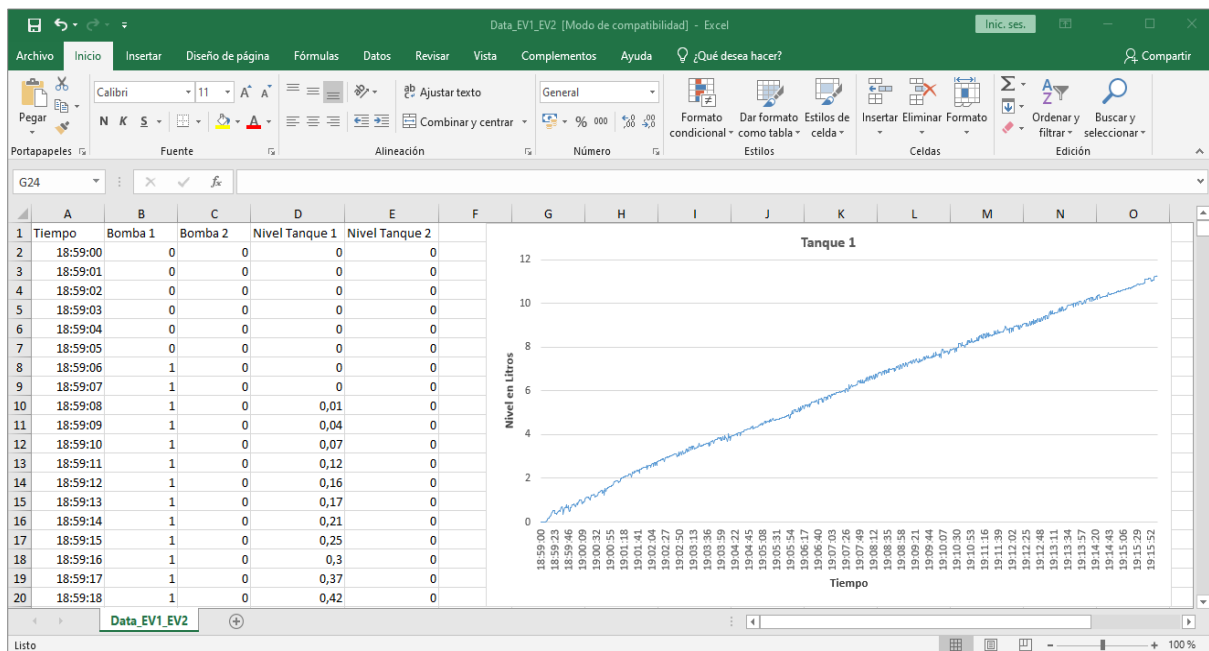


Figura 54. Archivo de Excel generado por LabVIEW el cual contiene el registro de datos obtenidos en el experimento de identificación.

En el **Anexo 12**, se presentan las gráficas de los datos obtenidos que representan la dinámica de llenado de los tanques interconectados en función del tiempo, para cada uno de los casos planteados.

6.2 Modelos matemáticos del sistema de tanques interconectados

Los modelos obtenidos para cada uno de los casos planteados, se detallan a continuación.

6.2.1 Caso 1: Llenado del tanque 1 con la bomba 1

En esta configuración de llenado se activa la bomba 1 y las electroválvulas 1 y 2, circulando el flujo de agua tal como se expone en la **Figura 55**.

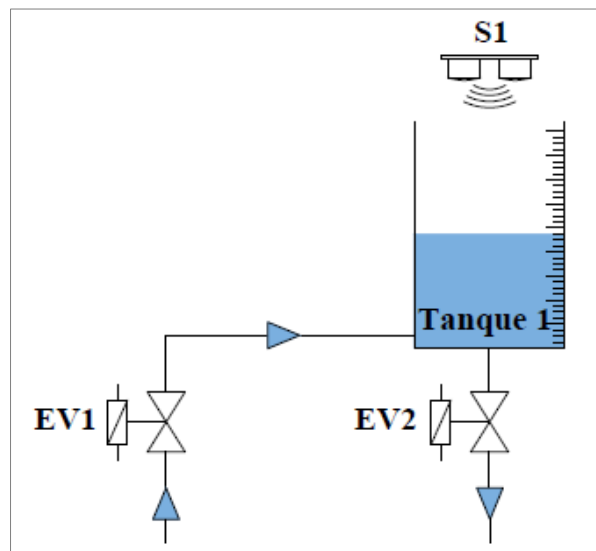


Figura 55. Esquema de llenado del tanque 1 con la bomba 1.

La función de transferencia de la **Ecuación 23** representa la dinámica de llenado del tanque 1 con la bomba 1.

$$G(s) = \frac{Kp}{Tp_1 * s + 1}$$

Donde:

- $Kp = 27,735$
- $Tp_1 = 1\,995,3$

$$G(s) = \frac{27,735}{1\,995,3 s + 1} \quad (23)$$

Al analizar la **Ecuación 23** se observa que el sistema alcanza el 63,2 % del valor final en estado estable (27,735 litros) a los 1 995,3 segundos. Esto debido a que, en base al conjunto de datos obtenidos en el proceso experimental, la herramienta Identificación de Sistemas de MATLAB® determina el punto de equilibrio del sistema dinámico (nivel del tanque) para el cual el flujo de entrada es igual al flujo de salida; por lo tanto, en base a este punto de equilibrio el software linealiza el modelo y determina la función de transferencia que representa la dinámica de nivel del tanque en ese punto en específico.

El modelo obtenido tiene un 95,74 % de precisión, en relación con los datos medidos. Esto se puede observar en la **Figura 56**, donde se hace una comparación entre los datos recopilados en la experimentación y el modelo matemático estimado.

En la **Figura 57**, se puede observar la respuesta del sistema ante una señal de entrada del tipo escalón, así como la ausencia de ceros y la presencia de un polo con parte real negativa ubicado en el semiplano izquierdo del plano complejo “s”, lo que indica que el sistema es estable.

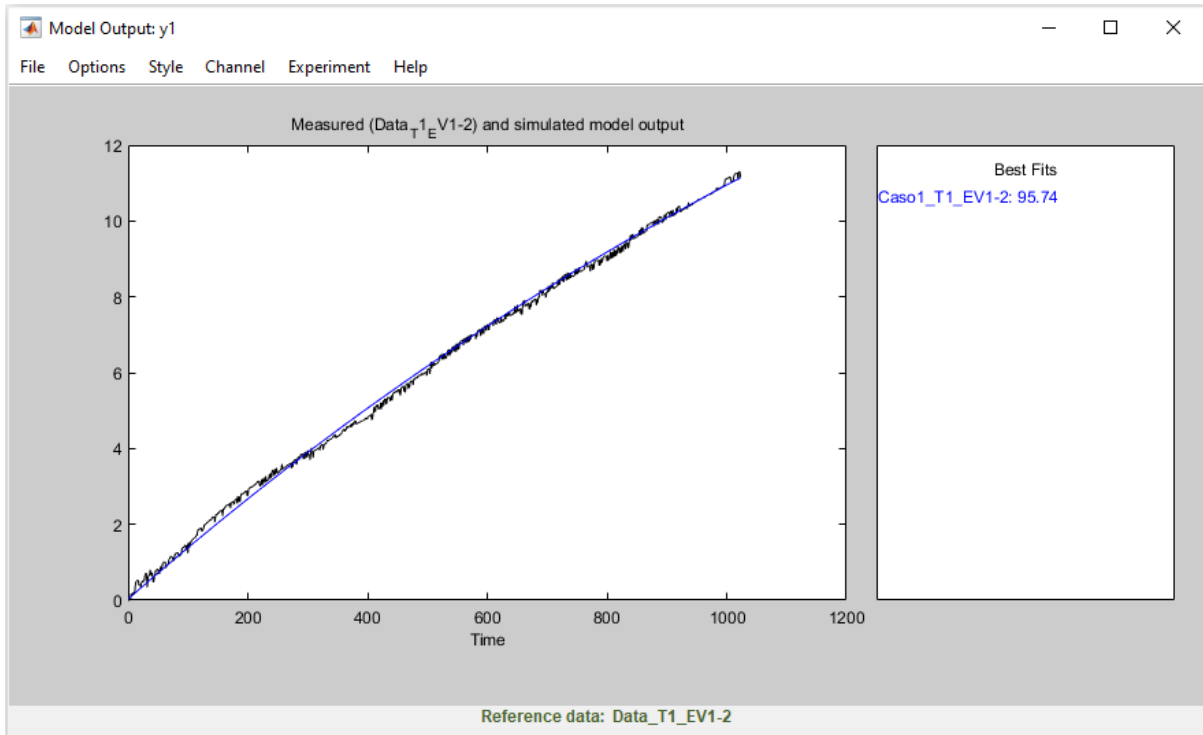


Figura 56. Validación del modelo obtenido para el tanque 1 (caso 1).

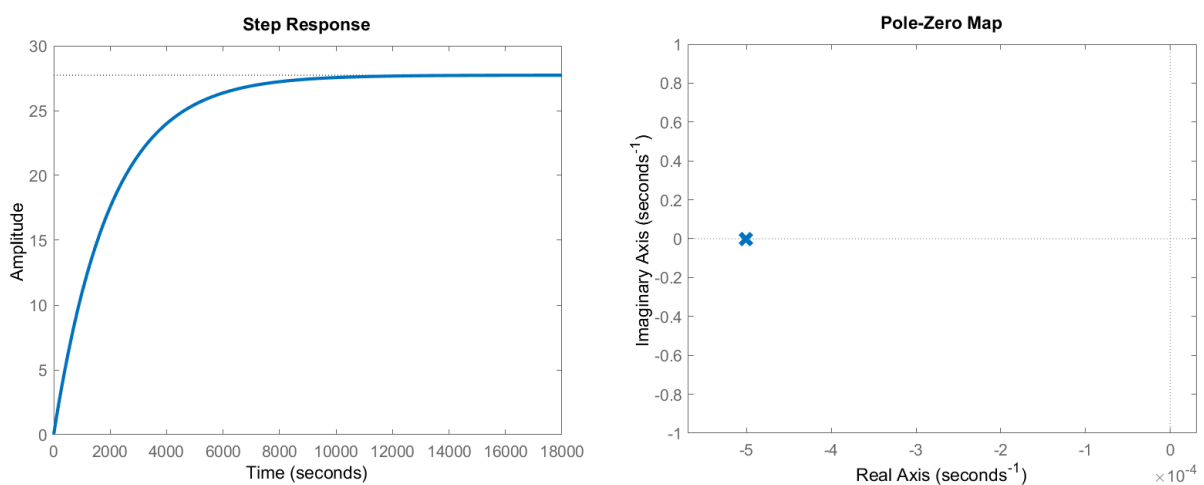


Figura 57. Respuesta del sistema a la señal escalón y diagrama de polos y ceros (caso 1).

6.2.2 Caso 2: Llenado de los tanques interconectados 1 y 2 con la bomba 1

En esta configuración de llenado se activa la bomba 1 y las electroválvulas 1, 4 y 5, circulando el flujo de agua tal como se expone en la **Figura 58**. El objetivo en este caso, es controlar el nivel de agua en el último tanque, es decir, el tanque 2.

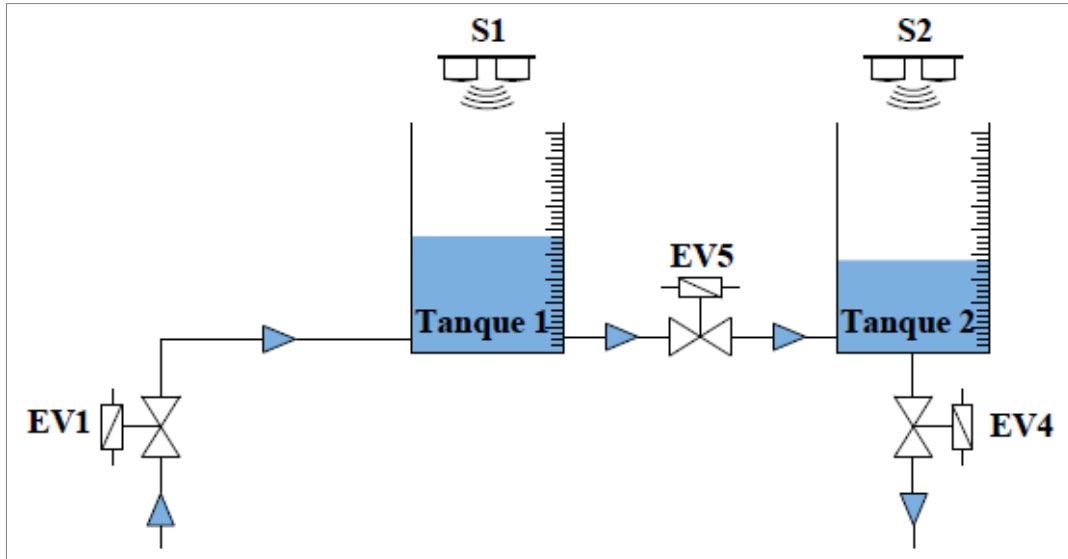


Figura 58. Esquema de llenado de los tanques interconectados 1 y 2 con la bomba 1.

La función de transferencia de la **Ecuación 24** representa la dinámica de llenado del tanque 2 con la bomba 1. En este caso se tiene un retardo de tiempo en la salida del sistema de 86,84 segundos, debido al acople que conecta a los dos tanques, ya que el flujo de agua no circula hacia el tanque 2, hasta que el nivel del tanque 1 alcanza la altura del acople.

$$G(s) = \frac{Kp}{Tp_1 * s + 1} * exp(-Td * s)$$

Donde:

- $Kp = 9,3496$
- $Tp_1 = 1\ 086,9$
- $Td = 86,838$

$$G(s) = \frac{9,3496}{1\ 086,9 s + 1} * exp(-86,84 s) \quad (24)$$

Al analizar la **Ecuación 24** se observa que el sistema alcanza el 63,2 % del valor final en estado estable (9,3496 litros) a los 1 086,9 segundos. En esta configuración de llenado a diferencia del caso 1 en el que se tiene un solo tanque, el sistema presenta una respuesta más lenta debido a que se tiene dos tanques interconectados; por lo cual, el punto de equilibrio

calculado para el tanque 2 es de 9,3496 litros. En base a este punto de equilibrio se linealiza el modelo y se determina la función de transferencia.

El modelo obtenido para el tanque 2 tiene un 96,03 % de precisión, en relación con los datos medidos. Esto se puede observar en la **Figura 59**, donde se hace una comparación entre los datos recopilados en la experimentación y el modelo matemático estimado.

En la **Figura 60**, se puede observar la respuesta del sistema para el tanque 2 ante una señal de entrada del tipo escalón, así como la ausencia de ceros y la presencia de un polo con parte real negativa ubicado en el semiplano izquierdo del plano complejo “s”, lo que indica que el sistema es estable.

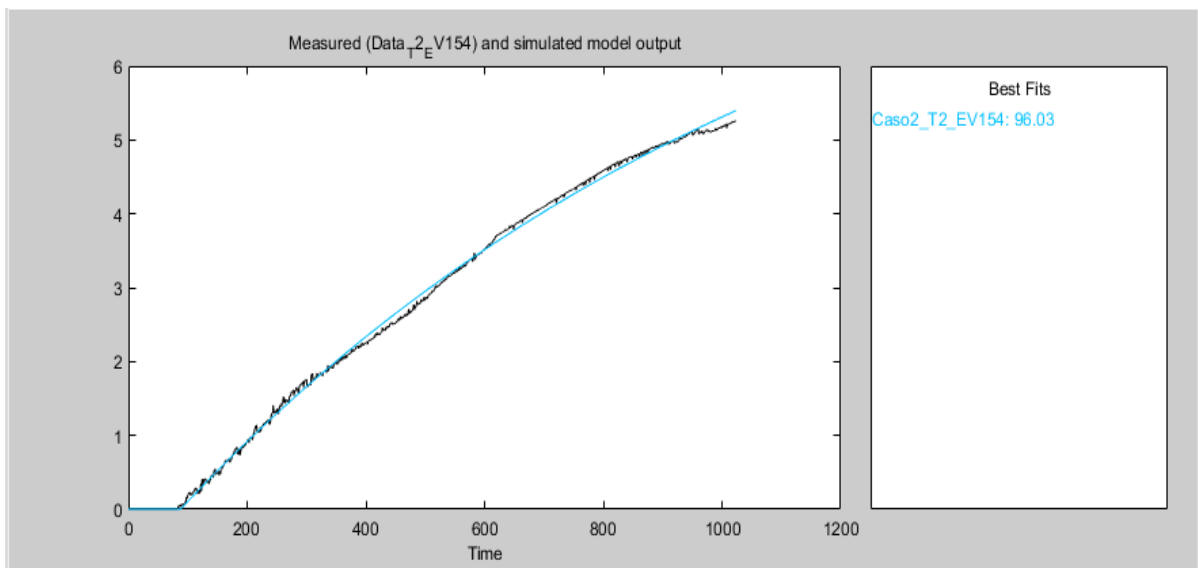


Figura 59. Validación del modelo obtenido para el tanque 2 (caso 2).

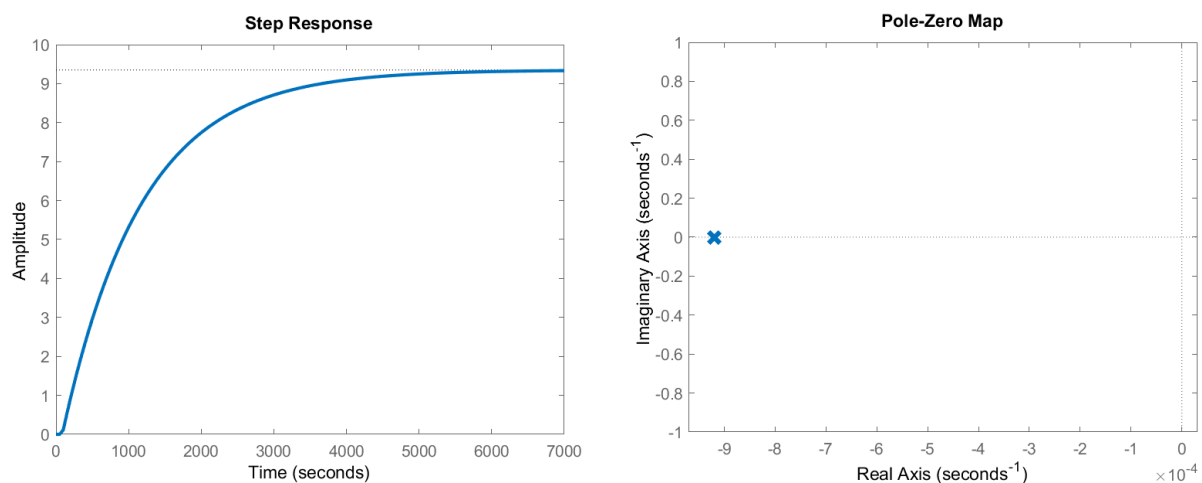


Figura 60. Respuesta del sistema a la señal escalón y diagrama de polos y ceros para el tanque 2 (caso 2).

En el **Anexo 13**, se proporciona la función de transferencia estimada para el tanque 1 en esta configuración de llenado, en caso de que se requiera analizar su comportamiento.

6.2.3 Caso 3: Llenado del tanque 2 con la bomba 2

En esta configuración de llenado se activa la bomba 2 y las electroválvulas 3 y 4, circulando el flujo de agua tal como se expone en la **Figura 61**.

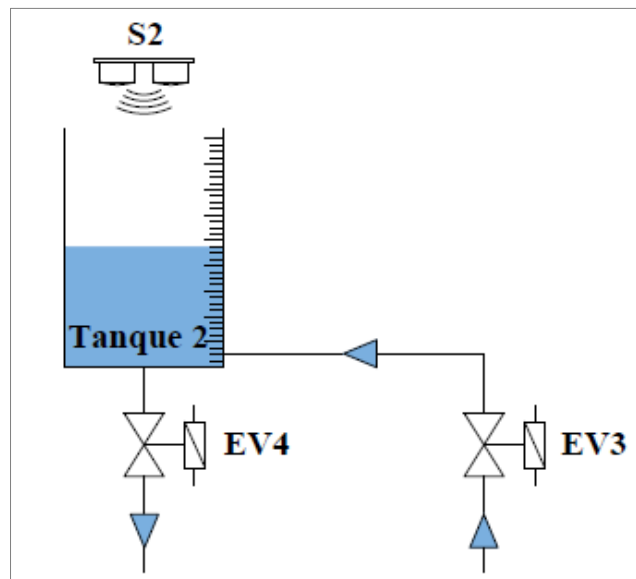


Figura 61. Esquema de llenado del tanque 2 con la bomba 2.

La función de transferencia de la **Ecuación 25** representa la dinámica de llenado del tanque 2 con la bomba 2. En este caso se tiene un retardo de tiempo de 18,47 segundos debido a que el sistema presenta una dinámica de llenado lenta, por lo que el sensor no detecta una variación en el nivel hasta que transcurre este tiempo.

$$G(s) = \frac{Kp}{Tp_1 * s + 1} * exp(-Td * s)$$

Donde:

- $Kp = 29,805$
- $Tp_1 = 5\,932,8$
- $Td = 18,47$

$$G(s) = \frac{29,805}{5\,932,8 s + 1} * exp(-18,47 s) \quad (25)$$

El modelo obtenido tiene un 96,7 % de precisión, en relación con los datos medidos. Esto se puede observar en la **Figura 62**, donde se hace una comparación entre los datos recopilados en la experimentación y el modelo matemático estimado.

En la **Figura 63**, se puede observar la respuesta del sistema ante una señal de entrada del tipo escalón, así como la ausencia de ceros y la presencia de un polo con parte real negativa

ubicado en el semiplano izquierdo del plano complejo “s”, lo que indica que el sistema es estable.

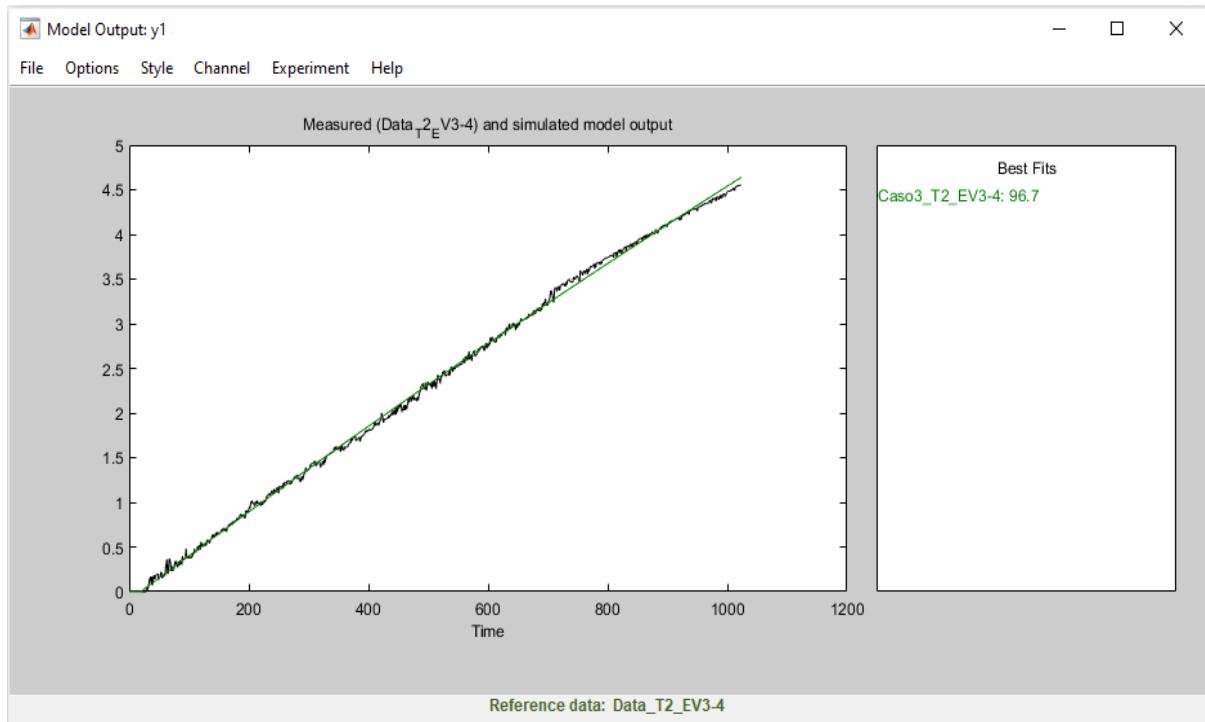


Figura 62. Validación del modelo obtenido para el tanque 2 (caso 3).

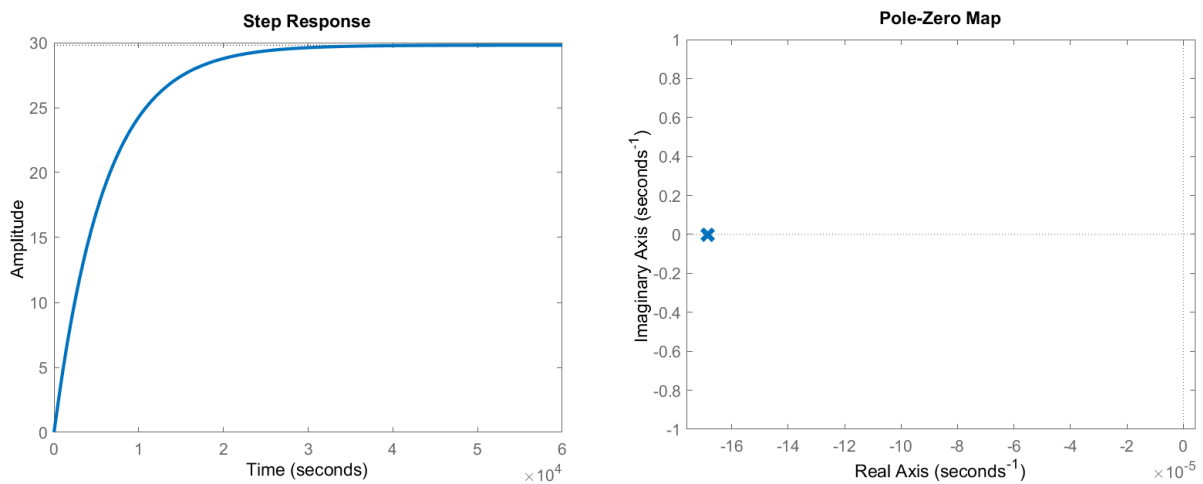


Figura 63. Respuesta del sistema a la señal escalón y diagrama de polos y ceros (caso 3).

6.2.4 Caso 4: Llenado de los tanques interconectados 1 y 2 con la bomba 2

En esta configuración de llenado se activa la bomba 2 y las electroválvulas 2, 3 y 5, circulando el flujo de agua tal como se expone en la **Figura 64**. El objetivo es controlar el nivel de agua en el último tanque, es decir, el tanque 1.

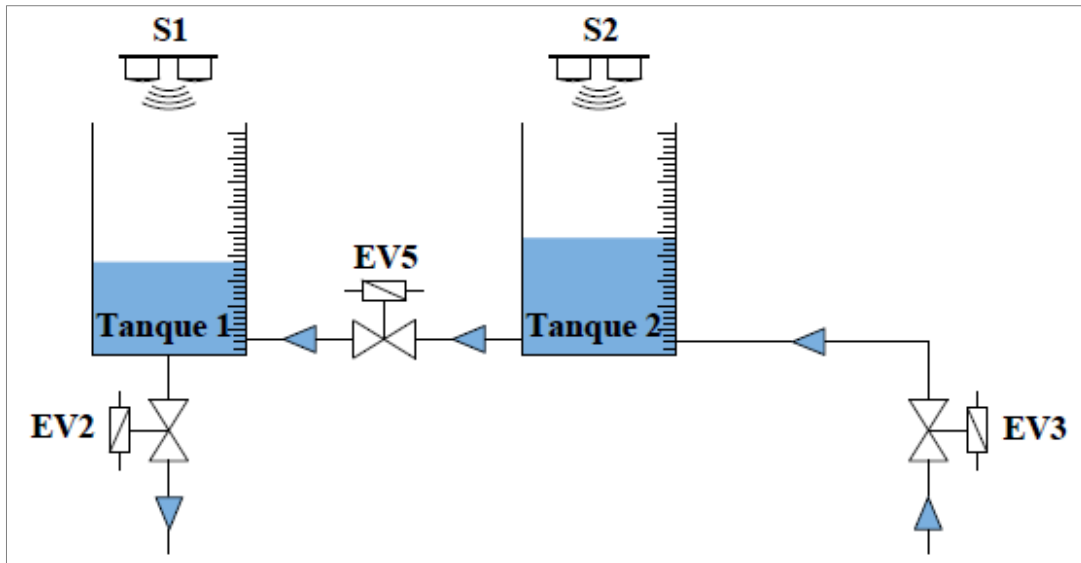


Figura 64. Esquema de llenado de los tanques interconectados 1 y 2 con la bomba 2.

La función de transferencia de la **Ecuación 26** representa la dinámica de llenado del tanque 1 con la bomba 2. En este caso se tiene un retardo de tiempo en la salida del sistema de 133,2 segundos, debido al acople que conecta a los dos tanques, ya que el flujo de agua no circula hacia el tanque 1, hasta que el nivel del tanque 2 alcanza la altura del acople.

$$G(s) = \frac{Kp}{Tp_1 * s + 1} * exp(-Td * s)$$

Donde:

- Kp = 5,4898
- Tp₁ = 863,4
- Td = 133,2

$$G(s) = \frac{5,4898}{863,4 s + 1} * exp(-133,2 s) \quad (26)$$

El modelo obtenido para el tanque 1 tiene un 95,97 % de precisión, en relación con los datos medidos. Esto se puede observar en la **Figura 65**, donde se hace una comparación entre la curva obtenida a partir de los datos recopilados en la experimentación y la curva obtenida del modelo matemático.

En la **Figura 66**, se puede observar la respuesta del sistema para el tanque 1 ante una señal de entrada del tipo escalón, así como la ausencia de ceros y la presencia de un polo con parte real negativa ubicado en el semiplano izquierdo del plano complejo “s”, lo que indica que el sistema es estable.

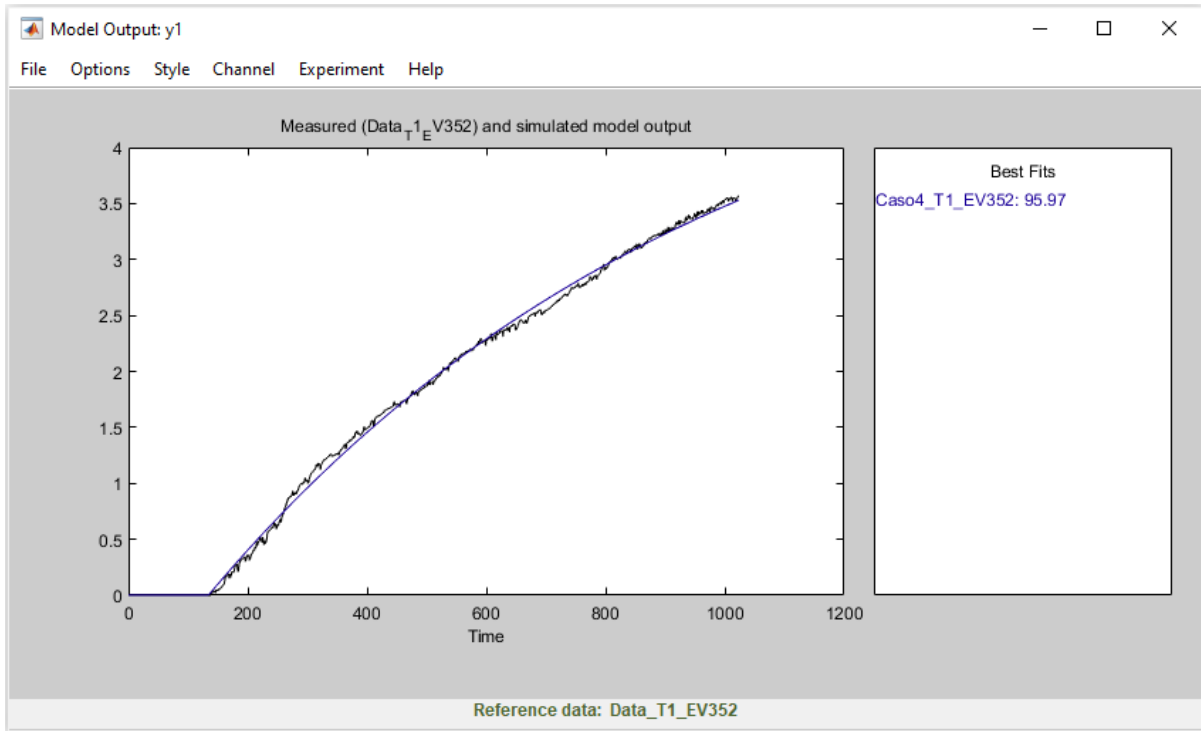


Figura 65. Validación del modelo obtenido para el tanque 1 (caso 4).

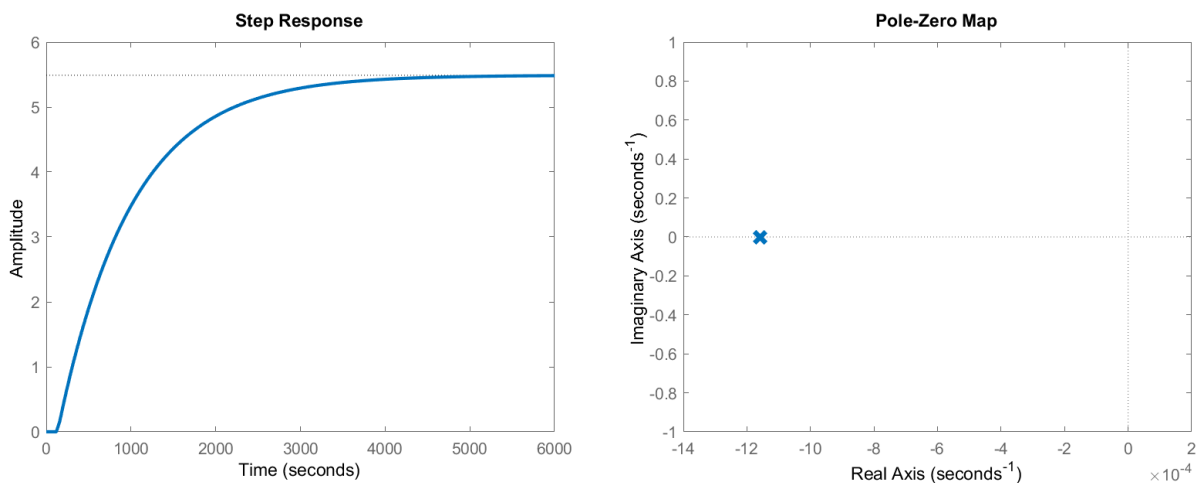


Figura 66. Respuesta del sistema a la señal escalón y diagrama de polos y ceros para el tanque 1 (caso 4).

En el **Anexo 14**, se proporciona la función de transferencia estimada para el tanque 2 en esta configuración de llenado, en caso de que se requiera analizar su comportamiento.

6.3 Diseño del controlador avanzado MPC

En cada uno de los casos planteados se ejecutó varias sintonizaciones, con el fin de determinar los mejores parámetros para un control MPC óptimo. Se elaboró una tabla con los parámetros y los errores obtenidos de cada sintonización ejecutada con el algoritmo desarrollado.

Es importante mencionar que el proceso de llenado de los tanques es lento, debido a sus características constructivas.

6.3.1 Controlador MPC para el caso 1

6.3.1.1 Parámetros obtenidos

Los parámetros para el MPC del tanque 1 se muestran en la **Tabla 6**. El mínimo error obtenido en la sintonización con el algoritmo MAGO fue del 4,82 %, siendo este control el más eficiente.

Tabla 6. Parámetros del MPC para el tanque 1 (caso 1).

Tiempo de muestreo	Horizonte de predicción	Horizonte de control	Pesos			Restricciones			Error %
			Variable manipulada	Incremento de la var. manipulada	Variable de salida	Variable de holgura	Variable manipulada		
Ts	Hp	Hc	u	Δu	y	ECR	Mín	Máx	
1	29	5	1	0,1	1	100	0	1	46,20
1	40	5	1	0,1	1	100	0	1	40,90
1	49	4	1	0,1	1	100	0	1	36,44
1	69	7	1	0,1	1	100	0	1	30,20
1	40	5	1	0,1	1,5	100	0	1	21,50
1	40	5	1	0,5	2	100	0	1	12,87
1	40	5	1	0,1	3	100	0	1	7,87
1	40	5	1	0,1	4	100	0	1	6,45
1	19	3	1	0,1	5	100	0	1	6,37
1	41	6	1	0,1	5	100	0	1	4,82

6.3.1.2 Implementación del controlador MPC en Simulink

En la **Figura 67**, se muestra el diagrama del sistema de control en lazo cerrado del MPC implementado en Simulink, ubicando en éste la función de transferencia obtenida en la identificación de sistemas.

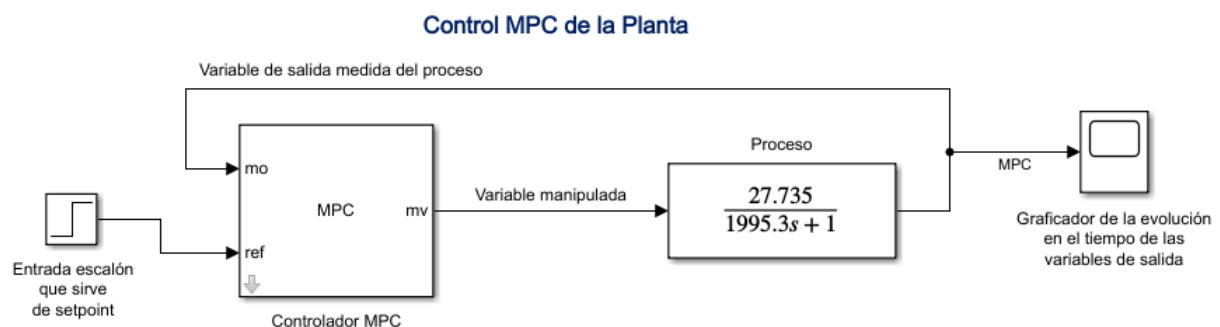


Figura 67. Diagrama de control del MPC en Simulink para el tanque 1 (caso 1).

En la **Figura 68**, se presenta la comparación y el resultado de tres controladores MPC sintonizados con el algoritmo diseñado, para el control de nivel en el tanque 1 (caso 1). La línea azul representa la dinámica del sistema sin control, a través de la función de transferencia obtenida mediante identificación de sistemas, la línea entrecortada de color rojo indica el setpoint y la línea de color verde, marrón y negro, corresponde al MPC sintonizado con un error del 4,82 %, 12,87 % y 21,5 %, respectivamente. En esta gráfica se puede observar que, el control MPC que presentó un resultado de sintonización con un error del 4,82 % sigue la trayectoria del modelo de la planta y alcanza el setpoint en un tiempo menor, estabilizándose a los 128 segundos, mientras que los demás controladores, que presentaron un error del 12,87 % y 21,5 % son menos eficientes a comparación del primero. Entre mayor es el error de sintonización mayor es el error en estado estacionario del sistema de control.

Por otro lado, en la **Figura 69** se puede observar cómo el controlador MPC que presentó el mínimo error en su sintonización, sigue la trayectoria del modelo de la planta sin sobrepasar los límites de operación máximos del actuador, alcanzando el estado estacionario para el nivel asignado en el menor tiempo posible, presentando un control eficiente.

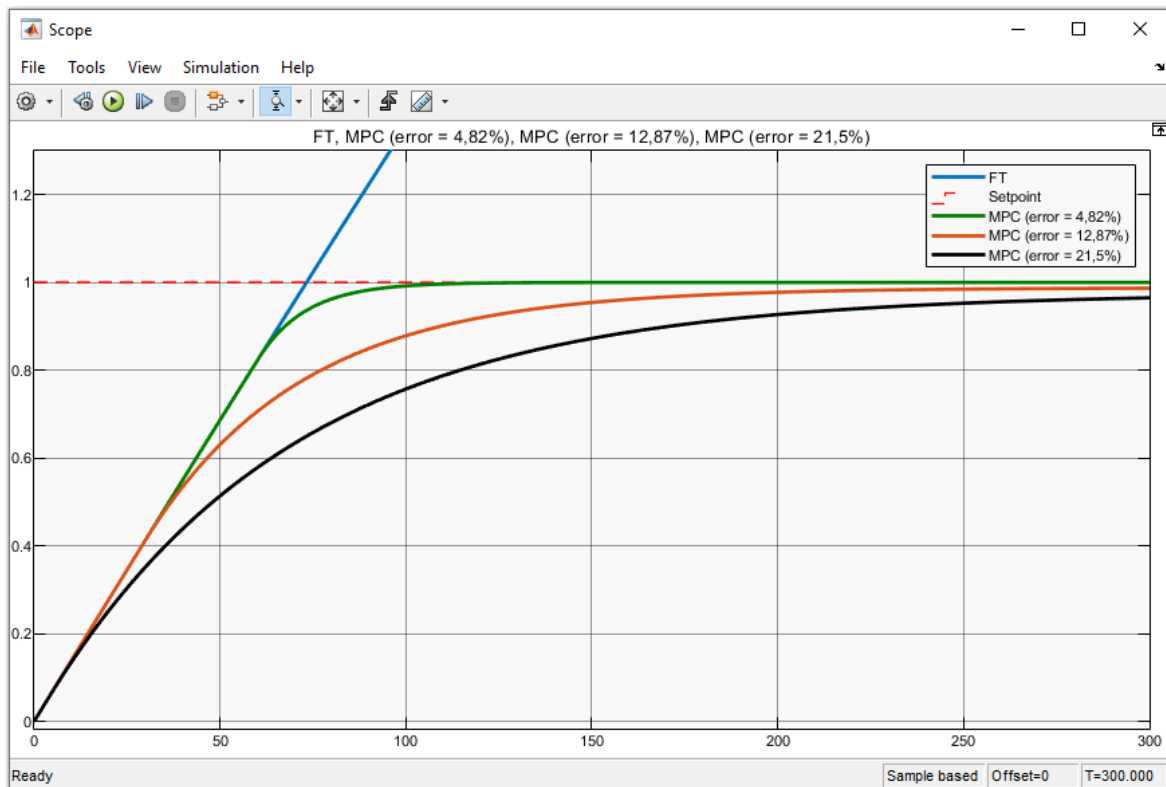


Figura 68. Gráfica comparativa de los resultados de sintonización obtenidos para el tanque 1 (caso 1) con el algoritmo diseñado.

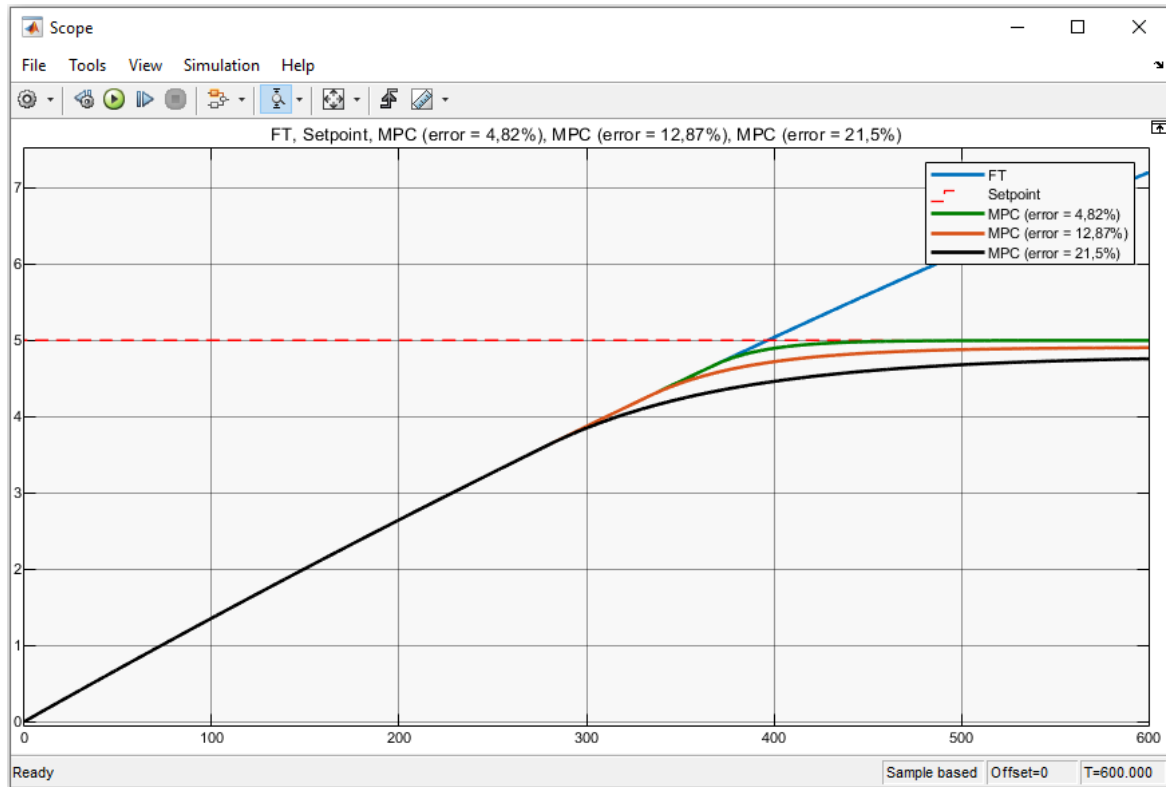


Figura 69. Comportamiento del controlador MPC para el tanque 1 (caso 1) con los diferentes errores de sintonización determinados por el algoritmo desarrollado.

6.3.2 Controlador MPC para el caso 2

6.3.2.1 Parámetros obtenidos

En esta configuración de llenado se tiene una entrada (caudal entregado por la bomba 1) y dos salidas (tanque 1 y 2), el objetivo en este caso es controlar el nivel en el último tanque, el cual corresponde al tanque 2.

Los parámetros para el MPC del tanque 2 se muestran en la **Tabla 7**. El mínimo error obtenido en la sintonización con el algoritmo MAGO fue del 3,10 %, siendo este control el más eficiente.

Tabla 7. Parámetros del MPC para el tanque 2 (caso 2).

Tiempo de muestreo	Horizonte de predicción	Horizonte de control	Variable manipulada	Pesos			Restricciones			Error %
				Incremento de la var. manipulada	Variable de salida	Variable de holgura	Variable manipulada			
Ts	Hp	Hc	u	Δu	y	ECR	Mín	Máx	%	
1	89	17	1	0,1	1	100	0	1	41,67	
1	93	15	1	0,1	1	100	0	1	41,56	
1	95	14	1	0,1	1	100	0	1	40,32	
1	97	22	1	0,1	1	100	0	1	39,36	

Tiempo de muestreo	Horizonte de predicción	Horizonte de control	Pesos				Restricciones			Error %
			Variable manipulada	Incremento de la var. manipulada	Variable de salida	Variable de holgura	Variable manipulada			
			u	Δu	y	ECR	Mín	Máx		
T_s	H_p	H_c								
1	96	12	1	0,1	2	100	0	1	30,75	
1	99	10	1	0,1	2	100	0	1	28,69	
1	109	16	1	0,1	2	100	0	1	20,32	
1	118	27	1	0,1	3	100	0	1	12,90	
1	104	12	1	0,8	5	100	0	1	5,63	
1	138	14	1	0,1	5	100	0	1	3,10	

6.3.2.2 Implementación del controlador MPC en Simulink

En la **Figura 70**, se muestra el diagrama del sistema de control en lazo cerrado del MPC implementado en Simulink, ubicando en éste la función de transferencia y el retardo de tiempo.

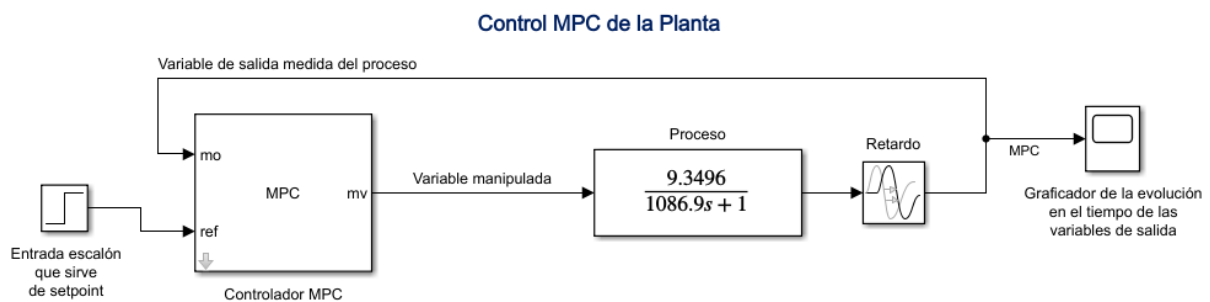


Figura 70. Diagrama de control del MPC en Simulink para el tanque 2 (caso 2).

En la **Figura 71**, se presenta la comparación y el resultado de tres controladores MPC sintonizados con el algoritmo diseñado, para el control de nivel en el tanque 2 (caso 2). La línea azul representa la dinámica del sistema sin control, a través de la función de transferencia obtenida mediante identificación de sistemas, la línea entrecortada de color rojo indica el setpoint y la línea de color verde, marrón y negro, corresponde al MPC sintonizado con un error del 3,1 %, 12,9 % y 20,3 %, respectivamente. En esta gráfica se puede observar que, el control MPC que presentó un resultado de sintonización con un error del 3,1 % sigue la trayectoria del modelo de la planta y alcanza el setpoint en un tiempo menor, estabilizándose a los 300 segundos, mientras que los demás controladores, que presentaron un error del 12,9 % y 20,3 % son menos eficientes a comparación del primero. Entre mayor es el error de sintonización mayor es el error en estado estacionario del sistema de control.

Por otro lado, en la **Figura 72** se puede observar cómo el controlador MPC que presentó el mínimo error en su sintonización, sigue la trayectoria del modelo de la planta sin sobrepasar

los límites de operación máximos del actuador, alcanzando el estado estacionario para el nivel asignado en el menor tiempo posible, presentando un control eficiente.

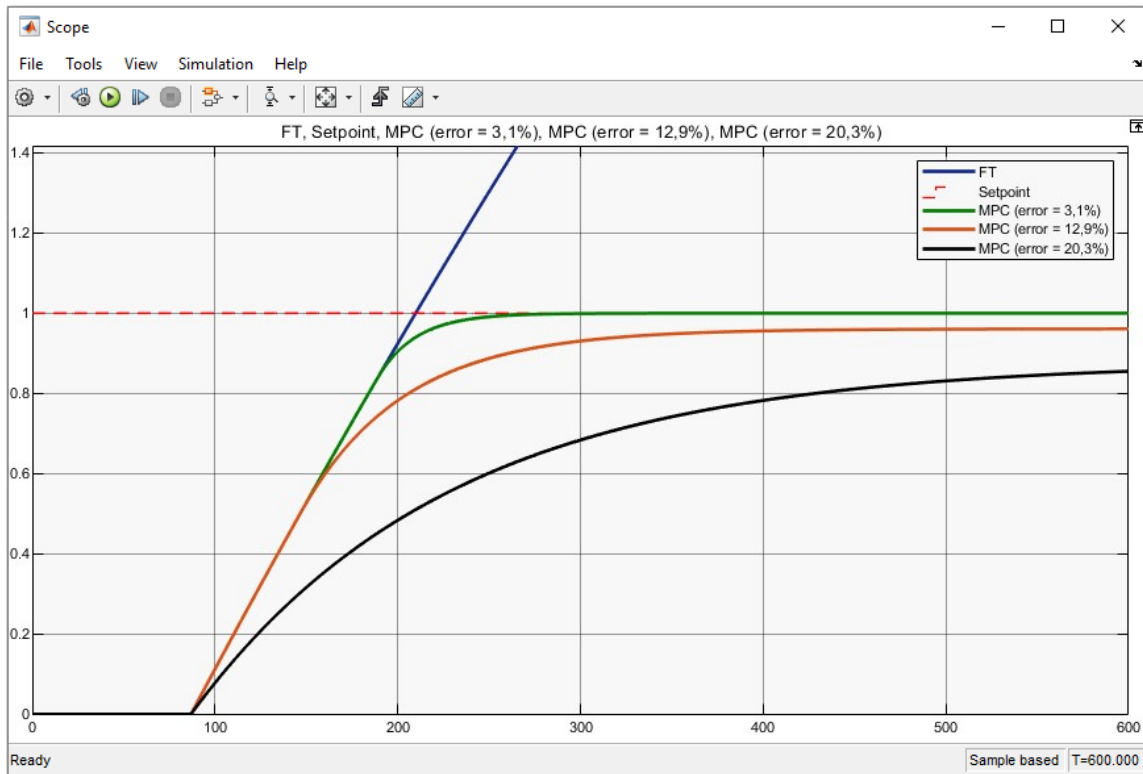


Figura 71. Gráfica comparativa de los resultados de sintonización obtenidos para el tanque 2 (caso 2) con el algoritmo diseñado.

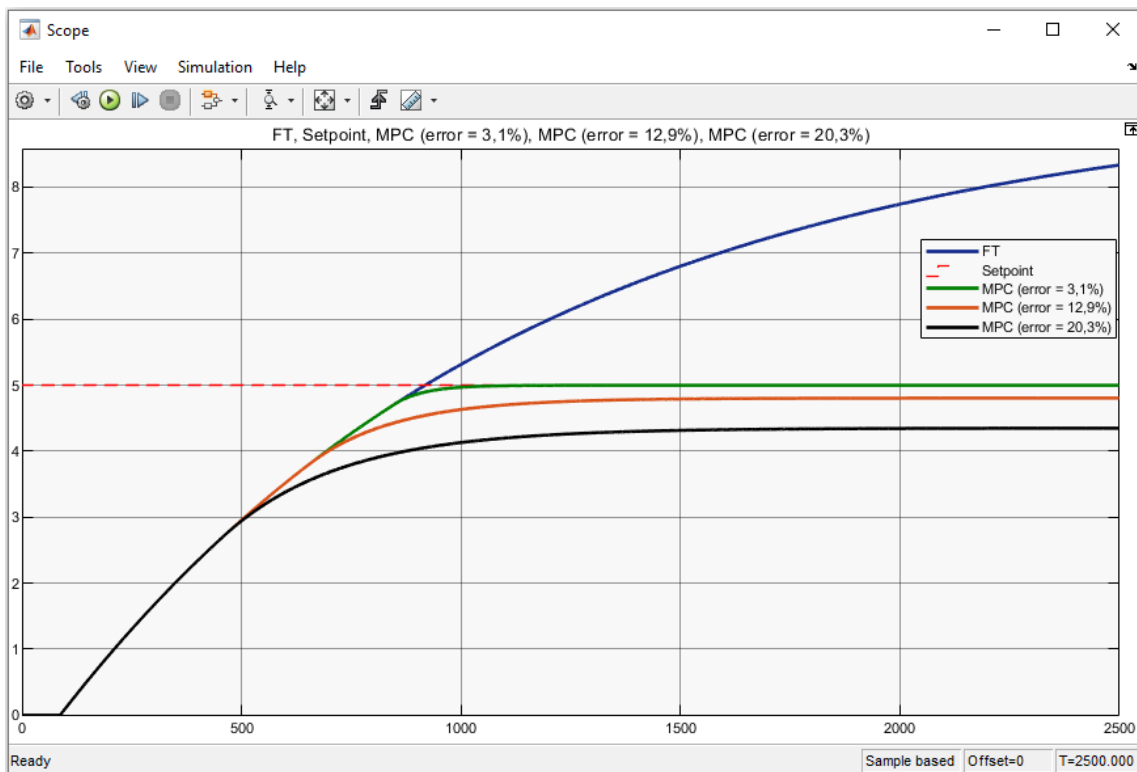


Figura 72. Comportamiento del controlador MPC para el tanque 2 (caso 2) con los diferentes errores de sintonización determinados por el algoritmo desarrollado.

6.3.3 Controlador MPC para el caso 3

6.3.3.1 Parámetros obtenidos

Los parámetros para el MPC del tanque 2 se muestran en la **Tabla 8**. El mínimo error obtenido en la sintonización con el algoritmo MAGO fue del 2,43 %, siendo este control el más eficiente.

Tabla 8. Parámetros del MPC para el tanque 2 (caso 3).

Tiempo de muestreo	Horizonte de predicción	Horizonte de control	Variable manipulada	Pesos			Restricciones		Error %
				Incremento de la var. manipulada	Variable de salida	Variable de holgura	Variable manipulada		
Ts	Hp	Hc	u	Δu	y	ECR	Mín	Máx	%
1	30	6	1	0,1	1	100	0	1	50,43
1	39	5	1	0,1	1	100	0	1	47,93
1	30	6	1	0,1	2	100	0	1	41,94
1	40	15	1	0,1	2	100	0	1	34,60
1	39	12	1	0,1	3	100	0	1	18,35
1	52	7	1	0,1	4	100	0	1	7,19
1	47	9	1	0,1	5	100	0	1	5,04
1	98	10	1	0,1	6	100	0	1	2,43

6.3.3.2 Implementación del controlador MPC en Simulink

En la **Figura 73**, se muestra el diagrama del sistema de control en lazo cerrado del MPC implementado en Simulink, ubicando en éste la función de transferencia y el retardo de tiempo.

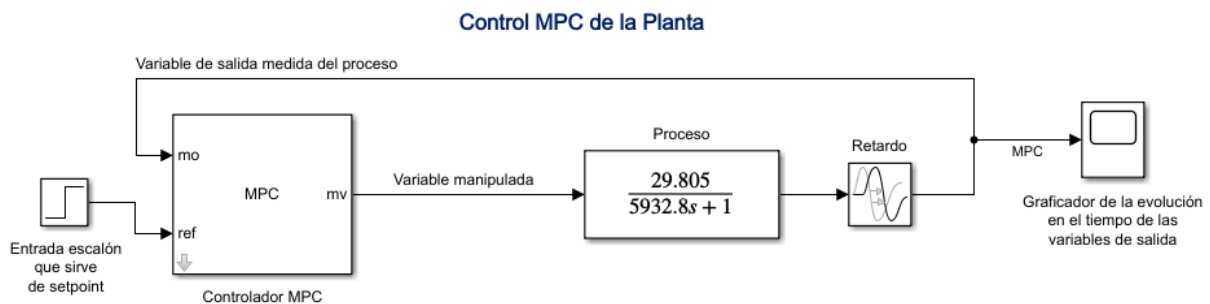


Figura 73. Diagrama de control del MPC en Simulink para el tanque 2 (caso 3).

En la **Figura 74**, se presenta la comparación y el resultado de tres controladores MPC sintonizados con el algoritmo diseñado, para el control de nivel en el tanque 2 (caso 3). La línea azul representa la dinámica del sistema sin control, a través de la función de transferencia obtenida mediante identificación de sistemas, la línea entrecortada de color rojo indica el setpoint y la línea de color verde, marrón y negro, corresponde al MPC sintonizado con un

error del 2,43 %, 7,19 % y 18,35 %, respectivamente. En esta gráfica se puede observar que, el control MPC que presentó un resultado de sintonización con un error del 2,43 % sigue la trayectoria del modelo de la planta y alcanza el setpoint en un tiempo menor, estabilizándose a los 325 segundos, mientras que los demás controladores, que presentaron un error del 7,19 % y 18,35 % son menos eficientes a comparación del primero. Entre mayor es el error de sintonización mayor es el error en estado estacionario del sistema de control.

Por otro lado, en la **Figura 75** se puede observar cómo el controlador MPC que presentó el mínimo error en su sintonización, sigue la trayectoria del modelo de la planta sin sobrepasar los límites de operación máximos del actuador, alcanzando el estado estacionario para el nivel asignado en el menor tiempo posible, presentando un control eficiente.

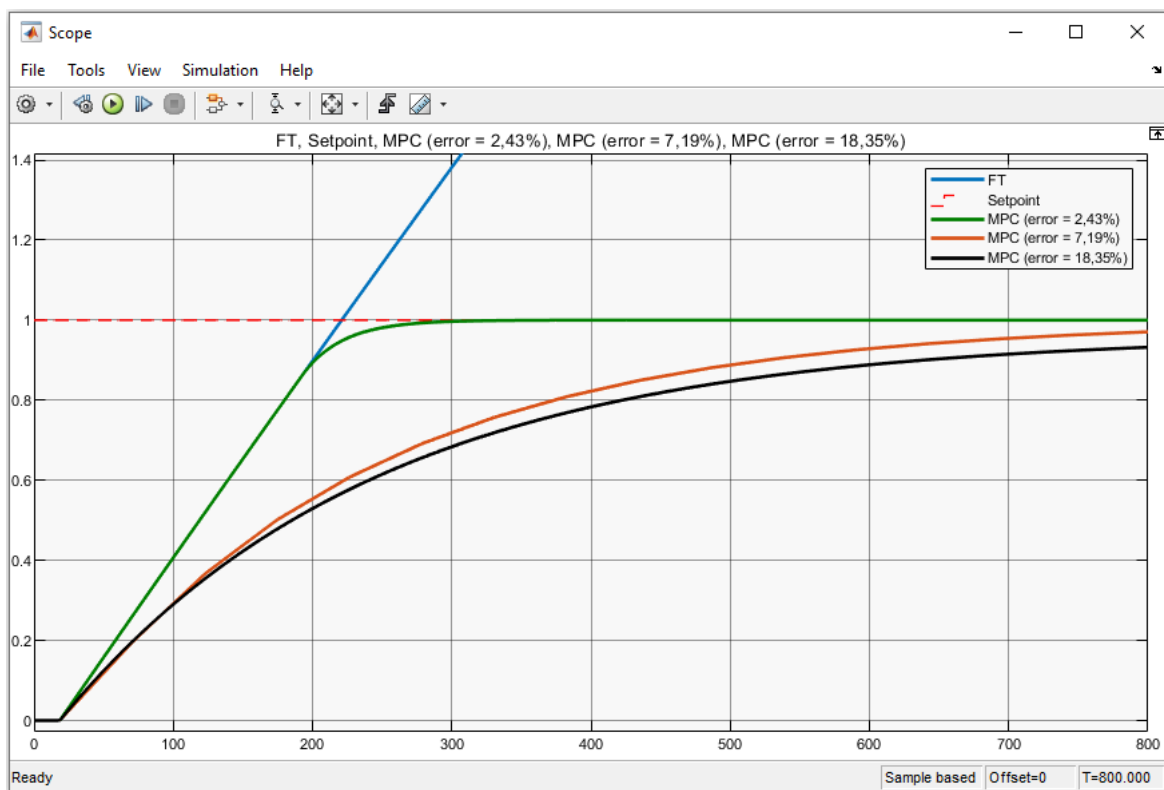


Figura 74. Gráfica comparativa de los resultados de sintonización obtenidos para el tanque 2 (caso 3) con el algoritmo diseñado.

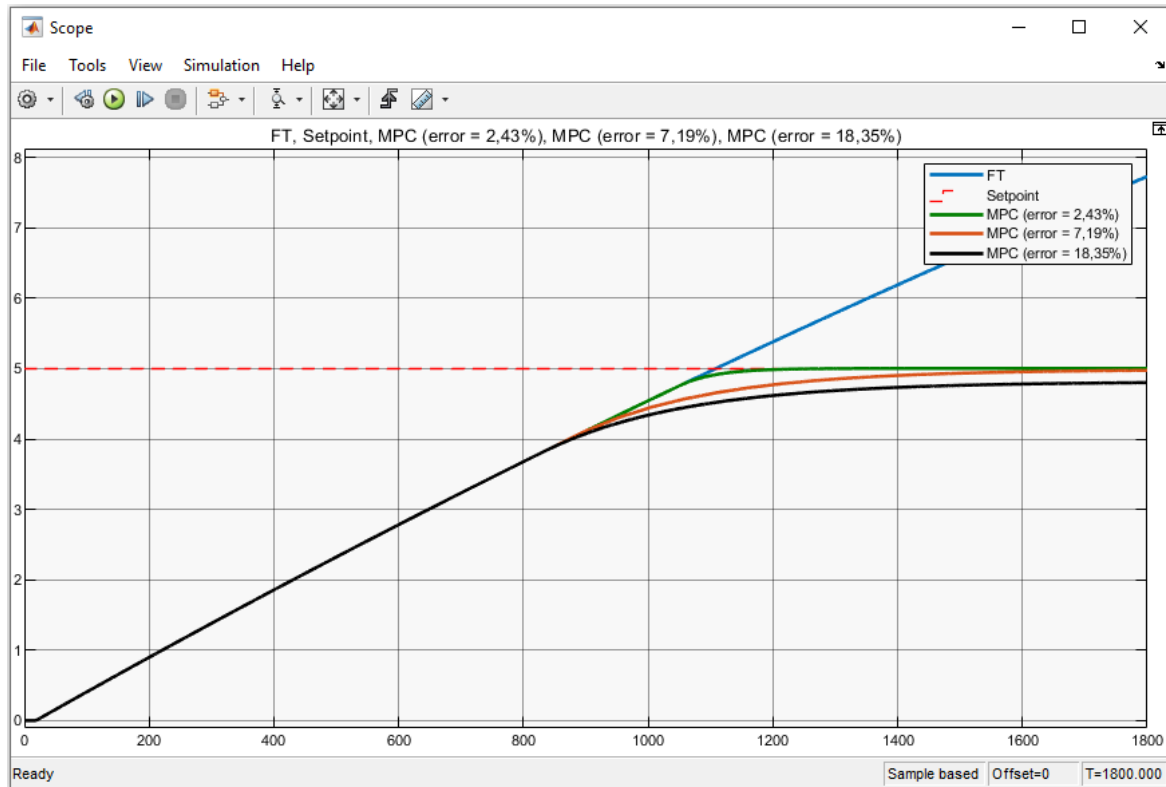


Figura 75. Comportamiento del controlador MPC para el tanque 2 (caso 3) con los diferentes errores de sintonización determinados por el algoritmo desarrollado.

6.3.4 Controlador MPC para el caso 4

6.3.4.1 Parámetros obtenidos

En esta configuración de llenado se tiene una entrada (caudal entregado por la bomba 2) y dos salidas (tanque 1 y 2), el objetivo en este caso es controlar el nivel en el último tanque, el cual corresponde al tanque 1.

Los parámetros para el MPC del tanque 1 se indican en la **Tabla 9**. El mínimo error obtenido en la sintonización con el algoritmo MAGO fue del 2,81 %, siendo este control el más eficiente.

Tabla 9. Parámetros del MPC para el tanque 1 (caso 4).

Tiempo de muestreo	Horizonte de predicción	Horizonte de control	Variable manipulada	Pesos			Restricciones			Error
				Incremento de la var. manipulada	Variable de salida	Variable de holgura	Variable manipulada			
Ts	Hp	Hc	u	Δu	y	ECR	Mín	Máx	%	
1	213	25	1	0,1	1	100	0	1	34,77	
1	240	24	1	0,1	1	100	0	1	31,13	
1	206	21	1	0,1	2	100	0	1	27,94	
1	241	22	1	0,1	2	100	0	1	25,35	

Tiempo de muestreo	Horizonte de predicción	Horizonte de control	Pesos				Restricciones			Error %
			Variable manipulada	Incremento de la var. manipulada	Variable de salida	Variable de holgura	Variable manipulada			
			u	Δu	y	ECR	Mín	Máx		
T_s	H_p	H_c								
1	196	31	1	0,1	3	100	0	1	21,57	
1	209	25	1	0,1	4	100	0	1	8,55	
1	208	27	1	0,1	5	100	0	1	5,17	
1	200	21	1	0,1	5,5	100	0	1	2,81	

6.3.4.2 Implementación del controlador MPC en Simulink

En la **Figura 76**, se muestra el diagrama del sistema de control en lazo cerrado del MPC implementado en Simulink, ubicando en éste la función de transferencia y el retardo de tiempo.

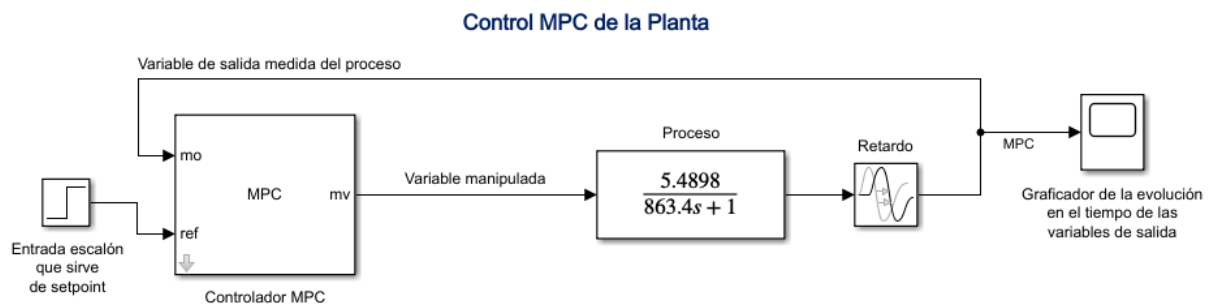


Figura 76. Diagrama de control del MPC en Simulink para el tanque 1 (caso 4).

En la **Figura 77**, se presenta la comparación y el resultado de tres controladores MPC sintonizados con el algoritmo diseñado, para el control de nivel en el tanque 1 (caso 4). La línea azul representa la dinámica del sistema sin control, a través de la función de transferencia obtenida mediante identificación de sistemas, la línea entrecortada de color rojo indica el setpoint y la línea de color verde, marrón y negro, corresponde al MPC sintonizado con un error del 2,81 %, 8,55 % y 21,57 %, respectivamente. En esta gráfica se puede observar que, el control MPC que presentó un resultado de sintonización con un error del 2,81 % sigue la trayectoria del modelo de la planta y alcanza el setpoint en un tiempo menor, estabilizándose a los 400 segundos, mientras que los demás controladores, que presentaron un error del 8,55 % y 21,57 % son menos eficientes a comparación del primero. Entre mayor es el error de sintonización mayor es el error en estado estacionario del sistema de control.

Por otro lado, en la **Figura 78** se puede observar cómo el controlador MPC que presentó el mínimo error en su sintonización, sigue la trayectoria del modelo de la planta sin sobrepasar los límites de operación máximos del actuador, alcanzando el estado estacionario para el nivel asignado en el menor tiempo posible, presentando un control eficiente.

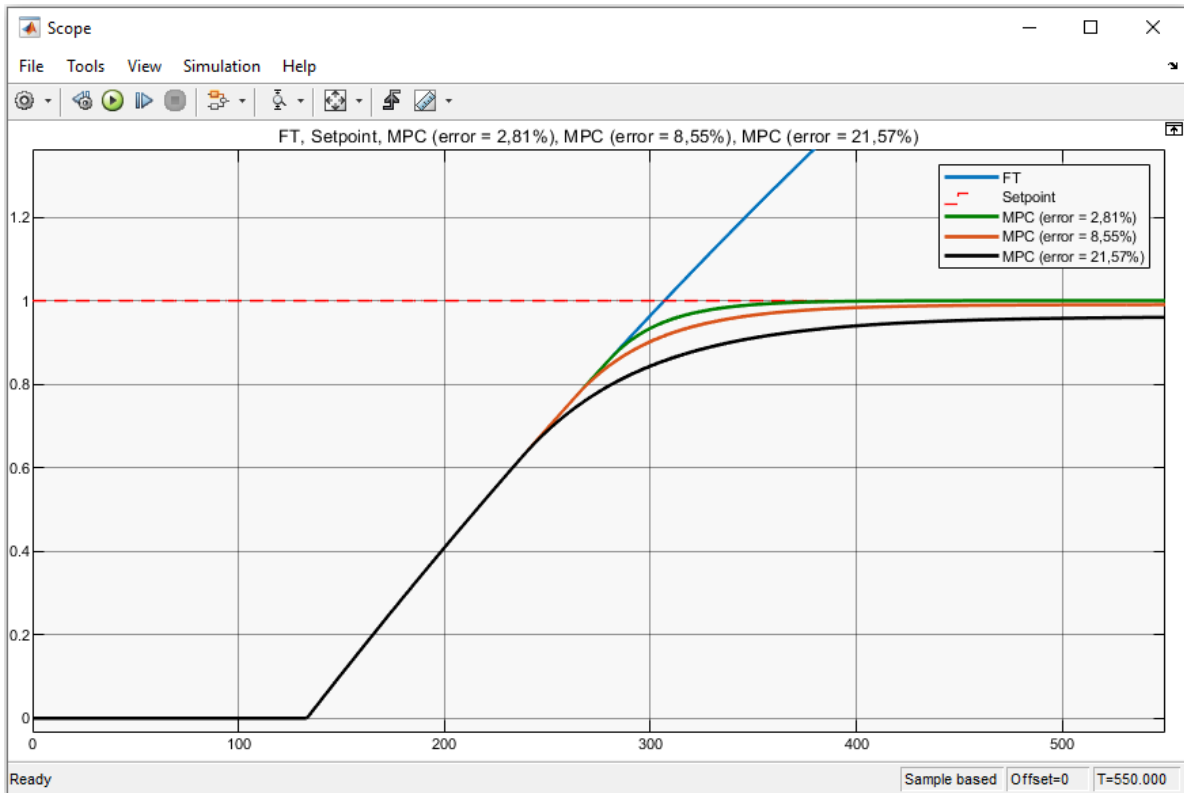


Figura 77. Gráfica comparativa de los resultados de sintonización obtenidos para el tanque 1 (caso 4) con el algoritmo diseñado.

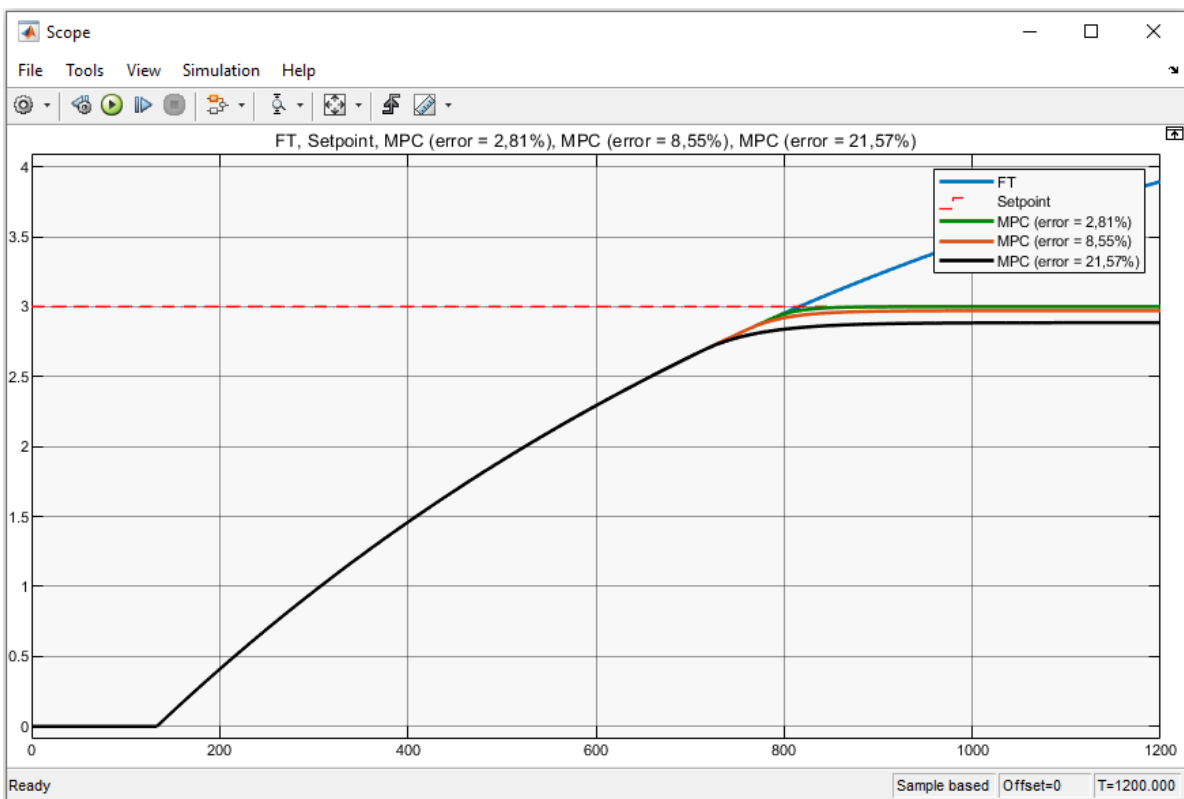


Figura 78. Comportamiento del controlador MPC para el tanque 1 (caso 4) con los diferentes errores de sintonización determinados por el algoritmo desarrollado.

6.4 Diseño del controlador clásico PID

En cada uno de los casos planteados se ejecutó varias sintonizaciones con el fin de determinar los mejores parámetros para un control PID óptimo. Es importante mencionar que el proceso de llenado de los tanques es lento, debido a sus características constructivas.

6.4.1 Controlador PID para el caso 1

6.4.1.1 Implementación del controlador PID en Simulink

En la **Figura 79**, se muestra el diagrama del sistema de control PID en lazo cerrado implementado en Simulink, ubicando en éste la función de transferencia obtenida en la identificación de sistemas.

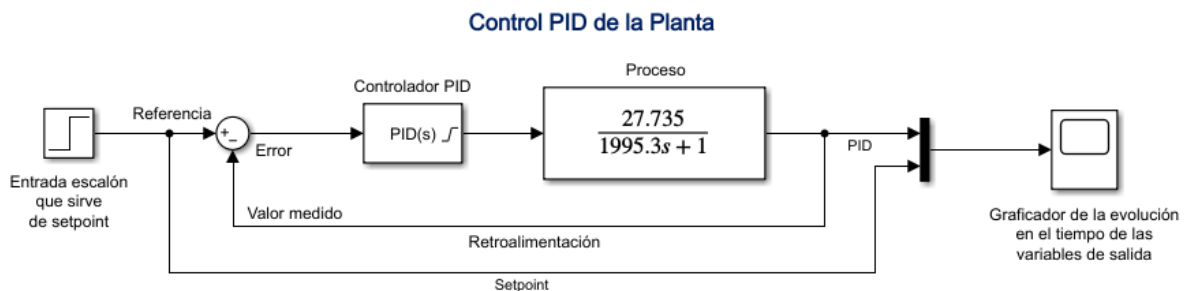


Figura 79. Diagrama de control PID en Simulink para el tanque 1 (caso 1).

6.4.1.2 Parámetros obtenidos

En la **Figura 80**, se presentan dos curvas de control obtenidas con la aplicación PID Tuner de Simulink, la línea continua muestra la respuesta de la sintonización automática (Tuned response) y la línea entrecortada la respuesta del controlador diseñado (Block response), además se presenta un cuadro con los parámetros del PID y los índices de rendimiento.

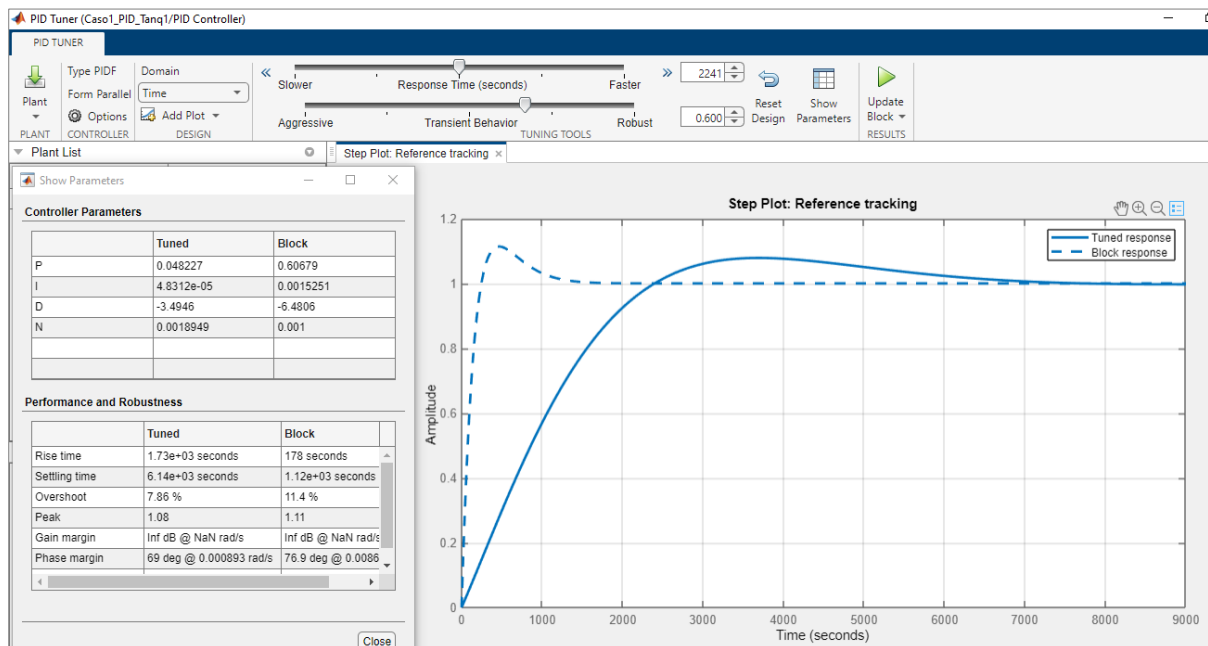


Figura 80. Sintonización del controlador PID para el tanque 1 (caso 1).

Los parámetros obtenidos para el control PID del tanque 1 se muestran en la **Tabla 10**.

Tabla 10. Parámetros del controlador PID para el tanque 1 (caso 1).

Constante	Tanque 1
Proporcional (P)	0,606793197743396
Integral (I)	0,00152506829616581
Derivativa (D)	-6,48063389276603

En la **Figura 81**, se puede observar la respuesta del sistema con el controlador PID diseñado para un setpoint de 1, alcanzado esta referencia a los 246 segundos y estabilizándose alrededor de los 1 120 segundos.

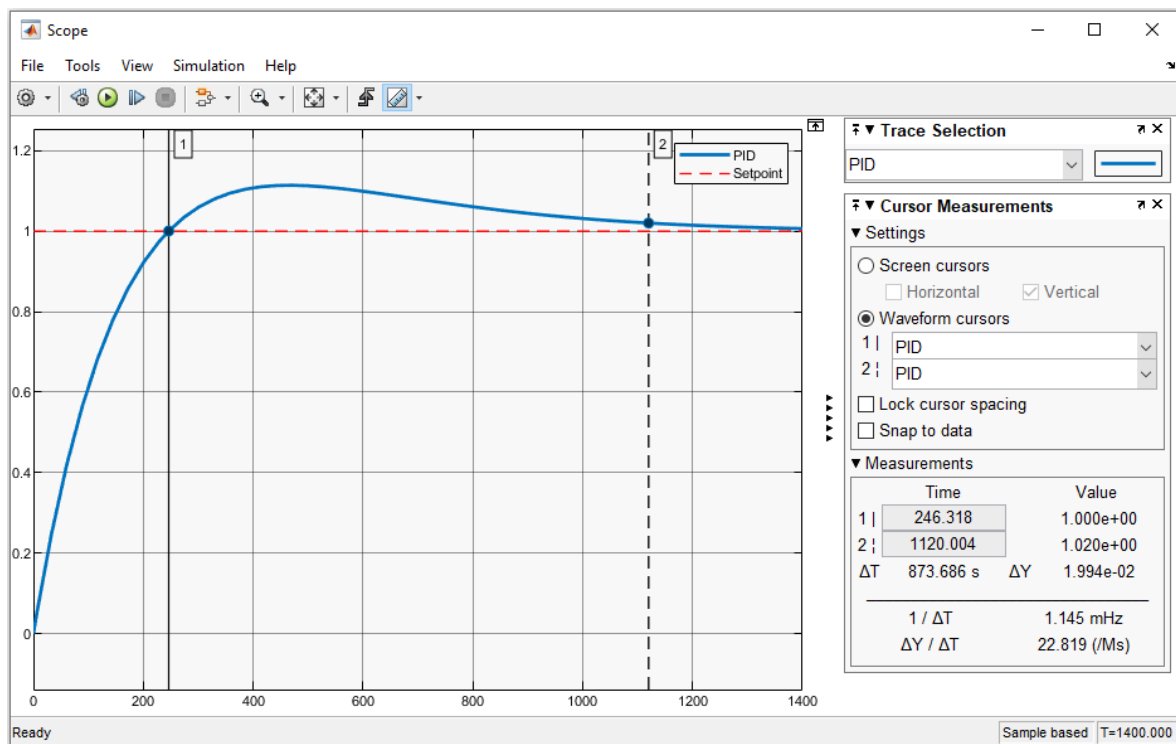


Figura 81. Respuesta del controlador PID diseñado para el tanque 1 (caso 1).

Como ya se había mencionado en el apartado de metodología del presente proyecto, se consideró los límites de saturación del actuador en el diseño del controlador PID para todos los casos planteados, empleando una estrategia conocida como anti-windup, ya que de no ser así en el proceso real podemos llegar a tener un comportamiento no deseado de la variable controlada.

En la **Figura 82**, se muestra el control PID para el nivel del tanque 1 (caso 1), sin considerar el límite de saturación del actuador (sin método anti-windup). La línea de color rojo representa la dinámica de llenado del tanque 1 sin control, operando la bomba a su máxima

capacidad, la línea de color azul corresponde a la respuesta del controlador PID diseñado para este sistema y la línea entrecortada de color verde indica el setpoint. En esta gráfica se puede observar que el PID funciona perfectamente para un setpoint de 1, sin sobrepasar el límite de operación máximo del actuador. Sin embargo, para un setpoint o nivel diferente, en este caso de 14 litros, el controlador PID presenta un comportamiento irreal en simulación, ya que el nivel de llenado del tanque para 1 litro y 14 litros se alcanza en el mismo tiempo, lo cual no es posible.

Por otro lado, en la **Figura 83** se muestra el mismo control PID, pero considerando el límite de saturación del actuador, teniendo como resultado un control eficiente. En esta gráfica se puede observar el comportamiento del controlador PID para un setpoint de 3 y 14 litros, alcanzando el nivel asignado sin sobrepasar el límite de operación máximo del actuador.

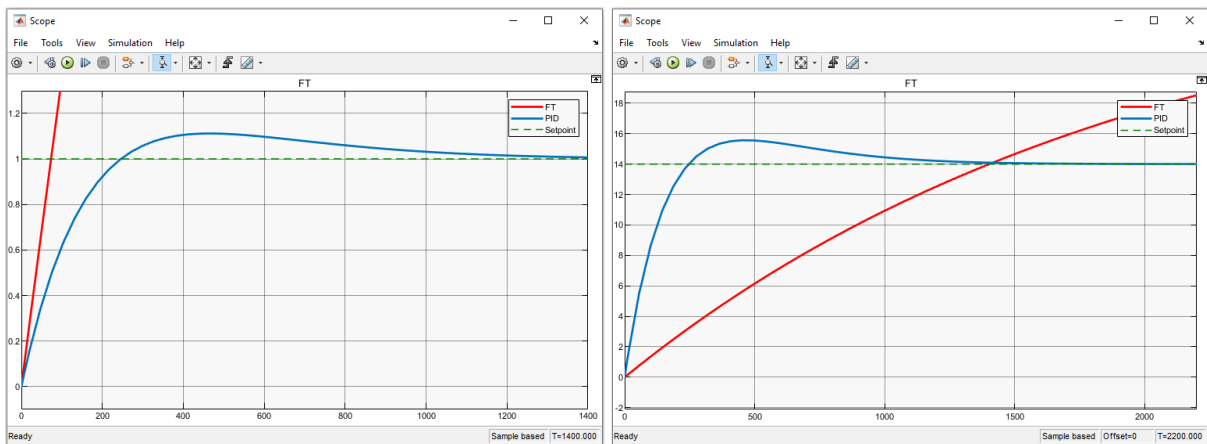


Figura 82. Control PID del tanque 1 sin implementar el método anti-windup.

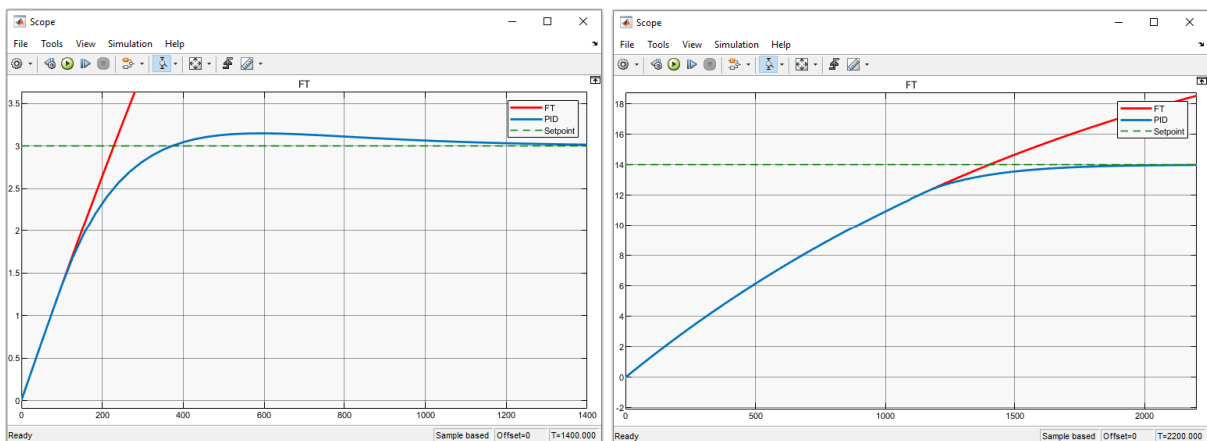


Figura 83. Control PID del tanque 1 implementando el método anti-windup.

6.4.2 Controlador PID para el caso 2

6.4.2.1 Implementación del controlador PID en Simulink

En la **Figura 84**, se muestra el diagrama del sistema de control PID en lazo cerrado implementado en Simulink, ubicando en éste la función de transferencia y el retardo de tiempo.

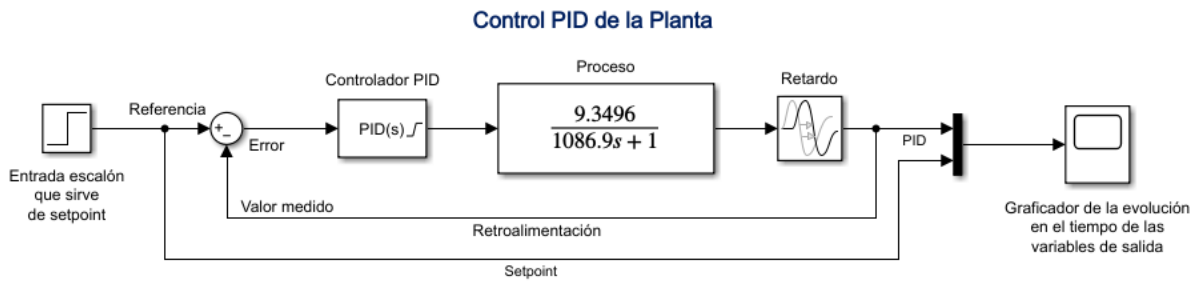


Figura 84. Diagrama de control PID en Simulink para el tanque 2 (caso 2).

6.4.2.2 Parámetros obtenidos

En la **Figura 85**, se presentan dos curvas de control obtenidas con la aplicación PID Tuner de Simulink, la línea continua muestra la respuesta de la sintonización automática (Tuned response) y la línea entrecortada la respuesta del controlador diseñado (Block response), además se presenta un cuadro con los parámetros del PID y los índices de rendimiento.

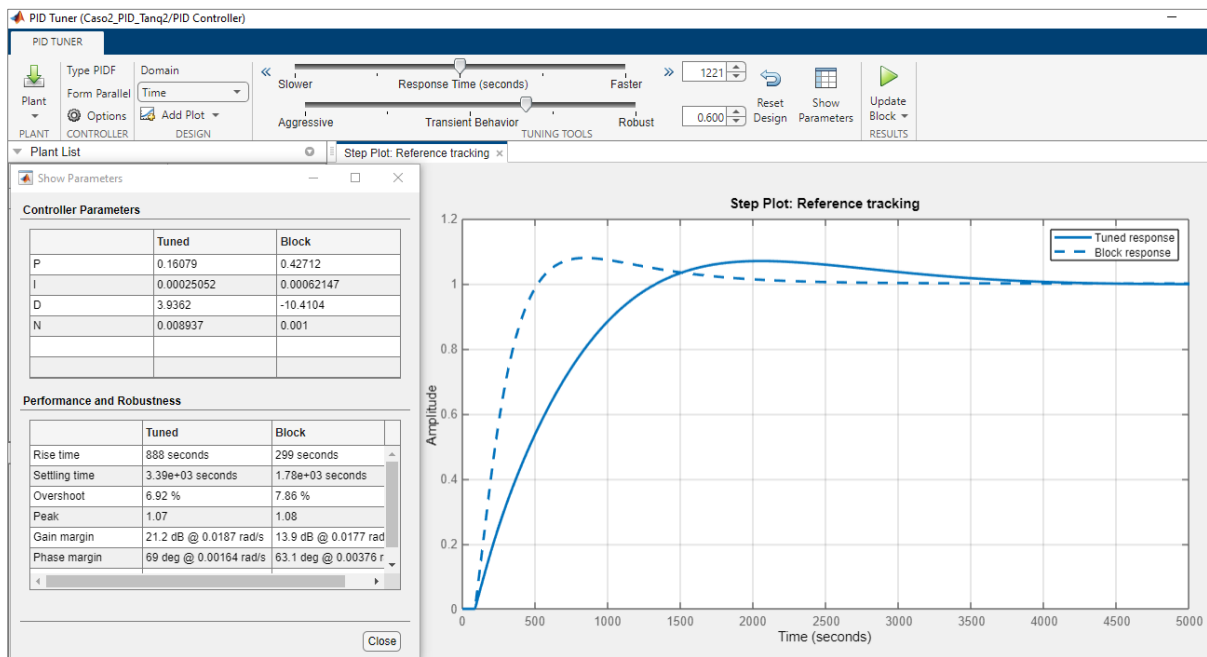


Figura 85. Sintonización del controlador PID para el tanque 2 (caso 2).

Los parámetros obtenidos para el control PID del tanque 2 se muestran en la **Tabla 11**.

Tabla 11. Parámetros del controlador PID para el tanque 2 (caso 2).

Constante	Tanque 2
Proporcional (P)	0,427124523488005
Integral (I)	0,000621471834437484
Derivativa (D)	-10,410403716964

En la **Figura 86**, se puede observar la respuesta del sistema con el controlador PID diseñado para un setpoint de 1, alcanzado esta referencia a los 520 segundos y estabilizándose alrededor de los 1 780 segundos.

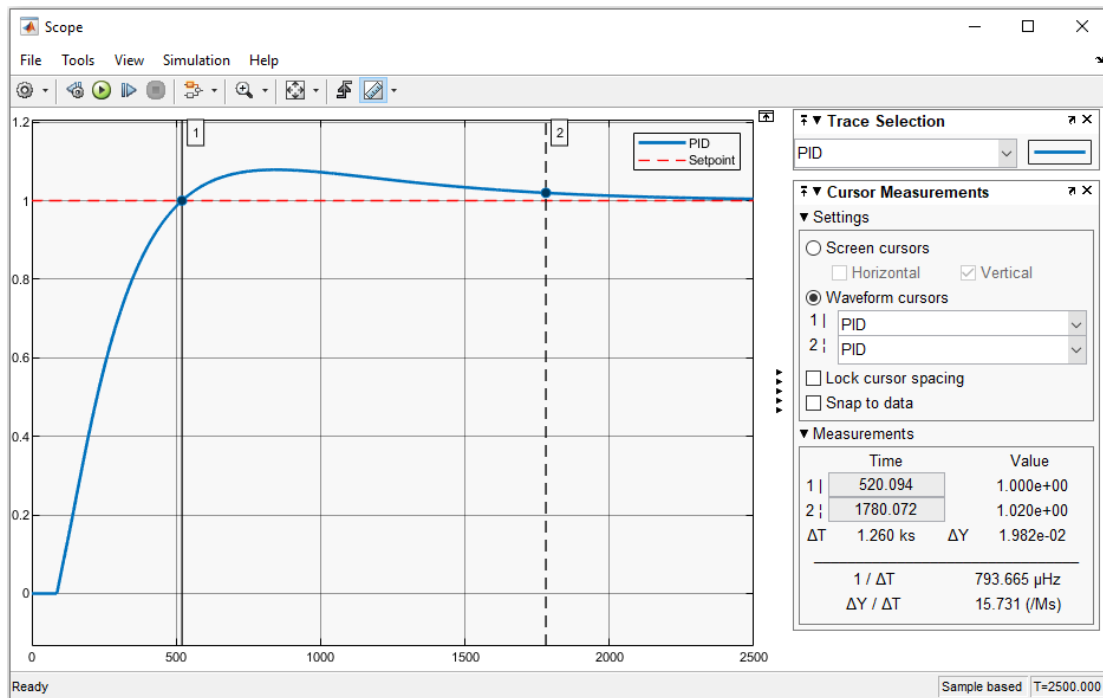


Figura 86. Respuesta del controlador PID diseñado para el tanque 2 (caso 2).

6.4.3 Controlador PID para el caso 3

6.4.3.1 Implementación del controlador PID en Simulink

En la **Figura 87**, se muestra el diagrama del sistema de control PID en lazo cerrado implementado en Simulink, ubicando en éste la función de transferencia y el retardo de tiempo.

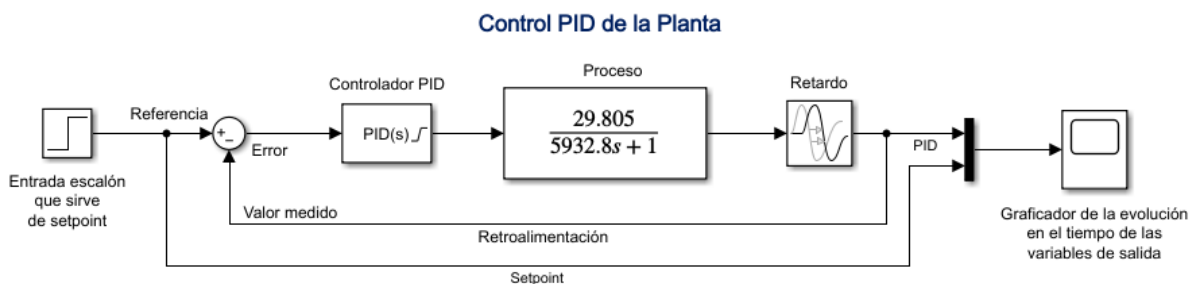


Figura 87. Diagrama de control PID en Simulink para el tanque 2 (caso 3).

6.4.3.2 Parámetros obtenidos

En la **Figura 88**, se presentan dos curvas de control obtenidas con la aplicación PID Tuner de Simulink, la línea continua muestra la respuesta de la sintonización automática (Tuned response) y la línea entrecortada la respuesta del controlador diseñado (Block response), además se presenta un cuadro con los parámetros del PID y los índices de rendimiento.

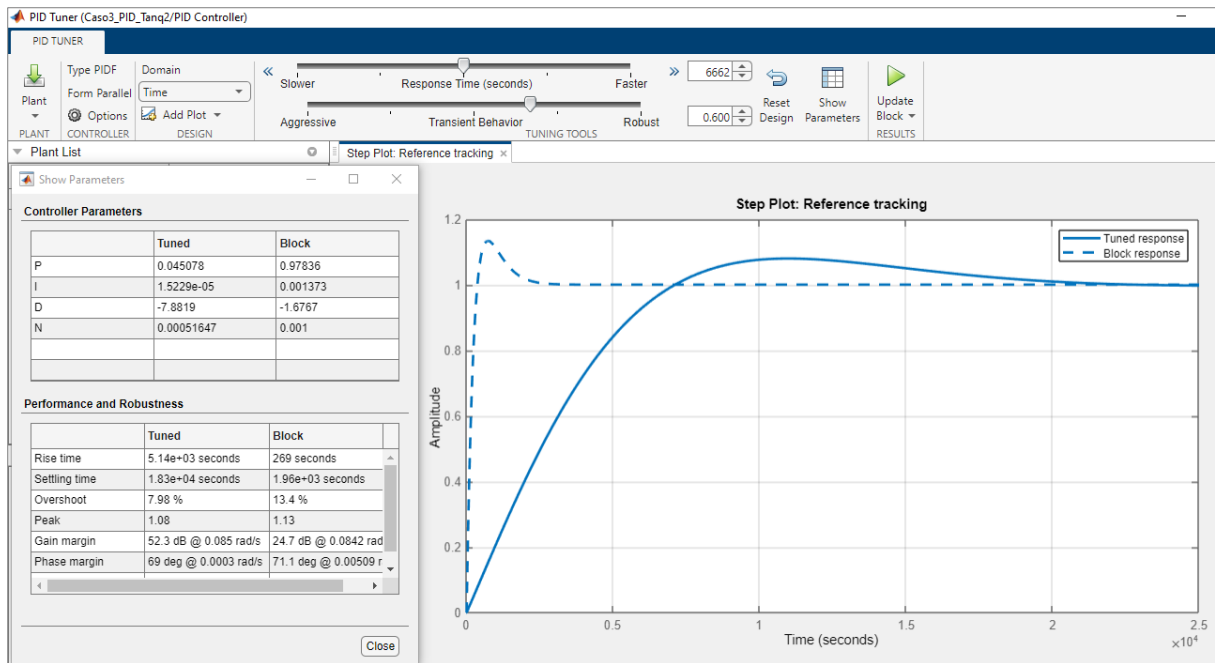


Figura 88. Sintonización del controlador PID para el tanque 2 (caso 3).

Los parámetros obtenidos para el control PID del tanque 2 se muestran en la **Tabla 12**.

Tabla 12. Parámetros del controlador PID para el tanque 2 (caso 3).

Constante	Tanque 2
Proporcional (P)	0,978357533056197
Integral (I)	0,00137304956237647
Derivativa (D)	-1,67670476841092

En la **Figura 89**, se puede observar la respuesta del sistema con el controlador PID diseñado para un setpoint de 1, alcanzado esta referencia a los 392 segundos y estabilizándose alrededor de los 1 960 segundos.

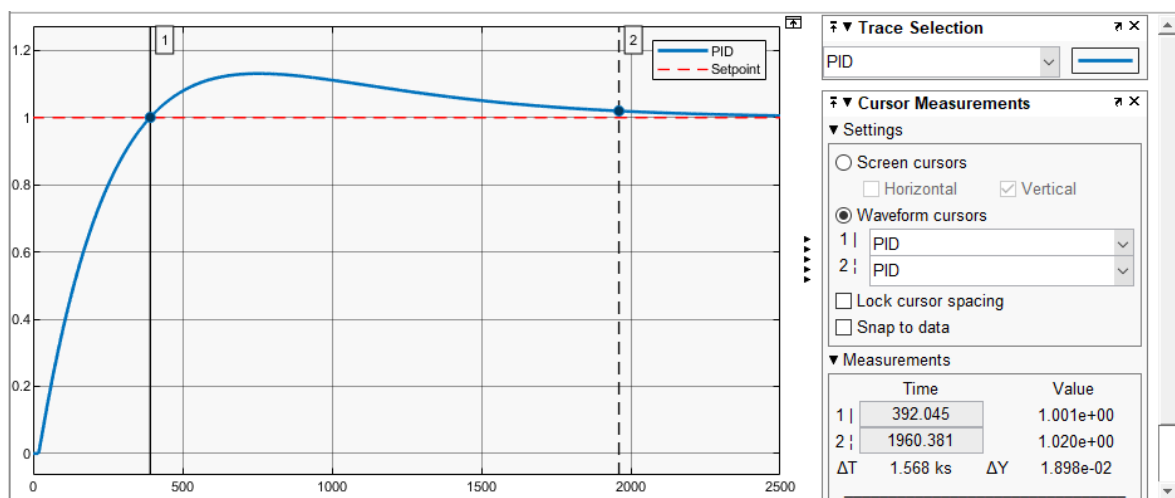


Figura 89. Respuesta del controlador PID diseñado para el tanque 2 (caso 3).

6.4.4 Controlador PID para el caso 4

6.4.4.1 Implementación del controlador PID en Simulink

En la **Figura 90**, se muestra el diagrama del sistema de control PID en lazo cerrado implementado en Simulink, ubicando en éste la función de transferencia y el retardo de tiempo.

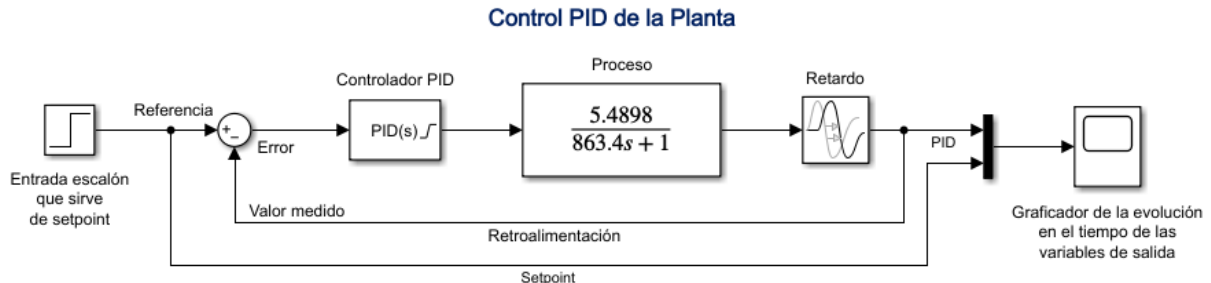


Figura 90. Diagrama de control PID en Simulink para el tanque 1 (caso 4).

6.4.4.2 Parámetros obtenidos

En la **Figura 91**, se presentan dos curvas de control obtenidas con la aplicación PID Tuner de Simulink, la línea continua muestra la respuesta de la sintonización automática (Tuned response) y la línea entrecortada la respuesta del controlador diseñado (Block response), además se presenta un cuadro con los parámetros del PID y los índices de rendimiento.

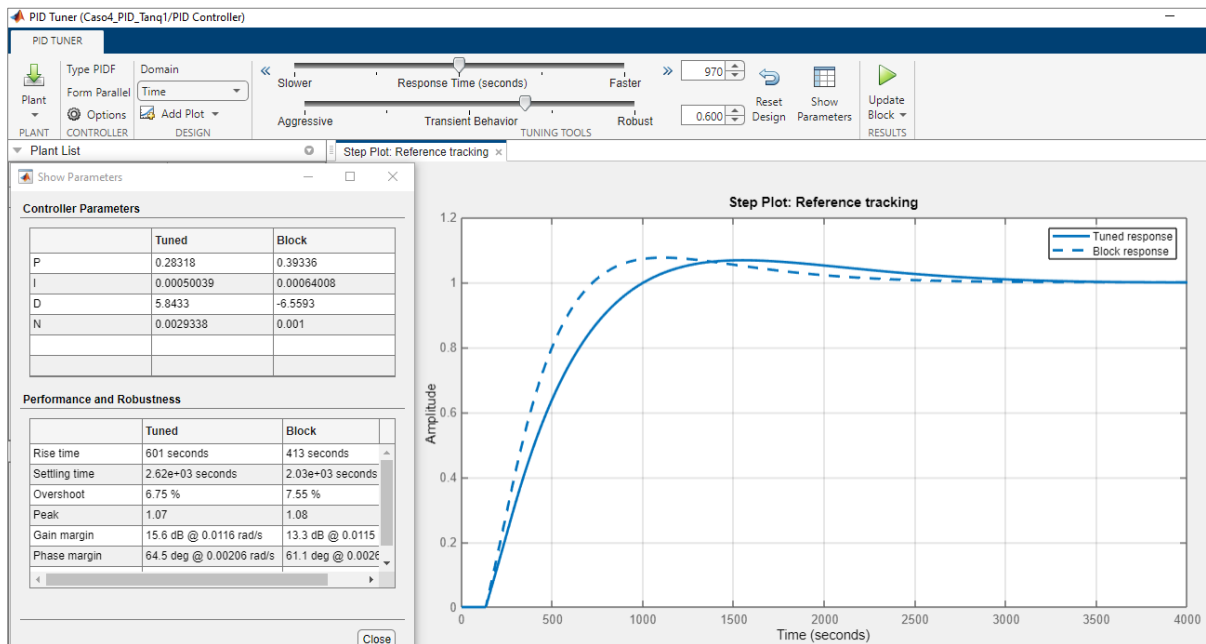


Figura 91. Sintonización del controlador PID para el tanque 1 (caso 4).

En este caso, al igual que en el caso 2, en los que se tiene grandes retardos de tiempo, los controladores PID diseñados son las más ideales, ya que sí se intenta hacer que el control sea más agresivo para que se minimice el tiempo en que el sistema alcanza el estado estable, teniendo una respuesta más rápida, el controlador se vuelve inestable, tal como se puede observar en la **Figura 92**.

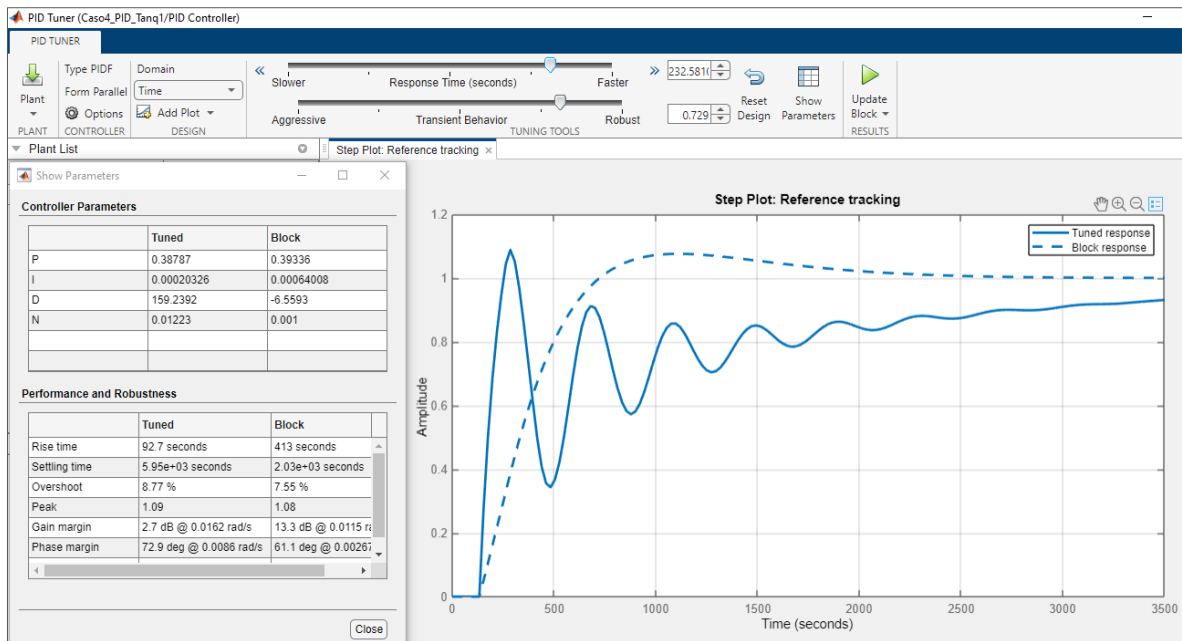


Figura 92. Análisis de la respuesta del controlador PID para el tanque 1 (caso 4), minimizando el tiempo en que el sistema alcanza el estado estable.

Los parámetros obtenidos para el control PID del tanque 1 se muestran en la **Tabla 13**.

Tabla 13. Parámetros del controlador PID para el tanque 1 (caso 4).

Constante	Tanque 1
Proporcional (P)	0,393364198559904
Integral (I)	0,000640083306836919
Derivativa (D)	-6,5592841093836

En la **Figura 93**, se puede observar la respuesta del sistema con el controlador PID diseñado para un setpoint de 1, alcanzado esta referencia a los 725 segundos y estabilizándose alrededor de los 2 030 segundos.

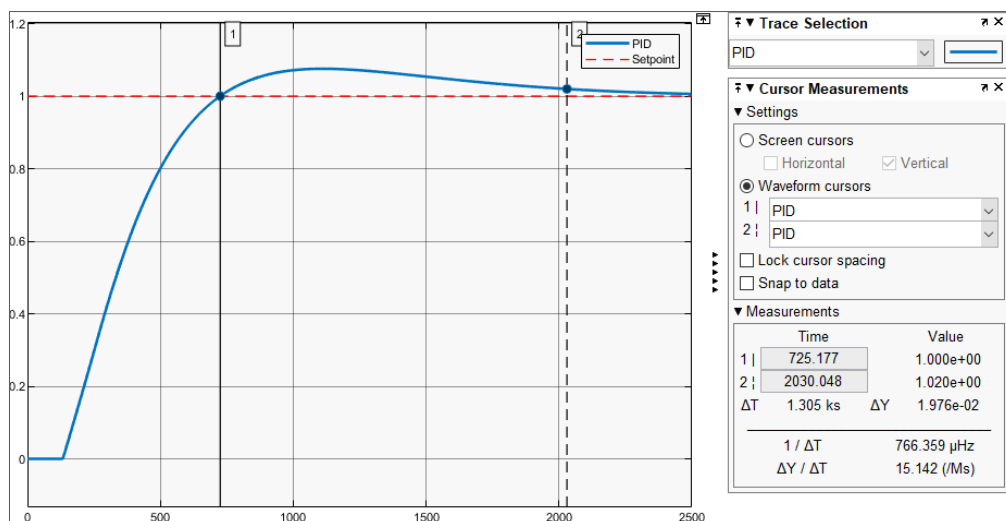


Figura 93. Respuesta del controlador PID diseñado para el tanque 1 (caso 4).

6.5 Evaluación de desempeño y comparación del controlador avanzado MPC con el controlador clásico PID

Los criterios de evaluación para comparar el desempeño de los controladores PID y MPC diseñados, son los siguientes:

- **Tiempo de subida o de levantamiento:** tiempo necesario para que la respuesta del sistema pase del 10 % al 90 % de su valor final (setpoint).
- **Tiempo de estabilización:** tiempo que se requiere para que la respuesta del sistema se estabilice alrededor de un rango máximo de $\pm 2\%$ del valor final y permanezca dentro de él.
- **Sobrepaso o sobreimpulso:** valor pico máximo que alcanza la respuesta del sistema medido a partir de su valor final.

También se evaluó la respuesta al cambio en la referencia y la capacidad de rechazo a las perturbaciones.

6.5.1 Comparación del controlador PID con el MPC para el caso 1

En la **Figura 94** y **95**, se muestra la respuesta de los controladores diseñados para el tanque 1 ante un setpoint de 5 y 12 litros, respectivamente.

Cabe señalar que aunque el controlador MPC parece presentar una respuesta agresiva ante un nivel de 12 litros, este comportamiento se da porque la dinámica del sistema es bastante lenta, lo cual se puede verificar observando la **Figura 96**, en la cual se hizo un acercamiento para analizar mejor esta respuesta, pudiendo observar que el sistema pasa de los 11,8 a 12 litros en un lapso de tiempo de 86 segundos, tiempo suficiente para que el actuador realice las respectivas acciones de control, siendo lo más óptimo posible.

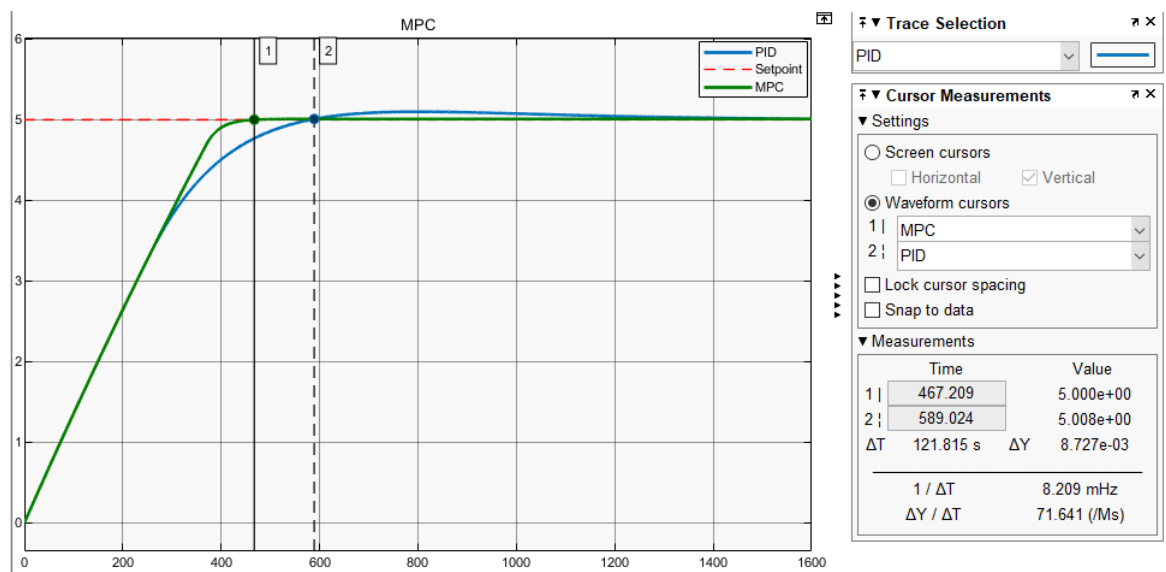


Figura 94. Comparación de los controladores diseñados para el tanque 1 (caso 1) a una referencia de 5 litros.

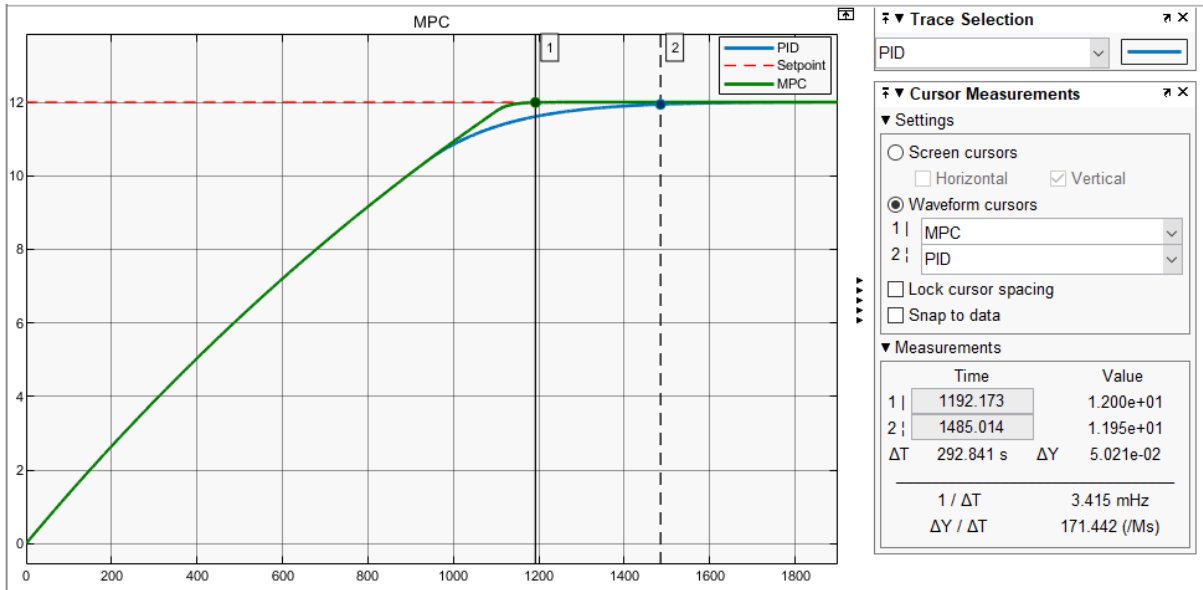


Figura 95. Comparación de los controladores diseñados para el tanque 1 (caso 1) a una referencia de 12 litros.

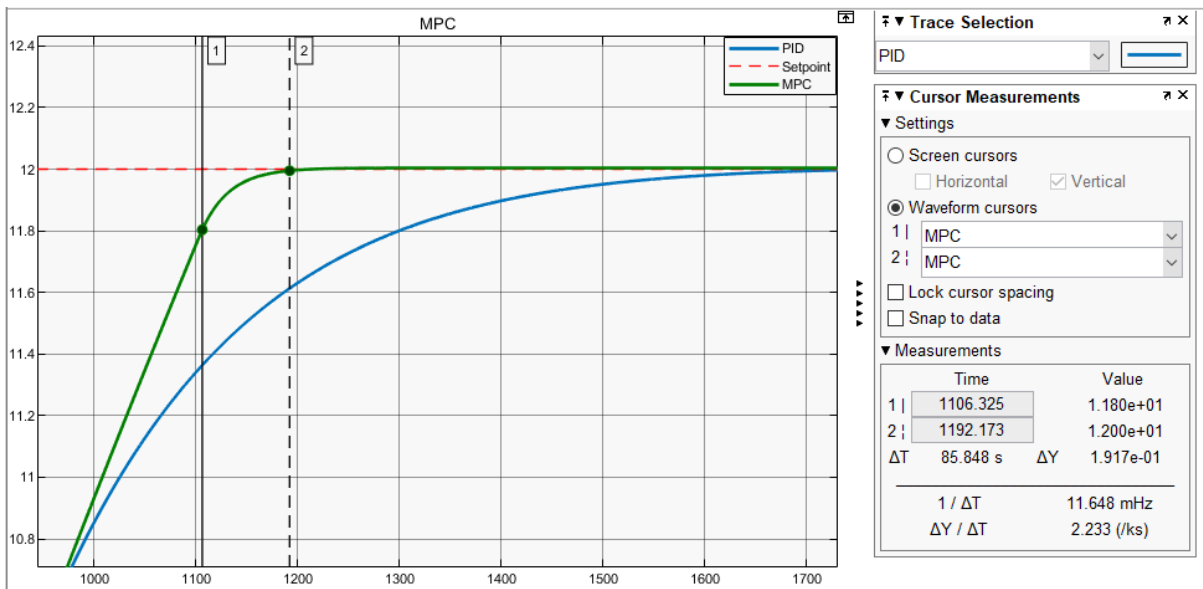


Figura 96. Análisis de la respuesta del controlador MPC para un nivel de referencia de 12 litros.

En la **Tabla 14**, se muestra los criterios de evaluación analizados para comparar el desempeño de los controladores diseñados, en la cual se puede observar que el tiempo de estabilización del PID para un setpoint de 5 y 12 litros, es respectivamente de 122 segundos (2,03 min) y 293 segundos (4,88 min) mayor en comparación al tiempo de estabilización del MPC; por lo tanto, es evidente que el controlador avanzado MPC es mucho más eficiente que el PID.

Tabla 14. Características de desempeño del controlador MPC y PID para el tanque 1 (caso 1).

Setpoint (litros)	Controlador	Tiempo de Subida (s)	Tiempo de Estabilización (s)	Sobrepaso máximo (%)
5	MPC	318	467	0
5	PID	363	589	2
12	MPC	896	1 192	0
12	PID	904	1 485	0

En la **Figura 97**, se muestra la respuesta de los controladores diseñados ante perturbaciones, donde se observa que el MPC presenta una mayor eficiencia. Ante una señal de perturbación de -0,5 que representa un aumento en el flujo de descarga por sobrepresión en la electroválvula, el MPC regresa a su setpoint en un tiempo de 192 segundos (3,2 min), mientras que el PID rechaza esta misma perturbación en un lapso de 242 segundos (4,03 min).

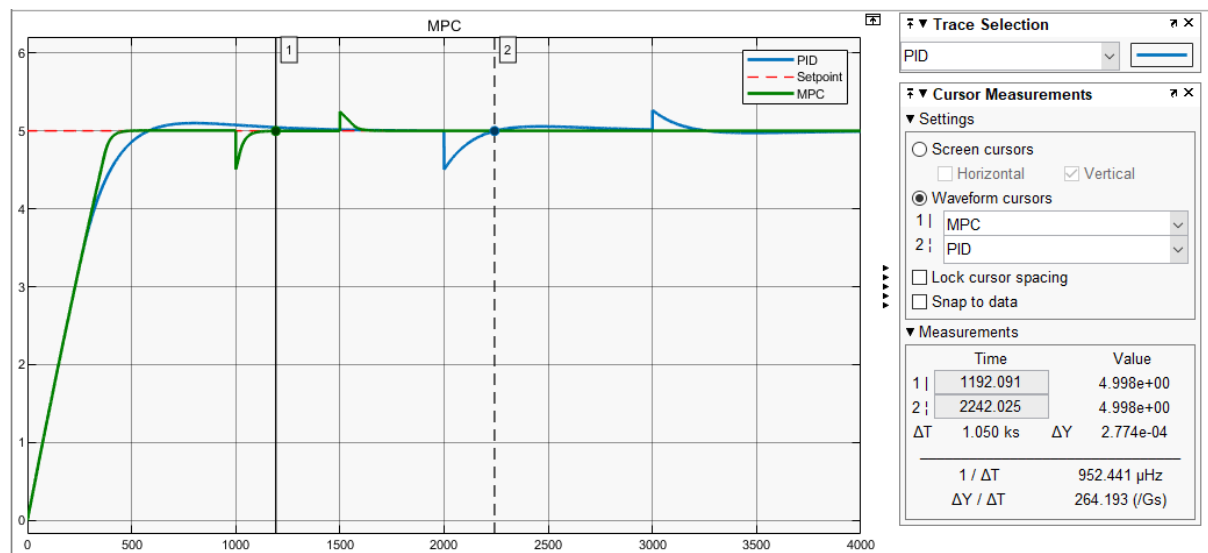


Figura 97. Análisis de rechazo de perturbaciones en el control de nivel del tanque 1 (caso 1).

6.5.2 Comparación del controlador PID con el MPC para el caso 2

En la **Figura 98** y **99**, se muestra la respuesta de los controladores diseñados para el tanque 2 ante un setpoint de 4 y 8 litros, respectivamente. Este sistema posee un retardo de tiempo en su salida de 86,84 segundos.

Los criterios de evaluación analizados para comparar el desempeño de los controladores diseñados se muestran en la **Tabla 15**, en la cual se puede observar que el tiempo de estabilización del PID para un setpoint de 4 y 8 litros, es respectivamente de 815 segundos (13,58 min) y 948 segundos (15,8 min) mayor en comparación al tiempo de estabilización del MPC; por lo tanto, es evidente que el controlador MPC es mucho más eficiente que el PID.

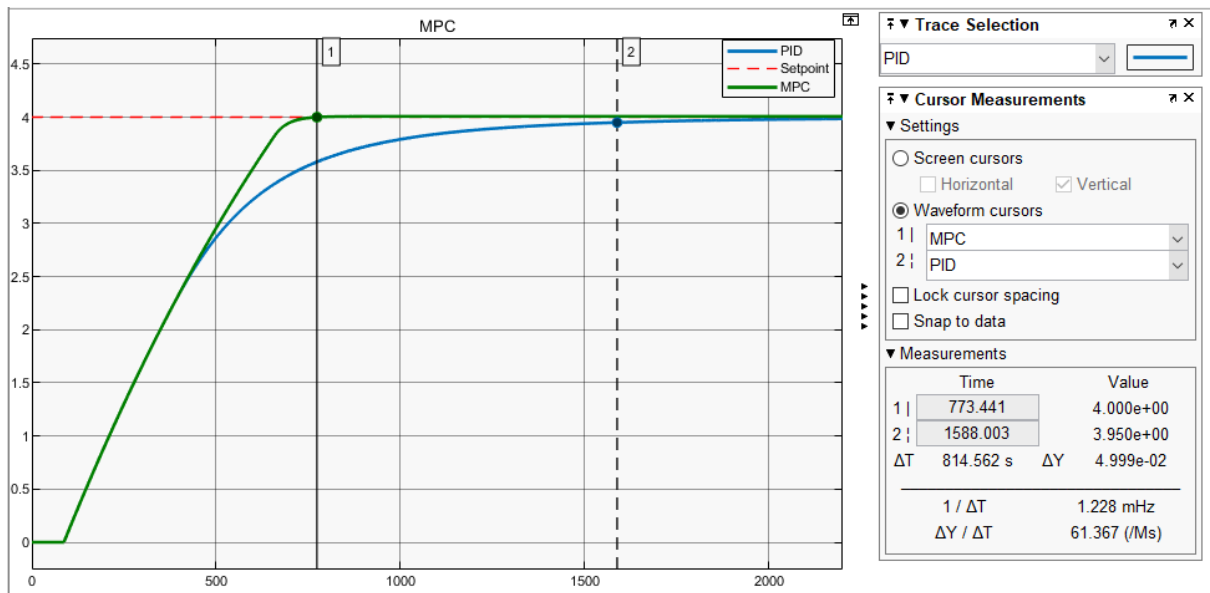


Figura 98. Comparación de los controladores diseñados para el tanque 2 (caso 2) a una referencia de 4 litros.

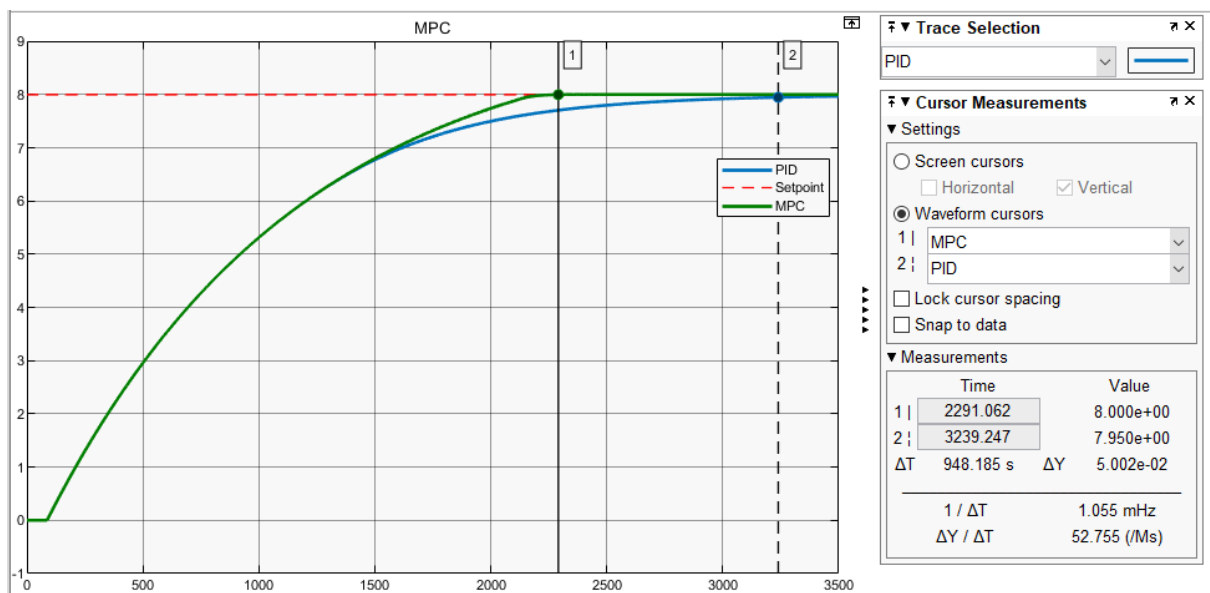


Figura 99. Comparación de los controladores diseñados para el tanque 2 (caso 2) a una referencia de 8 litros.

Tabla 15. Características de desempeño del controlador MPC y PID para el tanque 2 (caso 2).

Setpoint (litros)	Controlador	Tiempo de Subida (s)	Tiempo de Estabilización (s)	Sobrepaso máximo (%)
4	MPC	481	773	0,05
4	PID	655	1 588	0,12
8	MPC	1 501	2 291	0,05
8	PID	1 558	3 239	0

En la **Figura 100**, se muestra la respuesta de los controladores ante las perturbaciones, en la cual se puede observar que el MPC presenta una mayor capacidad de rechazo de estas señales indeseadas que afectan la salida del sistema. Ante una señal de perturbación de -0.5, el MPC regresa a su setpoint en un tiempo de 336 segundos (5,6 min), mientras que el PID rechaza esta misma perturbación en un lapso de 520 segundos (8,67 min).

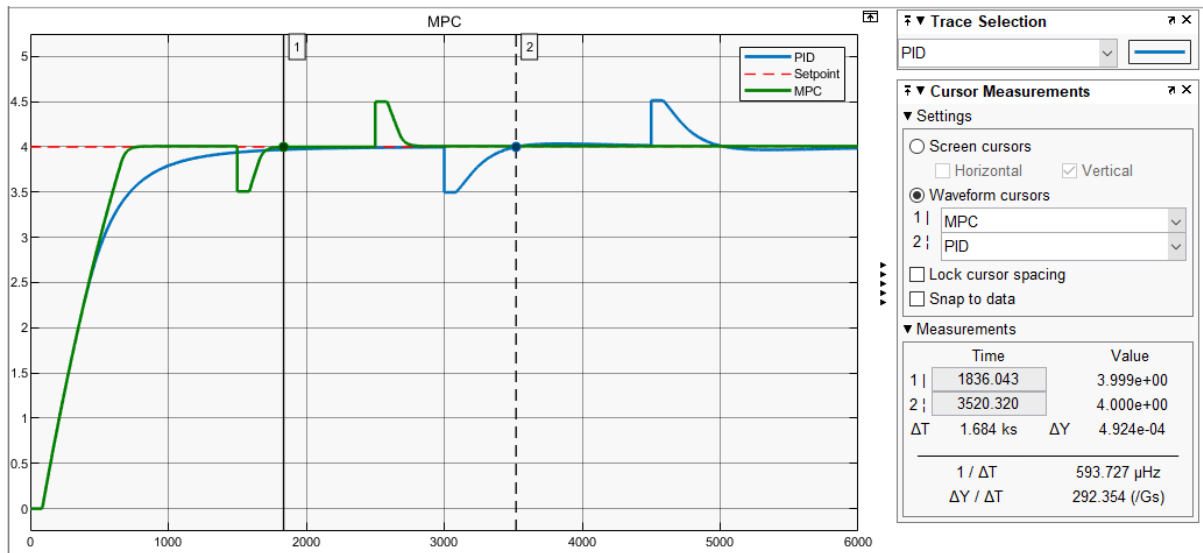


Figura 100. Análisis de rechazo de perturbaciones en el control de nivel del tanque 2 (caso 2).

6.5.3 Comparación del controlador PID con el MPC para el caso 3

En la **Figura 101** y **102**, se muestra la respuesta de los controladores diseñados para el tanque 2 ante un setpoint de 5 y 12 litros, respectivamente. Este sistema posee un retardo de tiempo en su salida de 18,47 segundos.

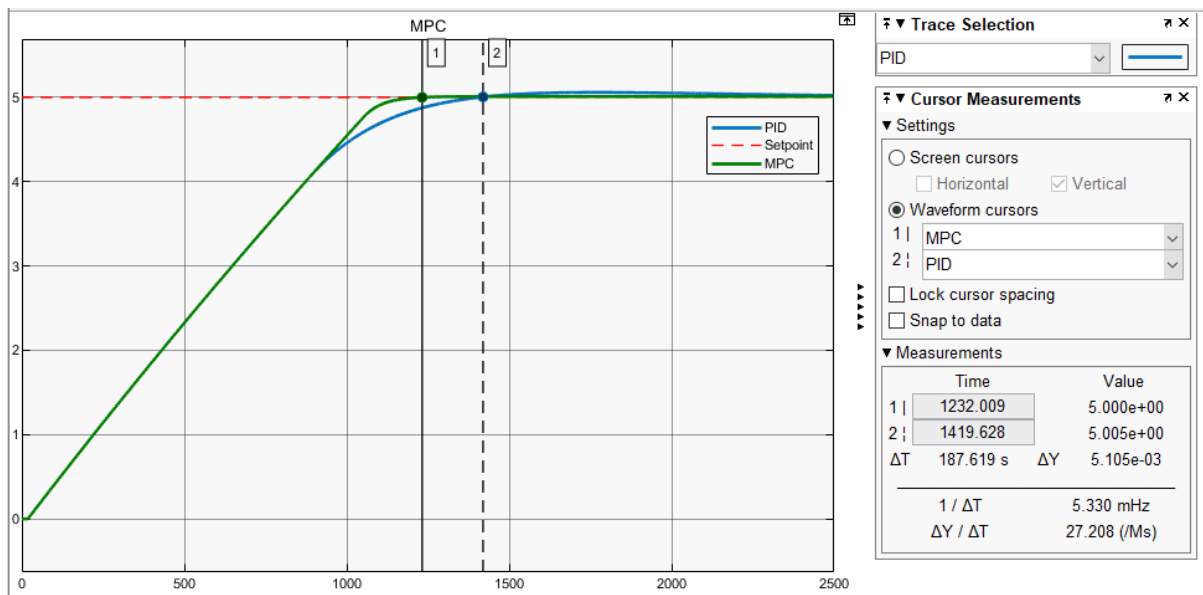


Figura 101. Comparación de los controladores diseñados para el tanque 2 (caso 3) a una referencia de 5 litros.

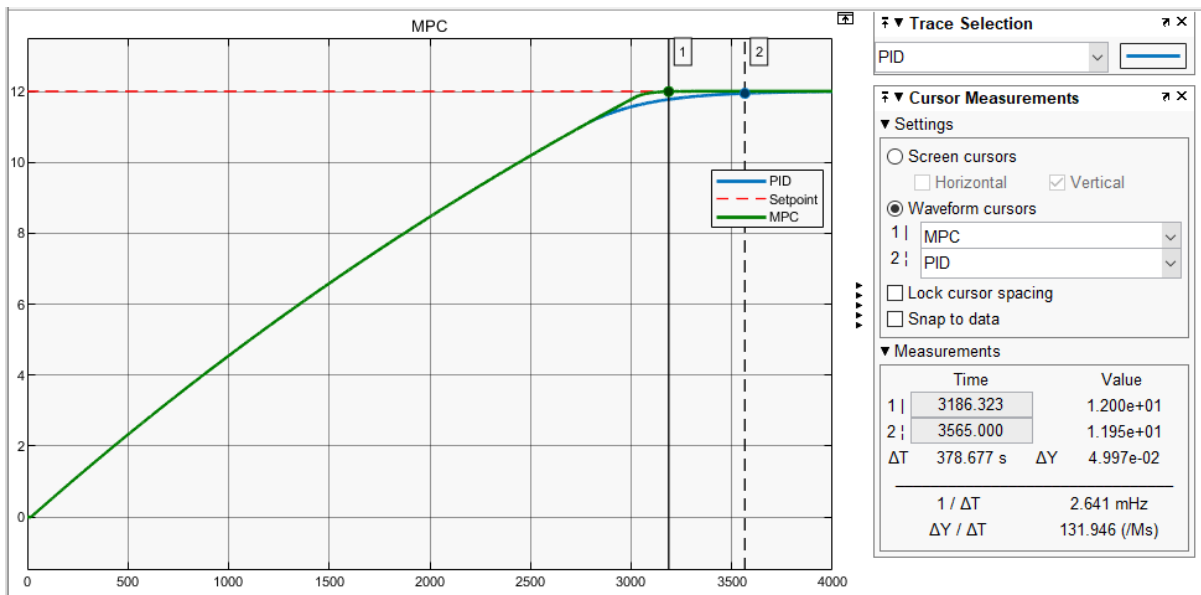


Figura 102. Comparación de los controladores diseñados para el tanque 2 (caso 3) a una referencia de 12 litros.

Los criterios de evaluación analizados para comparar el desempeño de los controladores diseñados se muestran en la **Tabla 16**, en la cual se puede observar que el tiempo de estabilización del PID para un setpoint de 5 y 12 litros, es respectivamente de 187 segundos (3,12 min) y 379 segundos (6,32 min) mayor en comparación al tiempo de estabilización del MPC; por lo tanto, es evidente que el controlador avanzado MPC es mucho más eficiente que el PID.

Tabla 16. Características de desempeño del controlador MPC y PID para el tanque 2 (caso 3).

Setpoint (litros)	Controlador	Tiempo de Subida (s)	Tiempo de Estabilización (s)	Sobrepaso máximo (%)
5	MPC	871	1 232	0
5	PID	897	1 419	1,2
12	MPC	2 425	3 186	0
12	PID	2 425	3 565	0

En la **Figura 103**, se muestra la respuesta de los controladores ante las perturbaciones, en la cual se puede observar que el MPC presenta una mayor capacidad de rechazo de estas señales indeseadas que afectan la salida del sistema. Ante una señal de perturbación de -0,5 el MPC regresa a su setpoint en un tiempo de 285 segundos (4,75 min), mientras que el PID rechaza esta misma perturbación en un lapso de 380 segundos (6,33 min).

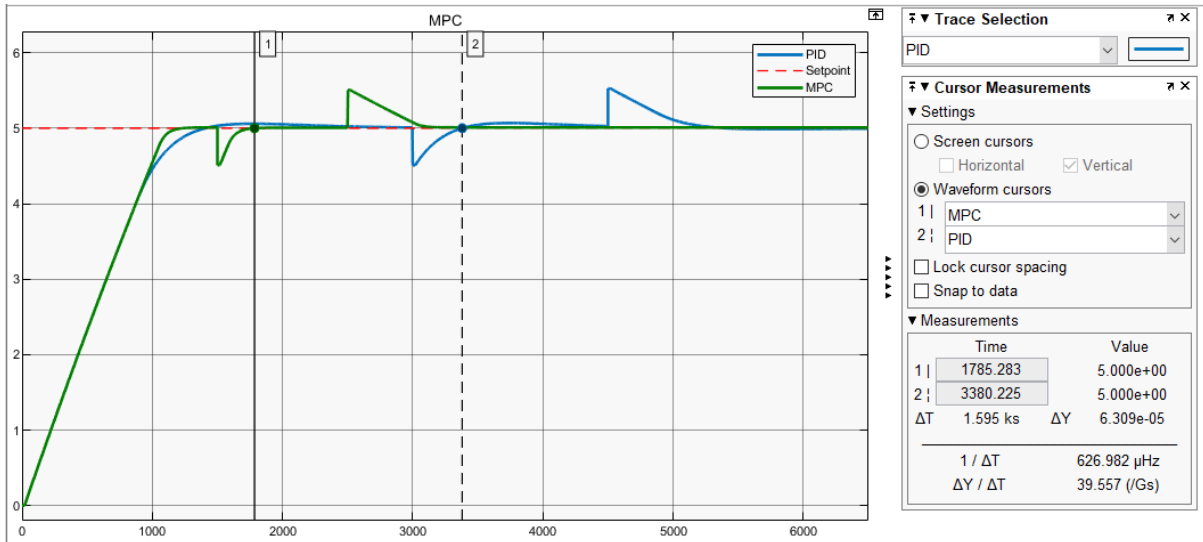


Figura 103. Análisis de rechazo de perturbaciones en el control de nivel del tanque 2 (caso 3).

6.5.4 Comparación del controlador PID con el MPC para el caso 4

En la **Figura 104**, 105 y 106, se muestra la respuesta de los controladores diseñados para el tanque 1 ante un setpoint de 2, 3,5 y 5 litros, respectivamente. Este sistema posee un retardo de tiempo en su salida de 133,2 segundos.

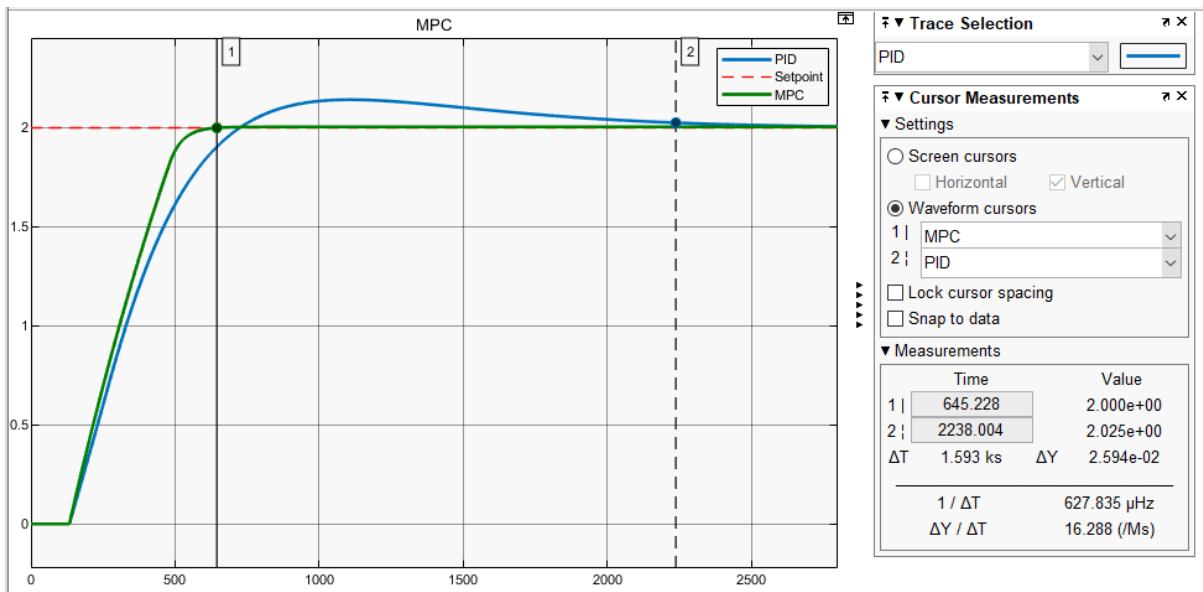


Figura 104. Comparación de los controladores diseñados para el tanque 1 (caso 4) a una referencia de 2 litros.

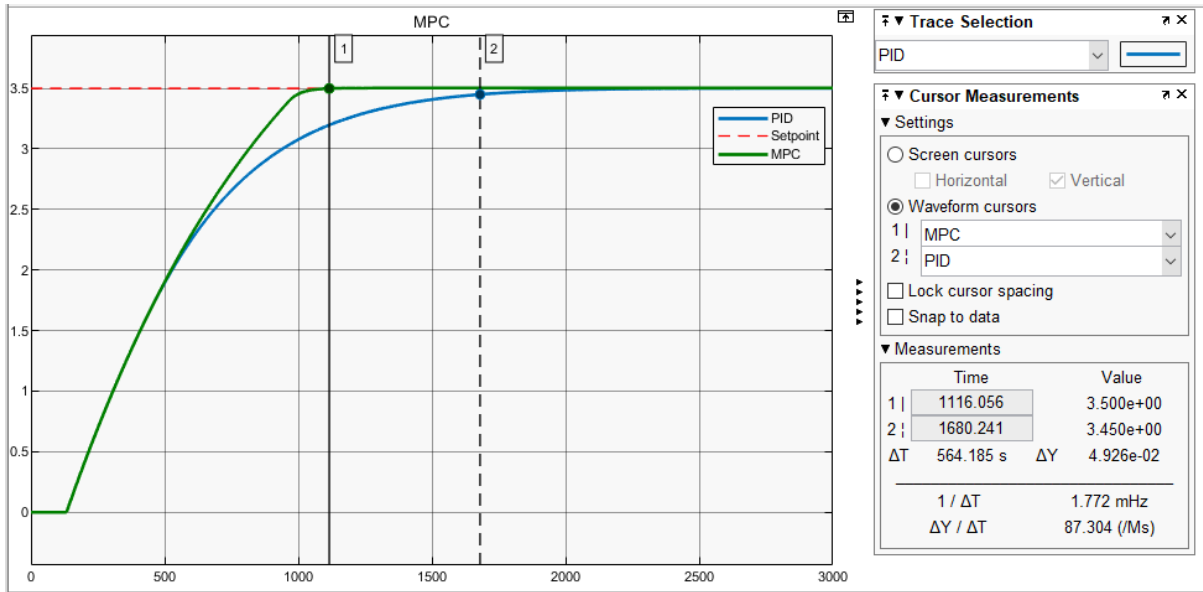


Figura 105. Comparación de los controladores diseñados para el tanque 1 (caso 4) a una referencia de 3,5 litros.

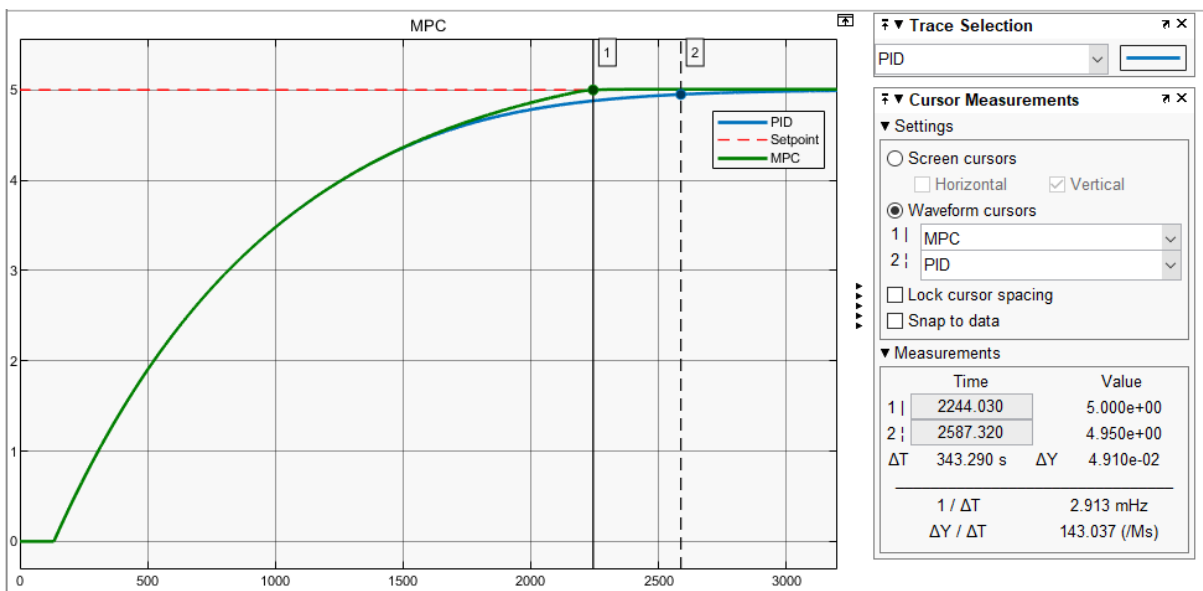


Figura 106. Comparación de los controladores diseñados para el tanque 1 (caso 4) a una referencia de 5 litros.

Los criterios de evaluación analizados para comparar el desempeño de los controladores diseñados se muestran en la **Tabla 17**, en la cual se puede observar que el tiempo de estabilización del PID para un setpoint de 2, 3,5 y 5 litros, es respectivamente de 1 593 segundos (26,55 min), 564 segundos (9,4 min) y 343 segundos (5,72 min) mayor en comparación al tiempo de estabilización del MPC; por lo tanto, es evidente que el controlador avanzado MPC es mucho más eficiente que el PID.

Tabla 17. Características de desempeño del controlador MPC y PID para el tanque 1 (caso 4).

Setpoint (litros)	Controlador	Tiempo de Subida (s)	Tiempo de Estabilización (s)	Sobrepaso máximo (%)
2	MPC	312	645	0
2	PID	410	2 238	7,1
3,5	MPC	679	1 116	0,03
3,5	PID	875	1 680	0,03
5	MPC	1 396	2 244	0,02
5	PID	1 408	2 587	0

En la **Figura 107**, se muestra la respuesta de los controladores ante las perturbaciones, en la cual se puede observar que el MPC presenta una mayor capacidad de rechazo de estas señales indeseadas que afectan la salida del sistema. Ante una señal de perturbación de -0,5 el MPC regresa a su setpoint en un tiempo de 500 segundos (8,33 min), mientras que el PID rechaza esta misma perturbación en un lapso de 695 segundos (11,58 min).

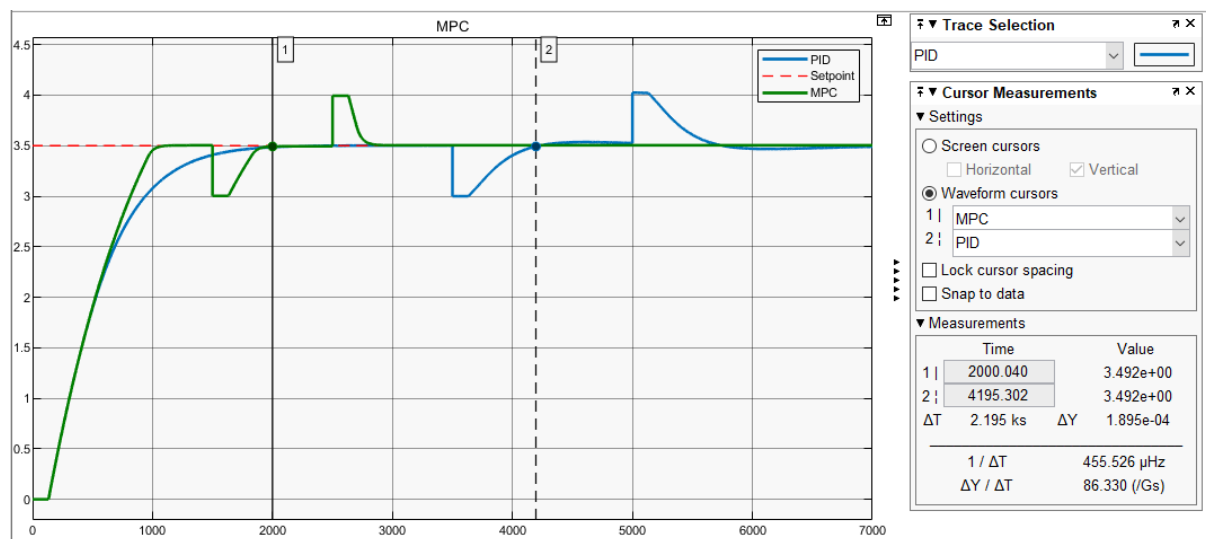


Figura 107. Análisis de rechazo de perturbaciones en el control de nivel del tanque 1 (caso 4).

6.6 Interfaz gráfica de usuario (GUI)

La GUI desarrollada es intuitiva y permite al usuario seleccionar cada uno de los casos propuestos y simular la planta con los controladores PID y MPC diseñados, además permite modificar los parámetros preestablecidos del MPC y ejecutar nuevas simulaciones con los datos ingresados por el usuario.

En la **Figura 108** y 109, se presenta la simulación de la planta para el tanque 2 (caso 3) con los controladores diseñados para un setpoint de 1 y 3 litros, respectivamente. En esta última gráfica, se puede observar que el controlador MPC se estabiliza a los 734 segundos, mientras que la curva de control PID se estabiliza alrededor de los 1 656 segundos.

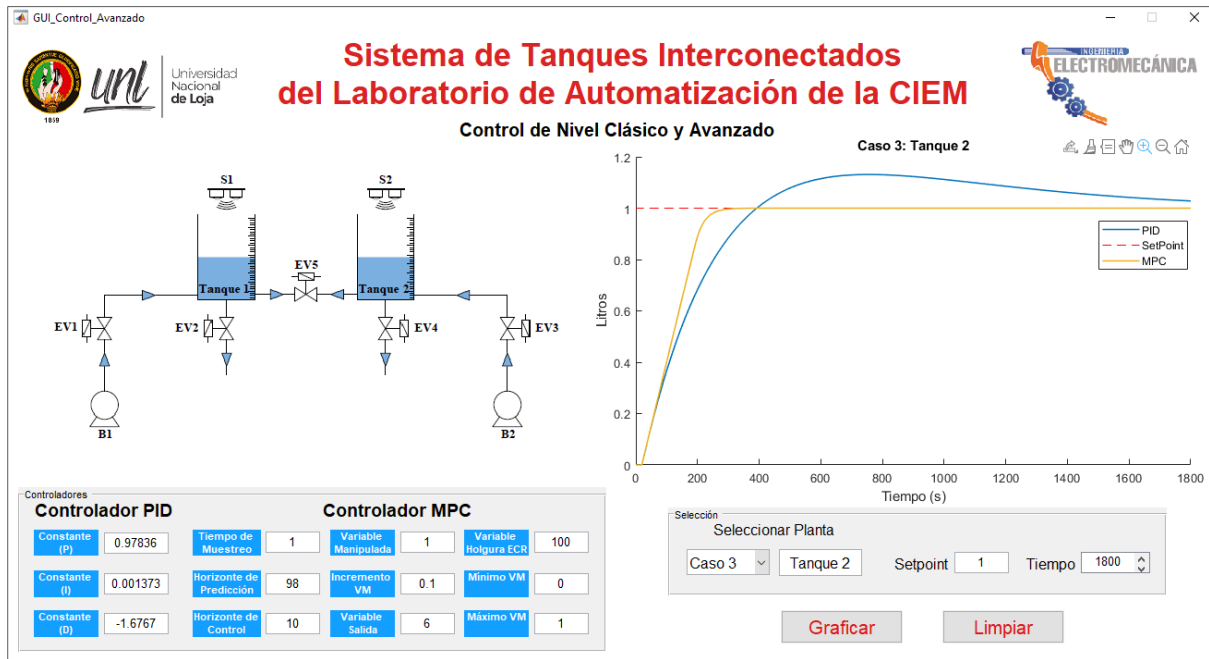


Figura 108. Simulación de la respuesta del controlador PID y MPC para el tanque 2 (caso 3) a un setpoint de 1 empleando la GUI.

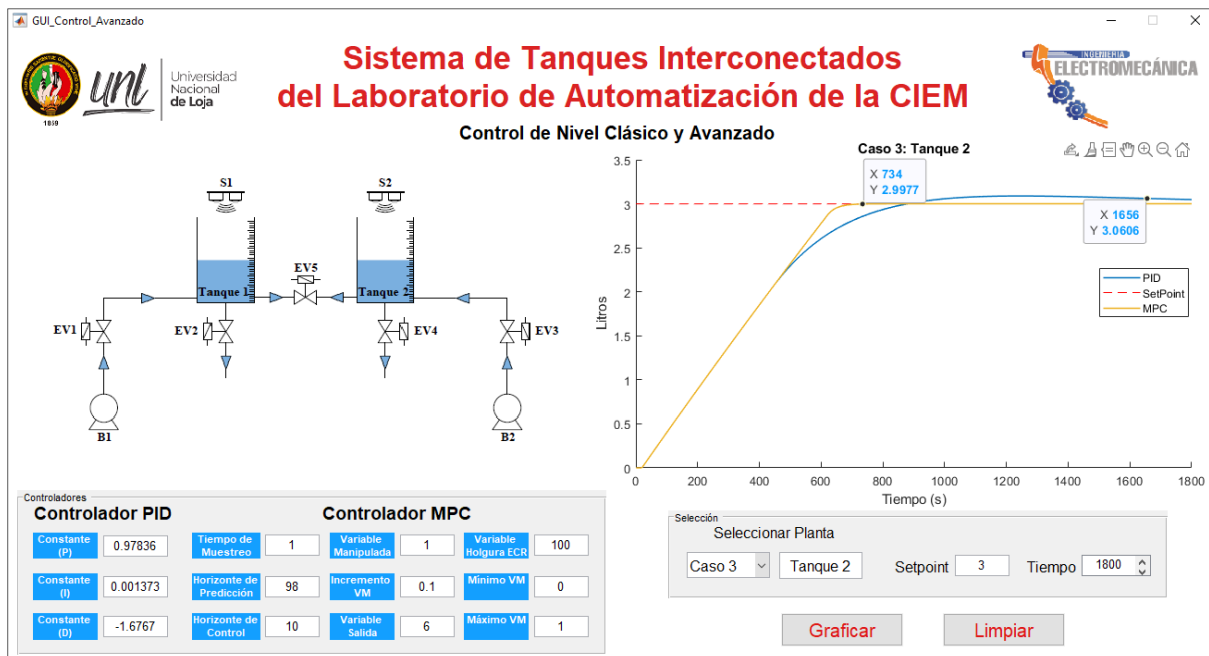


Figura 109. Comparación de la respuesta del controlador PID con el MPC para el tanque 2 (caso 3) a un setpoint de 3 empleando la GUI.

La GUI también permite modificar los parámetros del controlador MPC y ejecutar una nueva simulación ingresando los nuevos valores en los cuadros que contienen estos parámetros, con el objetivo de que el usuario pueda realizar una sintonización manual y comparar resultados. En la **Figura 110**, se evidencia un ejemplo de lo mencionado, donde se observa que al disminuir de 5 a 1 el peso a la variable de salida del controlador MPC para el tanque 1 (caso 1), este último presenta un error en el setpoint en estado estable del 6 % (0,94 litros), además de ser ineficiente comparado con el primer controlador diseñado, el cual se estabiliza a los 128 segundos y no presenta ningún error en estado estable.

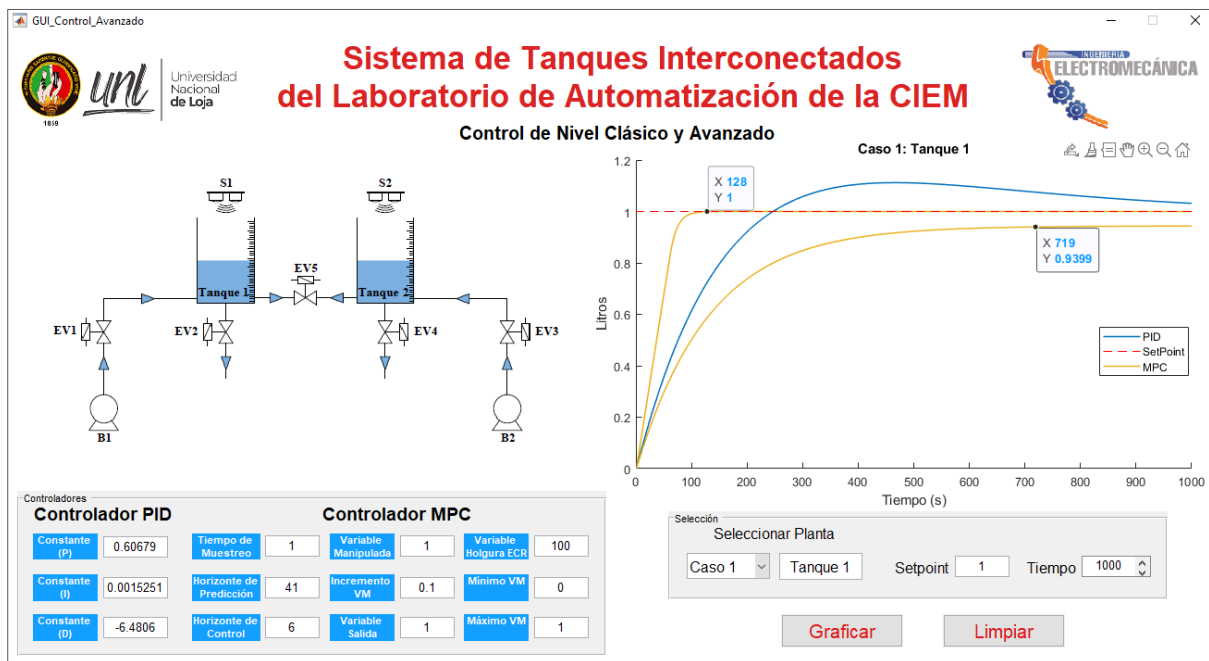


Figura 110. Comparación de las curvas del controlador MPC para el tanque 1 (caso 1), obtenidas de la sintonización manual realizada con la GUI.

7. Discusión

El control de procesos es parte esencial de toda industria basada en la producción, donde el control predictivo toma relevancia por las ventajas que presenta frente a otros controladores. Entorno a ello, se planteó diseñar este controlador de nivel avanzado para el sistema prototipo de tanques interconectados que posee la carrera de Ingeniería Electromecánica, debido a que solo cuenta con un control on-off y no considera la dinámica compleja del sistema. Al desarrollar este proyecto, se determinó el proceso necesario para el diseño del controlador avanzado MPC y se corroboró los resultados obtenidos con la teoría existente.

Inicialmente, el banco de pruebas construido para la medición y control de nivel de líquidos, poseía cuatro tanques interconectados entre sí y contaba con un controlador PID, donde la acción de control se realizaba mediante las electroválvulas, las cuales se encargaban de abrir y cerrar el paso de agua, para controlar el nivel en los tanques. Luego, en otro proyecto, se realizó una serie de modificaciones al banco mencionado, obteniendo la estructura actual y contando sólo con dos tanques interconectados, cuyo objetivo es controlar y supervisar de forma remota (comunicación inalámbrica) el proceso de llenado de los tanques; sin embargo, este posee únicamente un control on-off, como ya se ha mencionado. Finalmente, en el presente trabajo de titulación se ha diseñado un controlador avanzado para este sistema, logrando obtener un control de nivel eficiente en simulación, el mismo que puede ser implementado en un futuro proyecto, para su estudio y análisis.

Con el experimento realizado sobre el sistema de tanques interconectados, se obtuvo los datos necesarios para efectuar la identificación de sistemas y obtener su modelo matemático. Al contar con el modelo, se pudo diseñar el controlador avanzado MPC y realizar su sintonización aplicando un algoritmo evolutivo, obteniendo los resultados deseados.

En concordancia con las recomendaciones dadas por Capuano (2021) para la asignación de pesos y demás parámetros del MPC, se pudo evidenciar con los errores obtenidos en la sintonización con el algoritmo desarrollado, que al incrementar el peso de la variable de salida o controlada, el MPC prioriza el seguimiento de la misma, haciendo que el error en el setpoint en estado estable sea mínimo, en este caso de cero. Por lo tanto, la prioridad asignada al peso de la variable manipulada fue de 1 (prioridad media) y el peso de la variable controlada fue entre 5 y 6 (prioridad superior a la media). Referente al tiempo de muestreo, se asignó un valor de 1 segundo en todos los casos planteados, de acuerdo a lo sugerido por el autor mencionado, para sistemas de control de procesos; obteniendo como resultado un equilibrio entre rendimiento y esfuerzo computacional, debido a que un valor menor al recomendado, ocasionaba que el esfuerzo computacional sea muy elevado a la hora de ejecutar la simulación.

Moromenacho & Rosero (2010) en su proyecto de tesis referido al “Diseño e implementación del control predictivo en tanques acoplados”, el cual emplea el MPC Toolbox de MATLAB® para su diseño al igual que en el presente proyecto, implementa un controlador con la planta modelada en base a ecuaciones y otro con la planta identificada, donde este último responde mejor a comparación del primero, justificando la identificación de sistemas al obtener resultados aproximados y bastante aceptables, comparando los resultados de la simulación con respecto a la implementación. Por lo que, se espera que el controlador diseñado tenga una respuesta similar en el proceso real a los obtenidos en simulación, en caso de que en un futuro proyecto sea implementado.

En los resultados obtenidos del presente trabajo de titulación, se observó que el MPC presenta un mejor desempeño a comparación del controlador PID, siendo el control avanzado más eficiente en todos los casos analizados. Es así que en los procesos donde se tiene retardos de tiempo considerables, se pudo verificar en concordancia con la literatura existente, que el MPC presenta una respuesta mucho más rápida que el PID, debido a que este último no está diseñado para trabajar con sistemas que presentan retardos de tiempo elevados, ya que este atraso en la salida del sistema, ocasiona que la respuesta del controlador PID sea mucho más lenta cuanto mayor es el retardo; por otro lado, el MPC también mostró mayor capacidad de rechazo de perturbaciones.

Existen varios trabajos de investigación, donde se compara el control predictivo con el control PID. Buelvas & Pastrana (2023) en su proyecto de grado, llevan a cabo el diseño de un controlador predictivo para tres tanques acoplados, sujetos a retardos de tiempo, donde realizan una comparación con un controlador PID, concluyendo que el control MPC presenta mejores valores de desempeño que el PID. Gutarra (2016) también presenta un proyecto similar al mencionado, el cual concluye que “los controladores predictivos presentan mejor desempeño, mejor calidad y mejor ahorro de la energía en comparación al control PID”. Los resultados obtenidos en los proyectos mencionados, al igual que en el presente trabajo, evidencia que el control MPC es más eficiente que el control PID.

8. Conclusiones

- Los modelos obtenidos para las diferentes configuraciones de llenado mediante identificación de sistemas, presentaron un porcentaje de exactitud mayor al 95 % con relación a los datos reales obtenidos de la planta, representando lo más preciso posible la dinámica del sistema objeto de estudio, lo cual garantiza el buen desempeño del controlador MPC, ya que éste hace uso del modelo para predecir el comportamiento futuro de la planta y determinar las acciones de control.
- El controlador MPC diseñado para el sistema de tanques interconectados es más eficiente a comparación del controlador PID en todos los casos analizados mediante simulación, en especial en los procesos que conllevan un retardo de tiempo significativo en su salida, tal como señala la teoría. Por ejemplo, en el llenado del tanque 2 (caso 2), cuyo sistema posee un retardo de tiempo en su salida de 86,84 s, presenta una diferencia en el tiempo de estabilización de 815 s (13,58 min) y 948 s (15,8 min) para un nivel de 4 litros y 8 litros, respectivamente, en comparación del MPC con el PID. El MPC también presentó mayor capacidad de rechazo de perturbaciones.
- Al considerar las limitaciones físicas del actuador en los controladores PID y MPC diseñados, se logró obtener un control óptimo sin sobrepasar los límites de operación máximos del actuador, presentando un desempeño adecuado en simulación ante cualquier valor de referencia asignado.
- Al incluir en el MPC las restricciones y pesos en la variable manipulada y controlada, se optimizó el rendimiento del controlador, pudiendo operar en el límite de las restricciones en caso de ser necesario, lo que hace que se obtenga una mejor respuesta en el control a comparación del PID, siendo esto más evidente en procesos con una dinámica lenta como es el caso del sistema estudiado.

9. Recomendaciones

- Al momento de estimar el modelo matemático de la planta mediante identificación de sistemas, es importante considerar que la función de transferencia a obtener sea de primer o segundo orden y que ésta represente lo más exacto posible la dinámica real del sistema estudiado, ya que modelos con un orden superior se vuelven complejos de controlar con un PID y en el caso del MPC se incrementa el costo computacional.
- Si se desea diseñar un controlador para un sistema que presenta un gran retardo de tiempo, se debe crear estrategias de control ideales para estos casos, como el control predictivo basado en modelo (MPC) o compensadores de retardo como el predictor de Smith, debido a que este atraso en la dinámica del proceso provoca que el sistema en lazo cerrado del control PID presente un comportamiento lento.
- Para futuros proyectos sería interesante implementar este controlador avanzado, para evaluar su comportamiento en la práctica, debido a que este trabajo investigativo abarca el diseño, estudio y comparación del controlador predictivo basado en modelo con el controlador PID mediante simulación, más no su implementación física.
- Emplear el método anti-windup en el diseño del controlador PID, con el fin de mejorar su desempeño y que presente un comportamiento real, para que al momento de su implementación física se obtenga resultados aceptables y aproximados a los obtenidos en simulación, ya que, de no ser así en el proceso real se puede llegar a tener un comportamiento no deseado de la variable controlada.
- Es importante incluir en el diseño del controlador MPC las restricciones físicas de la planta, sobre la variable controlada y manipulada, principalmente por razones de seguridad y para evitar daños en los equipos. A su vez también se puede establecer otras restricciones como el incremento de control, para forzar que la respuesta del proceso tenga ciertas características de acuerdo a lo requerido por el diseñador.

10. Bibliografía

- Altec. (2021, octubre). *Clasificación de medidores de nivel*.
<https://altecdust.com/clasificacion-de-medidores-de-nivel/>
- Amador Soto, G. J. (2019). *Control de procesos industriales con minimización del consumo energético*. <https://repositorio.unal.edu.co/handle/unal/69608>
- Bohórquez, K. B., Fonseca, D. F., & Gutiérrez, S. (2017). *Sistema didáctico para el control de nivel con tanques acoplados* [Universidad Católica de Colombia].
<http://hdl.handle.net/10983/15297>
- Buelvas Sierra, O. A., & Pastrana Arellano, A. (2023). Controlador predictivo de nivel para un sistema hidráulico de tres tanques acoplados sujetos a retardos de tiempo. *instname:Universidad Antonio Nariño*.
<http://repositorio.uan.edu.co/handle/123456789/7636>
- Capuano, A. (2021). *Development of an Adaptive Model Predictive Control for platooning safety in Battery Electric Vehicles* [Politecnico di Torino].
<https://webthesis.biblio.polito.it/19499/>
- Castaño Giraldo, S. A. (2020). *Control realimentado: Lazo abierto y lazo cerrado. Sistemas dinámicos de primer orden. Sistemas de segundo orden. Control PID. Anti Windup en un control PID*. <https://controlautomaticoeducacion.com/category/control-realimentado/>
- Castaño Giraldo, S. A. (2021). *Análisis de sistemas: Función de transferencia. Retardo o tiempo muerto de un sistema*.
<https://controlautomaticoeducacion.com/category/analisis-de-sistemas/>
- Castillo, F. J. (2014). *Diseño y construcción de un sistema de tanques acoplados para la medición y control automático de nivel de líquidos* [Universidad Nacional de Loja].
<https://dspace.unl.edu.ec/handle/123456789/11998>
- Córdova, S. M. (2017). *Modelo de mejora continua para el proceso de molienda en el Ingenio Azucarero Monterrey "MALCA"* [Universidad del Azuay].
<http://dspace.uazuay.edu.ec/handle/datos/7027>
- Fimbres, O. (2018, febrero). *Parámetros de control en celdas de flotación*.
<https://es.linkedin.com/pulse/parámetros-de-control-en-celdas-flotación-omar-fimbres>
- González, D., & González, A. I. (2023). Influencia de las Restricciones en la Síntesis de un Controlador Predictivo. *Revista Politécnica*, 51(1), Article 1.
<https://doi.org/10.33333/rp.vol51n1.03>

- Gutarra, P. S. (2016). *Modelación y control de un sistema de cuatro tanques acoplados* [Universidad de Piura]. <http://repositorio.concytec.gob.pe/handle/20.500.12390/189>
- Guzmán, G. F. (2018). *Implementación de un controlador predictivo basado en modelo con restricciones sobre un microcontrolador de gama alta* [Universidad Politécnica Salesiana]. <http://dspace.ups.edu.ec/handle/123456789/15136>
- Hernández, J. A., & Carmona, C. (2014). *Una Herramienta para la Optimización de Sistemas Complejos*. <https://www.researchgate.net/publication/274696769>
- Hernández, J. A., & Ospina, J. D. (2010). A multi dynamics algorithm for global optimization. *Mathematical and Computer Modelling*, 52(7), 1271-1278. <https://doi.org/10.1016/j.mcm.2010.03.024>
- Jara, C. E., & Maldonado, J. L. (2015). *Diseño e implementación de un controlador PID y un MPC aplicado al posicionamiento lineal de un sistema neumático en bucle cerrado*. <http://dspace.ups.edu.ec/handle/123456789/7754>
- Moromenacho, E. O., & Rosero, C. Z. (2010). *Diseño e implementación del control predictivo en tanques acoplados* [Escuela Politécnica Nacional]. <http://bibdigital.epn.edu.ec/handle/15000/2119>
- Mundo Minero. (2021, enero). *Tres razones para modernizar tu celda de flotación con Metso: Outotec*. <https://mundominero.com.pe/tres-razones-para-modernizar-tu-celda-de-flotacion-con-metso-outotec/>
- Oliden, J. C. (2016). *Desarrollo de un controlador predictivo basado en modelo para plataforma industrial* [Universidad de Piura]. <https://hdl.handle.net/11042/5449>
- Ortiz, P. A., Ramírez, J. L., & Cardona, L. (2011). *Modelo matemático y control de un sistema de fluidos*. Instituto Tecnológico Metropolitano. <https://doi.org/10.22430/9789588743080>
- Sánchez Ortiz, H. A. (2022, abril 8). *Control PID y sintonización en Matlab*. <https://www.youtube.com/watch?v=u3svlFCVjU0>
- Sánchez, V. A. (2018). *Diseño de un sistema de control predictivo multivariable aplicado a un proceso hidráulico de cuatro tanques acoplados* [Pontificia Universidad Católica del Perú]. <https://tesis.pucp.edu.pe/repositorio/handle/20.500.12404/13563>
- Sendoya, D. F. (2013). ¿Qué es el Control Predictivo y Hacia Dónde se Proyecta? *Publicaciones e Investigación*, 7, 53-59. <https://doi.org/10.22490/25394088.1106>
- Tlachi, E. (2018). *Algoritmo de control predictivo aplicado a el control de nivel de líquido de un sistema de 4 tanques acoplados* [Benemérita Universidad Autónoma de Puebla]. <https://repositorioinstitucional.buap.mx/handle/20.500.12371/7626>

Torres, J. L., & Chanoluisa, S. X. (2015). *Diseño e implementación de un sistema de control en una planta de llenado de tanques a escala usando un transmisor de nivel y algoritmo PID para pruebas en el laboratorio de automatización industrial* [Universidad Politécnica Salesiana]. <http://dspace.ups.edu.ec/handle/123456789/10414>

11. Anexos

Anexo 1. Programación en diagrama de bloques para el VI diseñado en LabVIEW 2019.

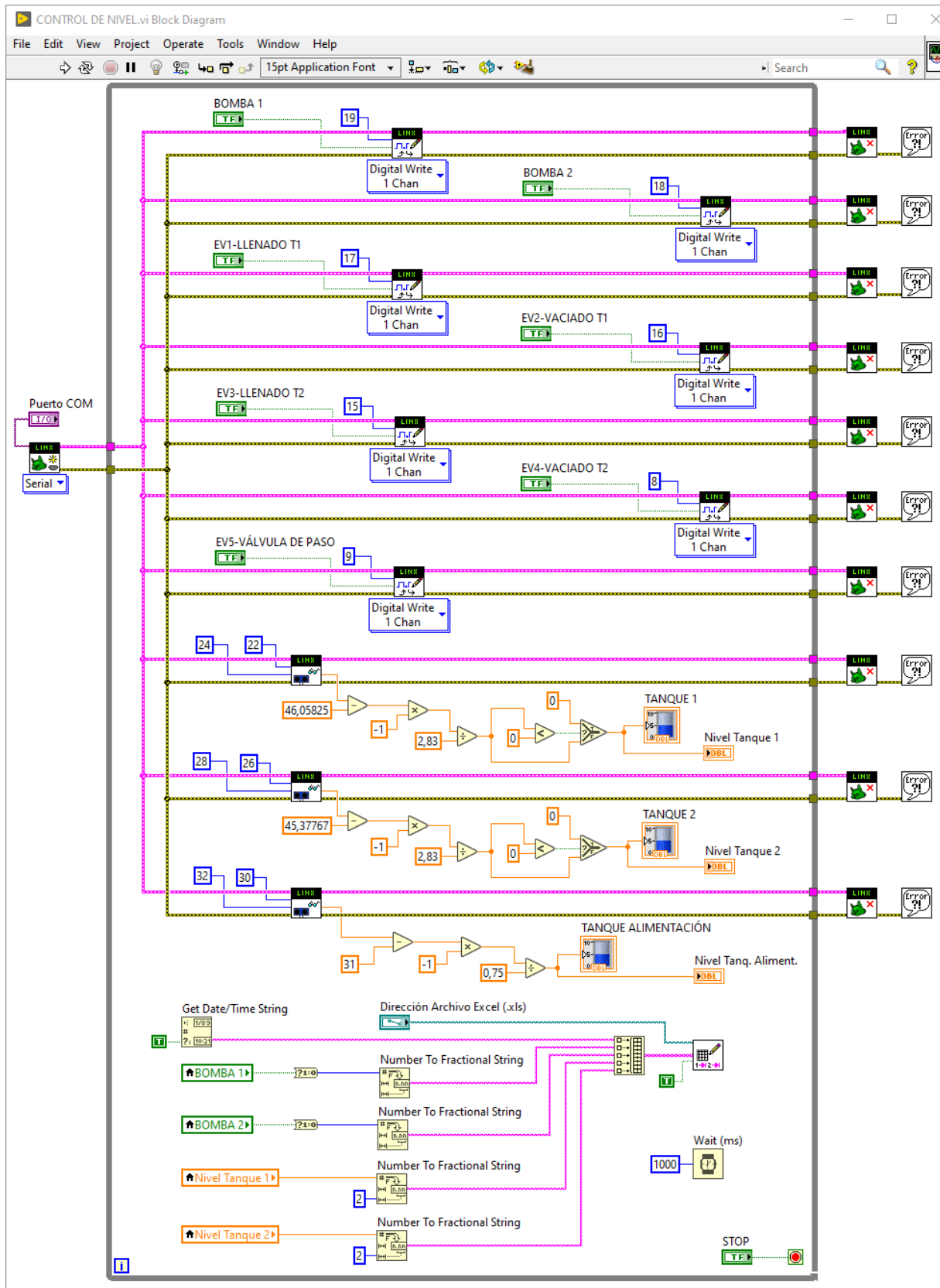


Figura 111. Programación en el diagrama de bloques para el instrumento virtual (VI) diseñado.

Anexo 2. Algoritmo para la función objetivo.

```
function [error, y, y1] = funcion_obj_mpc(Valor)

% Definir la función de transferencia del sistema
num = [27.735]; % Numerador de la función de transferencia
den = [1995.3 1]; % Denominador de la función de transferencia
G = tf(num, den);

% Definir los parámetros del controlador MPC
Ts = 1; % Tiempo de muestreo
Hp = fix(Valor(1)); % Horizonte de predicción
Hc = fix(Valor(2)); % Horizonte de control
weights = struct('ManipulatedVariables', 1, 'ManipulatedVariablesRate', 0.1,
'OutputVariables', 5, 'ECR', 100);
MV = struct('Min', 0, 'Max', 1); % Límites de la variable manipulada

% Tiempo final de simulación (definir parámetro)
Tfinal = 100;

% Crear un objeto de controlador MPC
mpcobj = mpc(G, Ts, Hp, Hc, weights, MV);

% Definir la señal de entrada escalón
u = ones(Tfinal/Ts, 1);

% Simular la respuesta del controlador MPC a la entrada escalón
y1 = sim(mpcobj, Tfinal+1, u);

% Graficar la respuesta del sistema a una entrada constante de 1
t = 0:Ts:Tfinal; % Vector tiempo (pasos "0:Ts", tiempo_simulación "Tfinal")
y = step(G, t);

% Cálculo del error (función objetivo) entre y & y1
error=sum(abs((y-y1))/length(y))*100

end
```


Anexo 3. Programación para evaluar el MPC.

```
close all
clc
tic

%Creación de la variable de salida donde se almacenan los resultados
exist salida2
if ans == 0
    salida2 = 0
end

%Definir los límites de evaluación para los parámetros del MPC
lb =[1  4 ]; %Límites inferiores
ub =[40  10]; %Límites superiores

n = 40;
ng = 25;
%salida2 = [Horizonte_predicción Horizonte_control Error], [0 0 0]

%Función 01
fprintf('Error Numerico');
fobj=@funcion_obj_mpc;
Erro01=MAGO(fobj,lb,ub,n,ng);
[error,y,y1] = funcion_obj_mpc(Erro01.sol);
x=length(salida2(:,1))+1;
salida2(x,1) = Erro01.sol(1)
salida2(x,2) = Erro01.sol(2)
salida2(x,3) = error

close all
figure; plot (y1); title ('Respuesta MPC');
toc
```

Anexo 4. Algoritmo por Multidinámicas para Optimización Global (MAGO).

```
function [magostr, mejorind]=MAGO(fobj,lb,ub,n,ng)

% Algoritmo evolutivo multi especie para optimizar la función fobj.

% fobj: función objetivo.
% lb: límite inferior (vector fila) del espacio de búsqueda.
% ub: límite superior (vector fila) del espacio de búsqueda.
% n: tamaño de la población.

np=length(lb); % Número de parámetros de la función objetivo.
f=zeros(n,ng); % En este vector se almacena todas las evaluaciones de la función
objetivo.
% Generación de la población inicial:
card=zeros(ng,3);
% Cargar Pob
Pob=mvrnif(n,lb,ub);
save('Pob', 'Pob')
% Generación de la población inicial. Se genera aleatoriamente
% de manera uniforme sobre todo el espacio de búsqueda.
NPob=zeros(size(Pob)); % En esta matriz se guarda la población intermedia en cada
generación.
Pobev=zeros(n,np,ng); % En esta matriz se guarda todas las poblaciones del
proceso.
cent=zeros(1,n); % Este centinela se utiliza para no evaluar dos veces un mismo
individuo.
% f=zeros(n,ng+1);

i=1; % Contador del ciclo de evolución.
while i<=ng;
% Evaluación de la población

j=1; % Contador del ciclo de evaluación de la función objetivo.
    while j<=n
        if cent(j)==0
            x=Pob(j,:);
            f(j,i)=fobj(x);
        end
        j=j+1;
    end
    cent=zeros(1,n);

%% ORDENAMIENTO PARA MINIMIZACIÓN
    [a,b]=sort(f(:,i)); % Ordenamiento del vector con las evaluaciones de la
función objetivo.
%% ORDENAMIENTO PARA MAXIMIZACIÓN
    % [a,b]=sort(f(:,i),'descend'); % Ordenamiento del vector con las evaluaciones de
la fobj.
%%
    mejorind(i,:)=Pob(b(1),:); % Selección del mejor individuo.
    mejorf(i)=f(b(1),i); % Valor de la función objetivo para el mejor individuo.

    S=cov(Pob); % Matriz de covarianzas de la población actual.
    Sd=sqrt(diag(S)); % Matriz diagonal con las varianzas de la población.
    m=mean(Pob); % Media de la población.
    mini=m-0.5*Sd; % Extremo izquierdo G1.
    maxi=m+0.5*Sd; % Extremo derecho G1.
    mini1=m-1*Sd; % Extremo izquierdo G2.
```

```

    max1=m+1*Sd; % Extremo derecho G2.
    R1=restriction(Pob,mini,maxi); % Se determina cuáles individuos caen en la
región 1.
    R2=restriction(Pob,mini1,maxi1); % Se determina cuáles individuos caen en la
región 2.
    n1=sum(R1); % Cardinalidad del grupo 1: es la suma de los que caen en la región
1 menos 1.

% Cálculo de la distancia en la primera generación
if n1 < 0
    n1 = 0;
end
    n2=sum(R2)-n1-1; % Cardinalidad del grupo 2: es la suma de los que caen en la
región 2 menos 1.
if n2 < 0
    n2 = 0;
end

n3=n-n1-n2; % Cardinalidad del grupo 3: son los que no pertenecen ni al grupo 1 ni
al grupo 2.

card(i,:)= [n1,n2,n3]; % Cardinalidades de las tres dinámicas.
Sn=1/norm(S,inf)*S; % Matriz F de las ecuaciones (2) y (3).
K(:, :, i)=S; % En esta matriz se guarda la covarianza de la Población de cada
generación.
NPob(1:1,:)= Pob(b(1:1),:);

% Se grafica el valor de la funcional de costo de cada generación
% plot(i,fobj(NPob(1:1,:)),'LineWidth', 2, ...)

if n1>0
    NPob(2:n1+1,:)=Pob(b(2:n1+1),:); % Con esto se descarta el mejor individuo.
end

if n1>0
    j=1;

    while j<=n1
        if j==1
            diferencias(j,:)=zeros(1,np);
        else
            diferencias(j,:)=(Sn*(mejorind(i,:)-NPob(j,:))')'; % Cálculo de
% las diferencias de la ecuación (2).
        end
        j=j+1;
    end

    NPob(1:n1,:)=NPob(1:n1,:)+ diferencias(1:n1,:); % Esta es la élite
% transformada de acuerdo con la ecuación (2).

% Se obliga a que todos los individuos de la generación permanezcan dentro de la
región de búsqueda.

[p,q] = size(NPob);

for k = 1:q
    for l = 1:p
        if NPob(l,k) > ub(1,k)
            NPob(l,k) = ub(1,k);
        end
    end
end

```

```

        end
        if NPob(1,k) < lb(1,k)
            NPob(1,k) = lb(1,k);
        end
    end
end

clear diferencias

j=1;
while j<=n1
    x=NPob(j,:);
    fN=fobj(x); % Si los individuos transformados
    if fN>f(b(j),i)
        % CONTROL DE MINIMIZACION
        % if fN<f(b(j),i)%
        % CONTROL DE MAXIMIZACION
        NPob(j,:)=Pob(b(j),:);
        % f(j,i)=f(b(j),i);
        % cent(j)=1;
    else
        % f(j,i)=fN;
        % cent(j)=1; % El centinela indica que el individuo ya ha sido evaluado y su
        % evaluación guardada.
    end
    j=j+1;
end
end
reloj=clock; % Estos comandos se usan para generar números aleatorios.
            % La semilla se fija según el reloj.

randn('state',reloj(6));

if n2==0 && n3~=0 && n1~=0
    NPob(n1+1:n,:)=mvrnif(n3,lb,ub); % Generación del grupo 3 sin grupo 2.
elseif n3==0 && n2~=0 && n1~=0
    NPob(n1+1:n,:)=mvrnif(n2,mini,maxi); % Generación del grupo 2 sin grupo 3.
elseif n2~=0 && n3~=0 && n1~=0
    NPob(n1+1:n1+n2,:)=mvrnif(n2,mini,maxi); % Generación del grupo 2.
    NPob(n1+n2+1:n,:)=mvrnif(n3,lb,ub); % Generación del grupo 3.
    % Si n2=0 y n3=0, basta con el calculo de n.
elseif n2~=0 && n3~=0 && n1==0
    NPob(1:1,:)= Pob(b(1:1),:);
    NPob(n1+2:n1+n2,:)=mvrnif(n2-1,mini,maxi); % Generación del grupo 2.
    NPob(n1+n2+1:n,:)=mvrnif(n3,lb,ub); % Generación del grupo 3.
    % Si n2=0 y n3=0, basta con el calculo de n.
elseif n2==0 && n3~=0 && n1==0
    NPob(1:1,:)= Pob(b(1:1),:);
    NPob(n1+2:n,:)=mvrnif(n3-1,lb,ub); % Generación del grupo 3 sin grupo 2.
elseif n3==0 && n2~=0 && n1==0
    NPob(1:1,:)= Pob(b(1:1),:);
    NPob(n1+2:n,:)=mvrnif(n2-1,mini,maxi); %Generación del grupo 2 sin grupo 3.
end

[p,q] = size(Pob);
for k = 1:q
    for l = 1:p
        if NPob(l,k) > ub(1,k)

```

```

        NPob(1,k) = ub(1,k);
    end
    if NPob(1,k) < lb(1,k)
        NPob(1,k) = lb(1,k);
    end
end
end

Pob=NPob; % La población consolidada se guarda en la matriz Pob.
fbestd(i,:)=f(b(1),i);
f(b(:,i));

[p,q] = size(Pob);

for k = 1:q
    for l = 1:p
        if Pob(l,k) > ub(1,k)
            Pob(l,k) = ub(1,k);
        end
        if Pob(l,k) < lb(1,k)
            Pob(l,k) = lb(1,k);
        end
    end
end

l_mejor=fbestd(i);

if l_mejor<=0.9
    break
end

i=i+1;
%i

end

sol=Pob(1,:);
fbest=f(b(1),ng);
inds = [fbestd mejorind];
magostr=struct('inds',inds,'fbestd',fbestd,'mejorf',mejorf,'fbest',fbest,'sol',sol);

function B=mvrnif(n,lb,ub)

% n: número de vectores a generar
d=length(lb); % dimensión
C=rand([n,d]);

LB=ones(n,d)*diag(lb);
UB=ones(n,d)*diag(ub);
B=C.*(UB-LB)+LB;

function R=restriction(M,lb,ub)

[n,m]=size(M);
[n1,m1]=size(lb);
R=zeros(1,n);

for i=1:n

```

```

    for j=1:n1
        z(j)=prod((M(i,:)<=ub(j,:))*1)*prod((M(i,:)>=lb(j,:))*1);
    end
    R(i)=prod(z);
end

```

Anexo 5. Código para cargar las funciones de transferencia al Workspace de MATLAB®.

```

FT_tanq1_caso1 = tf([27.735], [1995.3 1]);
FT_tanq2_caso2 = tf([9.3496], [1086.9 1], 'OutputDelay', 86.84);
FT_tanq2_caso3 = tf([29.805], [5932.8 1], 'OutputDelay', 18.47);
FT_tanq1_caso4 = tf([5.4898], [863.4 1], 'OutputDelay', 133.2);

```

Anexo 6. Programación para abrir el Simulink y cargar las imágenes a la GUI.

```

% --- Executes just before GUI_Control_Avanzado is made visible.
function GUI_Control_Avanzado_OpeningFcn(hObject, eventdata, handles, varargin)

%% Abrir el simulink
warning('off','all');
find_system('name','PID_tipo')
open_system('PID_tipo.slx')

%%cargar Imagenes
axes(handles.Logo)
[x,map] = imread('Logo.png');
image(x)
colormap(map)
axis off
hold on

axes(handles.Ingenieria)
[x,map] = imread('Electromecanica.png');
image(x)
colormap(map)
axis off
hold on

axes(handles.Tanques)
[x,map] = imread('Tanques Interconectados.png');
image(x)
colormap(map)
axis off
hold on

set(handles.graficar,'Enable','off')

```

Anexo 7. Programación para verificar el caso seleccionado en la GUI y cargar los parámetros al controlador.

```
% --- Executes on selection change in listBox1.
function listBox1_Callback(hObject, eventdata, handles)
indiceSeleccionado = get(hObject, 'Value');
opciones = get(hObject, 'String');
switch opciones{indiceSeleccionado}
    case 'Caso 1'
        %% Activar el boton de grafica
        set(handles.graficar, 'Enable', 'on')

        %% Cargar el listado de tanques
        set(handles.Lista_tanques, 'String', 'Tanque 1')

        %% Constantes PID Caso 1 tanque 1
        Kp = num2str(0.606793197743396);
        Ki = num2str(0.00152506829616581);
        Kd = num2str(-6.48063389276603);
        set(handles.P, 'String', Kp);
        set(handles.I, 'String', Ki);
        set(handles.D, 'String', Kd);

        %% Constantes MPC Caso 1 tanque 1
        Ts = num2str(1);
        HP = num2str(41);
        HC = num2str(6);
        VM = num2str(1);
        RVM = num2str(0.1);
        VS = num2str(5);
        ECR= num2str(100);
        VM_min = num2str(0);
        VM_max = num2str(1);
        Enviar_datos(handles, Ts, HP, HC, VM, RVM, VS, ECR, VM_min, VM_max);

        %% Acciones para la opción 1
        disp('Acciones para la opción 1');
```

Anexo 8. Función enviar datos para la GUI.

```
function Enviar_datos(handles, Ts, HP, HC, VM, RVM, VS, ECR, VM_min, VM_max)
    set(handles.Ts, 'String', Ts);
    set(handles.HP, 'String', HP);
    set(handles.HC, 'String', HC);
    set(handles.VM, 'String', VM);
    set(handles.RVM, 'String', RVM);
    set(handles.VS, 'String', VS);
    set(handles.ECR, 'String', ECR);
    set(handles.VM_min, 'String', VM_min);
    set(handles.VM_max, 'String', VM_max);
end
```

Anexo 9. Función extraer datos para la GUI.

```
function [Ts, HP, HC, VM, RVM, VS, ECR, VM_min, VM_max] = Extraer_datos(handles)
    Ts = str2double(get(handles.Ts, 'String')); % Tiempo de muestreo
    HP = str2double(get(handles.HP, 'String')); % Horizonte de predicción
    HC = str2double(get(handles.HC, 'String')); % Horizonte de control
    VM = str2double(get(handles.VM, 'String')) % Variable Manipulada
    RVM = str2double(get(handles.RVM, 'String')) % Incremento Variable
Manipulada
    VS = str2double(get(handles.VS, 'String')) % Variable de salida
    ECR = str2double(get(handles.ECR, 'String')) % ECR
    VM_min = str2double(get(handles.VM_min, 'String')) % VM min
    VM_max = str2double(get(handles.VM_max, 'String')) % VM max
end
```

Anexo 10. Programación para el controlador clásico PID de la GUI.

```
% --- Executes on button press in graficar.
function graficar_Callback(hObject, eventdata, handles)
%% Obtener el valor seleccionado del Listbox
indiceSeleccionado = get(handles.listbox1, 'Value');
valoresListbox = get(handles.listbox1, 'String');
valorSeleccionado = valoresListbox{indiceSeleccionado};

% Utilizar el valor seleccionado del Listbox
disp(['El valor seleccionado del Listbox es: ', valorSeleccionado])

switch valoresListbox{indiceSeleccionado}
    case 'Caso 1'
        %% LLamar al archivo PID tipo para graficar
        %% Controlador PID Caso 1 Tanque 1
        %%Funcion de transferencia
        num = [27.735];
        den = [1995.3 1];
        H = tf(num, den);

        % Entrada del setpoint
        setpoint = str2double(get(handles.SetPoint, 'String')); % Valor del
setpoint ingresado por el usuario

        % Control PID Caso 1
        tiempo = str2double(get(handles.Tiempo, 'String')); % Tiempo de muestreo

        %Constantes del PID
        Kp=str2double(get(handles.P, 'String'))
        Ki=str2double(get(handles.I, 'String'))
        Kd=str2double(get(handles.D, 'String'))

        set_param('PID_tipo/Setpoint', 'After', num2str(setpoint))
        set_param('PID_tipo/Retardo', 'DelayTime', num2str(0))

        set_param('PID_tipo/PID Controller', 'P', num2str(Kp))
        set_param('PID_tipo/PID Controller', 'I', num2str(Ki))
        set_param('PID_tipo/PID Controller', 'D', num2str(Kd))

        vector_num = sprintf('[%g]', num(1));
        vector_den = sprintf('[%g %g]', den(1), den(2));
```



```

set_param('PID_tipo/Transfer Fcn','Numerator',[num2str(vector_num)])
set_param('PID_tipo/Transfer Fcn','Denominator',[num2str(vector_den)])

set_param('PID_tipo','StopTime',num2str(tiempo))
set_param('PID_tipo','SimulationCommand','start')
pause(1);

x=evalin('base','out.salida');
x=x(1:tiempo)

t = 0:1:tiempo;
axes(handles.axes2)
hold on
plot(x, 'LineWidth', 1.2)
plot(t, setpoint*ones(size(t)), 'r--', 'LineWidth', 0.75);

```

Anexo 11. Programación para el controlador avanzado MPC de la GUI.

```

%% Controlador MPC Caso 1 Tanque 1
num = [27.735];
den = [1995.3 1];
G = tf(num, den);

% Definir los parámetros del controlador MPC
[Ts, HP, HC, VM, RVM, VS, ECR, VM_min, VM_max] = Extraer_datos(handles);
weights = struct('ManipulatedVariables', VM, 'ManipulatedVariablesRate',
RVM, 'OutputVariables', VS, 'ECR', ECR);
MV = struct('Min', VM_min, 'Max', VM_max); %Límites de la variable
manipulada

% Crear un objeto de controlador MPC
mpcobj = mpc(G, Ts, HP, HC, weights, MV);

% Definir la señal de entrada escalón
Tfinal = tiempo; % tiempo final de simulación
u = ones(Tfinal/Ts, 1); % señal de entrada escalón
u = setpoint*u;

% Simular la respuesta del controlador MPC a la entrada escalón
y2 = sim(mpcobj, tiempo, u);
hold on
plot(y2, 'LineWidth', 1.2)

title("Caso 1: Tanque 1")
xlabel("Tiempo (s)")
ylabel("Litros")
legend("PID", "SetPoint", "MPC")

% Acciones para la opción 1
disp('Acciones para la opción 1');

```

Anexo 12. Gráficas de la dinámica de nivel del sistema de tanques interconectados con los datos obtenidos en el experimento de identificación.

- **Caso 1: Llenado del tanque 1 con la bomba 1**

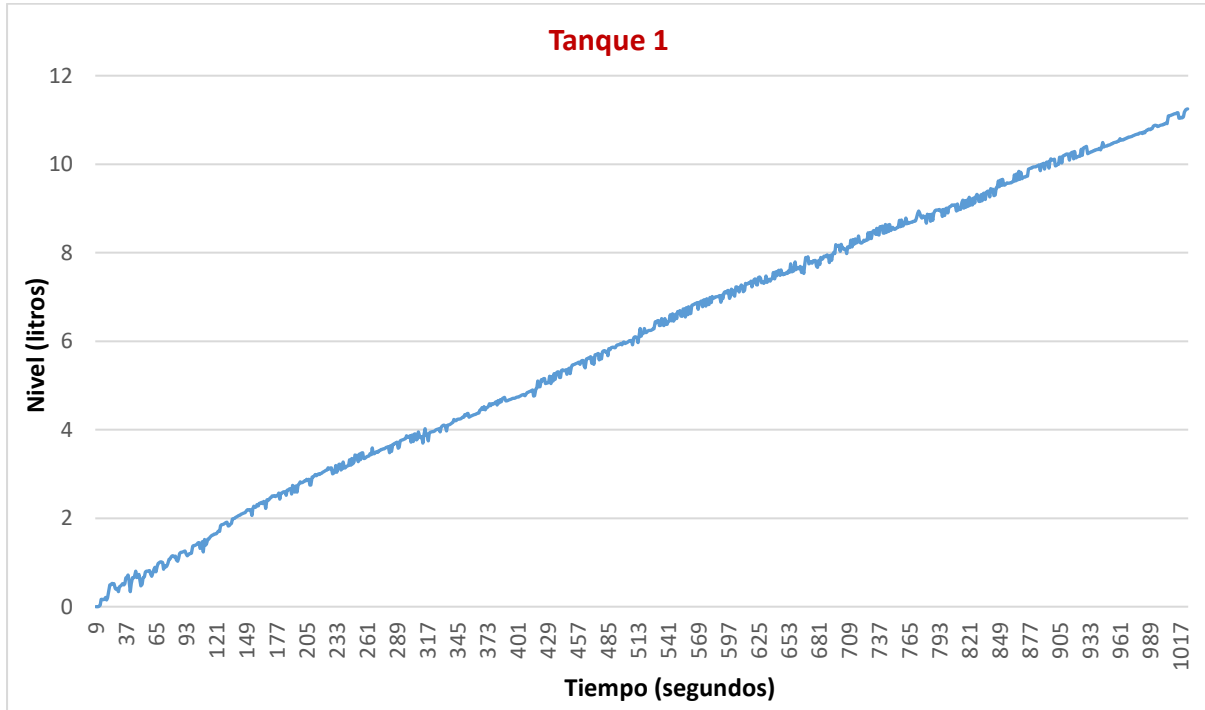


Figura 112. Comportamiento dinámico de llenado del tanque 1 (caso 1).

- **Caso 2: Llenado del tanque 1 y 2 con la bomba 1**

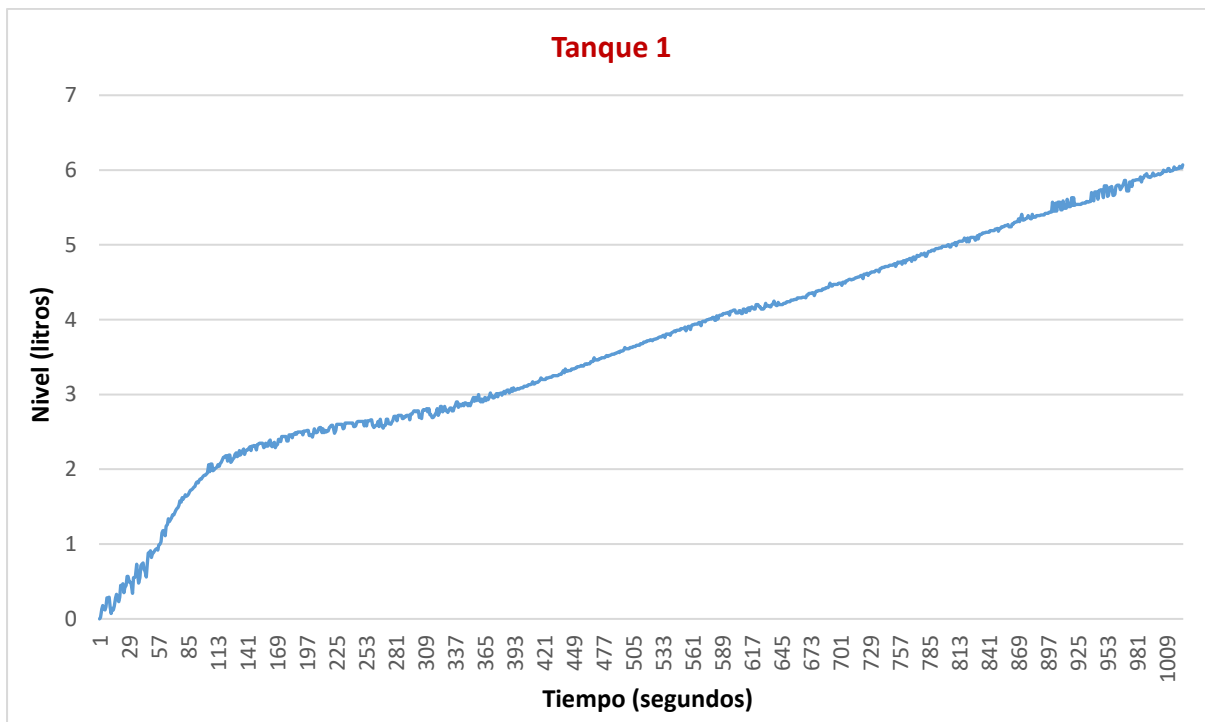


Figura 113. Comportamiento dinámico de llenado del tanque 1 (caso 2).

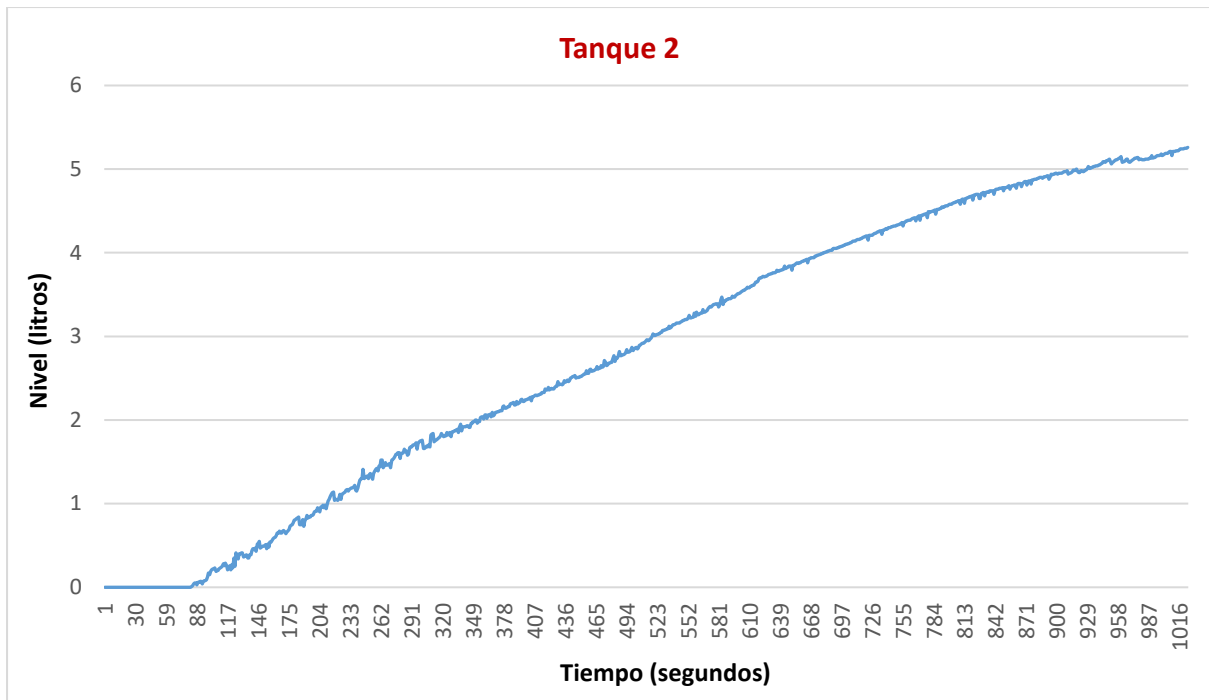


Figura 114. Comportamiento dinámico de llenado del tanque 2 (caso 2).

- **Caso 3:** Llenado del tanque 2 con la bomba 2

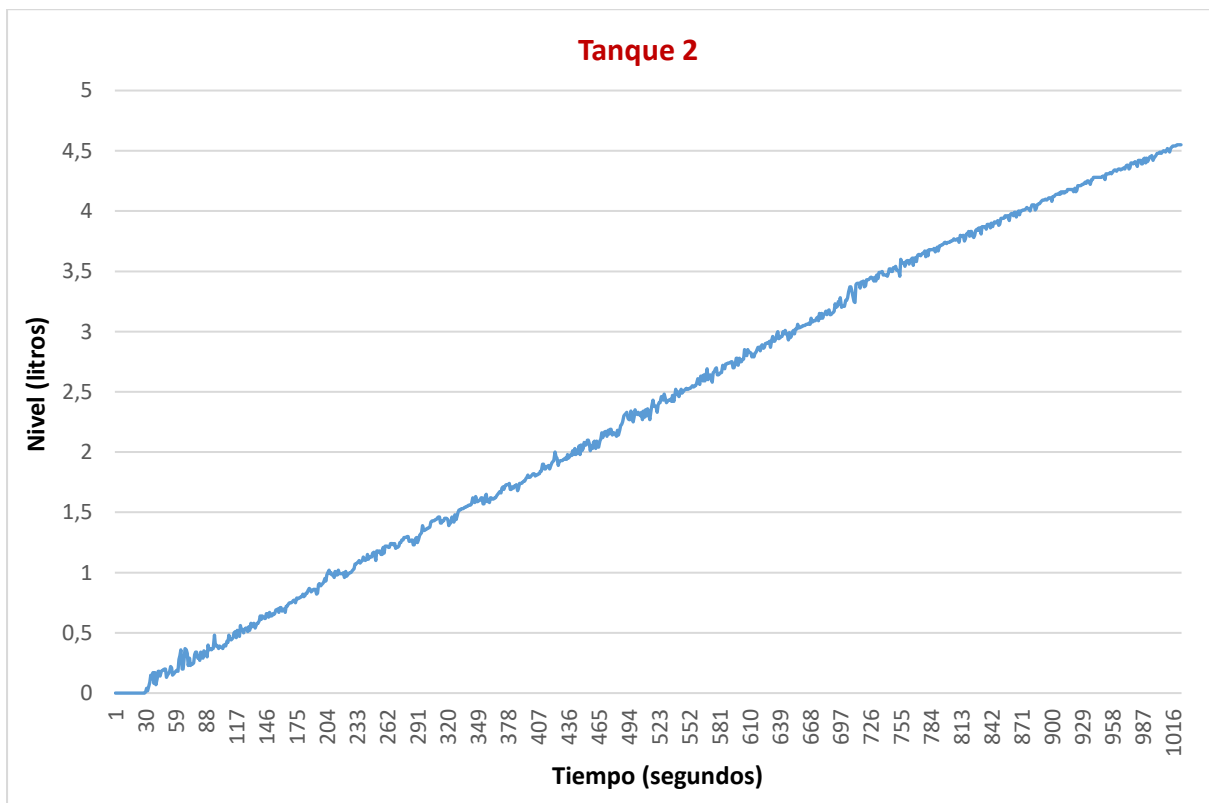


Figura 115. Comportamiento dinámico de llenado del tanque 2 (caso 3).

▪ **Caso 4:** Llenado del tanque 1 y 2 con la bomba 2

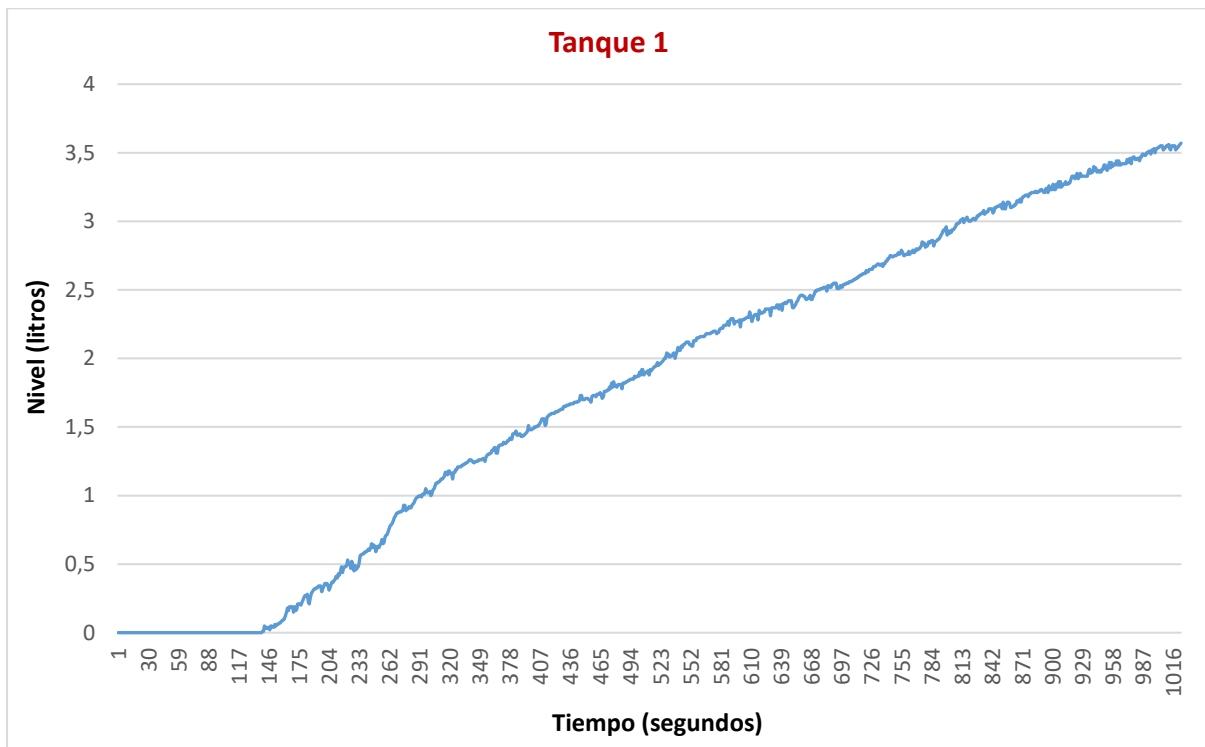


Figura 116. Comportamiento dinámico de llenado del tanque 1 (caso 4).

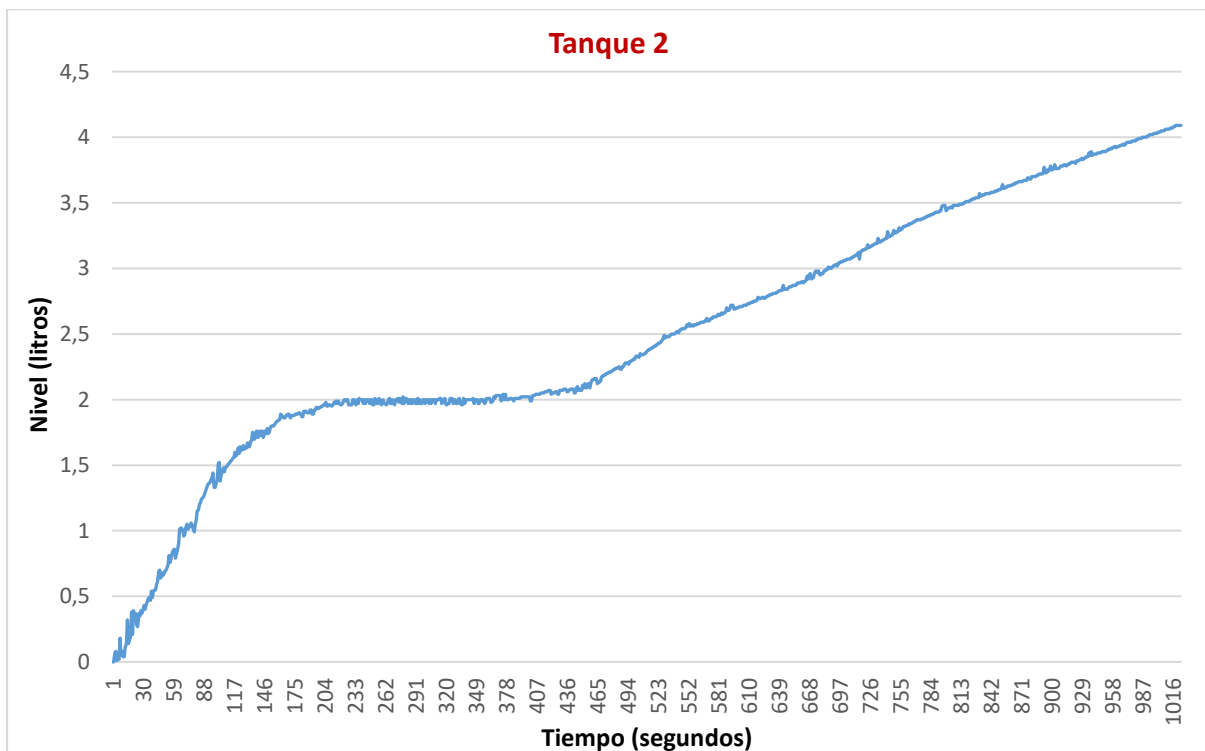


Figura 117. Comportamiento dinámico de llenado del tanque 2 (caso 4).

Anexo 13. Función de transferencia del tanque 1 (caso 2).

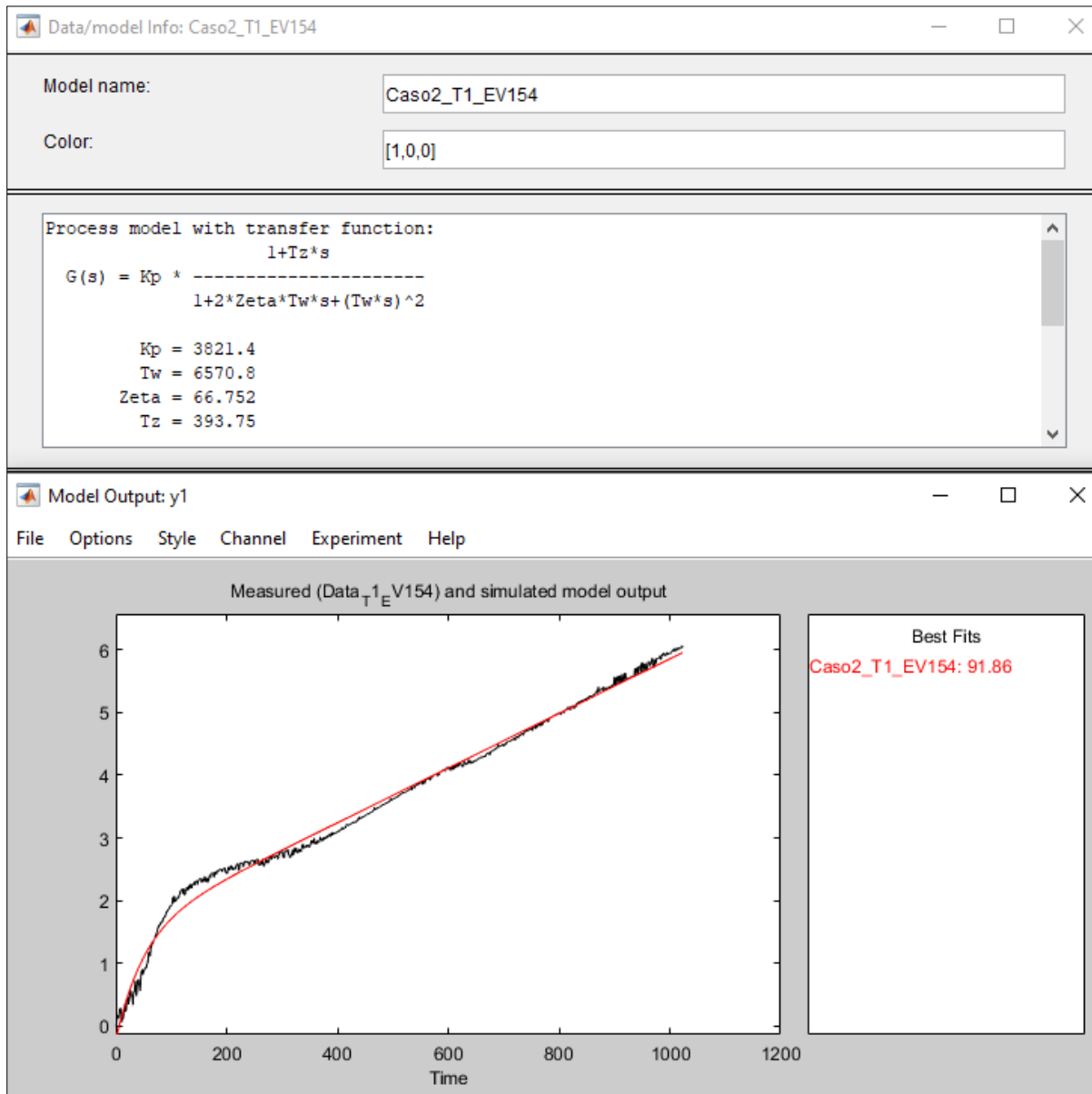


Figura 118. Función de transferencia del tanque 1 (caso 2).

Anexo 14. Función de transferencia del tanque 2 (caso 4).

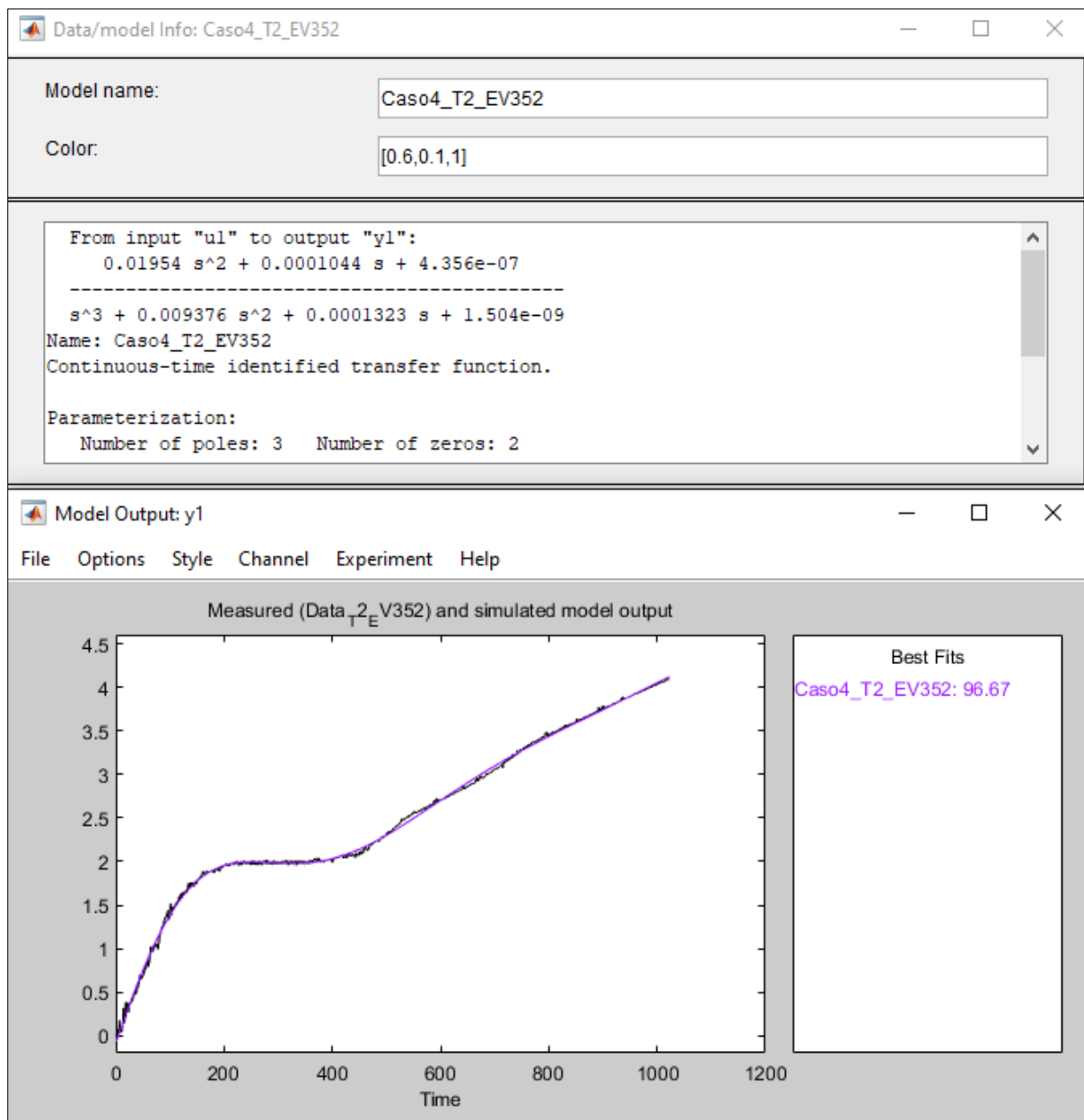


Figura 119. Función de transferencia del tanque 2 (caso 4).

Anexo 15. Interfaz gráfica de usuario (GUI).

Anexo 16. Instrumento virtual diseñado en LabVIEW para la recolección de datos.

Anexo 17. Algoritmo desarrollado en MATLAB® para la sintonización del MPC.

Los Anexo 15, 16 y 17 mencionados se encuentran en el siguiente enlace:

<https://drive.google.com/drive/folders/12tEtm9DRx5Ias4aSzlVQg1ikYLhq-jMt?usp=sharing>

Anexo 18. Certificado de traducción del resumen.

Lic. Denise Estefania Robalino S.
0968190698
denise.robalino@educación.gob.ec
Loja - Ecuador

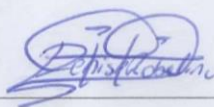
Loja, 14 de octubre del 2023

*La suscrita, Lic. Denise Estefania Robalino Sánchez, **DOCENTE EN CIENCIAS DE LA EDUCACIÓN MENCIÓN IDIOMA INGLÉS**, a petición de la parte interesada y en forma legal,*

C E R T I F I C A:

*Que, la traducción del trabajo de titulación, “**Diseño de un controlador avanzado para control de nivel en el sistema de tanques interconectados del Laboratorio de Automatización de la CIEM**”, de autoría del estudiante **Roger Javier Romero Jara**, con cédula de ciudadanía **No.1104124571**, ha sido realizado y aprobado por mi persona, LICENCIADA EN CIENCIAS DE LA EDUCACIÓN MENCIÓN IDIOMA INGLÉS, con número de registro del título en la Senescyt 1008 – 2018 – 1987009. El apartado del Abstract es una traducción textual del Resumen aprobado en español.*

Particular que comunico en honor a la verdad para los fines académicos pertinentes, facultando al portador del presente documento, hacer el uso legal pertinente.



Lic. Denise E. Robalino Sánchez
ENGLISH PROFESSOR