



UNIVERSIDAD DE IXTLAHUACA CUI, A. C.

LICENCIATURA EN INGENIERÍA EN
TELECOMUNICACIONES, SISTEMAS Y ELECTRÓNICA

INCORPORADA A LA
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
Clave 8968-66

**Diseño de un controlador MPC para un sistema
hidráulico de dos tanques conectados en
cascada.**

TESIS

QUE PARA OBTENER EL TÍTULO DE

**INGENIERO EN TELECOMUNICACIONES, SISTEMAS Y
ELECTRÓNICA**

PRESENTA

JOSÉ MANUEL DE LA CRUZ PÉREZ

ASESOR

M. en C. JESÚS NAMIGTLE JIMÉNEZ



Ixtlahuaca, Edo. de México, 2023



Universidad Nacional
Autónoma de México



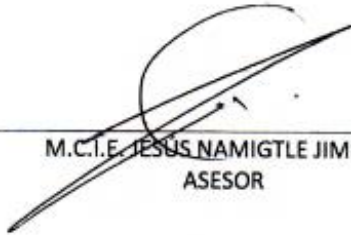
UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.


REVISORES



M.C.I.E. JESÚS NAMIGTLE JIMÉNEZ
ASESOR



M. EN A.T.I. KARINA BALDERAS PÉREZ
REVISOR



ING. E. JOSÉ JESÚS RAMÍREZ
MORQUECHO
REVISOR



DR. EN C. LUIS ANTONIO MIER QUIROGA



M.C.I.E. JORGE EDMUNDO MASTACHE
MASTACHE
REVISOR

TABLA DE CONTENIDOS

AGRADECIMIENTOS

RESUMEN

INDICE DE ILUSTRACIONES Y TABLA(S)

SÍMBOLOS Y ABREVIACIONES

1	PLANTEAMIENTO DEL PROBLEMA.....	11
1.1	Introducción.....	11
1.2	Problemática	12
1.3	Propuesta de solución.....	13
1.4	Justificación.....	14
1.5	Antecedentes	15
1.6	Hipótesis	17
1.7	Objetivos	17
1.7.1	Objetivo general.....	17
1.7.2	Objetivos específicos.....	17
1.8	Metodología	18
2	MARCO TEÓRICO	20
2.1	Sistema de control.....	20
2.1.1	Variables de los sistemas de control	20
2.2	Clasificación de los sistemas	21
2.2.1	Sistema de control lineal.....	21
2.2.2	Sistema de control no lineal.....	22
2.2.3	Sistema de control multivariable	23
2.2.4	Sistema en lazo abierto	23
2.2.5	Sistema en lazo cerrado	24
2.3	Requerimientos de un sistema de control	24
2.3.1	Estabilidad	24
2.3.2	Error o señal actuadora	25
2.3.3	Controlabilidad.....	25
2.3.4	Observabilidad.....	26
2.4	Formulación de un sistema de dos tanques.....	26
2.5	Control clásico.....	28
2.5.1	Controlador PID	28

2.6	Control Moderno	29
2.6.1	Control Predictivo por Modelo.....	29
2.6.1.1	Estrategia del control predictivo	29
2.6.1.2	Modelo en espacio de estados.....	30
2.6.1.2.1	Sistema de entrada única y salida única (SISO)	31
2.6.1.3	Diseño del modelo aumentado.....	32
2.6.1.4	Predicción de variables de estado y salida.....	33
2.7	Identificación de sistemas mediante Matlab®	34
2.7.1	Modelo de un sistema.....	35
2.7.1.1	Clasificación de modelos.....	35
2.7.2	Proceso de identificación.....	36
2.7.3	Métodos de identificación	37
3	DISEÑO Y SIMULACIÓN DEL CONTROLADOR.....	38
3.1	Diseño del prototipo	38
3.1.1	Bomba sumergible.....	40
3.1.2	Acondicionamiento del sensor de presión diferencial	41
3.2	Modelo de la planta.....	43
3.3	Verificación del sistema controlable y observable.....	55
3.4	Discretización del sistema y simulación.....	58
3.5	Diseño del modelo aumentado.....	64
4	RESULTADOS	66
4.1	Acoplamiento del sistema hidráulico.....	66
4.2	Análisis de resultados.....	67
4.2.1	Sintonización del controlador PID en Matlab®	71
4.2.2	Simulación del controlador PID.....	73
4.2.3	Desempeño del controlador MPC vs PID	75
5	CONCLUSIONES	77
5.1	Recomendaciones.....	77

REFERENCIAS

Anexo A-1 Especificaciones técnicas de sensor de presión MPX5010DP

Anexo A-2 Configuración del sensor de presión diferencial MPX5010DP

Anexo A-3 Código del controlador MPC a través de Matlab®

AGRADECIMIENTOS

En primera instancia, doy gracias a Dios por brindarme salud y fuerza para el logro de mis metas quien con su bendición me permitió tener tan buena experiencia en mi trayectoria académica.

Mi agradecimiento también va dirigido a mis padres por haberme forjado como la persona que soy en la actualidad. La mayoría de mis logros se los debo a ustedes incluyendo este proyecto de tesis. El apoyo de ustedes ha sido incondicional que no hay manera de agradecer el sacrificio que realizan día a día. Soy tan afortunado de tenerlos porque me han guiado a través de buenos ejemplos me formaron con reglas y algunas libertades, pero al final de cuentas, me motivaron constantemente para alcanzar mis anhelos. Agradezco a mi familia por esperar pacientemente y comprender que lo que estaba realizando es un proyecto del cual, tarde o temprano se obtendría su recompensa.

Así mismo agradezco a mis hermanos por la paciencia que me brindaron, por esos momentos de frustración en los que la situación se tornaba difícil, gracias por su apoyo y cariño durante esta etapa, espero que algún día me convierta en su ejemplo para que continúen con sus metas.

Al mismo tiempo quiero agradecer sinceramente a mi asesor de tesis M. en C. Jesús Namigtle Jiménez, su esfuerzo y dedicación. Sus conocimientos fueron parte fundamental para la realización de este proyecto, ha inculcado en mí un sentido de seriedad, responsabilidad y rigor académico.

Agradezco a mis docentes, personas con un gran conocimiento, quienes se esforzaron por ayudarme, sencillo no ha sido el proceso, pero gracias a su dedicación y por transmitir su sabiduría que los ha regido he logrado importantes objetivos.

RESUMEN

El control en la automatización se ha desempeñado de manera favorable e impactado en varios ámbitos como por ejemplo en la industria de procesos, y un gran avance en el hogar como es, el control de temperatura, control de presión, control de humedad, gestión de energía, sin olvidar el desarrollo en el ámbito académico donde es fundamental el diseño del control a través de la complejidad matemática.

Es importante narrar que existen diferentes controladores cada uno con distintas ventajas como son: la reducción del costo de mano de obra al automatizar procesos en los que interviene el ser humano, también hay un mejor monitoreo de los procesos, lo que hace que la detección de fallas se realice rápidamente, pero como se menciona anteriormente cada modelo contiene cierta complejidad para cumplir con los requerimientos en el diseño del controlador.

En este proyecto se diseña una técnica de control avanzado denominado "Control Predictivo por Modelo" (MPC), para un sistema hidráulico de dos tanques acoplados en cascada describiendo cada una de sus etapas, desde la construcción del prototipo (planta) de los tanques acoplados, así como el modelado del sistema a través del uso computacional en Matlab®, se desarrolla el algoritmo de control que consiste en la aplicación de ecuaciones y soluciones matemáticas, proponiendo diferentes valores para obtener la respuesta del sistema más óptima. Se concluirá con la programación del código en Matlab® para simular el comportamiento del controlador y realizar una comparación con un controlador PID con la finalidad de evaluar ventajas y desventajas que presente cada uno.

INDICE DE ILUSTRACIONES Y TABLA(S)

- Ilustración 1: Representación de un sistema de control.
- Ilustración 2: Representación gráfica de un sistema lineal.
- Ilustración 3: Representación gráfica de un sistema no lineal.
- Ilustración 4: Esquema de un sistema MIMO.
- Ilustración 5: Diagrama en bloques de un sistema en lazo abierto.
- Ilustración 6: Diagrama en bloques de un sistema en lazo cerrado.
- Ilustración 7: Gráfico de un sistema de control estable.
- Ilustración 8: Gráfico de un sistema de control inestable.
- Ilustración 9: Sistema de nivel de líquido.
- Ilustración 10: Sistema de dos tanques conectados en cascada.
- Ilustración 11: Esquema gráfico del comportamiento de un controlador MPC.
- Ilustración 12: Representación en bloques del sistema en espacio de estados.
- Ilustración 13: Sistema de adquisición de datos.
- Ilustración 14: Proceso de identificación.
- Ilustración 15: Representación en bloques del sistema hidráulico.
- Ilustración 16: Diseño del sistema hidráulico.
- Ilustración 17: Ensamble de la bomba sumergible.
- Ilustración 18: Diagrama de conexión bomba sumergible.
- Ilustración 19: Distribución de pines.
- Ilustración 20: Circuito nivel de tanque.
- Ilustración 21: Representación en bloques del sistema en lazo abierto.
- Ilustración 22: Comportamiento del nivel del sistema hidráulico.
- Ilustración 23: Exportación de datos de Excel-Matlab®.
- Ilustración 24: Carga de datos en Matlab®.
- Ilustración 25: Identificador de sistemas de Matlab®.
- Ilustración 26: Carga de datos en el identificador de sistemas.
- Ilustración 27: Formulario de import data.

Ilustración 28: Datos almacenados en Data Views.

Ilustración 29: Gráficos de entrada (u_1) vs salida (y_1).

Ilustración 30: Identificación de sistemas en Matlab®.

Ilustración 31: Rango de datos para entrenamiento.

Ilustración 32: Rango de datos para validación.

Ilustración 33: Carga de datos de entrenamiento y validación.

Ilustración 34: Gráfica con datos de entrenamiento y validación

Ilustración 35: Selección para estimar la función de transferencia.

Ilustración 36: Elección de polos y zeros.

Ilustración 37: Progreso de la estimación del modelo.

Ilustración 38: Generación del modelo.

Ilustración 39: Gráfica del sistema.

Ilustración 40: Interfaz de Simulink.

Ilustración 41: Creación de nuevo archivo.

Ilustración 42: Entorno de trabajo de Simulink.

Ilustración 43: Selección del bloque de entrada.

Ilustración 44: Asignación de parámetros a la entrada escalón del sistema.

Ilustración 45: Selección de espacio de estados en tiempo continuo.

Ilustración 46: Asignación matricial al bloque de espacio de estados.

Ilustración 47: Selección de espacio de estados en tiempo discreto.

Ilustración 48: Asignación matricial al bloque de espacio de estados.

Ilustración 49: Representación en bloques del sistema.

Ilustración 50: Comportamiento del sistema en tiempo continuo y tiempo discreto.

Ilustración 51: Respuesta del sistema con tiempo de simulación aumentado.

Ilustración 52: Armado del sistema hidráulico.

Ilustración 53: Controlador MPC con H_p cortos

Ilustración 54: Respuesta del controlador con $H_c=2$, $H_p=20,25,30$.

Ilustración 55: Respuesta del controlador con $H_c=20$, $H_p=20,25,30$.

Ilustración 56: Respuesta del controlador con $H_c=2$, $H_p=20,25,50$.

Ilustración 57: Ingreso a la herramienta pidTuner.

Ilustración 58: Entorno de la herramienta pidTuner

Ilustración 59: Importación de la planta.

Ilustración 60: Selección del Controlador PID.

Ilustración 61: Parámetros del controlador PID

Ilustración 62: Asignación de parámetros PID en Simulink.

Ilustración 63: Simulación del controlador PID.

Ilustración 64: Sintonización del controlador PID

Ilustración 65: Control PID vs MPC con referencia a 5 cm altura.

Ilustración 66: Control PID vs MPC con referencia a 5 cm altura (tiempo establecimiento PID).

Ilustración 67: Señal de error PID vs MPC

Tabla no.1: Respuesta transitoria de controlador PID vs MPC.

SIMBOLOS Y ABREVIACIONES

MPC	Controlador Predictivo por Modelo
PID	Proporcional Integral Derivativo
FPGA	Field Programmable Gate Array (Arreglo de compuertas lógicas programables)
FT	Función de Transferencia
Hc	Horizonte de control
Hp	Horizonte de predicción
L	Inductor
R	Resistencia
C	Capacitor
Pc	Matriz de controlabilidad
Po	Matriz de observabilidad
Vin	Voltaje de entrada
i	Corriente eléctrica
SISO	Única Entrada Única Salida
MIMO	Múltiples entradas Múltiples salidas
Kp	Ganancia proporcional
u(t)	Salida del controlador
e(t)	Señal de error
IMC	Control por Modelo Interno
D/A	Adquisición de Datos
V	Voltaje

1 PLANTEAMIENTO DEL PROBLEMA

1.1 Introducción

En el área de control y automatización se desarrollan distintas habilidades en el estudiante como son el diseño y modelado de sistemas a través de la ingeniería de control. Estas competencias permiten desarrollar técnicas complejas para aplicarlos al control moderno.

El control automático ha beneficiado eficazmente en la industria ya sea en el llenado de sustancias en tanques, control de motores para mejorar los procesos de producción, así como, un gran avance en el hogar, por ejemplo, control de presión, temperatura, humedad, gestión de energía, en donde comúnmente se trabaja con técnicas de control convencionales como Proporcional Integral Derivativa (PID) pero estas pueden quedar limitadas al momento de utilizarlas en otros sistemas más complejos.

La competencia que existe en la industria da origen a optimizar el control, y también a implementar nuevas estrategias de control avanzado que cumplan con los requisitos para ayudar en donde los otros tipos de control no han logrado solucionar técnicas como: control difuso, control predictivo, redes neuronales, algoritmos genéticos, etc.

Existe una técnica de control avanzado denominado Control Predictivo por Modelo (MPC) muy útil para la aplicación de sistemas con comportamiento no lineal como una planta de doble tanque acoplado, ya que predice respuestas futuras previamente calculadas por un modelo y horizonte de predicción.

El MPC es un tipo de control del cual se han implementado muchas aplicaciones, logrando gran aceptación tanto en la industria como en el ámbito académico. Actualmente varias aplicaciones de controladores predictivos están funcionando con éxito, en la industria de procesos y la robótica. (Salcedo & Correa, 2008)

El diseño de los controladores predictivos se basa en modelos matemáticos dinámicos del sistema a controlar, debido a esto, es importante tener una adecuada recopilación de datos del sistema experimental.

La diferencia entre las técnicas de los nuevos sistemas de control está en la complejidad matemática, quiere decir que cada vez más es importante incluir las especificaciones matemáticas en el diseño del controlador. El análisis de un estudio relacionado con las técnicas de control y desarrollo realizado por (Haruo, Toshiaki, & Mituhiko, 1998). Hacia la “Society of Instrumentation and Control Engineering” a 15 plantas en temas de control se determina que el MPC presenta importantes beneficios sobre otros. Es por esta razón que se decide diseñar un controlador MPC en el sistema hidráulico de dos tanques conectados en cascada.

1.2 Problemática

La implementación de procesos industriales necesita la puesta en marcha del control automático para lograr procesos de alta calidad, distintos procedimientos que requieren de la medición y el control de varios factores físicos. Es indispensable que por lo menos una variable del proceso cumpla con una referencia establecida y permanezca constante para hacer cumplir el mantenimiento y el control de estas variables.

En consecuencia, a este problema, se ha decidido diseñar el control de nivel que maneje circunstancias de procesos industriales, es decir, un proceso de tanques acoplados para un llenado, almacenamiento y distribución de algún fluido de manera adecuada.

El campo de la automatización es innovador modernizar las estrategias de control tradicionales con estrategias de control avanzado que permiten un mejor manejo del controlador.

1.3 Propuesta de solución

El desarrollo de este proyecto consiste en la construcción de un sistema muy conocido e implementado en el sector industrial, como es el de tanques acoplados. Este sistema está compuesto por dos tanques acoplados en cascada y un tanque de depósito, en donde el primer tanque es suministrado por algún fluido a través de una bomba sumergible, con el objetivo de nivelar el fluido con un controlador MPC.

Para lograr diseñar el control predictivo, es necesario formular el modelo matemático del sistema. Existen dos formas de obtener este modelo, una de ellas es a través de aplicación de leyes, teoremas y otra es mediante la técnica de identificación de sistemas en Matlab®. La identificación de sistemas es un método que se basa en la observación, que presenta el sistema en lazo abierto de manera experimental.

En la identificación de sistemas es fundamental la recopilación de los datos, para ello se utilizan tarjetas de adquisición de datos, como, por ejemplo, tarjeta Arduino, FPGA's. En este proyecto se propone utilizar Arduino Uno, debido a la precisión que este nos brinda en la obtención de los datos.

Cuando se obtienen los datos de la respuesta del sistema, estos son sometidos a valores de entrenamiento y validación en el dominio del tiempo, con la finalidad de proporcionarnos una función de transferencia.

Por otra parte, el control predictivo requiere de un algoritmo para ser implementado, existen diferentes tipos de control, como son: Control por Matriz Dinámica, Control Predictivo Funcional, Control Auto adaptativo de Predicción Extendida, Control Adaptativo de Horizonte Extendido, Control Predictivo Generalizado.

Nos enfocaremos en el estudio del Control Predictivo por Modelo, que consiste en obtener el modelo de la planta (función de transferencia tiempo continuo), discretizar la función con cierto tiempo de muestreo, convertir a modelo de estados, aplicar el horizonte de control y un horizonte de predicción para optimizar el controlador.

1.4 Justificación

Llevar a cabo el diseño de un controlador para el nivel de líquido en los tanques y el flujo entre ellos es un problema fundamental en los procesos industriales, ya que es una de las herramientas más utilizadas para la optimización de procesos.

Por esta razón se implementa una estrategia de control avanzado, para el sistema de dos tanques acoplados, en donde el principal desarrollo es el Controlador Predictivo por Modelo (MPC), se decidió utilizar este control, por qué, a diferencia de otros tipos, el MPC presenta ciertas ventajas como:

- Formulación en el dominio del tiempo.
- Permite ser estudiado con sistemas lineales, no lineales, multivariados.
- Brinda la posibilidad de incorporar restricciones.
- Permite controlar procesos con comportamiento dinámico.
- Tiene la capacidad de anticipar perturbaciones.

por lo tanto, se convierte en una técnica que tiene la capacidad de predecir el valor de las variables controladas a lo largo de un horizonte de predicción y de esta manera establecer su comportamiento en una referencia dada.

Un análisis entre la comparación de un controlador MPC y PID en una planta de temperatura por (Hernández, Díaz, & Pinzón, 2014), demuestra que el tiempo de establecimiento del control PID es de 2000 segundos, y del MPC es de 1000 segundos, tomando un 2% del set point, también se observa que el control MPC tiene mejor comportamiento dinámico, con un tiempo pico menor que el control PID, evidenciando la eficiencia del control predictivo.

1.5 Antecedentes

Es muy común implementar el sistema de dos o más tanques, al contar con este tipo de plantas se requiere tomar el control mediante alguna técnica que ejecute el trabajo en el nivel de alguna sustancia, así como el comportamiento del caudal e incluso la medición de la altura en los tanques.

Diferentes universidades han implementado configuraciones de tanques, un ejemplo de ello es en el año 2017 en Bogotá, D.C. la Universidad Católica de Colombia por medio de la facultad de Ingeniería presentaron un “Sistema didáctico para el control de nivel con tanques acoplados”, los autores de este trabajo son Karol Bibiana Bohorquez Valderrama, Diego Fernando Fonseca Sanabria, Santiago Gutiérrez Puertas.

El trabajo consiste en un reservorio que almacena el agua y con ayuda de una bomba sumergible transporta el líquido hacia el tanque principal, lógicamente este se encuentra acoplado con el segundo tanque. Para la medición de nivel en cada tanque emplean sensores de ultrasonido con el objetivo de calcular la distancia a la que se encuentra el líquido. (Bohorquez, Fonseca, & Gutiérrez, 2017)

La distinción que hay entre el trabajo antes mencionado con el presente proyecto, es el algoritmo de control MPC y en cuanto a la medición de altura utilizan sensores de ultrasonido, lo que respecta al presente proyecto es el uso de un sensor de presión diferencial.

También en el año 2015 en Ensenada, Baja California, México mediante el Congreso Nacional de Control Automático se presenta un “Control Predictivo Distribuido Óptimo Aplicado al Control de Nivel de un Proceso de Cuatro Tanques Acoplados”.

En el trabajo antes mencionado utilizan un sistema más complejo por que involucra el acoplamiento de 4 tanques, el algoritmo de control predictivo que utilizan (OMPC) va de acuerdo a la adaptación de su planta debido a la partición de un sistema en “n” subsistemas para la optimización de manera distribuida en las señales de control mediante una red LAN. (Sorcia Vázquez, y otros, 2015)

Realmente este trabajo es diferente al presente proyecto, debido al diseño del algoritmo y el acoplamiento de 4 tanques.

Para el año 2016 en la Universidad de Piura, Peter Saúl Gutarra Castillo desarrolló la “Modelación y control de un sistema de cuatro tanques acoplados”.

Su trabajo consistía en la implementación de equipos electrónicos de adquisición de datos mediante protocolos de comunicación ModBus. El desarrollo de su modelo matemático fue a través de leyes físicas y el algoritmo de control PI y MPC fue mediante Matlab®. (Gutarra, 2016)

Es importante señalar que en el trabajo antes mencionado se emplea un algoritmo de control diferente al presentado en este proyecto y la arquitectura de los tanques acoplados.

También existe el “Desarrollo de un sistema embebido de control de nivel predictivo basado en modelo aplicado a un módulo de tanques acoplados” por Eyder Jesús Revilla Cosi, en la Universidad Nacional de San Agustín de Arequipa en el año 2019 en Perú.

Su trabajo consiste en el acoplamiento de tres tanques, para la distribución de agua utiliza una bomba centrífuga, para la medición del nivel en el último tanque se ocupa un sensor de presión diferencial en cuanto a la adquisición de datos utilizan la NI myRIO 1900 por National Instruments. (Revilla, 2019)

La implementación del controlador se lleva a cabo con Matlab® en donde requieren de técnicas como horizonte de control, horizonte de predicción y el intervalo de control, los cuales están ligados a la dinámica del proceso que se quiere controlar. (Revilla, 2019)

El proyecto anterior a diferencia del presente proyecto utiliza diferentes técnicas de control, cada uno cuenta con su complejidad matemática para obtener con mayor precisión el diseño del controlador.

1.6 Hipótesis

Será posible diseñar un controlador predictivo por modelo (MPC) para medir el nivel de líquido en un sistema hidráulico de dos tanques conectados en cascada.

1.7 Objetivos

1.7.1 Objetivo general

Diseñar y evaluar un controlador MPC para el nivel en un sistema hidráulico de dos tanques conectados en cascada.

1.7.2 Objetivos específicos

- Construir e implementar el sistema experimental.
- Modelar a través de identificación de sistemas.
- Diseñar el controlador predictivo por modelo
- Simular y evaluar el controlador MPC con distintos horizontes de predicción y control.
- Comparar la respuesta del controlador MPC vs PID

1.8 Metodología

En este proyecto se desarrolla el MPC, para un sistema de dos tanques acoplados, este trabajo se orienta a un estudio de investigación. Cada una de las fases contiene una descripción clara para su desarrollo general.

FASE 1: Análisis sobre temáticas de control, a través de varias fuentes de información se realizó la investigación a los mejores candidatos en controladores para ser diseñado en relación a un sistema de dos tanques acoplados.

FASE 2: Construcción del prototipo, está enfocado en el armado de dos tanques con material de acrílico transparente interconectados en cascada, además de contar con un reservorio en donde se almacena el fluido para retroalimentar la entrada del primer tanque con una bomba sumergible. Estos tanques cuentan con un pequeño grifo que simulan a las válvulas de salida en cada tanque

FASE 3: El modelo matemático de la planta se obtiene con el método de identificación de sistemas en Matlab®. En donde se realiza la adquisición de datos con una tarjeta “Arduino® Uno”, que consiste en dejar trabajando el prototipo a manera de lazo abierto hasta obtener una estabilidad de nivel en ambos tanques.

Cuando los datos son leídos por la tarjeta Arduino®, estos serán extraídos en un archivo .XSL, para poder ser exportados en Matlab®. De esta manera los datos son procesados al identificador de sistemas, la única condición para lograr esto, es que la mitad de los datos recopilados deben ser sometidos a valores de entrenamiento y la otra mitad a valores de validación en el dominio del tiempo, para obtener con mayor precisión la función de transferencia de la planta.

FASE 4: Una vez obtenida la función de transferencia en tiempo continuo se aplica el método para convertir la FT a matrices de estado, ya que es importante tener en cuenta que antes de diseñar un controlador primero hay que determinar si el sistema es controlable y observable.

FASE 5: Simular y evaluar el comportamiento de la planta, ya que se tiene la función de transferencia en matrices de estados utilizamos Simulink en el Toolbox que Matlab® nos ofrece, en esta herramienta se evalúa el comportamiento de la función de transferencia a través de una gráfica, para ello se le colocara una entrada step como excitación al sistema.

FASE 6: El siguiente paso es convertir la función de transferencia de tiempo continuo a tiempo discreto con un intervalo de muestreo de 1s. Posteriormente, se simula el comportamiento del sistema en tiempo continuo y tiempo discreto, a través, de Simulink-Matlab® en diagrama de bloques con la finalidad de igualar su respuesta y continuar con el diseño del MPC en tiempo discreto.

FASE 7: Continuando con el diseño del MPC se aplica el método por modelo aumentado a las matrices de estado en tiempo discreto, lo siguiente es proponer un

horizonte de control (H_c) y un horizonte de predicción (H_p) para lograr obtener el modelo de predicción (matriz "F" y " Φ ").

FASE 8: Por último, se hace la recopilación de los cálculos obtenidos para programar los parámetros en el software de Matlab®, así mismo, se estiman diferentes referencias en el controlador para analizar la respuesta del sistema, en una comparación entre el MPC vs PID para determinar el comportamiento de cada controlador.

2 MARCO TEÓRICO

2.1 Sistema de control

La función de un sistema de control es mantener una relación preestablecida entre la salida y la entrada de referencia, realizando una comparación y utilizando la diferencia como medio de control. (S. Norman, 2006)

El objetivo de mantener valores establecidos se debe conseguir, a través del manejo de variables de control en el dominio del tiempo, para ello, se emplea un sistema de control que cuenta con elementos que influyen en el funcionamiento del mismo como se observa en la ilustración 1:

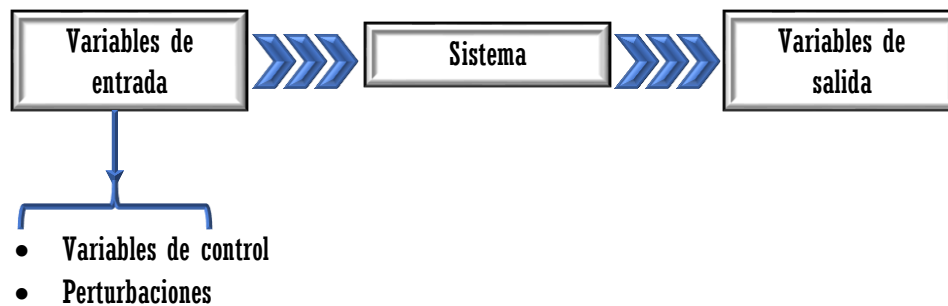


Ilustración 1: Representación de un sistema de control. Fuente: Elaboración propia

2.1.1 Variables de los sistemas de control

- Variable controlada: Es el tipo de variable que se encarga de medir y se controla. La variable controlada es la salida del sistema, (temperatura, velocidad flujo, nivel, presión, voltaje, corriente, etc.)
- Variable manipulada: Es aquella que se modifica por el controlador, a fin de afectar la variable controlada, (corriente para controlar la velocidad de un motor, presión para abrir o cerrar una válvula).
- Planta: Es la composición de objetos físicos que ha de ser controlado (horno de calentamiento, reactor químico, vehículo espacial, tanques de almacenamiento etc.)
- Perturbación o ruido: Es el tipo de señal que tiende a afectar el valor de la salida de un sistema. Las perturbaciones pueden ser internas o externas.
- Función de transferencia: La función de transferencia de un sistema se entiende como la relación entre la transformada de Laplace de la salida

(respuesta) y la transformada de Laplace de la entrada (excitación), en el entendido de que todas las condiciones iniciales son cero.

- Diagrama de bloques: Un diagrama de bloques es una representación gráfica de las funciones realizadas por cada componente del sistema, así como la descripción del flujo de las señales. A diferencia de una representación matemática, un diagrama de bloques tiene la ventaja de indicar en forma más realista el flujo de señales del sistema real.
- Set-Point: El Set-Point o punto de ajuste es el valor de la variable que se programa en el controlador para fijar la referencia donde se desea establecer la variable controlada. (Richard D., Robert B.,2020)

2.2 Clasificación de los sistemas

2.2.1 Sistema de control lineal

En un sistema lineal, la salida debe seguir el mismo comportamiento de la entrada, incluyendo cambios o modificaciones presentadas. Existen casos en lo que esto no se cumple y es complicado clasificarlos como sistemas lineales. (K, Ogata 2003)

Un ejemplo de un sistema lineal se muestra en la ilustración 2.

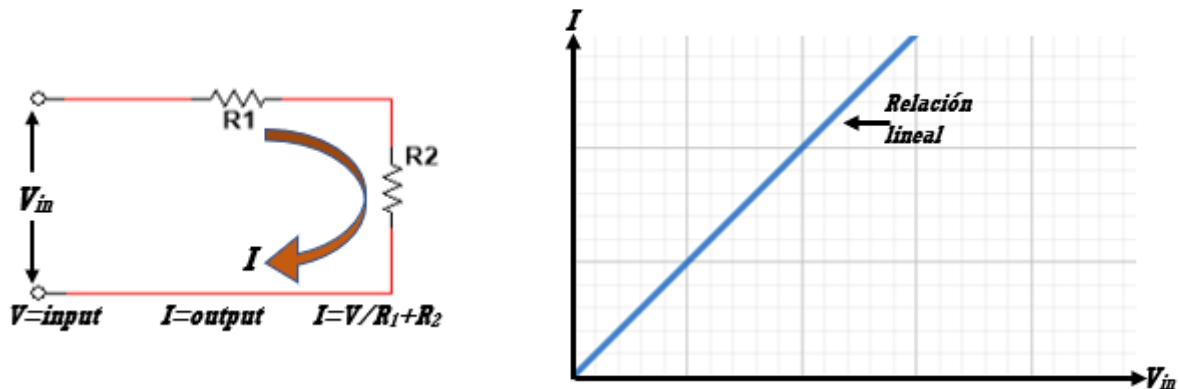


Ilustración 2: Representación gráfica de un sistema lineal. Fuente: Elaboración propia

2.2.2 Sistema de control no lineal

Son aquellos sistemas que cambian constantemente en el tiempo y que no siempre en su salida obtienen un único resultado, presentan mayor velocidad y precisión cuando se implementan de manera sencilla.

La ingeniería de control se encarga de estudiar el comportamiento de los sistemas dinámicos con entradas y sus respectivas modificaciones en las salidas a través del uso de la retroalimentación. (K, Ogata 2003)

Para entender mejor el concepto anterior se muestra un ejemplo en la siguiente ilustración.

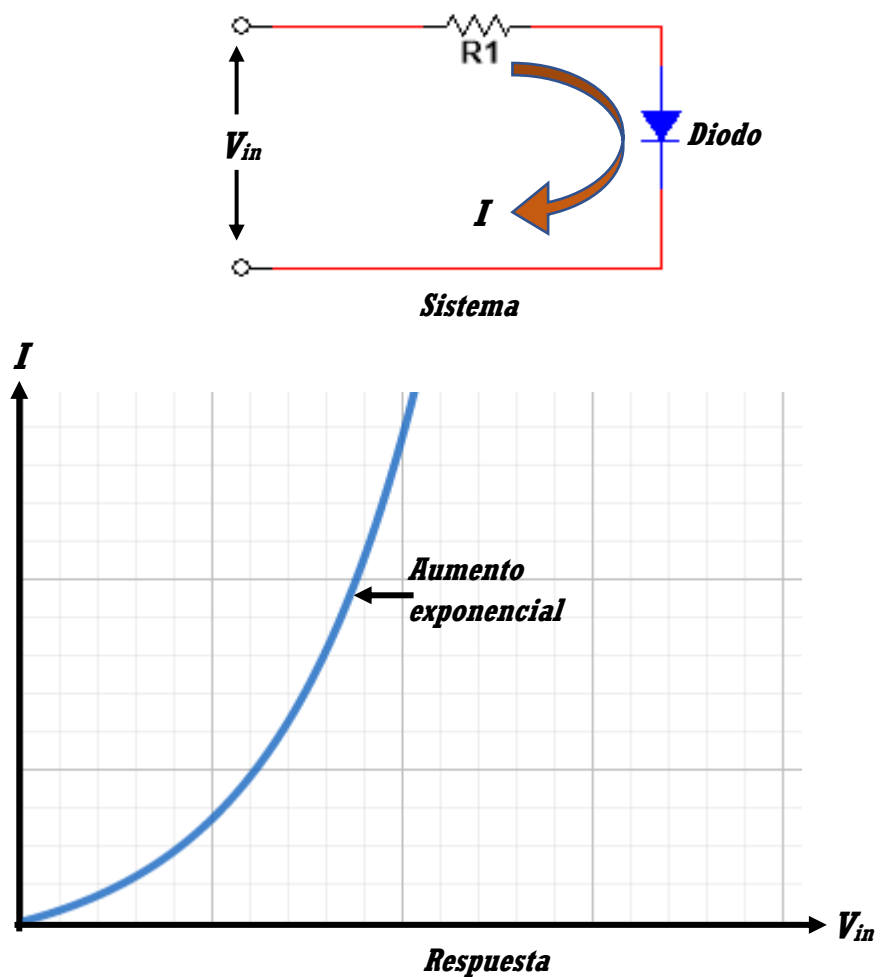


Ilustración 3: Representación gráfica de un sistema no lineal. Fuente: Elaboración propia

2.2.3 Sistema de control multivariable

Son los sistemas con múltiples variables de entrada o múltiples variables de salida son clasificados como sistemas de control multivariable. También conocidos como sistemas MIMO (multiple-input multiple-output). Una representación de este sistema se observa en la ilustración 4. (R. Gaviño, 2010).

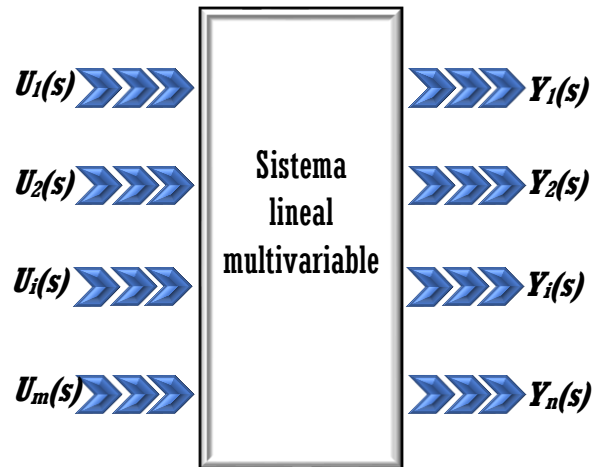


Ilustración 4: Esquema de un sistema MIMO. Fuente: Elaboración propia

2.2.4 Sistema en lazo abierto

Este sistema es empleado en la práctica solo si se conoce la relación entre la entrada y la salida, y no se presentan perturbaciones ni internas ni externas como lo muestra la ilustración 5. (R. Gaviño, 2010).

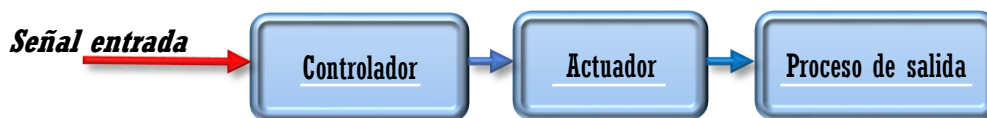


Ilustración 5: Diagrama en bloques de un sistema en lazo abierto. Fuente: Elaboración propia

Los elementos que lo integran son:

- Señal de entrada: Señal que excita el sistema de control.
- Controlador: Encargado de controlar el proceso.
- Actuador: Dispositivos que realizan el proceso.
- Salida: Señal que representa el comportamiento del sistema.

2.2.5 Sistema en lazo cerrado

En este sistema influye el uso de una acción de control de retroalimentación para disminuir el error del sistema. La retroalimentación es el resultado de la diferencia entre la señal de entrada y la salida, con el objetivo de reducir el error y lograr que la salida del sistema permanezca estable en dicha referencia. (ilustración 6).

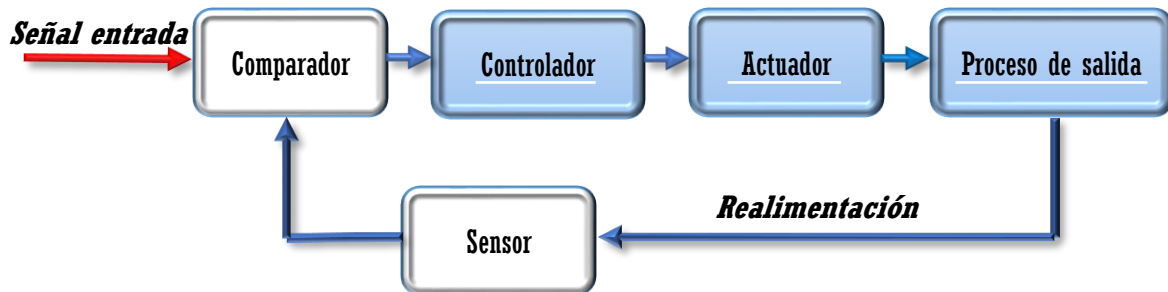


Ilustración 6: Diagrama en bloques de un sistema en lazo cerrado. Fuente: Elaboración propia

Los elementos que lo integran son:

- Señal de entrada: Señal que excita al sistema.
- Comparador: Encargado de comparar la señal de referencia con la señal medida en la salida.
- Controlador: Encargado de controlar el proceso.
- Actuador: Dispositivo mecánico encargado de realizar la operación del proceso.
- Sensor: Dispositivo encargado de medir la señal de salida para realimentarla y compararla con la señal de referencia.
- Perturbaciones: Señales no deseadas que afectan al funcionamiento del proceso.

2.3 Requerimientos de un sistema de control

2.3.1 Estabilidad

El comportamiento a una señal debe alcanzar y mantener un valor constante durante un periodo de tiempo, en el que el sistema debe tener la característica de ser estable, pues un sistema de control inestable podría producir oscilaciones inestables de gran amplitud o que esta señal tome valores que corresponden a los límites extremos, por lo tanto, una respuesta inestable es una opción indeseada, en las siguientes ilustraciones se muestra un gráfico con la definición anterior. (S. Norman, 2006)

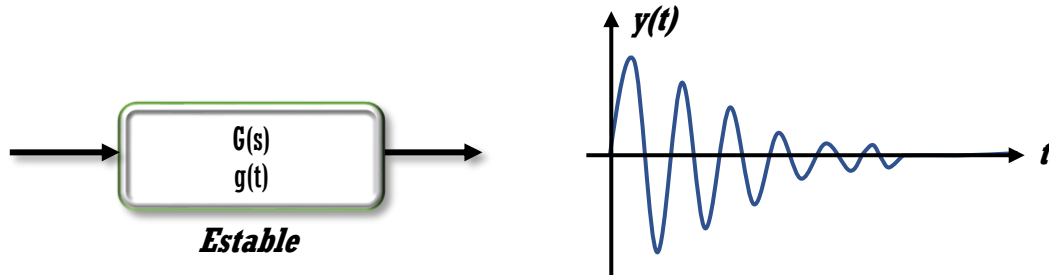


Ilustración 7: Grafico de un sistema de control estable. Fuente: Elaboración propia

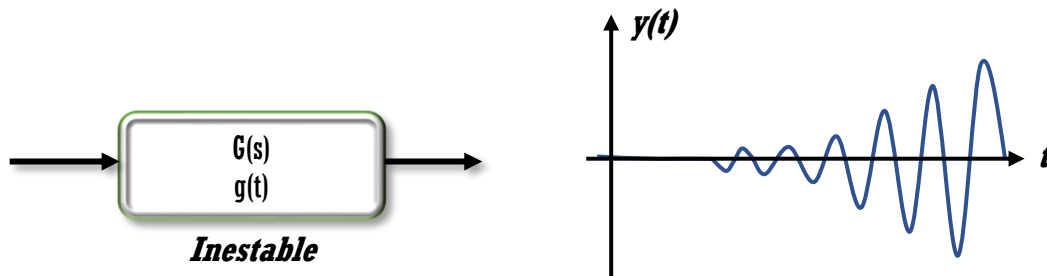


Ilustración 8: Grafico de un sistema de control inestable. Fuente: Elaboración propia

2.3.2 Error o señal actuadora

En todos los sistemas de control se presentan mínimos errores, con ayuda de la ingeniería de control se puede reducir ese error a casi cero, ya que siempre estarán presentes perturbaciones a los componentes del sistema. Sin embargo, un sistema de control puede ser exacto dentro de ciertos rangos, donde se es capaz de reducir cualquier error a un límite razonable. (S. Norman, 2006)

2.3.3 Controlabilidad

El concepto de controlabilidad describe la interacción que hay entre las entradas y las variables de estado de un sistema, es decir, señala si el comportamiento de un sistema puede ser controlado mediante sus entradas. Si todos los estados son controlables, entonces el sistema es completamente controlable

Por lo tanto, un sistema es controlable si manipulando una entrada es posible llevar un estado de una condición inicial a otra condición final en un tiempo finito, sin importar la trayectoria seguida. (K, Ogata 2003)

La matriz de controlabilidad es:

$$P_c = [B \ AB \ A^2B \ \dots \ A^{n-1}B] \quad [1]$$

2.3.4 Observabilidad

La observabilidad de un sistema explica la interacción que existe entre las salidas y las variables internas de un sistema, es decir, nos señala si el comportamiento interno de un sistema puede ser revelado mediante las variables de salida.

Por lo tanto, un sistema es observable si el vector de estados se puede determinar a partir del vector de salidas en un tiempo finito. (K, Ogata 2003)

La ecuación 2 muestra la matriz de observabilidad:

$$P_o = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \quad [2]$$

2.4 Formulación de un sistema de dos tanques

Los sistemas dinámicos hidráulicos comprenden fuerzas y movimientos por medio de líquidos. Los fundamentos en este tipo de sistemas son:

Principio de pascal: se basa en la presión que ejerce un fluido incompresible y en equilibrio dentro de un recipiente de paredes indeformables se transmite con igual intensidad en todas las direcciones y en todos los puntos del fluido.

En la ilustración 9 se observa un sistema de nivel de líquido en donde se consideran factores como son:

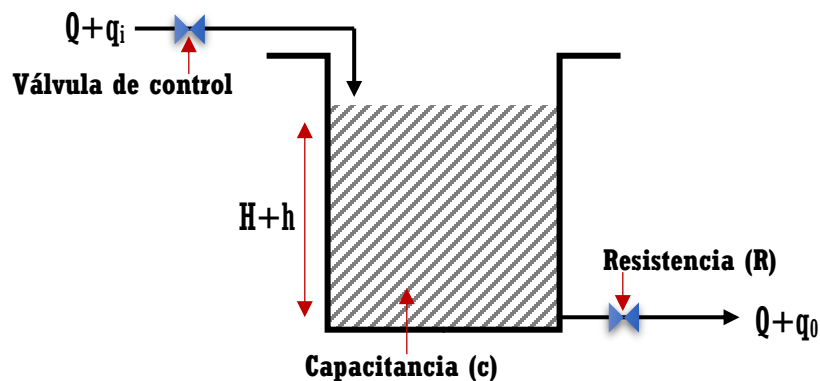


Ilustración 9: Sistema de nivel de líquido. Fuente: Elaboración propia

En la teoría de control moderno aparece la representación en variables de estado o espacio de estados.

Donde el sistema de tanques los podemos representar, basándonos en las ecuaciones diferenciales de primer orden que relacionan flujo y nivel en cada uno de los tanques, de la forma:

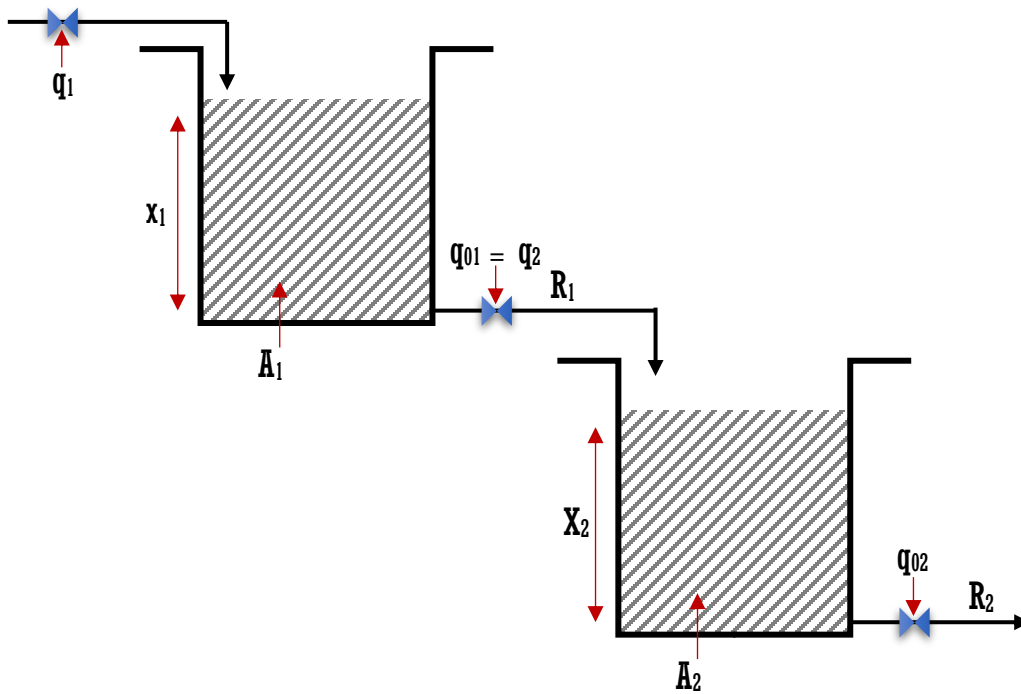


Ilustración 10: Sistema de dos tanques conectados en cascada. Fuente: Elaboración propia

$$\text{del primer tanque: } \frac{dx_1(t)}{dt} = \frac{1}{A_1} q_1(t) - \frac{1}{A_1 R_1} x(t)$$

$$\text{del segundo tanque: } \frac{dx_2(t)}{dt} = \frac{1}{A_2 R_1} x_1(t) - \frac{1}{A_2 R_2} x_2(t) \quad [3]$$

2.5 Control clásico

2.5.1 Controlador PID

La conjunción de una acción de control Proporcional, una acción de control Integral y una acción de control Derivativo se denomina acción de control Proporcional Integral Derivativo. Esta acción combinada tiene las ventajas de cada una de las tres acciones de control individuales. (K, Ogata 2003)

La relación entre la salida del controlador $u(t)$ y la señal de error $e(t)$ de la acción de control Proporcional Integral Derivativo está dada por:

$$u(t) = K_p e(t) + K_p T_d \frac{d e(t)}{dt} + \frac{K_p}{T_i} \int_0^t e(t) \quad [4]$$

En donde K_p es la ganancia proporcional, T_i es el tiempo integral y T_d es el tiempo Derivativo.

Si se aplica la transformada de Laplace se obtiene:

$$U(s) = K_p E(s) + K_p T_d s E(s) + \frac{1}{T_i s} E(s) \quad [4.1]$$

$$\frac{U(s)}{E(s)} = K_p \left(1 + T_d s + \frac{1}{T_i s} \right) \quad [4.2]$$

Si se aplica la transformada Z se obtiene:

$$U(z) = K_p E(z) + \frac{T_d}{T} (1 - z^{-1}) E(z) - \frac{T}{2T_i} \frac{(1 + z^{-1})}{(1 - z^{-1})} E(z) \quad [4.3]$$

$$\frac{U(z)}{E(z)} = K_p + \frac{T_d}{T} (1 - z^{-1}) - \frac{T}{2T_i} \frac{(1 + z^{-1})}{(1 - z^{-1})} \quad [4.4]$$

$$\frac{U(z)}{E(z)} = K_p \left(1 - \frac{T}{2T_i} + \frac{T_d}{T} - \frac{T_d}{T} z^{-1} + \frac{T}{T_i} \frac{1}{(1 - z^{-1})} \right) \quad [4.5]$$

2.6 Control Moderno

La teoría de control moderno emplea en sus diferentes etapas para el diseño de los controladores, un amplio número de ciencias y herramientas tales como álgebra lineal, teoría de vectores y matrices, cálculo diferencial y programación.

2.6.1 Control Predictivo por Modelo

El control predictivo basado en modelo (Model Based Predictive Control – MBPC o simplemente MPC) es una metodología de control que hace uso del modelo del proceso para predecir las salidas futuras de la planta y con ello optimizar las acciones de control futuras.

El control MPC desempeña ciertas ventajas sobre otros métodos que a continuación se enuncian:

- El MPC se considera como un método avanzado de control de procesos que ha impactado conforme a las tecnologías computacionales en donde es necesario que utilice de un modelo dinámico para predecir las trayectorias de salida. (Diehl, 2001).
- Optimiza el intervalo de tiempo actual, sin olvidar los intervalos de tiempo futuro, es decir, se optimiza un horizonte finito, mientras que el intervalo de tiempo actual se implementa. Por lo tanto, el MPC tiene la capacidad de predecir el comportamiento futuro y puede implementar acciones de control en consecuencia (Orukpe, 2012).
- Es capaz de controlar múltiples entradas y múltiples salidas (sistemas MIMO) debido a su complejidad matemática.
- Presenta una buena respuesta ante sistemas con dinámica compleja tales como: sistemas inestables, sistemas con retardo.

2.6.1.1 Estrategia del control predictivo

La metodología de todos los controladores pertenecientes a la familia MPC se caracteriza por la siguiente estrategia, representada en la ilustración 11.

El control predictivo usa un modelo del proceso que se desea controlar llamado modelo de predicción, el cual se utiliza para predecir el comportamiento del sistema en el tiempo a lo largo de un horizonte, llamado horizonte de predicción (H_p). De esta manera la única salida del proceso en el caso de sistemas SISO converge a un valor de referencia. (L. Wang, 2019).

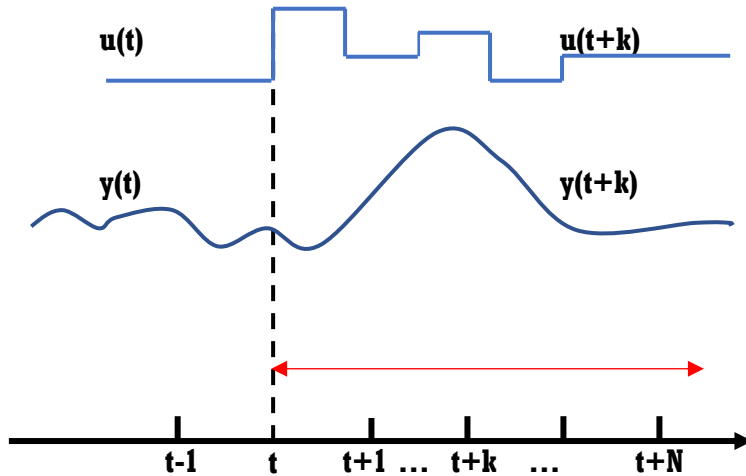


Ilustración 11: Esquema gráfico del comportamiento de un controlador MPC Fuente: Elaboración propia.

Las salidas futuras para un horizonte determinado N, llamado horizonte de predicciones predicen cada instante “t”.

2.6.1.2 Modelo en espacio de estados

Se considera que son ecuaciones diferenciales de primer orden usadas para modelar sistemas dinámicos.

De manera general, una ecuación diferencial de orden “n” puede ser descompuesta en “n” ecuaciones diferenciales de primer orden. Debido a que, en principio, las ecuaciones diferenciales de primer orden son más sencillas de resolver que las de orden superior, por lo tanto, se justifica este tipo de descomposición. (L. Wang, 2019).

En general un sistema puede representar como:

$$\dot{x}(t) = f(x, u, t) \quad [5]$$

$$y(t) = g(x, u, t) \quad [6]$$

Donde (ec. 5) es la ecuación de estado y la (ec. 6) es la ecuación de salida. Si los vectores de funciones “f” o “g” presentan el tiempo explícitamente, el sistema recibe el nombre de variante en el tiempo.

Ahora si las ecuaciones (5) y (6) son linealizadas alrededor del punto de operación o si el sistema es originalmente lineal se obtiene la siguiente ecuación de estado y ecuación de salida:

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad [7]$$

$$y(t) = C(t)x(t) + D(t)u(t) \quad [8]$$

Donde $A(t)$ es la matriz de estado, $B(t)$ es la matriz de entrada, $C(t)$ es la matriz de salida y $D(t)$ es la matriz de transmisión directa. Un diagrama de bloques representando las ecuaciones (7) y (8) se muestra en la ilustración 12. (L. Wang, 2019).

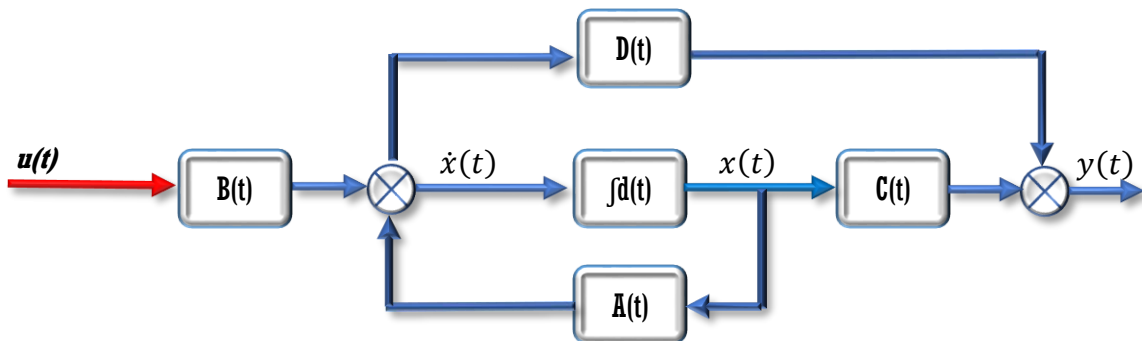


Ilustración 12: Representación en bloques del sistema en espacio de estados. Fuente: Elaboración propia

Considerando que el sistema también es invariante en el tiempo, las ecuaciones (7) y (8) pueden ser simplificadas quedando así el modelo en espacio de estados:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad [9]$$

$$y(t) = Cx(t) + Du(t) \quad [10]$$

2.6.1.2.1 Sistema de entrada única y salida única (SISO)

El estudio de sistemas con una única entrada y única salida comienza por tener descrita la siguiente forma:

$$x_m(k+1) = A_m x_m(k) + B_m u(k) \quad [11]$$

$$y(k) = C_m x_m(k) \quad [12]$$

Se tiene en cuenta que “ u ” es la variable manipulada o variable de entrada, “ y ” es la salida del proceso, y “ x_m ” es el vector de variables de estado. Este modelo de planta tiene “ $u(k)$ ” como entrada. Por lo tanto, es indispensable cambiar el modelo para que se adapte al diseño en el que está incrustado un integrador.

Es importante considerar que una formulación general de un modelo de espacio de estados tiene un término directo de la señal de entrada “ $u(k)$ ” a la salida “ $y(k)$ ” como:

$$y(k) = C_m x_m(k) + D_m u(k) \quad [13]$$

Sin embargo, debido al principio de control del horizonte, se requiere información actual de la planta para la predicción y el control, se asume implícitamente que la entrada “ $u(k)$ ” no puede afectar la salida “ $y(k)$ ” al mismo tiempo. Así, “ $D_m=0$ ” en el modelo de planta. (L. Wang, 2019).

Realizando una operación de diferencia en ambos lados se obtiene:

$$x_m(k + 1) - x_m(k) = A_m(x_m(k) - x_m(k - 1)) + B_m(u(k) - u(k - 1)) \quad [14]$$

Denotando la diferencia de la variable de estado por:

$$\Delta x_m(k + 1) = x_m(k + 1) - x_m(k); \quad \Delta x_m(k) = x_m(k) - x_m(k - 1) \quad [15]$$

y la diferencia de la variable de control por

$$\Delta u(k) = u(k) - u(k - 1) \quad [16]$$

Estos son los incrementos de las variables $x_m(k)$ y $u(k)$. Con esta transformación, la ecuación de espacio de estado es:

$$\Delta x_m(k + 1) = A_m \Delta x_m(k) + B_m \Delta u(k) \quad [17]$$

2.6.1.3 Diseño del modelo aumentado

Retomando la escritura del incremento de las variables (ec. 17), lo siguiente es la de incorporar “ $\Delta x_m(k)$ ” en la salida $y(k)$, también es necesario realizar un nuevo vector de estado que comprenda el cambio de la variable “ x_m ”. (ec. 18).

$$x(k) = [\Delta x_m(k) \ y(k)]^T \quad [18]$$

$$y(k + 1) - y(k) = C_m(x_m(k) - x_m(k - 1)) = C_m\Delta x_m(k)$$

$$x(k) = [\Delta x_m(k) \ y(k)]^T \quad [19]$$

Juntando (ec.17) con (ec.19) se obtiene las nuevas ecuaciones de estado y resultan las matrices aumentadas de estado (ec. 20) (L. Wang, 2019).

$$\begin{bmatrix} \Delta x_m(k + 1) \\ y(k + 1) \end{bmatrix} = \begin{bmatrix} A_m & 0_m^T \\ C_m A_m & 1 \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix} \Delta u(k)$$

$$y(k) = [0_m \ 1] \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} \quad [20]$$

2.6.1.4 Predicción de variables de estado y salida.

Con ayuda del modelo aumentado (ec. 20) se obtienen las predicciones de estado y salida para el instante presente “ k_i ”, estas predicciones se pueden apreciar en las siguientes ecuaciones:

$$x(k_i + 1 | k_i) = Ax(k_i) + B\Delta u(k_i)$$

$$x(k_i + 2 | k_i) = A^2x(k_i) + AB\Delta u(k_i) + B\Delta u(k_i + 1)$$

$$x(k_i + N_p | k_i) = A^{N_p}x(k_i) + A^{N_p-1}B\Delta u(k_i) + A^{N_p-N_c}B\Delta u(k_i + N_c - 1) \quad [21]$$

Donde N_c hace referencia al horizonte de control, el horizonte de control está asociado con el número de parámetros empleados para regir la trayectoria de control futura y N_p es el horizonte de predicción.

Implementando el mismo criterio para las predicciones de salida en el instante “ k_i ” se obtiene:

$$y(k_i + 1 | k_i) = CAx(k_i) + CB\Delta u(k_i)$$

$$y(k_i + 2 | k_i) = CA^2x(k_i) + CAB\Delta u(k_i) + B\Delta u(k_i + 1)$$

$$Y(k_i + N_p | k_i) = CA^{N_p}x(k_i) + CA^{N_p-1}B\Delta u(k_i) + CA^{N_p-N_c}B\Delta u(k_i + N_c - 1) \quad [22]$$

Respecto a la ecuación 22 es posible definir los vectores siguientes:

$$Y = [y(k_i + 1) \ y(k_i + 2) \ y(k_i + 3) \ \dots \ y(k_i + N_p)]^T$$

$$\Delta U = [\Delta u(k_i) \ \Delta u(k_i + 1) \ \Delta u(k_i + 2) \ \dots \ \Delta u(k_i + N_c - 1)]^T \quad [23]$$

En los sistemas de única entrada y única salida (SISO) la dimensión de Y es N_p por otro lado la dimensión de ΔU es N_c , (L. Wang, 2019), agrupando los vectores de acuerdo a la ec.22 se obtiene la siguiente matriz:

$$Y = F x(k_i) + \Phi \Delta U \quad [24]$$

Donde:

$$F = [CA \ CA^2 \ CA^3 \ \dots \ CA^{H_p}]^T, \quad [25]$$

$$\Phi = \begin{bmatrix} CB & 0 & 0 & 0 \\ CAB & CB & 0 & 0 \\ CA^2B & CAB & CB & 0 \\ CA^{H_p-1}B & CA^{H_p-2}B & CA^{H_p-3}B & CA^{H_p-H_c}B \end{bmatrix} \quad [26]$$

2.7 Identificación de sistemas mediante Matlab®

El modelo matemático de la planta, es un proceso que requiere de identificación de sistemas usando información de la entrada (voltaje de alimentación) y adquiriendo la variable de altura de nivel con la planta en lazo abierto. En la ilustración 13 se observa el diagrama que permite obtener el modelo de la planta en lazo abierto.

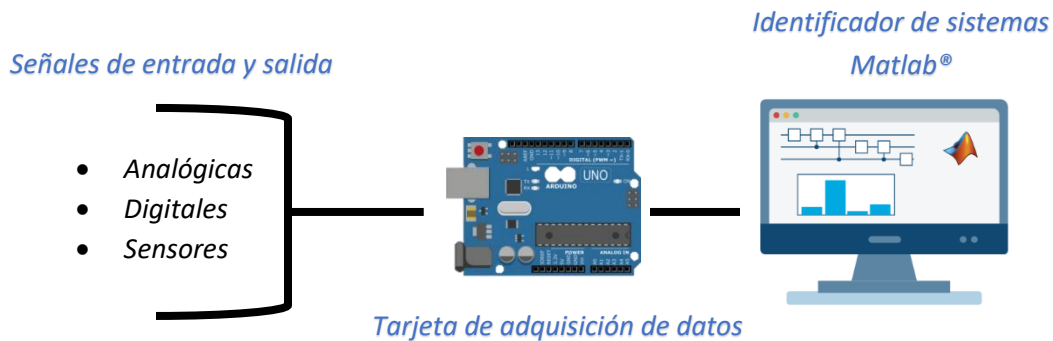


Ilustración 13: Sistema de adquisición de datos. Fuente: Elaboración propia

El identificador de sistemas es una herramienta disponible en el software de Matlab®, a través de esta herramienta es posible obtener un modelo matemático de cualquier sistema dinámico mediante la manipulación de señales de entrada y salida del sistema. (L. Aguirre, 2021)

2.7.1 Modelo de un sistema

El primer paso para llevar a cabo el estudio de un problema de control es la obtención del modelo matemático que se pretende controlar. Dicho modelo debe responder de la mejor manera posible ante una entrada $u(t)$ así como el proceso $y(t)$.

Para poder desarrollar la teoría de control se debe trabajar con modelos matemáticos ya sea modelos en variables de estado, funciones de transferencia, etc. Existen dos formas de generar dichos modelos:

- **Modelado físico:** Este desarrolló se rige mediante el uso de teoremas, leyes físicas fundamentales para interpretar el comportamiento del sistema, el punto en contra para el modelado de este tipo es que en ocasiones no todos los sistemas pueden ser modelados ya sea por la complejidad o porque las leyes que lo rigen se desconocen.
- **Identificación del sistema:** Este modo consiste en procesar los datos experimentales a través de registros de las señales de entrada y salida (*ver ilustración 13*). (L. Aguirre, 2021)

2.7.1.1 Clasificación de modelos.

Modelos mentales: Se basan en la percepción del hombre debido a que son el conjunto de imágenes e ideas que abarcan la mente humana.

Modelos matemáticos: Están constituidos por ecuaciones matemáticas en los cuales se puede conocer el comportamiento del sistema, es un caso de formalización simplificado para predecir su comportamiento.

Modelos análogos: Corresponden a procesos en distintas áreas por ejemplo un oscilador mecánico y un oscilador eléctrico son descritos por la misma ecuación diferencial, pero los coeficientes tendrán diferentes interpretaciones físicas.

Modelos Físicos: Se aplica a las construcciones teóricas de un sistema físico, así como al montaje a escala implementando objetos reales para improvisar el comportamiento de un sistema hidráulico de cierta complejidad.

2.7.2 Proceso de identificación.

De manera general el proceso de identificación se desarrolla a través de los siguientes pasos:

-Adquisición de datos entrada-salida: Consiste en excitar el sistema mediante la implementación de una señal de entrada y registrar el proceso de datos en un intervalo de muestreo.

Es importante mencionar que las señales con datos binarios son convenientes para identificar sistemas lineales.

-Análisis de datos capturados: Es indispensable estudiar y modificar los datos erróneos o innecesarios, eliminando tendencias a altas frecuencias o en el mejor de los casos aplicar una regresión lineal a los datos obtenidos.

-Elegir estructura del modelo: Tomar la decisión del modelo a utilizar, puede ser lineal, no lineal, en tiempo continuo, tiempo discreto, con la finalidad de determinar la estructura que mejor se adapte al sistema, es importante tener un conocimiento previo de las leyes, teoremas físicos que rigen al proceso.

-Validación del modelo: Finalmente el paso consiste en comparar el sistema sometiendo los datos recopilados a criterios para valorar la calidad del modelo.

De manera esquemática el proceso de identificación se muestra en la siguiente ilustración:

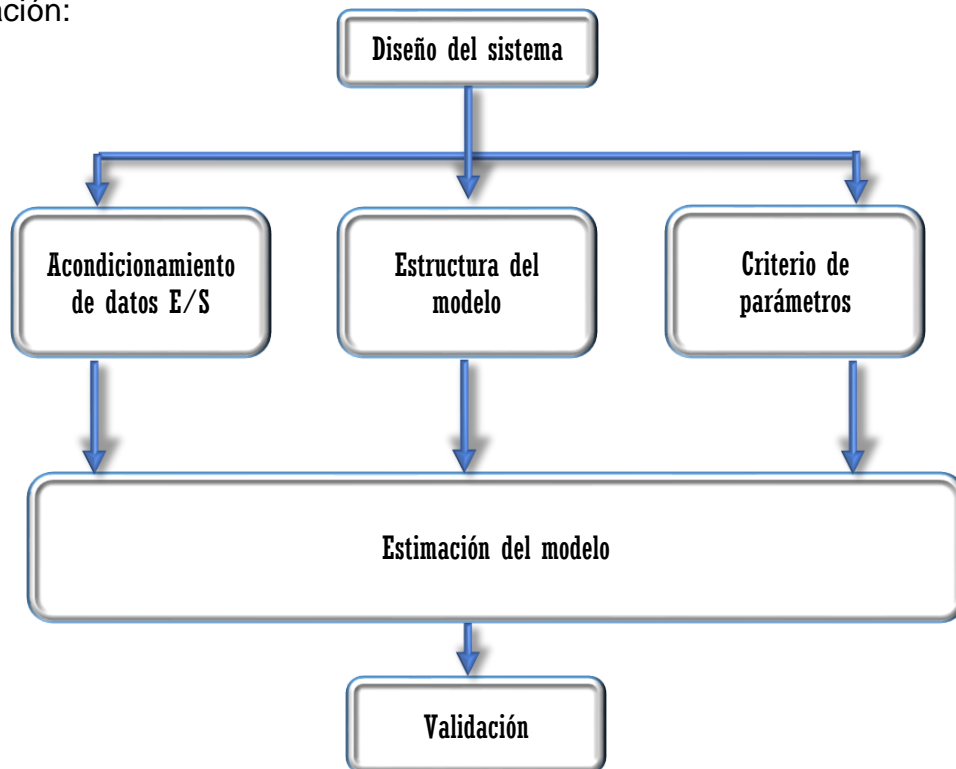


Ilustración 14:Proceso de identificación. Fuente: Elaboración propia

2.7.3 Métodos de identificación

Los métodos de identificación se clasifican de acuerdo al algoritmo utilizado, según el modelo obtenido se pueden clasificar en: (L. Aguirre, 2021)

- Métodos no paramétricos: Permiten obtener modelos bajo el estudio de análisis transitorios, análisis de la respuesta en frecuencia, análisis espectral entre otros.

Dependiendo de la aplicación se clasifican en:

- Métodos de identificación off-line: Este tipo de datos adquiridos en el sistema de estudio son almacenados, posteriormente transferidos a un computador para su evaluación y procesamiento, el cual es utilizado en este proyecto.
- Método de identificación on-line: En este método los datos son directamente procesados en tiempo real, que consiste en ajustar los parámetros implementando la teoría de control.

3 DISEÑO Y SIMULACIÓN DEL CONTROLADOR

3.1 Diseño del prototipo

En el diagrama de bloques siguiente se muestra el recorrido que hace el fluido a través del sistema hidráulico con los tanques acoplados, el tanque de depósito es almacenado con un fluido para suministrar al tanque no.1, la salida del tanque no.1 es la entrada del tanque no.2 y de esta manera alcanzar un estado estable en el nivel del fluido en los tanques 1 y 2, esto es el funcionamiento del sistema en cascada.

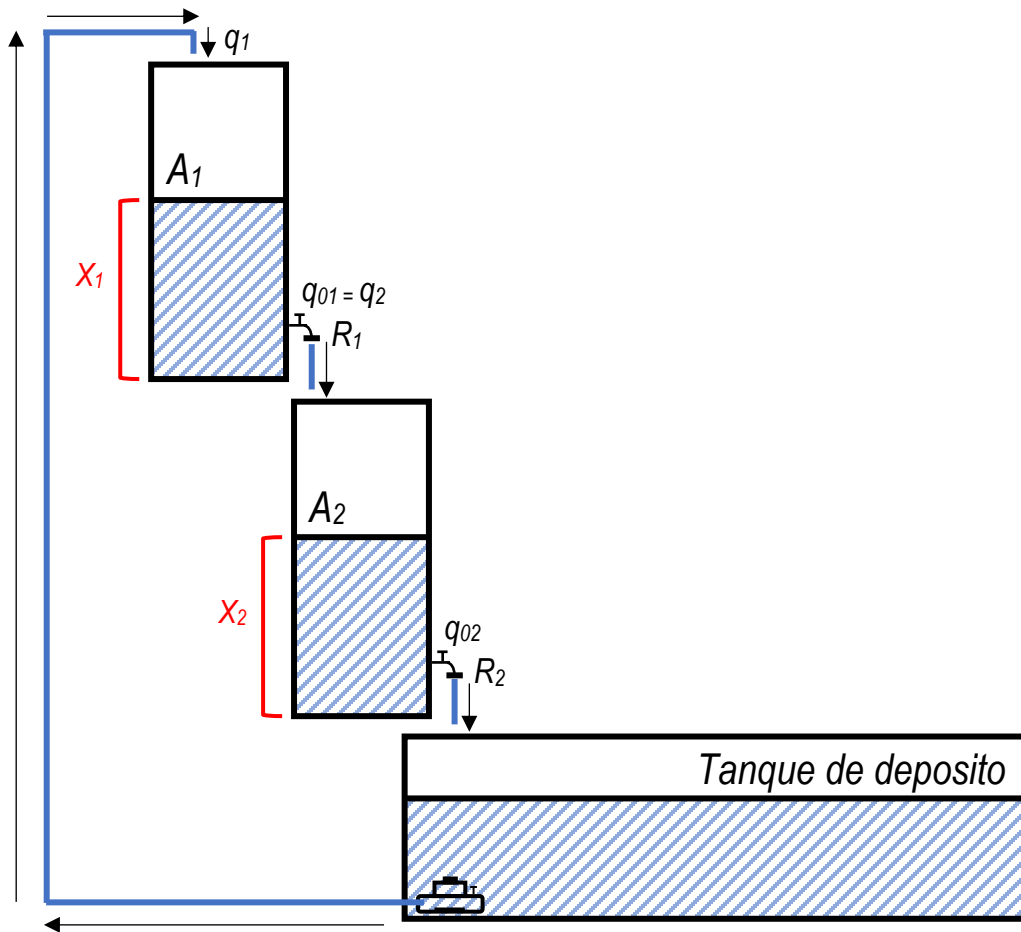


Ilustración 15: Representación en bloques del sistema hidráulico. Fuente: Elaboración propia

La construcción del sistema hidráulico consiste en la fabricación de una estructura de madera, con una base firme para el tanque de depósito, además de fijar una torre escalonada para el acoplamiento en cascada de los tanques, los tanques son elaborados en material de acrílico transparente con un espesor de 3mm.

El sistema de tanques acoplados es compuesto por una bomba sumergible que requiere de una alimentación de 3 a 5 volts, la bomba impulsa el fluido desde el tanque de depósito hasta la parte superior del sistema hidráulico, los tanques están

equipados con un grifo para regular el flujo de salida, además de contar con un sensor de presión diferencial para medir el nivel del fluido.

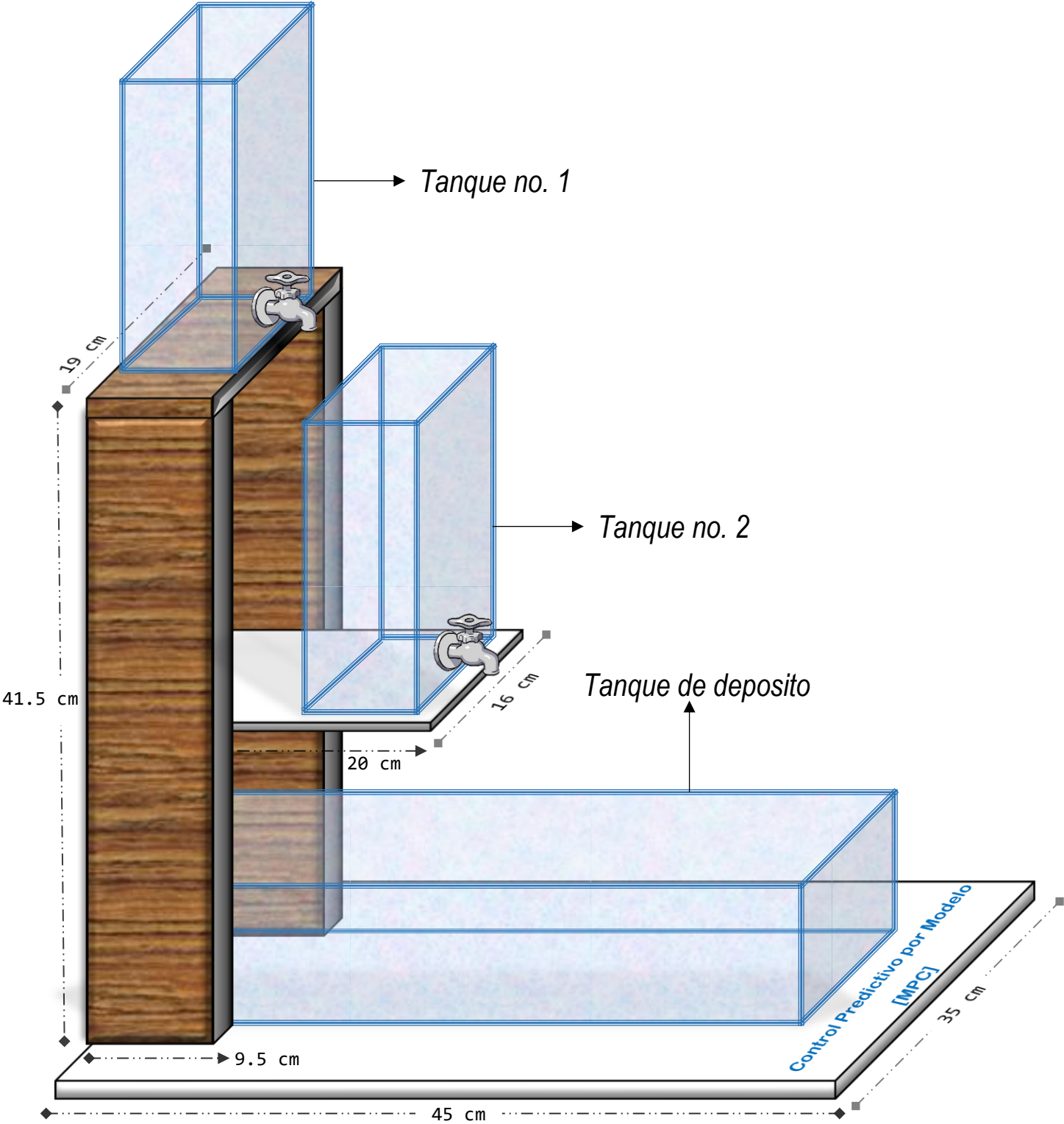


Ilustración 16: Diseño del sistema hidráulico. Fuente: Elaboración propia

3.1.1 Bomba sumergible

La bomba es totalmente sumergible y muy útil para aplicaciones donde se requiere extraer algún fluido. Esta funciona en un rango de tensión de 3 a 5 VCD. Su composición permite tener un flujo de hasta dos litros de agua por minuto (80-120 l/h).

De manera interna está conformado por un motor DC de 800 mA, en las terminales del motor están integrados unos capacitores cerámicos con la finalidad de mejorar la estabilidad del voltaje y prevenir el ruido eléctrico que la fuente de tensión puede generar.

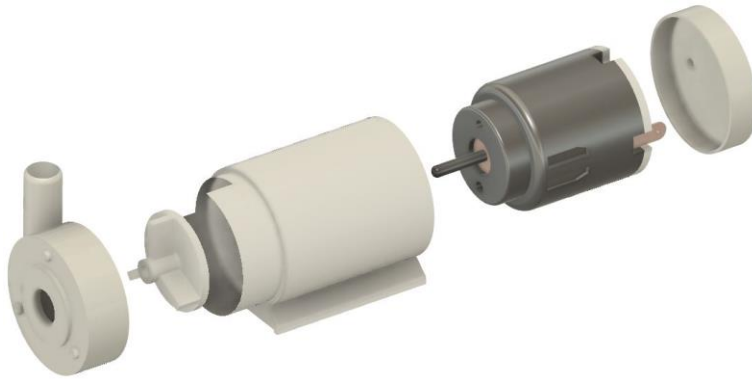


Ilustración 17: Ensamble de la bomba sumergible

Principales características de la bomba:

- Voltaje de trabajo: 3 – 5 VCD
- Caudal Bombeo máximo: 80-120 l/h.
- Corriente: 800mA
- Potencia: 0.4 -1.5 W

Para variar el flujo suministrado por la bomba conforme a los requisitos del sistema, la velocidad del motor se regula en proporción al voltaje de alimentación, en la siguiente ilustración se muestra la conexión de la bomba.

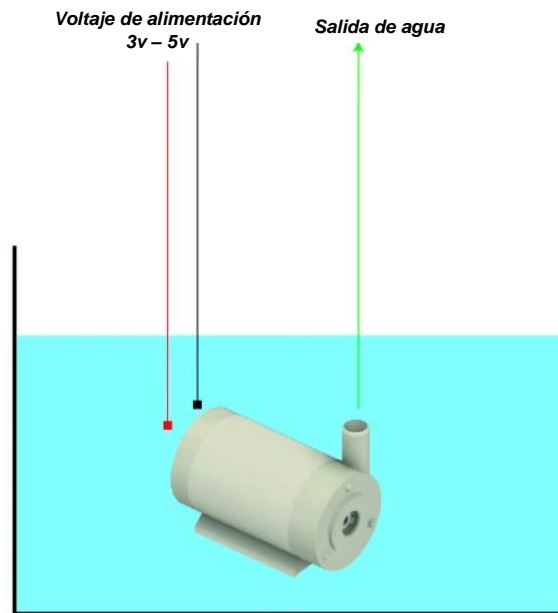


Ilustración 18: Diagrama de conexión bomba sumergible.

3.1.2 Acondicionamiento del sensor de presión diferencial

Es posible emplear una característica física para medir el nivel de algún fluido y es a través de un sensor de presión diferencial.

El sensor utilizado en este proyecto es de tipo MPX5010DP (Anexo A-1) ya que entrega un voltaje de 0v – 5v ideal para microcontroladores y en su salida proporciona una señal analógica precisa. Algunas de sus características son:

- Voltaje 5 Vdc.
- Rango 0 – 1.45 psi (0 – 10 Kpa).
- Temperatura compensada sobre -40 ° a +125 ° C.
- Comportamiento lineal (V vs Presión diferencial).

La distribución de pines se ilustra a continuación:

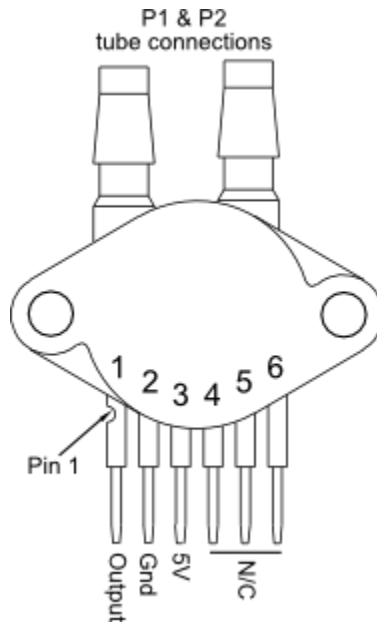


Ilustración 19: Distribución de pines.

El diagrama de conexión se muestra en la siguiente ilustración es importante mencionar que el sensor no puede entrar en contacto con el fluido, por lo tanto, se crea un colchón de aire en el interior del tubo que está conectado al sensor para evitar dañar la composición del sensor.

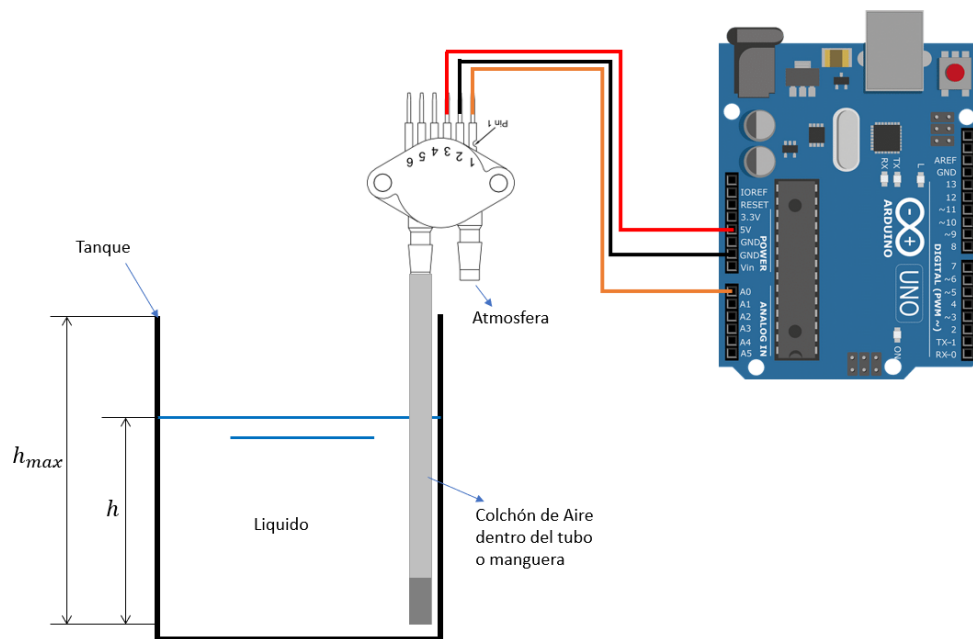


Ilustración 20: Circuito nivel de tanque.

3.2 Modelo de la planta

La obtención del modelo matemático del sistema se realiza a través de identificación de sistemas en Matlab® donde es indispensable disponer de un registro de datos del valor de la entrada y salida. En este proyecto las variables a considerar son el voltaje para alimentar la bomba que corresponde a 4 volts, y la salida que es el nivel del fluido en cm.

El primer aspecto a considerar es dejar trabajar el sistema hidráulico a manera de lazo abierto, para lograr realizar la adquisición de datos (D/A) como se muestra en la ilustración 21.



Ilustración 21: Representación en bloques del sistema en lazo abierto. Fuente: Elaboración propia

Para la adquisición de datos se realiza la programación en el software Arduino®, en este código se configuran los parámetros para hacer funcionar al sensor de presión diferencial MPX5010DP, (Anexo A-2) su función es otorgar el valor real del nivel conforme al tiempo así mismo se enlaza la comunicación entre la tarjeta Arduino® UNO y excel para el registro de datos, estos datos son leídos cada 10ms.

El sistema se logra establecer en una altura de 15 cm en un tiempo de 20 minutos, otorgando la lectura de 7251 datos, es decir, que en cada segundo hay un registro de 6 datos. En la ilustración 22 se muestra el resultado de acuerdo a la respuesta del sistema.

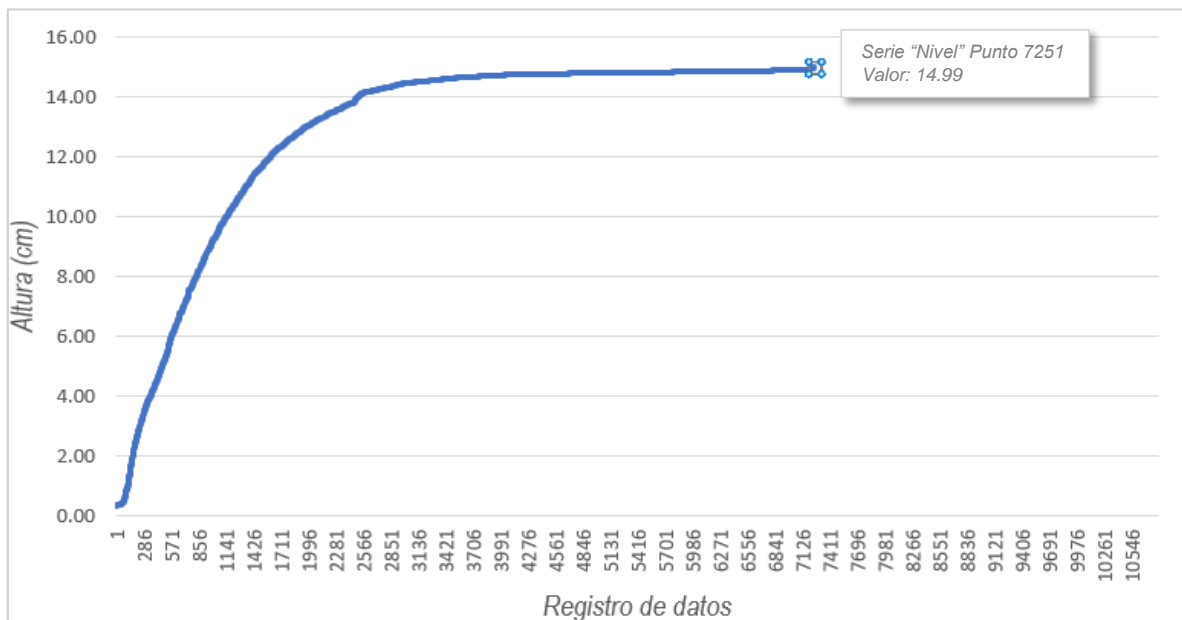


Ilustración 22: Comportamiento del nivel del sistema hidráulico. Fuente: Elaboración propia

Los datos obtenidos son exportados en Matlab® para poder ser sometidos al identificador de sistemas, solo se aplica ciertos comandos en el entorno de trabajo que a continuación se ilustran.

```
y= xlsread (`adquisicion_de_datos_lazo_abierto.xlsx`, `B2:B7252`);
x= xlsread (`adquisicion_de_datos_lazo_abierto.xlsx`, `C2:C7252`);
```

Ilustración 23:Exportación de datos de Excel-Matlab® Fuente: Elaboración propia

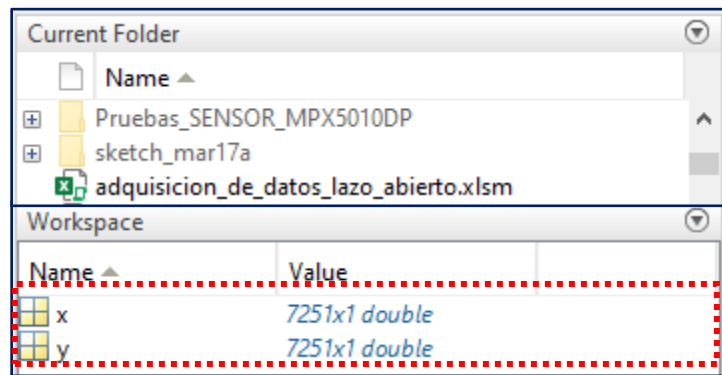


Ilustración 24:Carga de datos en Matlab® Fuente: Elaboración propia

Una vez que ya se tiene almacenado los datos en Matlab® se ingresa el comando "ident" para visualizar la interfaz gráfica del identificador. La siguiente imagen muestra el entorno del identificador de sistemas.

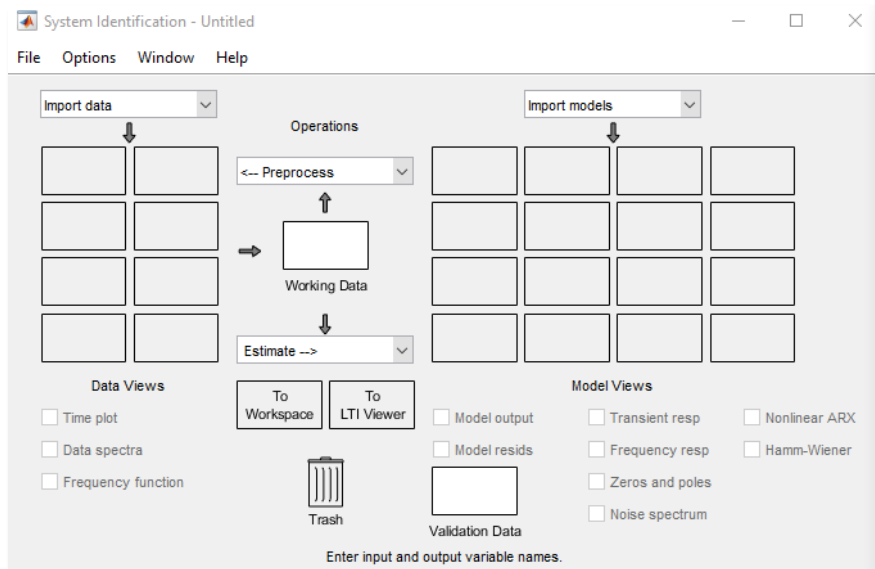


Ilustración 25:Identificador de sistemas de Matlab® Fuente: Elaboración propia

Lo siguiente es cargar los datos a través de la opción “import data → Time domain data” como se muestra en la ilustración 26.

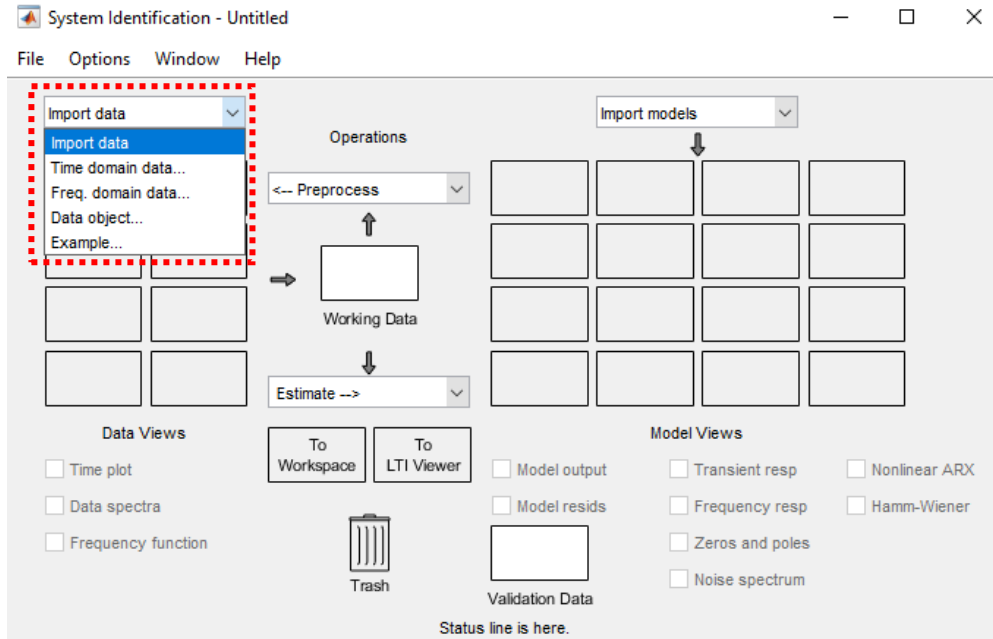


Ilustración 26: Carga de datos en el identificador de sistemas. Fuente: Elaboración propia

La ilustración 27 muestra un formulario, en este apartado se carga la entrada(x) y salida(y), finalmente se selecciona el icono de “import” y en “Data views” se puede observar los gráficos respecto a la entrada y salida.

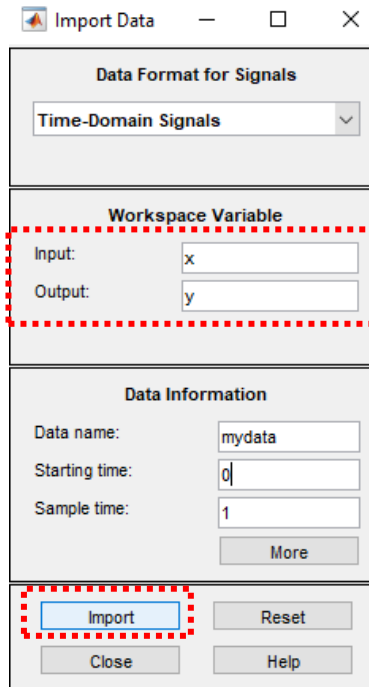


Ilustración 27: Formulario de import data Fuente: Elaboración propia

En la ilustración 28 se muestra con éxito los datos almacenados en el apartado de “Data Views”, para visualizar el comportamiento de los datos solo se marca la casilla de “Time plot”.

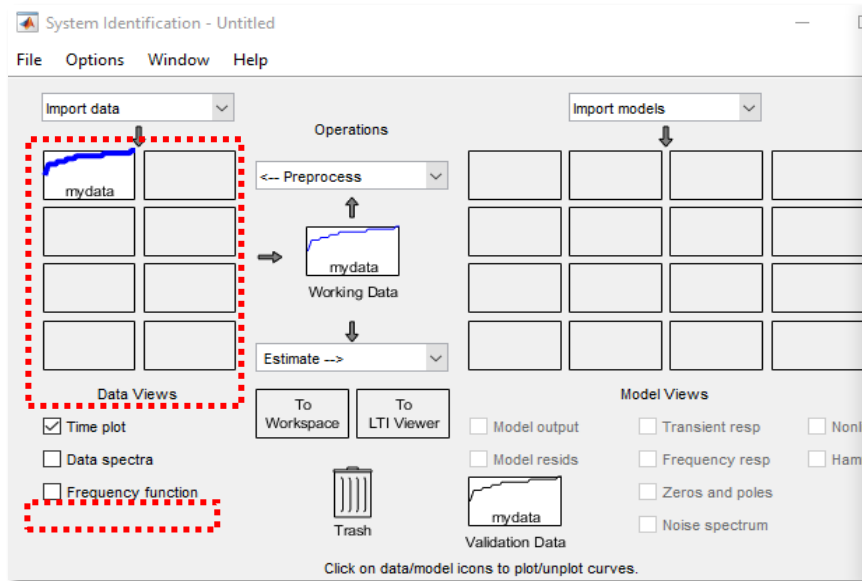


Ilustración 28: Datos almacenados en Data Views Fuente: Elaboración propia

En la ilustración 29 se muestra gráficamente el comportamiento de los datos almacenados en el identificador, donde “y1” hace referencia a la salida del sistema y “u1” hace referencia a la entrada del sistema.

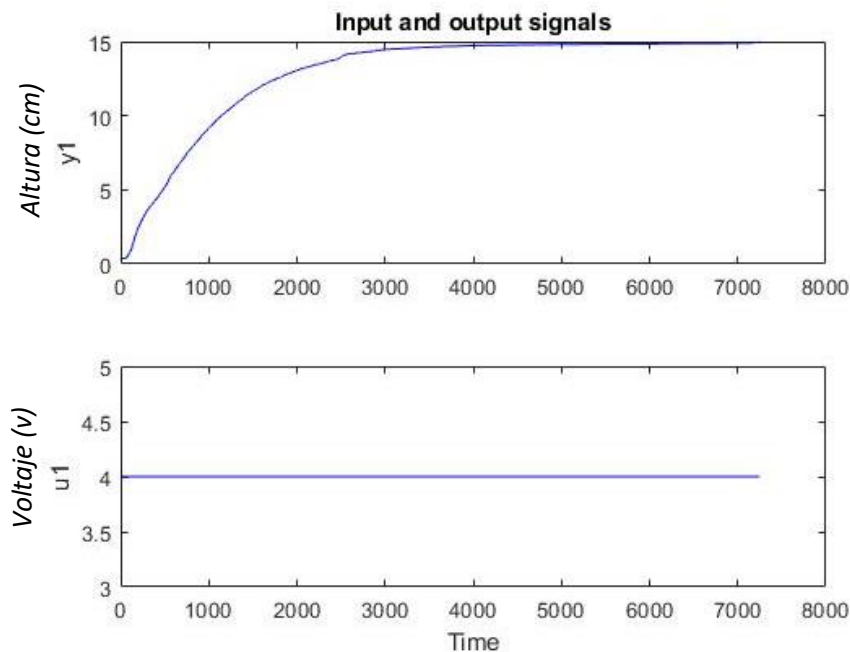


Ilustración 29: Gráficos de entrada (u1) vs salida (y1) Fuente: Elaboración propia

Ahora sigue el procesamiento de los datos, esto significa que una parte del registro es para trabajar el modelo de identificación y la otra parte para validar el modelo. Se selecciona la parte “Preprocess → Select range” como se muestra a continuación.

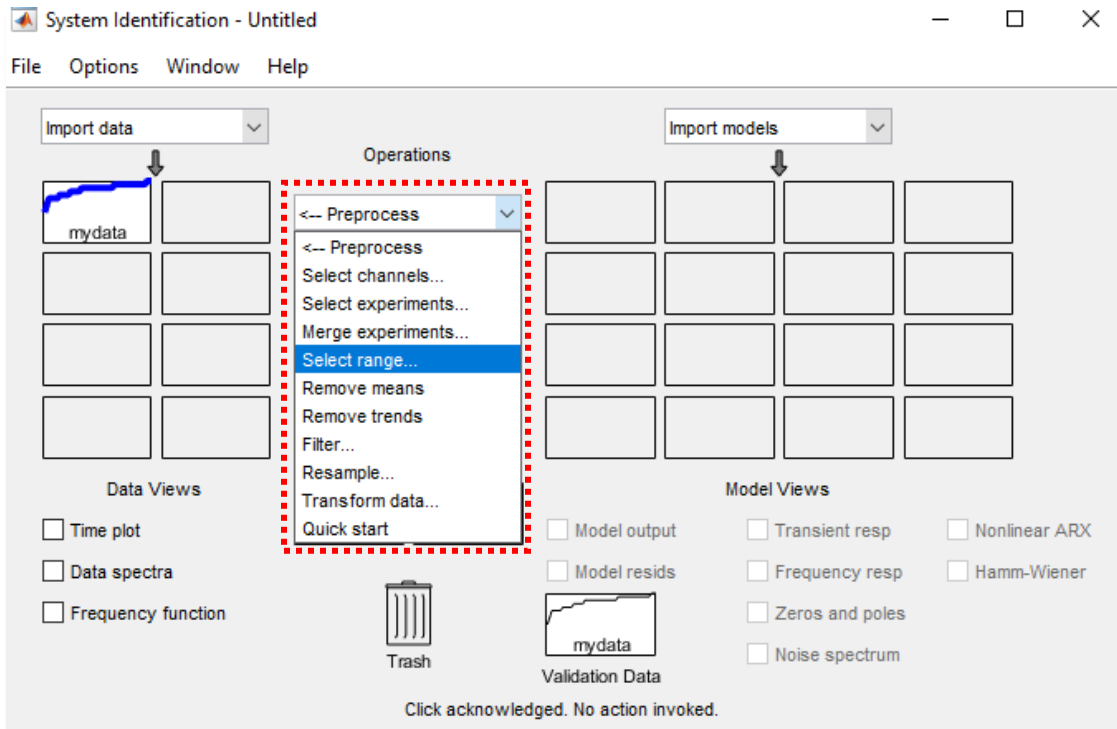


Ilustración 30: Identificación de sistemas en Matlab® Fuente: Elaboración propia

Posteriormente, se abren las ventanas de “Select Range” para agrupar los datos en dos partes la primera mitad de datos está destinada para generar el modelo (ilustración 31), el segundo intervalo de datos es destinada para validación del modelo. (ilustración 32)

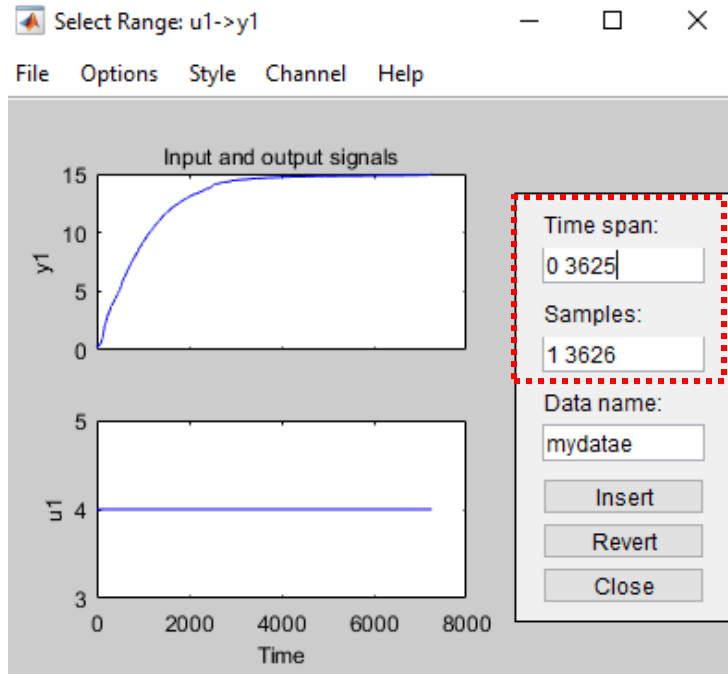


Ilustración 31: Rango de datos para entrenamiento Fuente: Elaboración propia

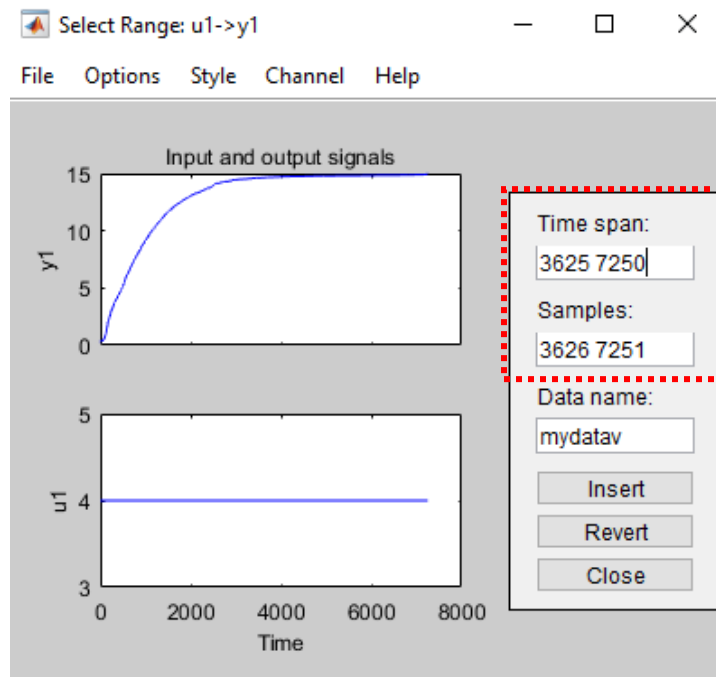


Ilustración 32: Rango de datos para validación Fuente: Elaboración propia

Cuando estos nuevos conjuntos de datos son generados y almacenados aparecen en las casillas del entorno de "Data views" así como en la casilla de "Working Data" y "Validation Data" donde se logra visualizar los gráficos para una mejor perspectiva.

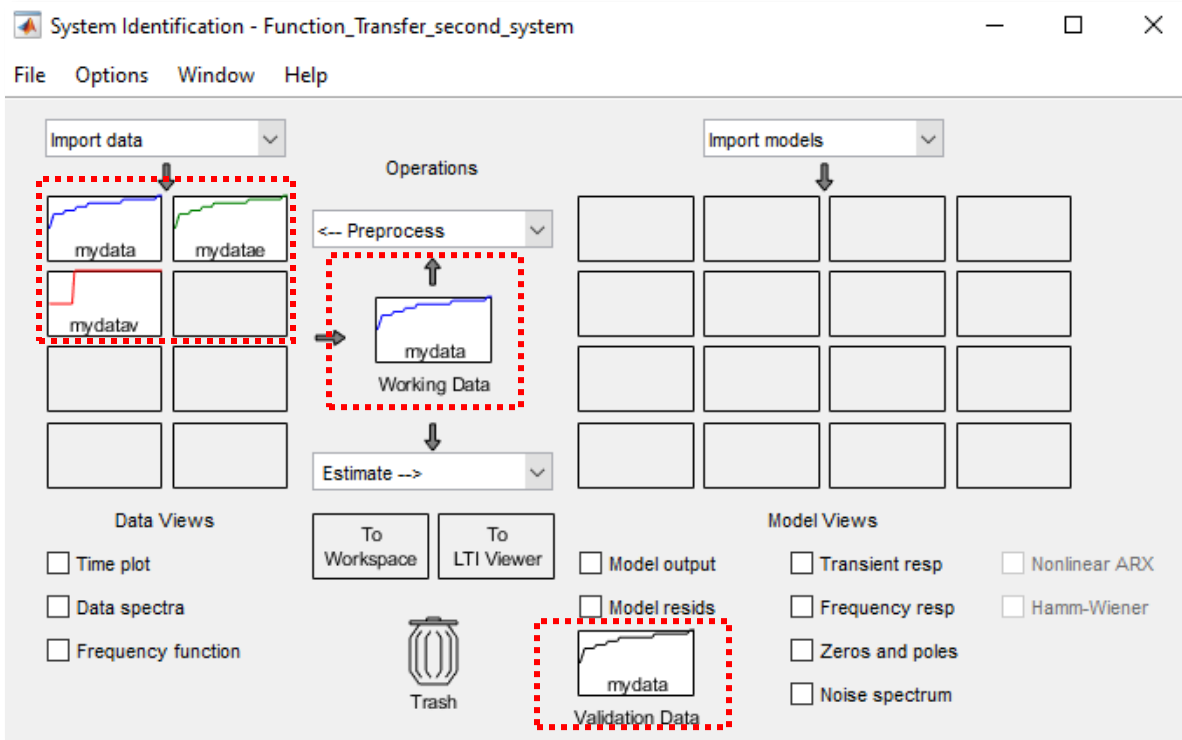


Ilustración 33: Carga de datos de entrenamiento y validación. Fuente: Elaboración propia

En la ilustración 34 se muestra el conjunto de datos para entrenamiento en color verde y la segunda parte los datos para validación en color rojo, esto se hace con el objetivo de obtener con mayor precisión la función de transferencia de la planta,

recordando que “y1” respecto al tiempo hace alusión a la salida del sistema, y “u1” respecto al tiempo hace referencia a la entrada del sistema.

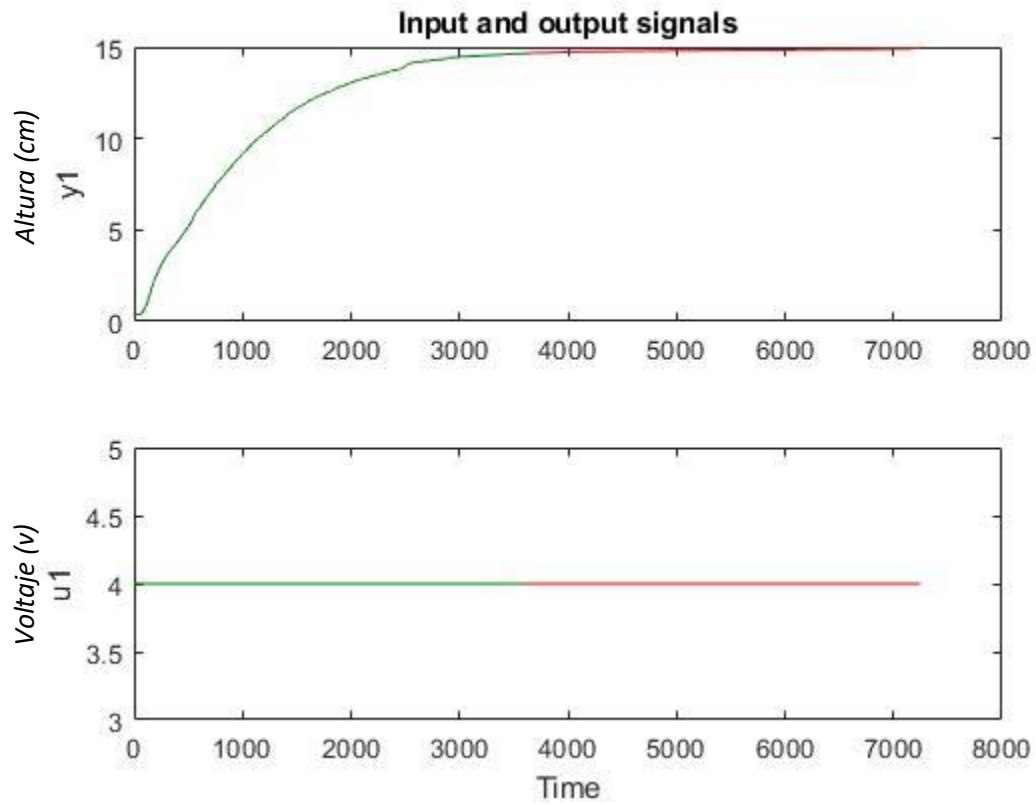


Ilustración 34: Gráfica con datos de entrenamiento y validación Fuente: Elaboración propia

La etapa siguiente es estimar el modelo de la planta solo se selecciona el apartado de “estímate” → “Transfer Function Models” y elegimos el número de polos y ceros que requiramos para el sistema como se muestra en la ilustración 35.

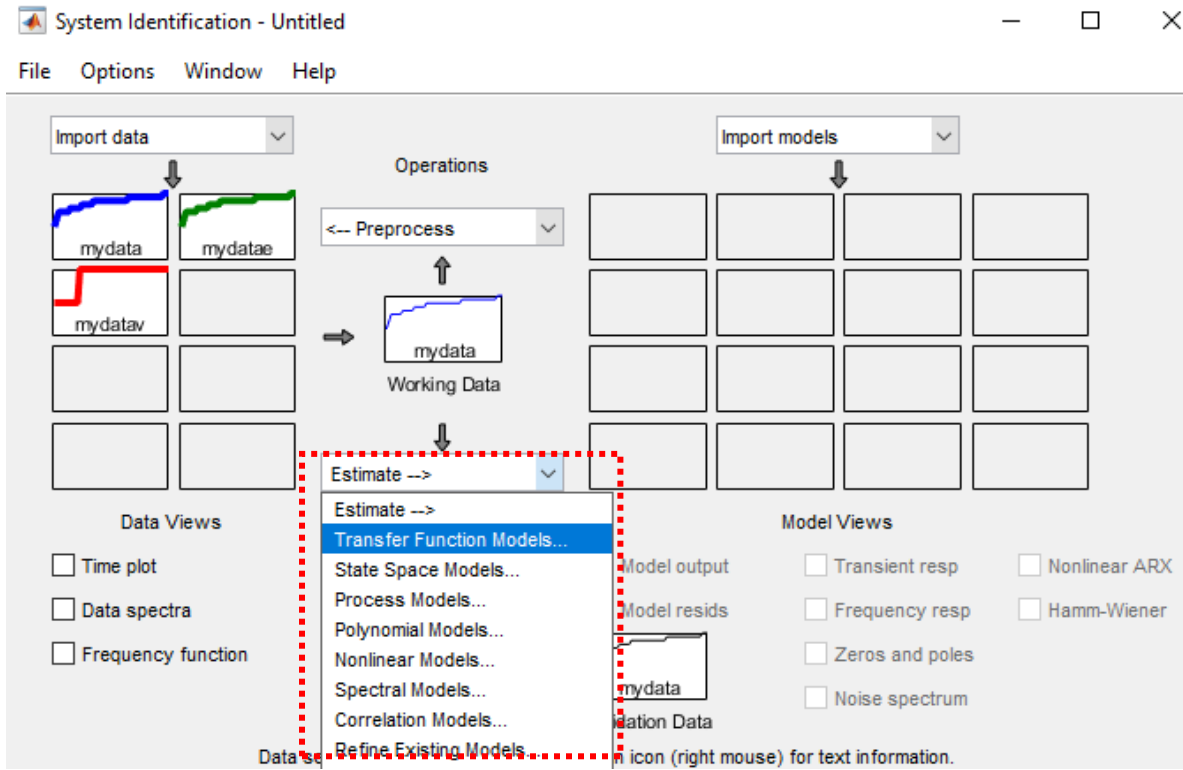


Ilustración 35: Selección para estimar la función de transferencia. Fuente: Elaboración propia

En la ilustración 36 se hace la elección del número de polos y número de zeros, en esta parte es opcional el elegir la forma del sistema (Función de transferencia), también se considera la elección del modelo en tiempo continuo y por último se selecciona el icono de “Estimate”.

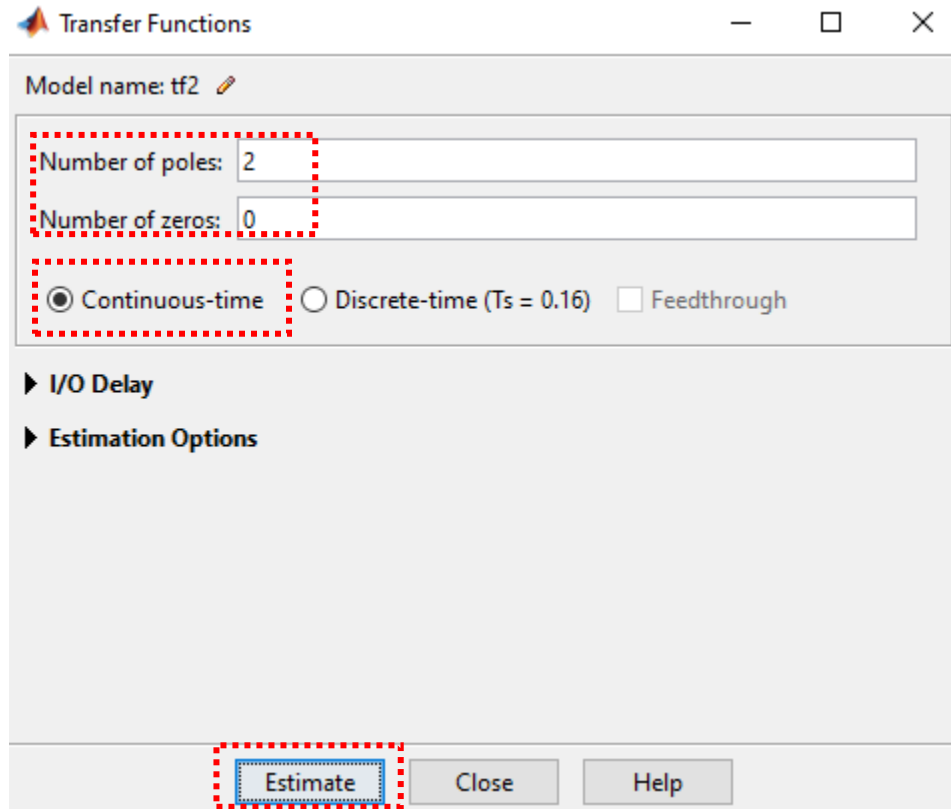


Ilustración 36: Elección de polos y zeros. Fuente: Elaboración propia

Cuando se selecciona el icono de “Estimate” el identificador de sistemas comienza a procesar todos los datos para el modelo como se muestra en la ilustración 37, al final del procesamiento se muestra en la pantalla un porcentaje de precisión acerca del modelo lo cual es eficazmente aceptable.

Plant Identification Progress

Transfer Function Identification
 Estimation data: Time domain data mydata
 Data has 1 outputs, 1 inputs and 7251 samples.
 Number of poles: 2, Number of zeros: 0
 Initialization Method: "iv"

Estimation Progress

Initializing model parameters...
 Initializing using 'iv' method...
 done.

Initialization complete.

Nonlinear least squares with automatically chosen line search method

Iteration	Cost	Norm of step	First-order optimality	Improvement (%)		
				Expected	Achieved	Bisections
0	0.0153008	-	2.87e+05	460	-	-
1	0.0146828	18.2	1.96e+07	460	4.04	2
2	0.0140381	30.6	7.74e+07	898	4.39	2
3	0.0134095	28.8	1.11e+08	1.96e+03	4.48	3
4	0.0128076	37.8	1.67e+08	2.65e+03	4.49	3
5	0.0123211	47.2	2.47e+08	3.41e+03	3.8	3

Result

Termination condition: Near (local) minimum, (norm(g) < tol).
 Number of iterations: 20, Number of function evaluations: 80

Status: Estimated using TFEST
 Fit to estimation data: 98.06%, FPE: 0.00460927

Ilustración 37: Progreso de la estimación del modelo. Fuente: Elaboración propia

Una vez que se configuran los parámetros del modelo aparece en una casilla “tf1” en el apartado de “Model Views”, esto quiere decir que el modelo se ha generado con éxito. (ilustración 38)

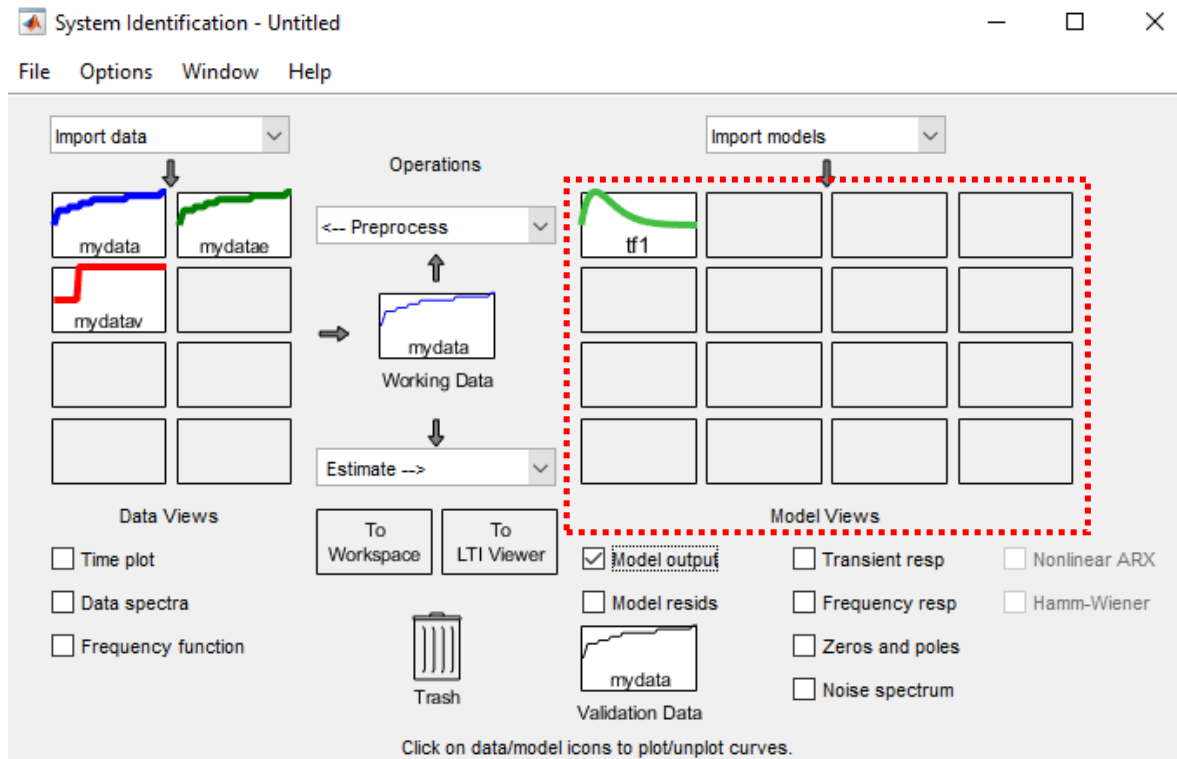


Ilustración 38: Generación del modelo. Fuente: Elaboración propia

Finalmente se selecciona la casilla “tf1” de la ilustración 38 para visualizar la función de transferencia del sistema en el “workspace” de Matlab®. En la ecuación 28 se observa un sistema de segundo orden y a través de ese sistema se continúa con el diseño del controlador.

$$G(s) = \frac{0.0002542}{s^2 + 0.01574 s + 6.847e^{-05}} \quad [28]$$

En la ilustración 39 se muestra el comportamiento del modelo (ec.28), se puede observar que alcanza un estado estable en una altura de 15 cm.

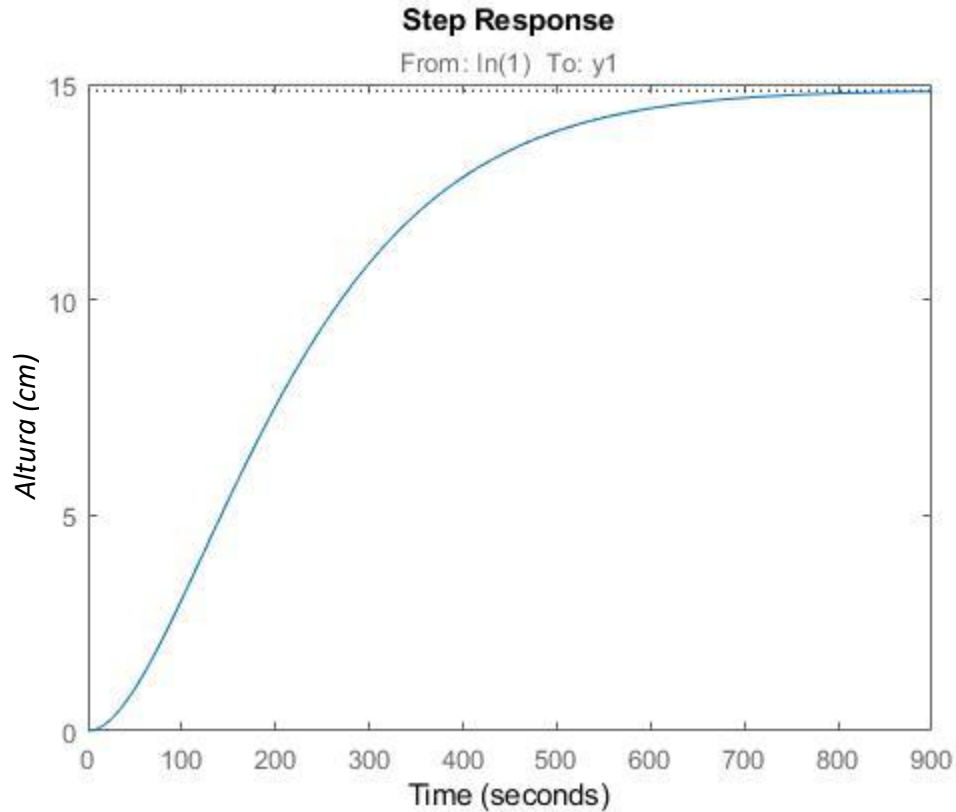


Ilustración 39:Grafica del sistema. Fuente: Elaboración propia

3.3 Verificación del sistema controlable y observable.

Estos métodos fueron descritos por Rudolf Kalman en 1960 con la finalidad de determinar si el control puede ser aplicado al sistema, por lo tanto, se hace el uso de técnicas matriciales a través del modelo en espacio de estados.

El uso de estas técnicas consiste en transformar el modelo de la planta (ec.28) a la siguiente forma:

$$\dot{x} = Ax + Bu,$$

$$y = Cx \tag{29}$$

En las siguientes ecuaciones se describe el procedimiento para llegar a la forma de la ecuación 29.

$$\frac{y(s)}{u(s)} = \frac{0.0002542}{s^2 + 0.01574 s + 6.847e^{-05}} \tag{30}$$

Posteriormente se reescribe de la forma:

$$y(s)(s^2 + 0.01574 s + 6.847e^{-05}) = 0.0002542u(s) \quad [31]$$

$$y''(t) + 0.01574 y' + 6.847e^{-05}y = 0.0002542u \quad [31.1]$$

$$y''(t) = -0.01574 y' - 6.847e^{-05}y + 0.0002542u \quad [31.2]$$

Para obtener la forma matricial en tiempo de estados se considera: (ver ec. (9)(10))

$$\begin{bmatrix} x'_1 \\ x'_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -6.847e^{-05} & -0.01574 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \quad ;$$

$$y = [0.0002542 \quad 0] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad [32]$$

Una vez obtenido el modelo en espacio de estados para diseñar el controlador al sistema, el primer aspecto a considerar es verificar si es controlable y observable entonces se analiza la matriz de controlabilidad que se describe en términos de “A” y “B” (ver ec.1)

Por lo tanto, si el determinante de Pc es distinto de cero el sistema es controlable, a continuación, se comprueba que el sistema es totalmente controlable.

Se tiene la matriz “A” y “B” de acuerdo al modelo de estados de la ecuación (32). Así como el resultado del producto de A*B.

$$A = \begin{bmatrix} 0 & 1 \\ -6.847e^{-05} & -0.01574 \end{bmatrix} \quad [33]$$

$$B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad [34]$$

$$A * B = \begin{bmatrix} 1 \\ -0.01574 \end{bmatrix} \quad [35]$$

Como el sistema tiene una entrada una salida la dimensión de la matriz de controlabilidad es de 2x2 y sus valores se muestran en la ecuación (36)

$$P_c = \begin{bmatrix} 0 & 1 \\ 1 & -0.01574 \end{bmatrix} \quad [36]$$

Finalmente se obtiene el resultado del determinante de la matriz de controlabilidad demostrando que el sistema es totalmente controlable.

$$\det\{P_c\} = \begin{vmatrix} 0 & 1 \\ 1 & -0.01574 \end{vmatrix} = -1 \quad [37]$$

El siguiente paso es determinar si el sistema es observable un sistema es completamente observable cuando el determinante de la matriz P_o es distinto de cero de acuerdo a la ecuación (2).

Se considera nuevamente al sistema de la ecuación (32) para demostrar que el sistema es observable teniendo como resultado un determinante diferente de cero. Se tiene a la matriz:

$$C = [0.0002542 \quad 0] \quad [38]$$

Y al producto de "C*A".

$$C * A = [0 \quad 0.0002542] \quad [39]$$

Por lo tanto, la matriz de observabilidad queda de la siguiente forma:

$$P_o = \begin{bmatrix} 0.0002542 & 0 \\ 0 & 0.0002542 \end{bmatrix} \quad [40]$$

Por último, se obtiene el resultado del determinante de la matriz de controlabilidad demostrando que el sistema es totalmente controlable.

$$\det\{P_o\} = \begin{vmatrix} 0.0002542 & 0 \\ 0 & 0.0002542 \end{vmatrix} = 6.46e^{-08} \quad [41]$$

3.4 Discretización del sistema y simulación.

Ahora que se tiene la representación del sistema en matrices de estado en tiempo continuo (ver ec. 32), se procede a discretizar. Un sistema discreto se representa por (ec. 42), con ayuda de la función `c2dm` de Matlab® y un periodo de muestreo de 1s tendremos el sistema discreto en matrices de estado (ec. 43).

$$\begin{aligned}x_{n+1} &= Ax_n + Bu_n \\ y_n &= Cx_n\end{aligned}\tag{42}$$

$$\begin{aligned}x_{n+1} &= \begin{bmatrix} 0.99997 & 0.99216 \\ -6.7933e^{-05} & 0.98435 \end{bmatrix} x_n + \begin{bmatrix} 0.49738 \\ 0.99216 \end{bmatrix} u_n \\ y_n &= [0.0002542 \quad 0] x_n\end{aligned}\tag{43}$$

Posteriormente en Simulink compararemos el sistema en tiempo continuo (ec.32) y tiempo discreto (ec.43) en representación de diagrama de bloques.

Primero se ejecuta el comando “*Simulink3*” en el *Command Window* de Matlab®, enseguida, se muestra la interfaz de Simulink como en la ilustración 40.

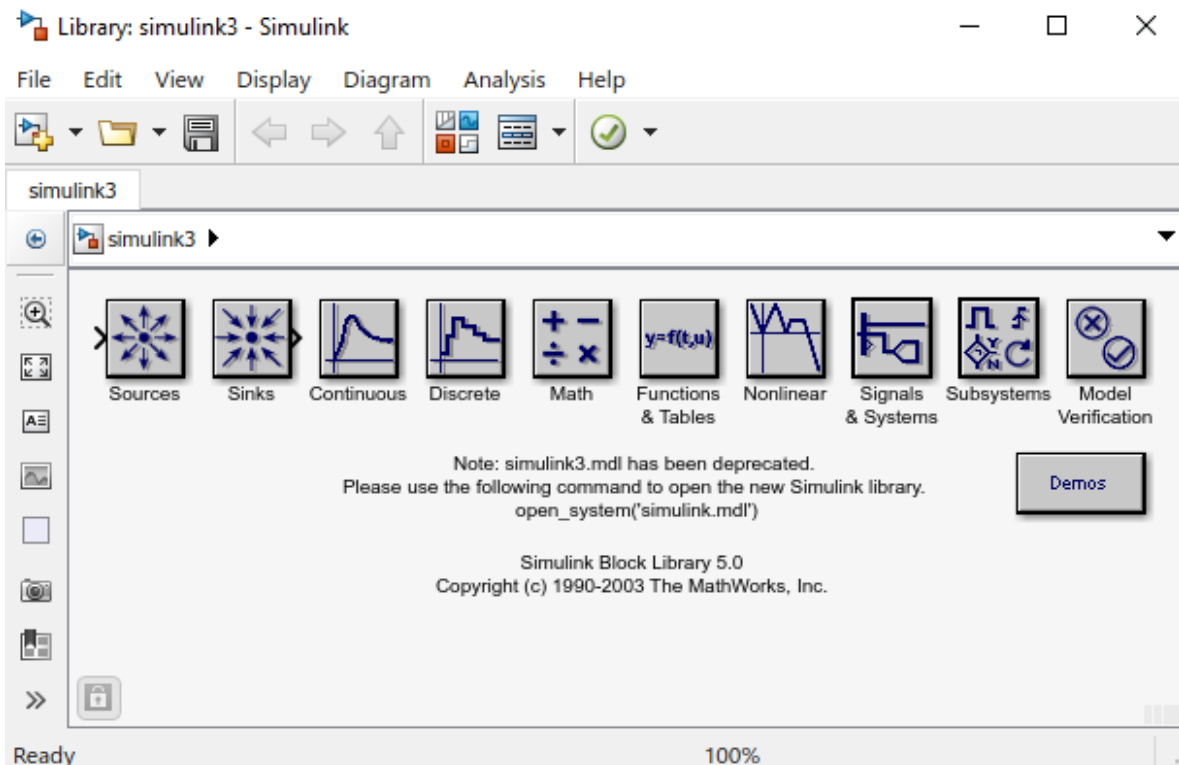


Ilustración 40: Interfaz de Simulink. Fuente: Elaboración propia

Se crea un archivo nuevo seleccionando la opción “File → New→Blank Model” como se muestra en la ilustración 41.

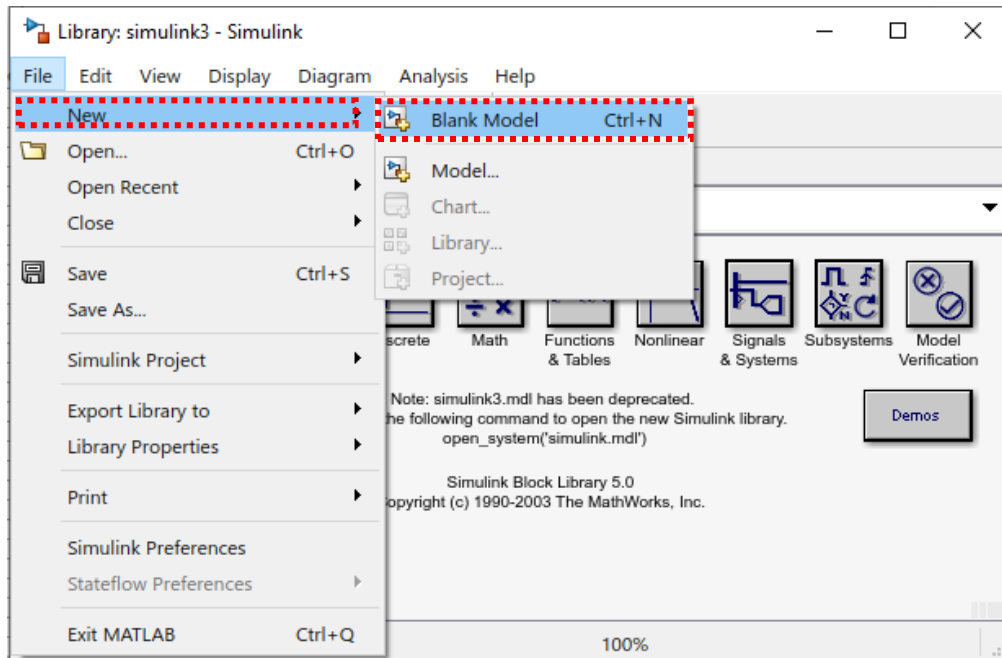


Ilustración 41: Creación de nuevo archivo. Fuente: Elaboración propia

Posteriormente se abre una nueva ventana de entorno de trabajo para diseñar el modelo a bloques del sistema continuo y sistema discreto, esto se muestra en la ilustración 42.

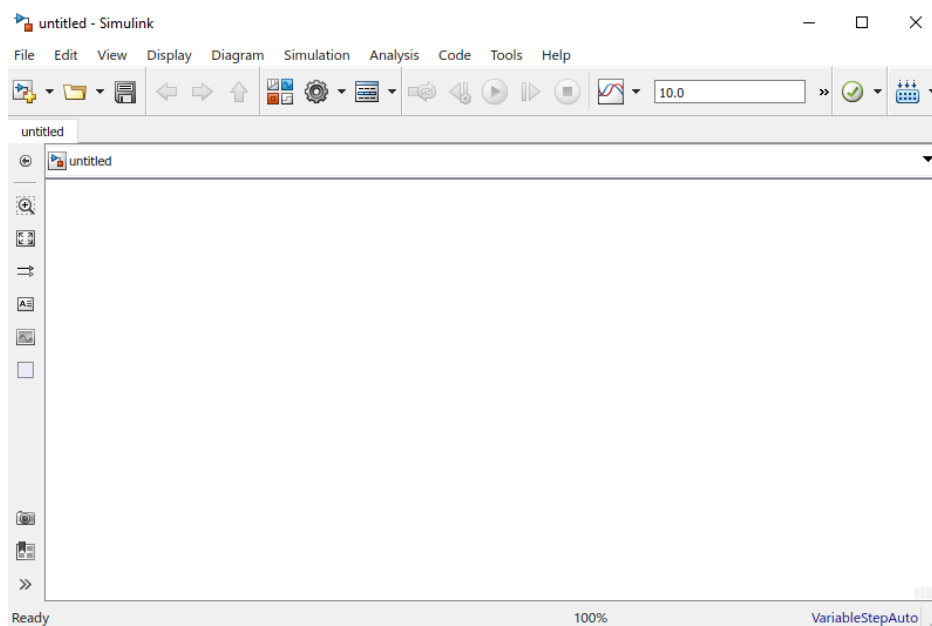


Ilustración 42: Entorno de trabajo de Simulink. Fuente: Elaboración propia

Inicialmente se selecciona el bloque de entrada al sistema para colocarlo en el área de trabajo, la entrada escalón (step) nos servirá para ambos sistemas (tiempo continuo y tiempo discreto), ya que tiene los mismos parámetros en su configuración

En la ilustración 43 se muestra la selección del bloque de entrada, en la librería “Source → Step”.

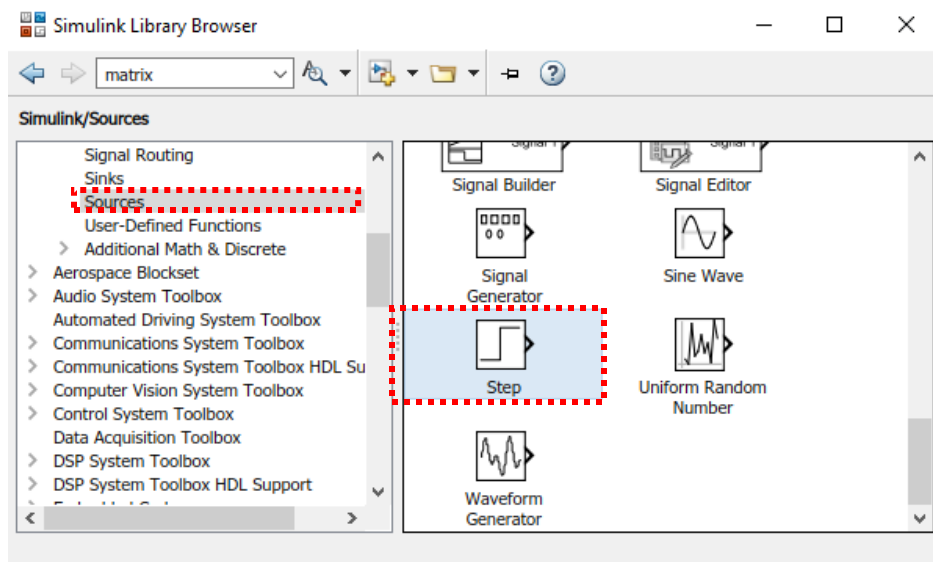


Ilustración 43: Selección del bloque de entrada. Fuente: Elaboración propia

Posteriormente en la ilustración 44 se configuran los parámetros para la entrada escalón, se coloca un valor de uno en el tiempo de salto debido al escalón unitario, en el valor final se coloca la cantidad de cuatro correspondiente al voltaje de alimentación (entrada del sistema).

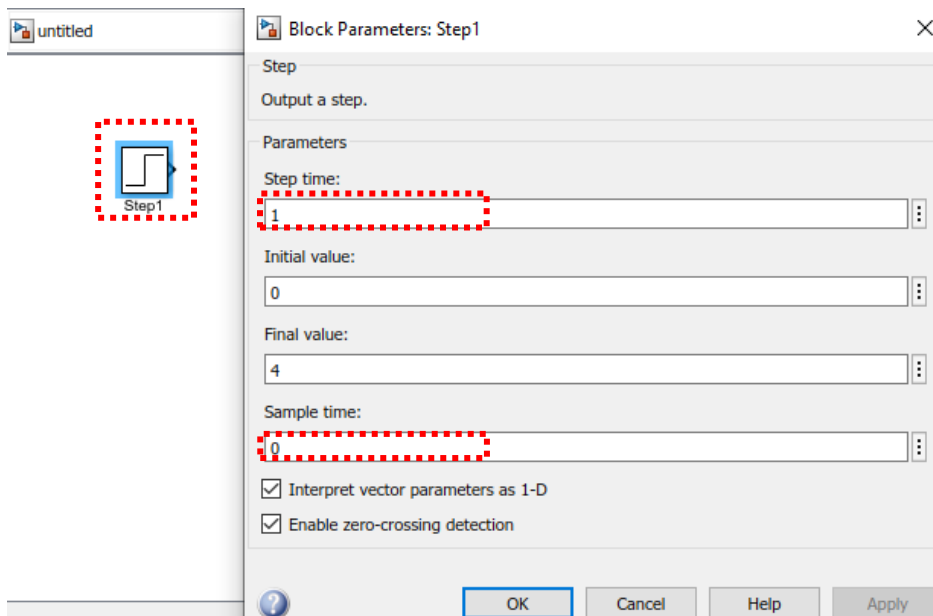


Ilustración 44: Asignación de parámetros a la entrada escalón del sistema. Fuente: Elaboración propia

La siguiente etapa para simular al modelo de estados en tiempo continuo es seleccionar en la librería el apartado “*Continuos*→ *State-Space*” como se muestra en la ilustración 45.

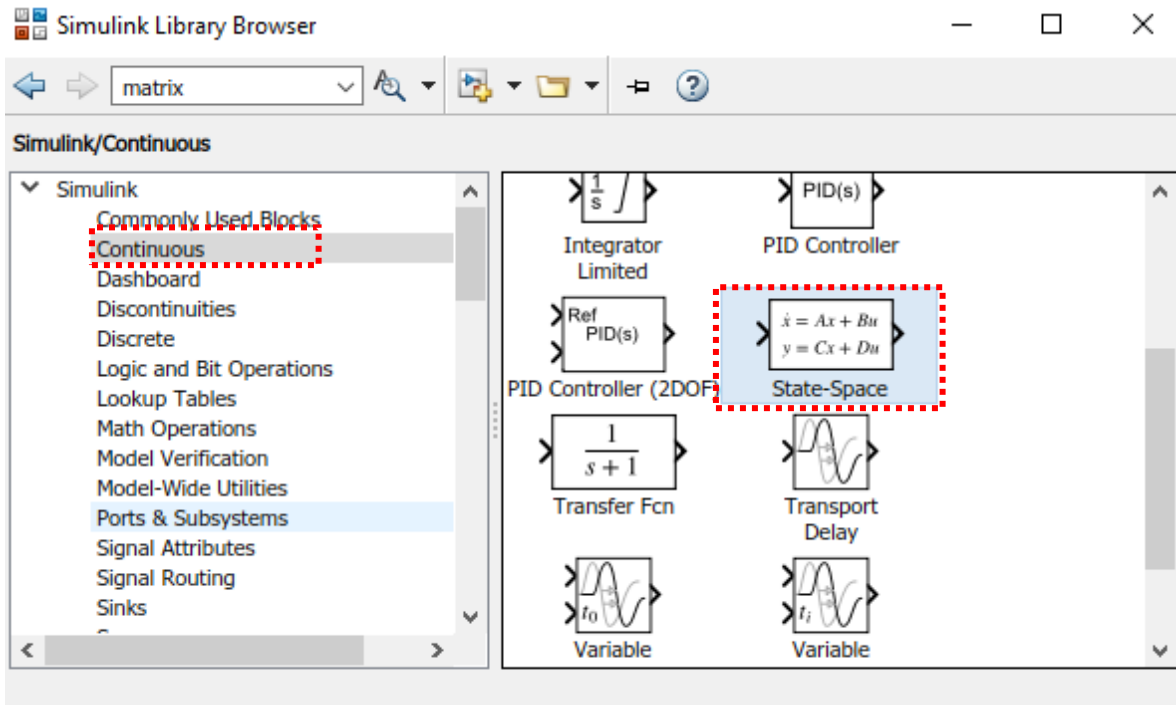


Ilustración 45: Selección de espacio de estados en tiempo continuo. Fuente: Elaboración propia

Lo siguiente es ingresar el modelo del sistema de la ecuación (32) a los parámetros del bloque “*State Space*”, mostrados en la ilustración 46.

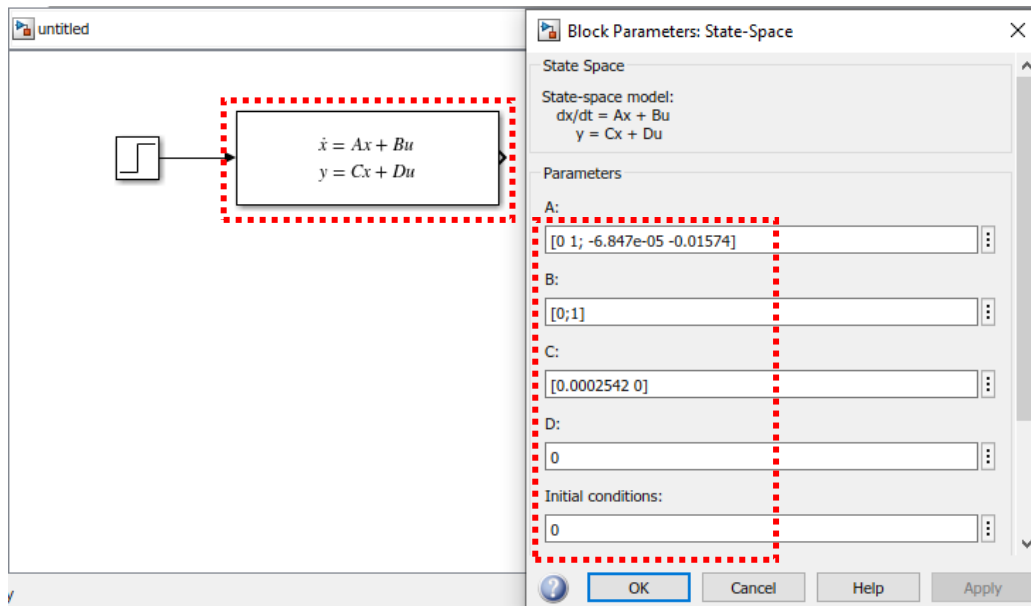


Ilustración 46: Asignación matricial al bloque de espacio de estados. Fuente: Elaboración propia

Se continúa con la asignación al modelo de estados en tiempo discreto nuevamente se selecciona en la librería de Simulink el apartado “Discrete → Discrete State-Space” como se muestra en la ilustración 47.

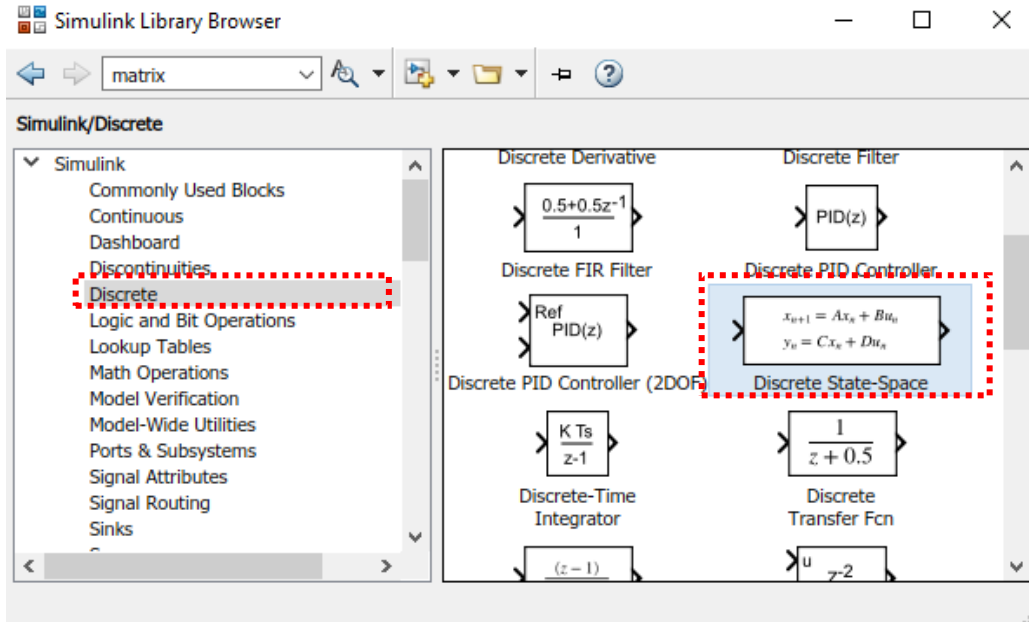


Ilustración 47: Selección de espacio de estados en tiempo discreto. Fuente: Elaboración propia

En la ilustración 48 se muestra la asignación de los datos matriciales correspondientes a la ecuación (43) en el bloque del sistema en tiempos discretos.

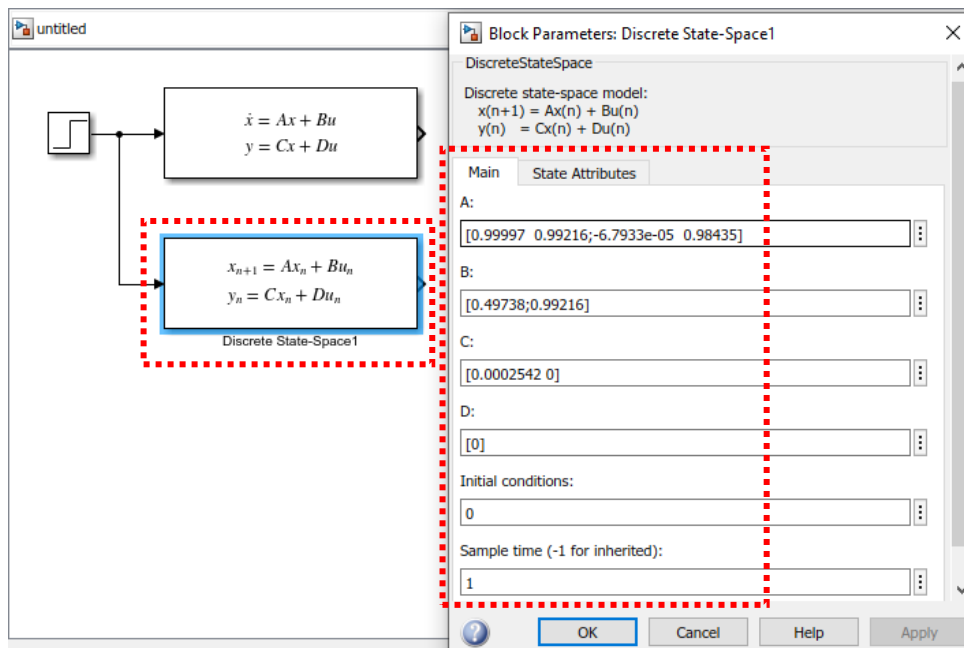


Ilustración 48: Asignación matricial al bloque de espacio de estados. Fuente: Elaboración propia

Finalmente, a la salida del sistema se agrega un bloque denominado “scope” para visualizar gráficamente el comportamiento de dichos sistemas. En la ilustración 49 se muestra el diagrama a bloques del sistema en tiempo continuo y discreto.

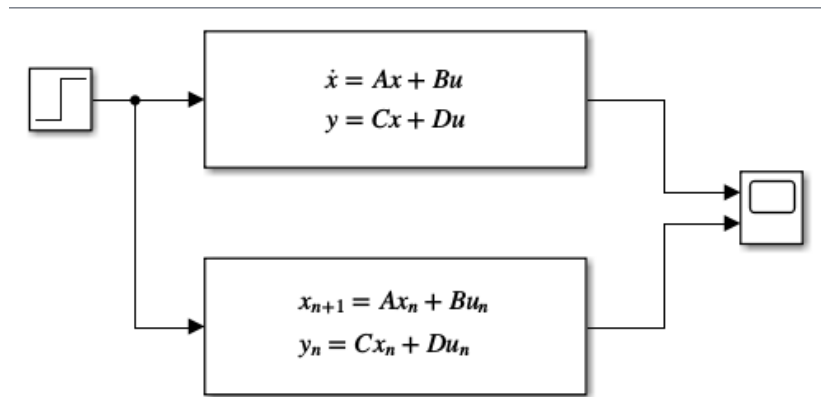


Ilustración 49: Representación a bloques del sistema. Fuente: Elaboración propia

Para una mejor visualización del sistema en tiempo continuo y discreto, se representan en una sola gráfica, validando la precisión de los cálculos obtenidos. (ilustración 50)

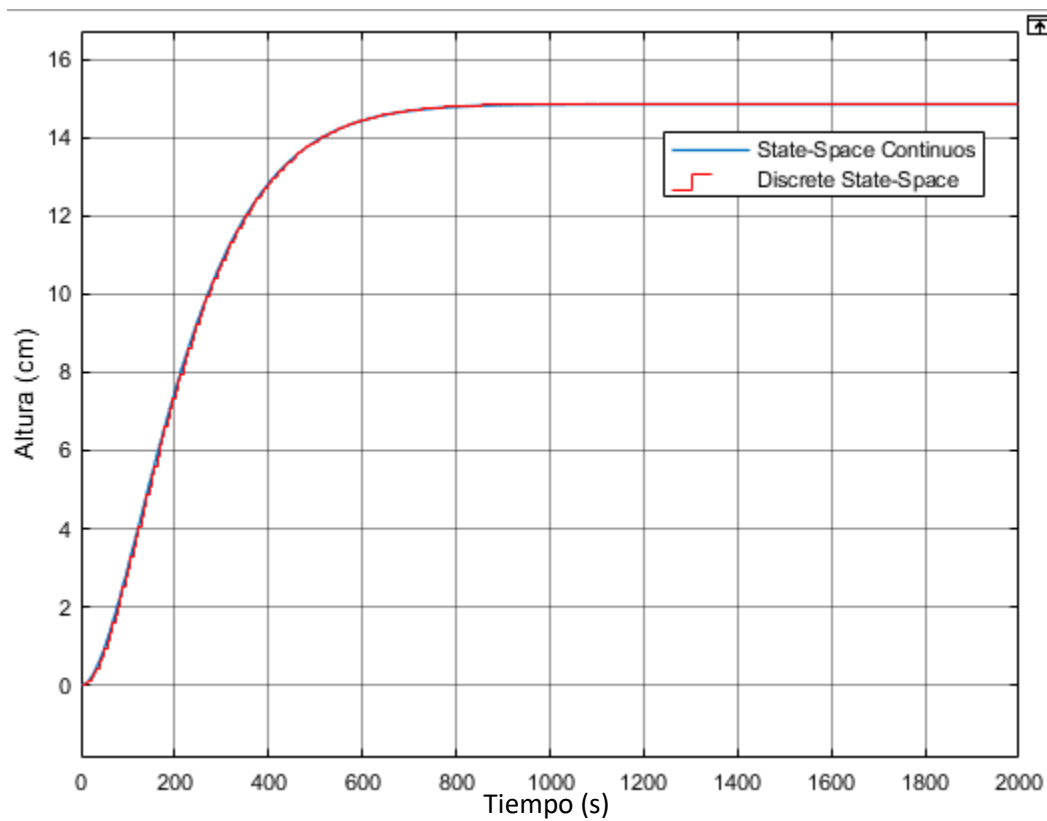


Ilustración 50: Comportamiento del sistema en tiempo continuo y tiempo discreto. Fuente: Elaboración propia

En la ilustración 51 se aumenta el tiempo de simulación demostrando que el modelo obtenido del sistema converge a la altura de 15 cm por lo tanto se continúa con el diseño del controlador usando el modelo del sistema en tiempo discreto(ec.43).

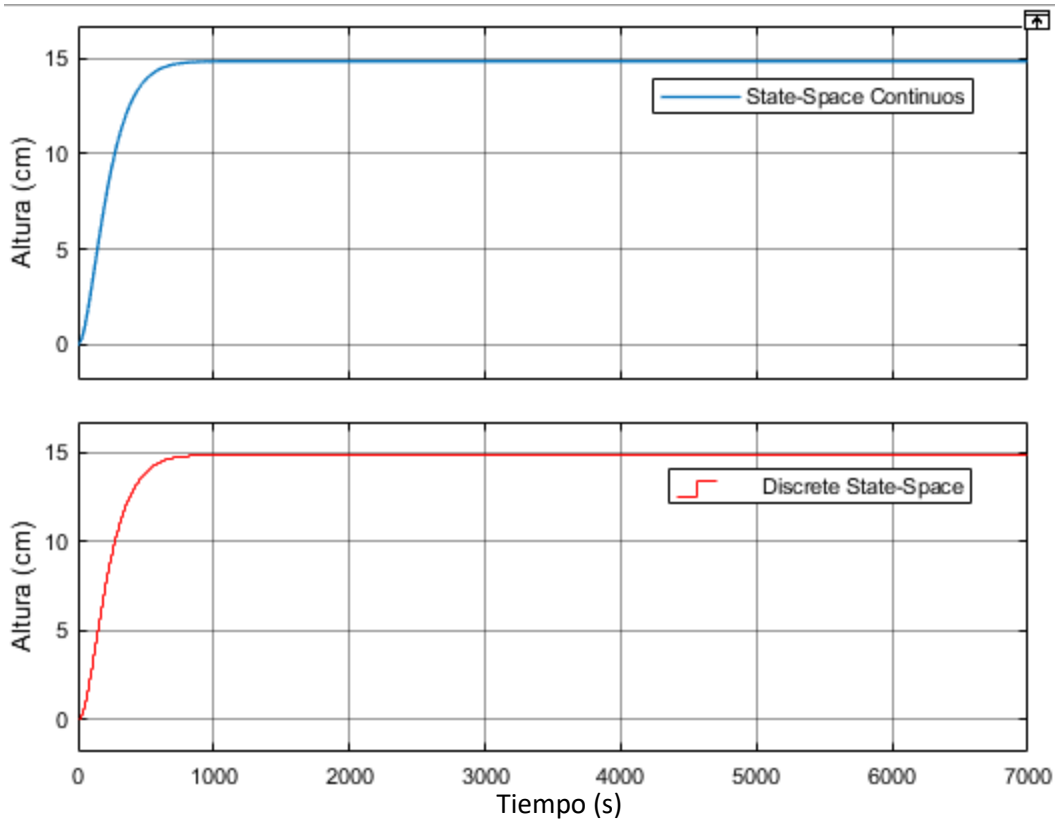


Ilustración 51: Respuesta del sistema con tiempo de simulación aumentado. Fuente: Elaboración propia

3.5 Diseño del modelo aumentado.

Para lograr llevar a cabo el diseño del MPC se desarrolla la metodología presentada por (L. Wang 2019) en donde se hace uso del sistema de modelo aumentado (ver ec. 20).

Este método consiste en trabajar con sistemas de una entrada y una salida (SISO), por lo tanto, el modelo aumentado del sistema queda de la siguiente forma (ec. 44).

$$\begin{bmatrix} \Delta x_m(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} 0.99997 & 0.99216 & 0 \\ -6.7933e^{-05} & 0.98435 & 0 \\ 0.00025419 & 0.00025221 & 1 \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} 0.49738 \\ 0.99216 \\ 0.00012643 \end{bmatrix} \Delta u(k)$$

$$y(k) = [0 \quad 0 \quad 1] \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} \quad [44]$$

Seguidamente se aplica la optimización del controlador predictivo donde se realiza el cálculo de la variable de control para la salida predicha del sistema. Esta predicción es descrita como el horizonte de predicción N_p . Lo siguiente es ajustar la trayectoria de la señal de control futura a través del horizonte de control N_c .

De acuerdo a la metodología (L. Wang 2019) la solución óptima para la señal de control está dada por:

$$\Delta U = (\Phi^T \Phi + R)^{-1} \Phi^T (R_s - Fx(ki)) \quad [45]$$

Donde se enuncia que el resultado de $(\Phi^T * R_s)$ es idéntico a la última columna de la matriz $(\Phi^T * F)$, y es así como se demuestra en las siguientes ecuaciones.

$$(\Phi^T * \Phi) = \begin{bmatrix} 0.009772 & 0.0086551 \\ 0.0086551 & 0.0076745 \end{bmatrix} \quad [46]$$

$$(\Phi^T * F) = \begin{bmatrix} 0.0013039 & 0.010346 & 0.3367 \\ 0.0011455 & 0.0091586 & 0.29091 \end{bmatrix} \quad [46.1]$$

$$(\Phi^T * R_s) = \begin{bmatrix} 0.3367 \\ 0.29091 \end{bmatrix} \quad [46.2]$$

Cabe resaltar que la obtención de estos cálculos matriciales se basa a través del algoritmo presentado en la literatura de (L. Wang 2019).

En los cálculos anteriores se realizó con un horizonte de control $N_c=2$ y un horizonte de predicción $N_p=20$, también es importante mencionar que en el diseño del MPC el horizonte de control debe ser menor al horizonte de predicción $N_c < N_p$.

Por último, se realiza la programación de los parámetros para el diseño del controlador en Matlab® (Anexo A-3), es decir, el modelo aumentado (ec. 44) y el modelo de predicción, (ec. 46, 46.1, 46.2).

4 RESULTADOS

4.1 Acoplamiento del sistema hidráulico.

En la ilustración 52 se muestra el armado del sistema hidráulico constituido por dos tanques conectados en cascada y un tanque de almacenamiento, en él se incorpora un sensor de presión diferencial para medir el nivel del fluido (altura).



Ilustración 52: Armado del sistema hidráulico Fuente: Elaboración propia

4.2 Análisis de resultados.

En esta sección se analiza la respuesta del controlador para distintas referencias (set-point) proponiendo un horizonte de control (H_c) corto, porque este debe ser menor al horizonte de predicción (H_p)

En la ilustración 53 se muestra el resultado de trabajar con horizontes de predicción cortos, se observa que la respuesta del controlador tiene un comportamiento inestable, sobrepasando los límites de referencia.

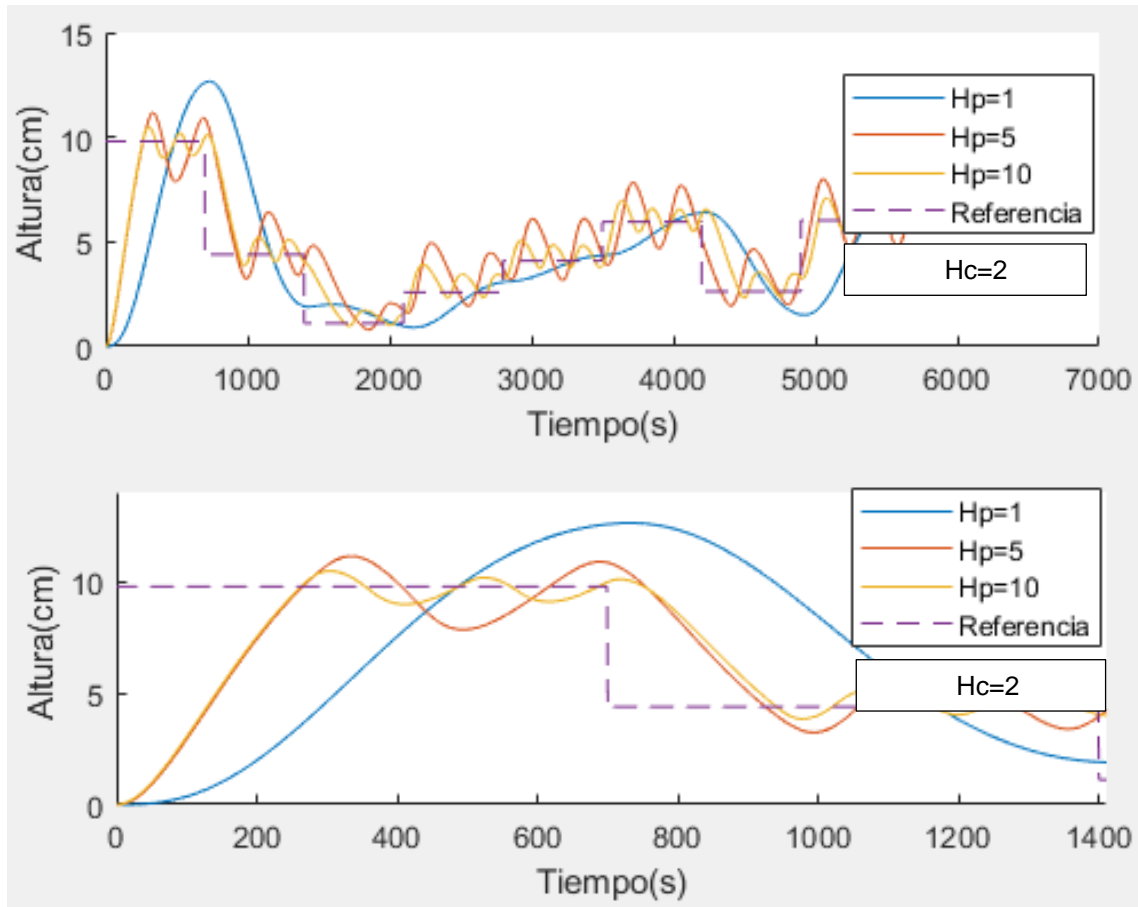


Ilustración 53: Controlador MPC con H_p cortos

En la ilustración 54 se muestra la respuesta del control con un $H_c=2$, $H_p=20,25,30$ en un mismo gráfico para tener una mejor visualización y comparación de diferentes valores propuestos. Así como un acercamiento a escala para un mejor análisis del controlador.

En la ilustración 54 se puede observar que el horizonte de control (H_c) corto ocasiona cambios mínimos en la acción de control, es decir, el controlador logra alcanzar al set-point de manera inmediata.

También se hace notar que el horizonte de predicción (H_p) igual a 20 toma mayor tiempo en acoplarse a la referencia, sin embargo, el horizonte de predicción (H_p) igual a 30 aumenta la capacidad de predicción del controlador, esto quiere decir que un horizonte de predicción (H_p) corto aumenta el tiempo en predecir la salida de la planta.

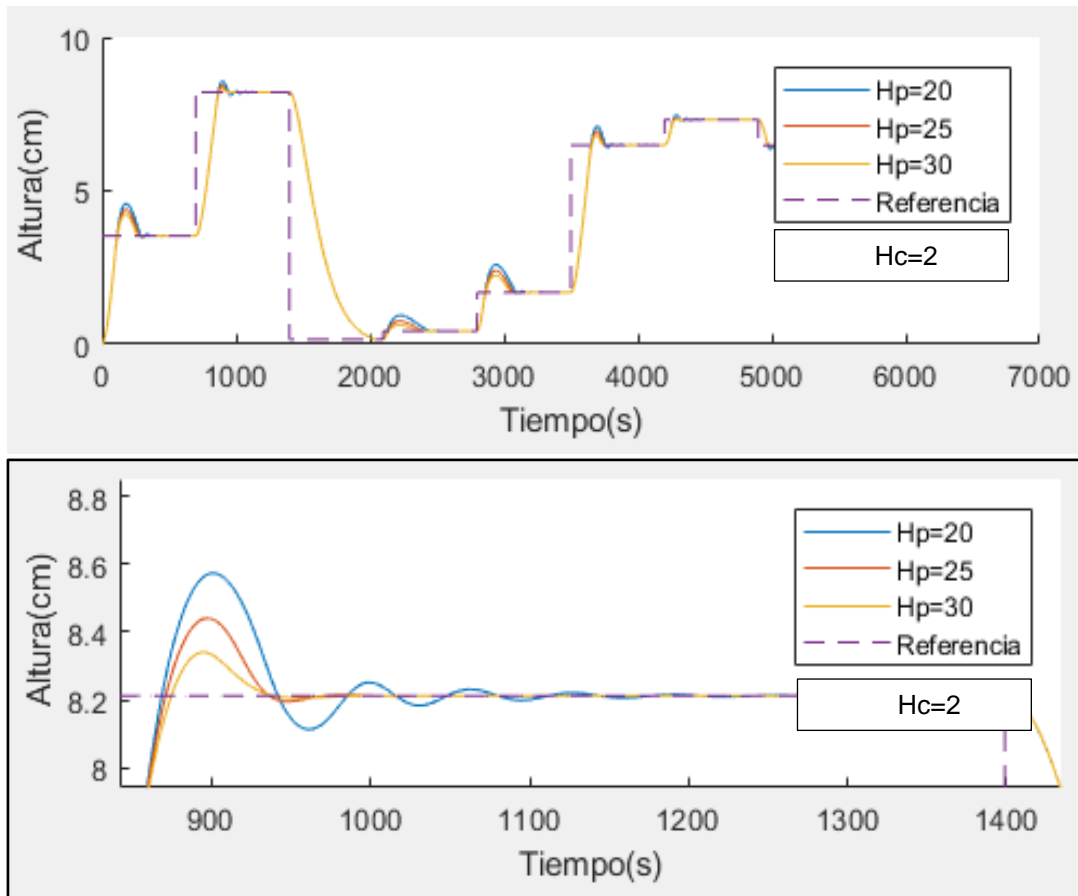


Ilustración 54: Respuesta del controlador con $H_c=2$, $H_p=20,25,30$. Fuente: Elaboración propia

De la misma manera en la ilustración 55 se analiza la respuesta del controlador con un acercamiento a escala, en esta ocasión se utiliza un horizonte de control (H_c) igual a 20, se hace notar que un horizonte de control largo produce cambios agresivos, es decir, causa la oscilación del sistema a la acción del control.

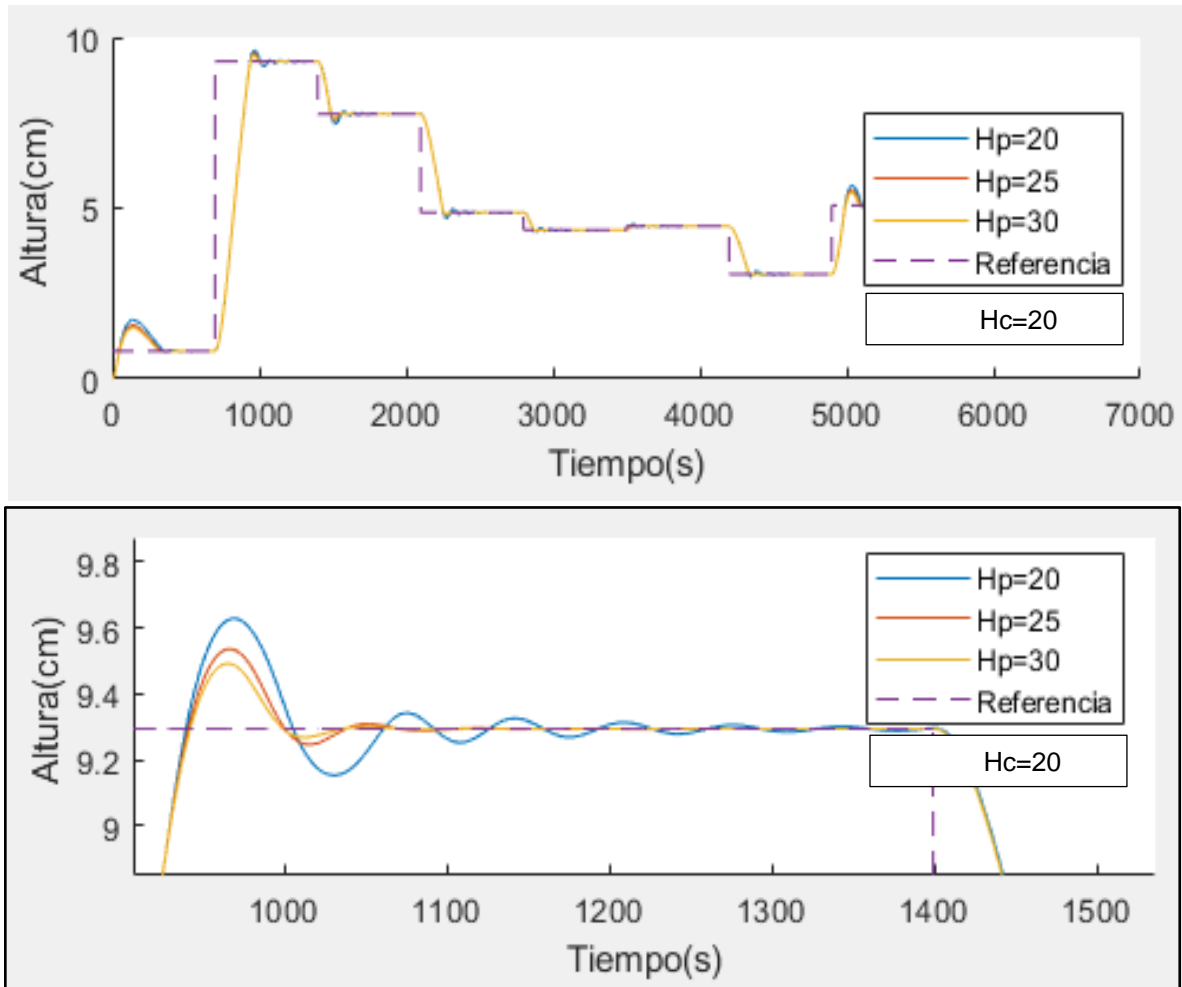


Ilustración 55: Respuesta del controlador con $H_c=20$, $H_p=20,25,30$. Fuente: Elaboración propia

Un último análisis consiste en someter el controlador con un horizonte de control (H_c) corto igual a 2 y un horizonte de predicción (H_p) igual a 20,25 y 50 en el mismo gráfico para comparar la respuesta a la acción de control, en la ilustración 56 se observa que el horizonte de predicción largo ($H_p=50$) converge de manera inmediata, esto es debe a que entre más grande sea la diferencia entre los valores de H_c y H_p , mejor será el control de nivel en la planta.

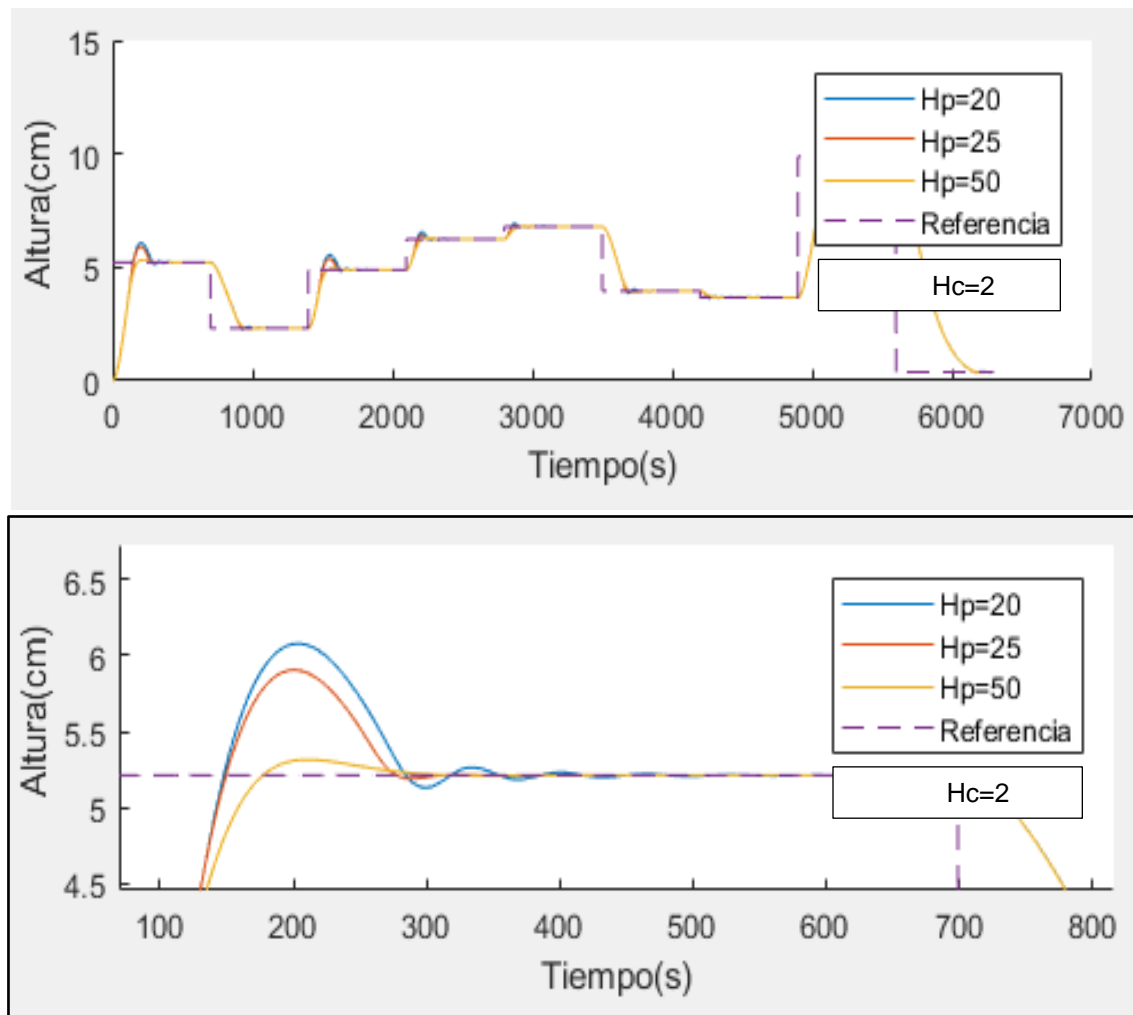


Ilustración 56: Respuesta del controlador con $H_c=2$, $H_p=20,25,50$. Fuente: Elaboración propia

4.2.1 Sintonización del controlador PID en Matlab®

El diseño del controlador PID se realiza sobre la función de transferencia del sistema empleando la herramienta PIDtuner de MATLAB®, esta herramienta permite sintonizar un controlador PID de manera gráfica.

Para ingresar a esta herramienta es necesario tener el modelo del sistema almacenado en el workspace (G) e ingresar el comando “*pidTuner*” como se muestra en la ilustración 57:

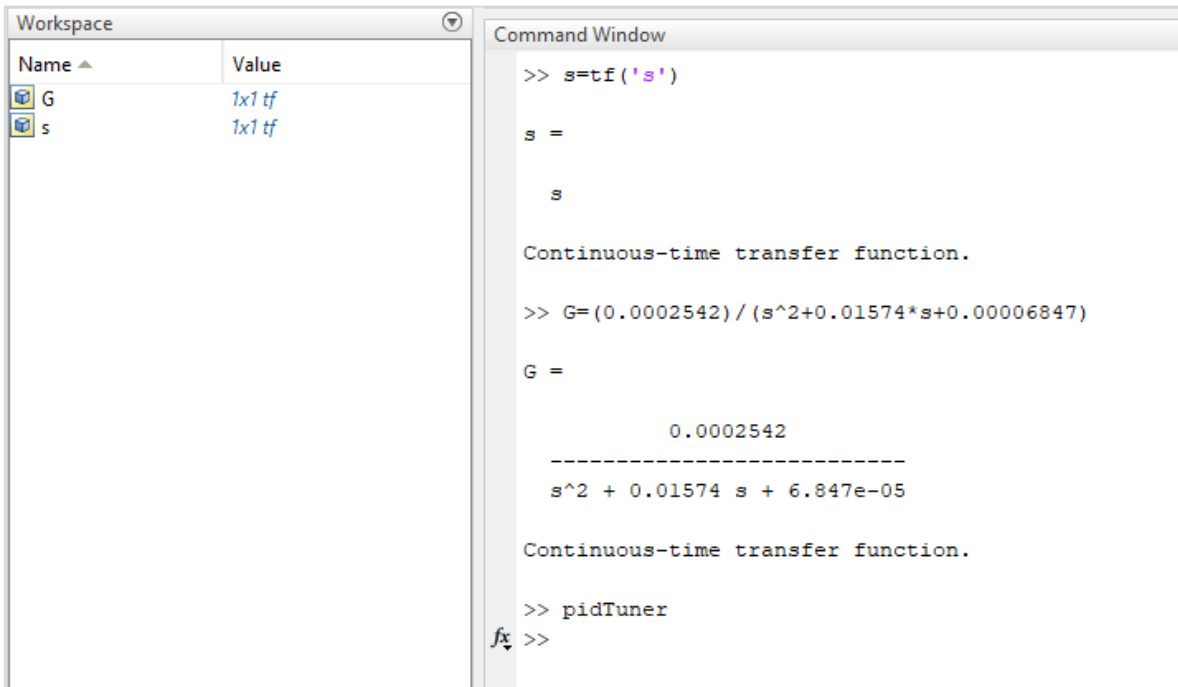


Ilustración 57: Ingreso a la herramienta pidTuner. Fuente: Elaboración propia

Al momento de ingresar el comando en el entorno de Matlab® se aprecia una ventana del pidTuner (ilustración 58).

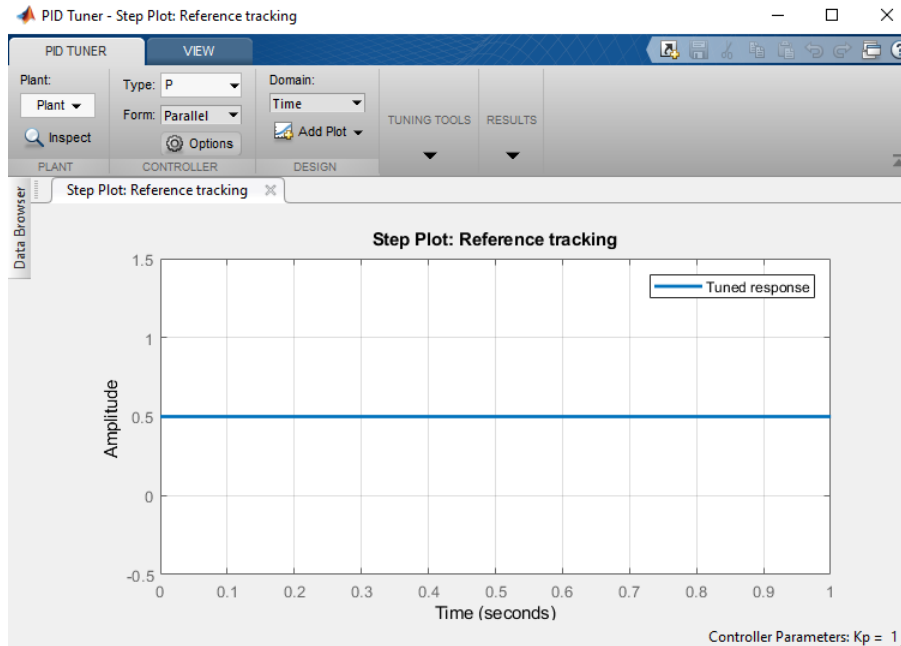


Ilustración 58:Entorno de la herramienta pidTuner. Fuente: Elaboración propia

Continuando con la sintonización del controlador se requiere importar el modelo del sistema, esto se realiza en la opción de “Plant” → “Import”. (ilustración 59).

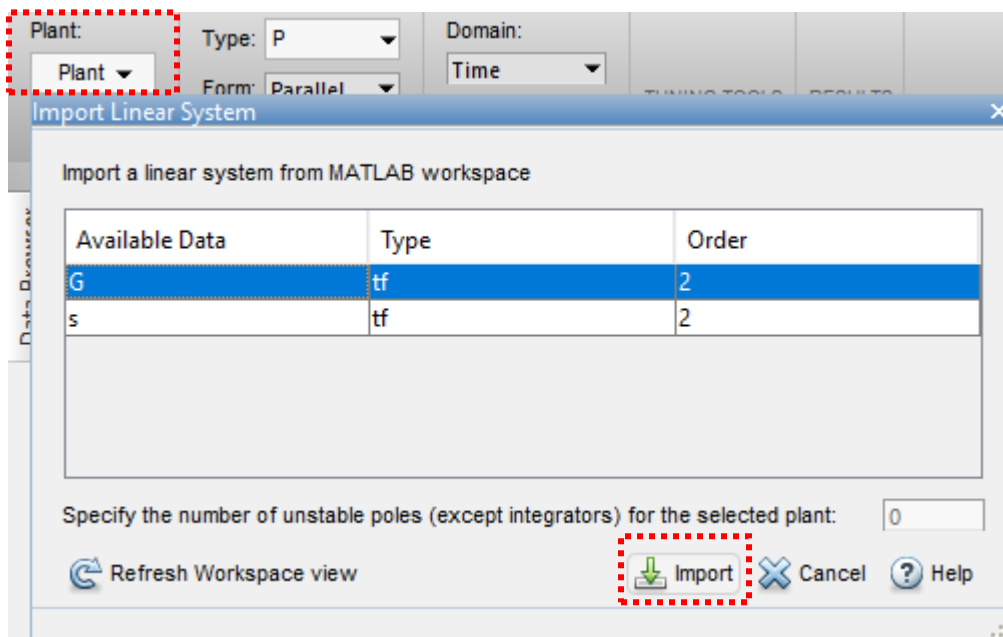


Ilustración 59: Importación de la planta. Fuente: Elaboración propia

Posteriormente, se elige el tipo de controlador deseado, en este proyecto se utiliza el PID para compararlo con el controlador MPC. En la ilustración 60 se observa que se selecciona el icono “Type” → “PID”.

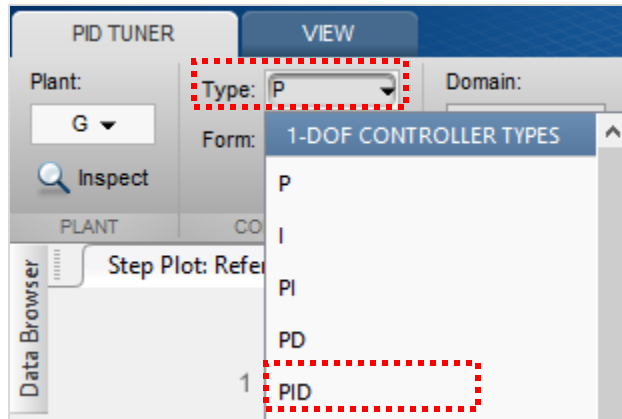


Ilustración 60: Selección del Controlador PID. Fuente: Elaboración propia

En seguida de seleccionar el tipo de controlador PID nos arroja una ventana con los parámetros del controlador, estos parámetros son los que se utilizarán en la simulación. (ilustración 61)

Controller Parameters	
	Tuned
Kp	0.98785
Ki	0.0054637
Kd	39.4542

Ilustración 61: Parámetros del controlador PID. Fuente: Elaboración propia

4.2.2 Simulación del controlador PID

De acuerdo a los parámetros obtenidos, se realiza un diagrama en bloques en Simulink de Matlab® para visualizar el comportamiento del controlador. Se ingresan los parámetros en el bloque “PID(s)”.

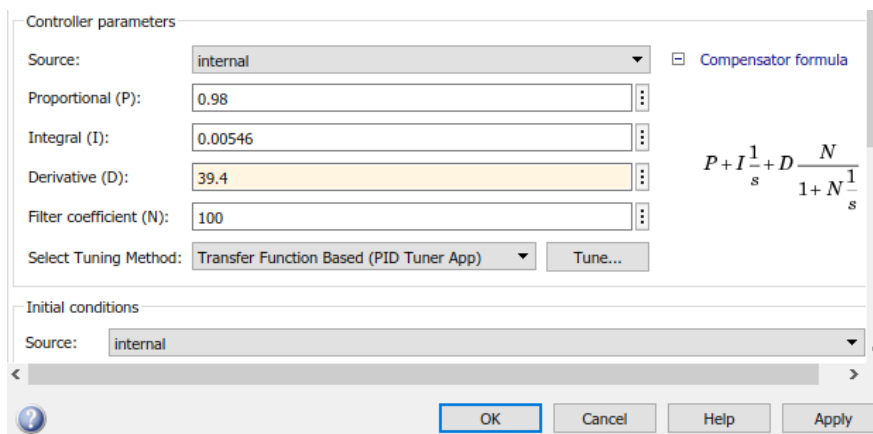


Ilustración 62:Asignación de parámetros PID en Simulink. Fuente: Elaboración propia

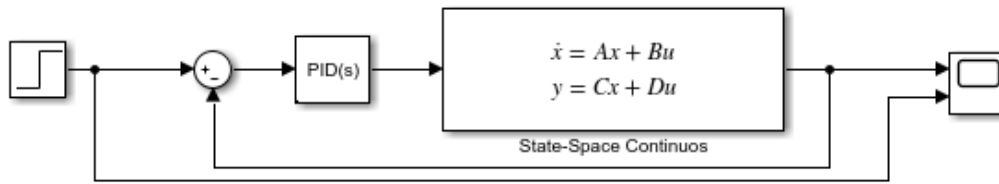


Ilustración 63: Simulación del controlador PID. Fuente: Elaboración propia

El comportamiento del controlador PID se observa en la ilustración 64, la finalidad de sintonizar este controlador es compararlo con el comportamiento del MPC.

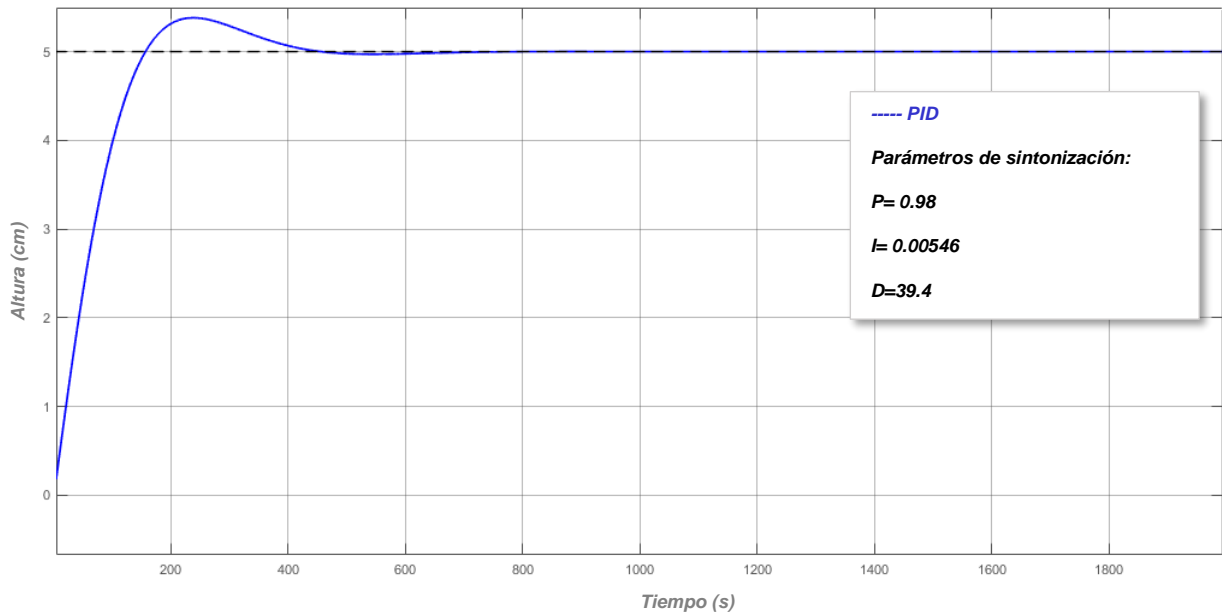


Ilustración 64: Sintonización del controlador PID. Fuente: Elaboración propia

4.2.3 Desempeño del controlador MPC vs PID

En este apartado el modelo del sistema es sometido a un controlador PID, con la finalidad de evaluar los inconvenientes, ventajas, desventajas que presenta cada controlador.

En la ilustración 65 se observa el comportamiento del controlador PID respecto al MPC, en el MPC se aplicó un horizonte de control igual a 2 y un horizonte de predicción igual a 30 con una referencia a 5 (altura en cm) del sistema, se denota que el controlador MPC logra establecerse a los 428 s con un sobreimpulso del 2%.

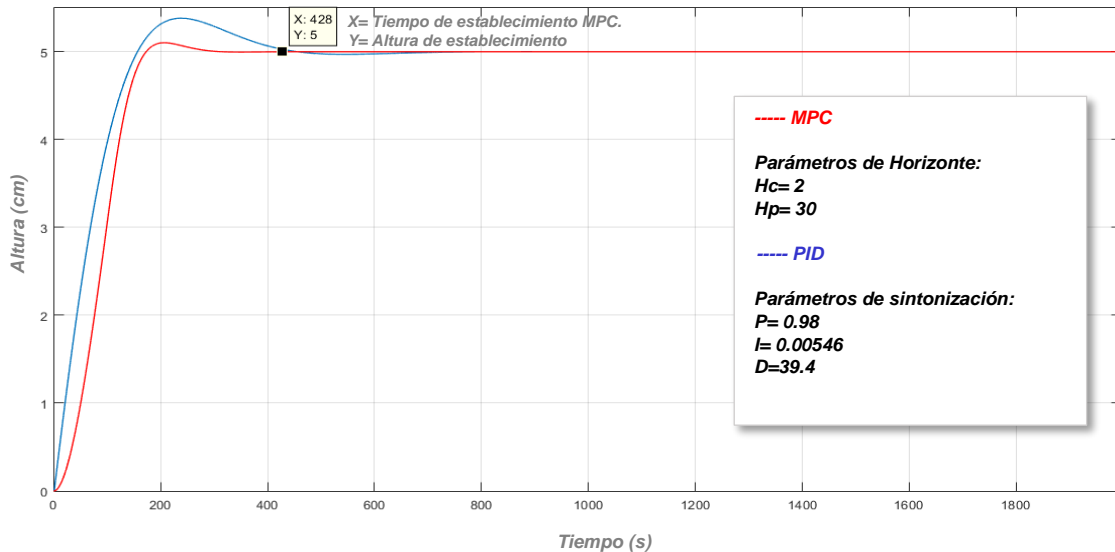


Ilustración 65: Control PID vs MPC con referencia a 5cm altura. Fuente: Elaboración propia

En la ilustración 66 se observa que el controlador PID logra establecerse a los 793s, con una diferencia de establecimiento del 46% respecto al MPC y un sobre impulso del 7.6%.

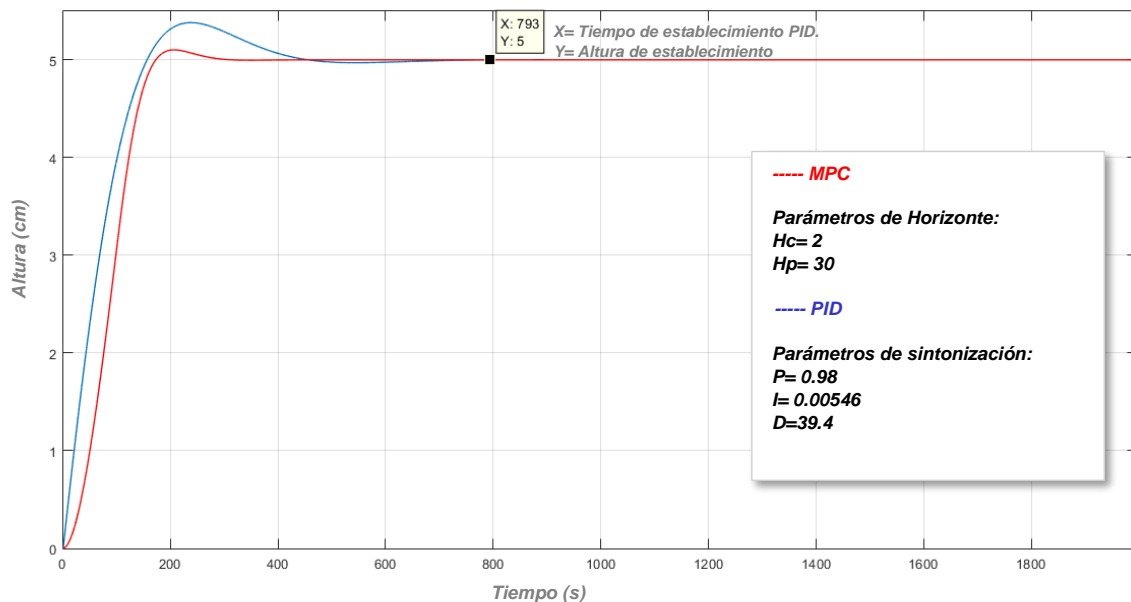


Ilustración 66: Control PID vs MPC con referencia a 5cm altura (tiempo establecimiento PID) Fuente: Elaboración propia

La respuesta transitoria de los controladores PID vs MPC se detallan en la Tabla no. 1

El sobre impulso máximo se calcula comúnmente como, un tanto por ciento del valor final de la respuesta, mediante la siguiente formula:

$$\text{Máximo sobre impulso porcentual} = \frac{c(tp) - c(\infty)}{c(\infty)} * 100\% \quad [47]$$

Donde:

$c(tp)$: representa el valor máximo o pico de la respuesta.

$c(\infty)$: representa el valor final de la respuesta.

Respuesta transitoria	PID	MPC
Sobre impulso (Mp)	7.6%	2%
Tiempo retardo (td)	56.4 s	88s
Tiempo de subida (tr)	157 s	173s
Tiempo pico (tp)	241 s	180s
Tiempo establecimiento (ts)	793 s	428s

Tabla no.1: Respuesta transitoria de controlador PID vs MPC.

Por otra parte, es importante mencionar que la señal de error se entiende como la diferencia entre la señal de entrada (referencia) y la salida proporcionada por el controlador, los errores en un sistema de control se atribuyen a factores como cambios en la referencia de entrada. En la ilustración 67 se observa la comparación de la señal de error entre el PID vs MPC, donde el MPC tiende rápidamente a cero en 326s mientras que el PID tarda 506 s.

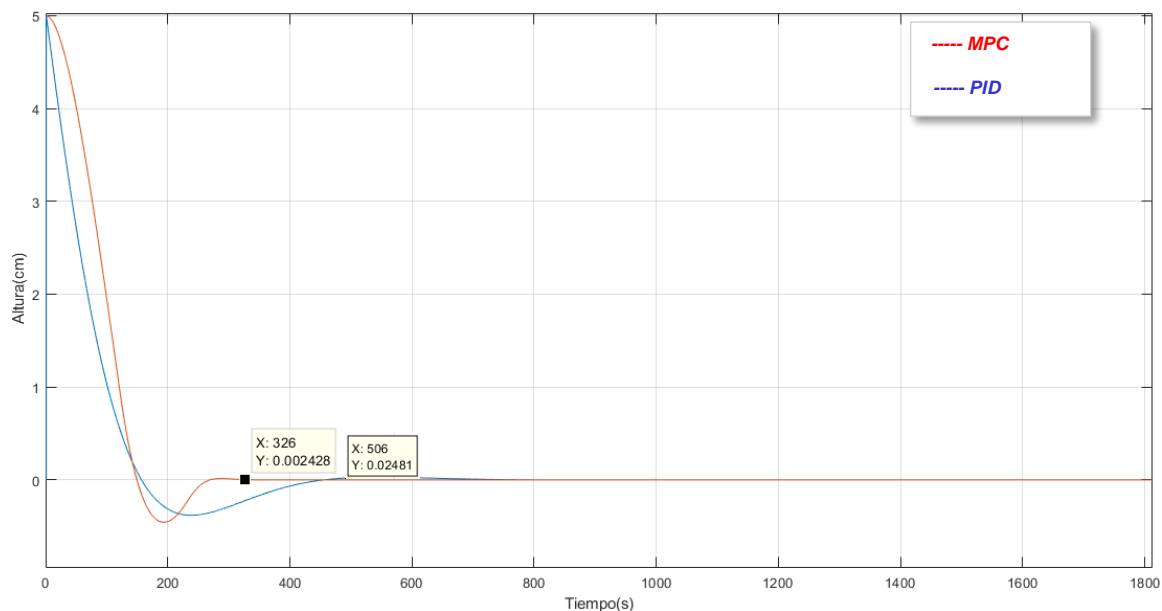


Ilustración 67: Señal de error PID vs MPC

5 CONCLUSIONES

-El método de identificación de sistemas es muy factible de utilizar porque requiere de datos experimentales para llevar a cabo un proceso más viable y preciso en la obtención del modelo.

-El controlador MPC comienza a controlar anticipándose al cambio de referencia demostrando lo que señala la teoría, esto se puede observar en la simulación proponiendo diferentes valores de H_p , por el hecho de que el controlador responde de manera efectiva ante diferentes referencias

-Al programar valores de horizonte de control mayores al horizonte de predicción la acción de control no se ejecuta, por lo tanto, se concluye que el horizonte de control (H_c) debe ser menor al horizonte de predicción (H_p).

-Cuando se trabaja con un (H_c) corto la acción de control presenta cambios mínimos, es decir, el controlador logra alcanzar al set-point de manera inmediata.

-A mayor diferencia entre los valores de H_c y H_p , mejor será el control de nivel en la planta.

-De acuerdo a los resultados, cuando el sistema es controlado por el PID se presentan sobre impulsos, en la situación del MPC se consigue eliminar en su totalidad el sobre impulso.

-El tiempo que tarda en establecer el control PID es de 793 segundos, mientras que el del controlador MPC es de 428 segundos evidenciando la eficiencia del Control Predictivo por Modelo (MPC).

-Se define que el controlador MPC es considerado en este proyecto como el mejor controlador, ya que las oscilaciones son prácticamente nulas en la salida del sistema.

5.1 Recomendaciones.

La recomendación para un trabajo futuro es la implementación del controlador a través de una interfaz gráfica con fines prácticos, a pesar de que existen varios factores con los cuales se calibra el controlador, es esencial considerar el horizonte de control, horizonte de predicción, sensor de presión diferencial por que con estos parámetros se obtiene una mejor sintonización del sistema.

REFERENCIAS.

Ioana, N. (2023). Estrategias avanzadas de control predictivo de modelos para procesos de evaporación en las industrias farmacéuticas. ScienceDirect, (10112), pp.1-8.

Kong, Y. (2022). MPC: un marco de control predictivo basado en datos y basado en el conocimiento para robots aéreos. IEEE Robotics and Automation Letters.

Sánchez, D. (2021). Estudio del diseño del control de nivel en un proceso de dos tanques acoplados mediante control predictivo (Ingeniero). PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ.

Bautista, O. (2022). Implementación de un MPC en un intercambiador de calor usando LabVIEW. PADI.

Wang, L. (2019). Model Predictive Control System Design and Implementation Using MATLAB®. Australia: Springer.

Zhang, B., Lin, F., Zhang, C., Liao, R., and Wang, Y.-X. (2020). Design and implementation of model predictive control for an open-cathode fuel cell thermal management system. Renewable Energy.

Sawant, P., Ho, E., and Pfafferott, J. (2020). Application and analysis of a model based controller for cooling towers in compression chiller plants.

Veloni, A. (2021). Digital Control Systems: Theoretical Problems and Simulation Tools. México: CRC Press.

Larco, A. (2021). INTRODUCCIÓN A LA IDENTIFICACIÓN DE SISTEMAS. , (1), pp.1-99

Richard D., Robert B. (2020). Ingeniería de control: Sistema de control moderno 13th edición. California: Pearson.

Norman, S. (2019). Control Systems Engineering. México: John Wiley & Sons.

M. Farina y G. Betti (2014). Distributed predictive control of continuous time systems, Systems & Control Letters.

Arnáez, E. (2014). Introducción al control moderno, Enfoque práctico del control moderno: con aplicaciones en Matlab (pp. 1-23). Lima: Universidad Peruana de Ciencias Aplicadas.

A. Harivignesh. (2014). System Identification and Design of controllers For a Hybrid Tank System, de IEEE International Conference on Innovations in Engineering and Technology.

J. Cyril, Design and implementation of model predictive control for a three-tank hybrid system, International Journal of Computer Science and Electronics Engineering.

Gaviño, R. (2010). Introducción a los sistemas de control: Conceptos, aplicaciones y simulación con MATLAB. México: Pearson.

Dorf, R, Bishop, R. (2005). Sistemas de control moderno. Madrid: Pearson.

Camacho. E.F., and Bordons C.,(2004). Model Predictive Control (2da ed). London: Springer – Verlag.

Bemporad, A., Ricker, N. L., and Owen, J. G. (2004). Model predictive control new tools for design and evaluation. In Proceedings of the 2004 American Control Conference, IEEE.

Ogata, K. (2003). Ingeniería de control moderna. México: Pearson.

Tajatsy H., Itoh T. y Araki M. (1998) "Future needs for the control theory in industries. Report and topics of the control technology survey in Japanese industry." Journal of Process Control.

Clarke D. W, Mohtadi C y Tuffs P.S. (1987) "Predictive Control Part I. The Basic Algorithm". Automatic.

Anexo A-1 Especificaciones técnicas de sensor de presión MPX5010DP



Freescall Semiconductor
Data Sheet: Technical Data

Pressure

MPX5010
Rev 13, 10/2012

Integrated Silicon Pressure Sensor On-Chip Signal Conditioned, Temperature Compensated and Calibrated

The MPxx5010 series piezoresistive transducers are state-of-the-art monolithic silicon pressure sensors designed for a wide range of applications, but particularly those employing a microcontroller or microprocessor with A/D inputs. This transducer combines advanced micromachining techniques, thin-film metallization, and bipolar processing to provide an accurate, high level analog output signal that is proportional to the applied pressure. The axial port has been modified to accommodate industrial grade tubing.

Features

- 5.0% Maximum Error over 0° to 85°C
- Ideally Suited for Microprocessor or Microcontroller-Based Systems
- Durable Epoxy Unibody and Thermoplastic (PPS) Surface Mount Package
- Temperature Compensated over -40° to +125°C
- Patented Silicon Shear Stress Strain Gauge
- Available in Differential and Gauge Configurations
- Available in Surface Mount (SMT) or Through-hole (DIP) Configurations

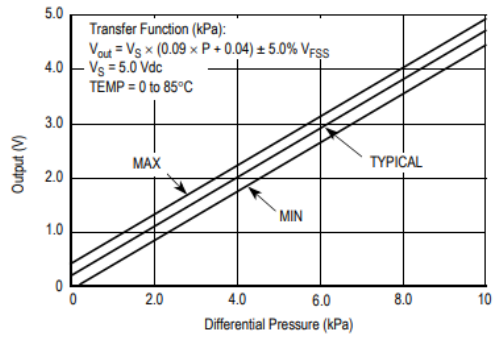
MPX5010 MPXV5010 MPVZ5010 Series

0 to 10 kPa (0 to 1.45 psi)
(0 to 1019.78 mm H₂O)
0.2 to 4.7 V Output

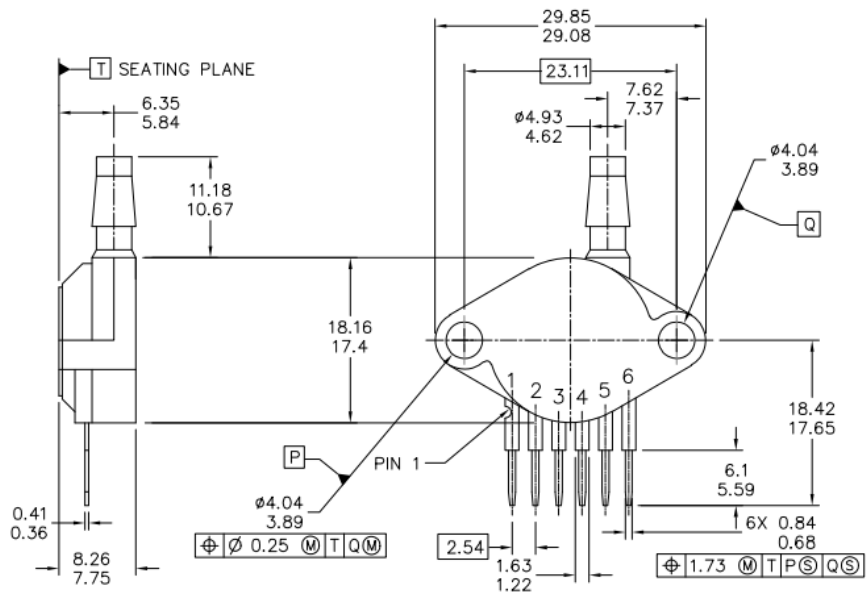
Application Examples

- Hospital Beds
- HVAC
- Respiratory Systems
- Process Control
- Washing Machine Water Level Measurement (Reference AN1950)
- Ideally Suited for Microprocessor or Microcontroller-Based Systems
- Appliance Liquid Level and Pressure Measurement

Characteristic	Symbol	Min	Typ	Max	Unit
Pressure Range	P _{OP}	0	—	10 1019.78	kPa mm H ₂ O
Supply Voltage ⁽¹⁾	V _S	4.75	5.0	5.25	Vdc
Supply Current	I _o	—	5.0	10	mAdc
Minimum Pressure Offset ⁽²⁾ @ V _S = 5.0 Volts	V _{off}	0	0.2	0.425	Vdc
Full Scale Output ⁽³⁾ @ V _S = 5.0 Volts	V _{FSS}	4.475	4.7	4.925	Vdc
Full Scale Span ⁽⁴⁾ @ V _S = 5.0 Volts	V _{FSS}	4.275	4.5	4.725	Vdc
Accuracy ⁽⁵⁾	—	—	—	±5.0	%V _{FSS}
Sensitivity	V/P	—	450 4.413	—	mV/mm mV/mm H ₂ O
Response Time ⁽⁶⁾	t _R	—	1.0	—	ms
Output Source Current at Full Scale Output	I _{O+}	—	0.1	—	mAdc
Warm-Up Time ⁽⁷⁾	—	—	20	—	ms
Offset Stability ⁽⁸⁾	—	—	±0.5	—	%V _{FSS}



PACKAGE DIMENSIONS



Anexo A-2 Configuración del sensor de presión diferencial MPX5010DP

El código fue desarrollado a través del software Arduino® debido a que este tipo de sensor entrega en su salida un voltaje de 0v-5v ideal para microcontroladores. Los parámetros del sensor se ajustaron de acuerdo a la hoja de especificaciones técnicas del fabricante. (anexo A-1)

```
double Level, Vout,P,Vs=5.0;
double aux;
double tolP=0.01;
int i, rho = 997;
double g=9.8;

void setup()
{
  Serial.begin(9600);
  Serial.println("CLEARDATA");
  Serial.println("LABEL, Hora, Nivel de Altura");
  Serial.println("RESETTIMER");
}

void loop()
{
  aux=0;
  for(i=0;i<10;i++){
    aux = aux + (float(analogRead(A0))*5.0/1023.0); //v
    delay(5);
  }

  Vout=aux/10.0;

  //Presión en Kpa según Datasheet

  P = ( Vout - 0.04*Vs ) / ( 0.09 * Vs ) + tolP; //kPa

  Level = ((P*1000)/(rho*g))*100;
  Level = 1.0628*Level - 0.06661;

  Serial.print("\n\nVoltaje:");
  Serial.print(Vout);
  Serial.println(" v");
  Serial.print("Presión:");
  Serial.print(P);
  Serial.println(" kPa");
  Serial.print("Nivel:");
  Serial.print(Level);
  Serial.println(" cm");

  delay(500);

  //Envio de Datos a Excel

  double lectura=Level;
  Serial.print("DATA, TIME, ");
  Serial.println(lectura);
  delay(10);
}
```

Anexo A-3 Código del controlador MPC a través de Matlab®

```
close all;
clear all;
clc;

Ap=[0.99997 0.99216;-6.7933e-05 0.98435];
Bp=[0.49738;0.99216];
Cp=[0.0002542 0];
Dp=0;

Np=20;
Np1=25; %20 25 30
Np2=30;

Nc=2; %2 3 4

[Phi_Phi,Phi_F,Phi_R,A_e, B_e,C_e]= Ganancias_MPC(Ap,Bp,Cp,Nc,Np);
[Phi_Phi1,Phi_F1,Phi_R1,A_e1, B_e1,C_e1]= Ganancias_MPC(Ap,Bp,Cp,Nc,Np1);
[Phi_Phi2,Phi_F2,Phi_R2,A_e2, B_e2,C_e2]= Ganancias_MPC(Ap,Bp,Cp,Nc,Np2);

[n,n_in]=size(B_e);

xm=[0;0];
xm1=[0;0];
xm2=[0;0];

Xf=zeros(n,1);
Xf1=zeros(n,1);
Xf2=zeros(n,1);

N_sim=700;

u=0; % u(k-1) =0
u_1=0;
u_2=0;

y=0;
y_1=0;
y_2=0;

%r1=ones(N_sim*6,1)*15;
```

```

for i=0:8
r=10*rand*ones(N_sim+i*N_sim,1);

for kk=1+i*N_sim:N_sim+i*N_sim;
DeltaU=inv(Phi_Phi+0.1*eye(Nc,Nc))*(Phi_R*r(kk)-Phi_F*Xf);
deltau=DeltaU(1,1);
u=u+deltau;
%r1(kk)=r;

if (u>4)
    u=4;
else if (u<=0)
    u=0;
else
    u=u;
end
end

u1(kk)=u;
y1(kk)=y;
xm_old=xm;
xm=Ap*xm+Bp*u;
y=Cp*xm;
Xf=[xm-xm_old;y];
%r1(1, kk)=r(kk+i*N_sim);
r1(kk)=r(kk-i*N_sim);

end

for kk1=1+i*N_sim:N_sim+i*N_sim;
DeltaU1=inv(Phi_Phi1+0.1*eye(Nc,Nc))*(Phi_R1*r(kk1)-Phi_F1*Xf1);
deltau1=DeltaU1(1,1);
u_1=u_1+deltau1;
%r1(kk)=r;

if (u_1>4)
    u_1=4;
else if (u_1<=0)
    u_1=0;
else
    u_1=u_1;
end
end

u1_1(kk1)=u_1;
y1_1(kk1)=y_1;
xm_old1=xm1;
xm1=Ap*xm1+Bp*u_1;
y_1=Cp*xm1;
Xf1=[xm1-xm_old1;y_1];

r1(kk1)=r(kk1-i*N_sim);
end

```

```

for kk2=1+i*N_sim:N_sim+i*N_sim;
DeltaU2=inv(Phi_Phi2+0.1*eye(Nc,Nc))*(Phi_R2*r(kk2)-Phi_F2*Xf2);
deltau2=DeltaU2(1,1);
u_2=u_2+deltau2;
%r1(kk)=r;

if (u_2>4)
    u_2=4;
else if (u_2<=0)
    u_2=0;
else
    u_2=u_2;
end
end

u1_2(kk2)=u_2;
y1_2(kk2)=y_2;
xm_old2=xm2;
xm2=Ap*xm2+Bp*u_2;
y_2=Cp*xm2;
Xf2=[xm2-xm_old2;y_2];

r1(kk2)=r(kk2-i*N_sim);

end

k=0:(N_sim-1+i*N_sim);
%fplot(y1,[0 3000])

end

figure
subplot(211)
hold
plot(k,y1,'DisplayName','Hp=20')
plot(k,y1_1,'DisplayName','Hp=25')
plot(k,y1_2,'DisplayName','Hp=50')
plot(k,r1,'--','DisplayName','Referencia')
legend('show')
xlabel('Tiempo(s)')
ylabel('Altura(cm)')

subplot(212)
hold
plot(k,u1,'DisplayName','Hp=20')
plot(k,u1_1,'DisplayName','Hp=25')
plot(k,u1_2,'DisplayName','Hp=30')
legend('show')
xlabel('Tiempo(s)')
ylabel('Voltaje (V)')

```