



# **UNIVERSIDAD NACIONAL DE LOJA**

## **ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES**

**CARRERA DE INGENIERÍA ELECTROMECAÁNICA**

### **TÍTULO:**

**DISEÑO DE UN SISTEMA DE CONTROL AVANZADO PARA EL BANCO DE  
PRESIÓN DEL LABORATORIO DE AUTOMATIZACIÓN INDUSTRIAL.**

Tesis de grado previo a optar  
por el Título de Ingeniero  
Electromecánico

### **AUTOR:**

**Paulo César Ramos Moreno**

### **DIRECTOR:**

**Ing. Julio César Cuenca Tinitana, Mg. Sc.**

**LOJA – ECUADOR**

**2015**

## **CERTIFICACIÓN**

**Ing. Julio César Cuenca Tinitana, Mg. Sc.**

**DIRECTOR DE TESIS**

### **CERTIFICA:**

Haber dirigido, asesorado, revisado y corregido el presente trabajo de tesis de grado, en su proceso de investigación cuyo tema versa en **“DISEÑO DE UN SISTEMA DE CONTROL AVANZADO PARA EL BANCO DE PRESIÓN DEL LABORATORIO DE AUTOMATIZACIÓN INDUSTRIAL”**, previa a la obtención del título de Ingeniero Electromecánico, realizado por el señor egresado: Paulo César Ramos Moreno, la misma que cumple con la reglamentación y políticas de investigación, por lo que autorizo su presentación y posterior sustentación y defensa.

Loja, 23 de febrero de 2015

---

**Ing. Julio César Cuenca Tinitana, Mg. Sc.**  
**DIRECTOR DE TESIS**

## **AUTORÍA**

Yo **PAULO CÉSAR RAMOS MORENO** declaro ser autor del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi tesis en Repositorio Institucional – Biblioteca Virtual.

**Autor:** Ramos Moreno Paulo César

**Firma:**  \_\_\_\_\_

**Cédula:** 1104191158

**Fecha:** 02 de abril de 2015

**CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DEL AUTOR, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.**

Yo **PAULO CÉSAR RAMOS MORENO**, declaro ser autor de la tesis titulada: **“DISEÑO DE UN SISTEMA DE CONTROL AVANZADO PARA EL BANCO DE PRESIÓN DEL LABORATORIO DE AUTOMATIZACIÓN INDUSTRIAL”**, como requisito para optar al grado de: **INGENIERO ELECTROMECAÁNICO**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional. Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, dos días del mes de abril del dos mil quince.

**Firma:** \_\_\_\_\_



**Autor:** Ramos Moreno Paulo César

**Cédula:** 1104191158

**Dirección:** Loja (Celi Román, calles Pedro V. Falconí y Ramón Burneo)

**Correo Electrónico:** paulocesar\_88@hotmail.com

**Teléfono:** 2713242

**DATOS COMPLEMENTARIOS.**

**Director de Tesis:** Ing. Julio César Cuenca Tinitana, Mg. Sc.

**Tribunal de Grado:** Ing. Milton Amable León Tapia, Mg. Sc.

Ing. José Fabricio Cuenca Granda, Mg. Sc.

Ing. Byron Agustín Solórzano Castillo, Mg. Sc.

## **PENSAMIENTO**

Lo maravilloso de aprender algo, es que nadie puede arrebatárnoslo.

B.B. King

Cada día sabemos más y entendemos menos.

Albert Einstein

## **DEDICATORIA**

A mis padres quienes me han apoyado para poder llegar a esta instancia de mis estudios, por su amor, trabajo y sacrificio en todos estos años, gracias a ustedes he logrado llegar hasta aquí.

## **AGRADECIMIENTO**

Mi más profundo y sincero agradecimiento a todas aquellas personas que con su ayuda han colaborado en la realización del presente trabajo, en especial al Ing. Julio César Cuenca director de esta tesis por su valiosa guía y asesoramiento continuo en la realización de la misma.

A mis compañeros por todos estos años de compartir momentos de diversión, felicidad, por su ayuda y apoyo siempre que fue necesario.

Un agradecimiento muy especial, merece la comprensión, paciencia y el ánimo recibidos de mi familia y amigos.

A todos ellos, muchas gracias.

## TABLA DE CONTENIDOS

CERTIFICACIÓN .....	II
AUTORÍA .....	III
CARTA DE AUTORIZACIÓN .....	IV
PENSAMIENTO .....	V
DEDICATORIA .....	VI
AGRADECIMIENTO .....	VII
TABLA DE CONTENIDOS .....	VIII
ÍNDICE DE FIGURAS .....	XI
a.- TÍTULO.....	13
b.- RESUMEN.....	14
c.- INTRODUCCIÓN.....	16
d.- REVISIÓN DE LITERATURA.....	17
d.1. CAPÍTULO I: Modelado Matemático e Identificación de Sistemas .....	17
d.1.1. Modelado matemático de sistemas físicos. ....	17
d.1.1.1. Modelado analítico. ....	18
d.1.2. Identificación de Sistemas.....	19
d.1.2.1. Métodos de Identificación .....	20
d.1.2.2. Identificación experimental utilizando la respuesta a la señal escalón. ....	20
d.1.2.2.1. Modelo de primer orden .....	22
d.1.2.2.2. Modelo de segundo orden: Método de Oldenbourg-Sartorius.....	23
d.1.2.2.3. Modelo de segundo orden: Método de Anderson .....	24
d.1.2.2.4. Modelos de primero y segundo orden con retraso puro.....	26
d.1.2.2.5. Modelo de orden $n$ : Método de Strej .....	26
d.1.2.3. Identificación con Secuencias binarias pseudo-aleatorias (SBSA) .....	28



d.2. CAPÍTULO II: Control Predictivo Basado en Modelo (MPC) .....	33
d.2.1. Perspectiva histórica.....	33
d.2.2. Conceptos básicos de control predictivo .....	33
d.2.3. Estrategia de los controladores predictivos .....	35
d.2.4. Elementos básicos .....	37
d.2.4.1. Modelo de predicción .....	37
d.2.4.1.1. Respuestas libre y forzada .....	41
d.2.4.2. Función objetivo .....	42
d.2.4.3. Obtención de la ley de control .....	44
d.3. CAPÍTULO III: Herramientas de Matlab aplicadas al control del proceso.....	46
d.3.1. Toolbox de Identificación de Sistemas (ident).....	46
d.3.2. Toolbox Model Predictive Controller (MPC) .....	49
d.3.2.1. Modelado MPC .....	49
d.3.2.2. Estimación del estado del controlador.....	54
d.3.2.3. Optimización del problema .....	58
d.3.2.4. Solucionador QP.....	63
d.3.2.5. GUI mpctool .....	65
d.3.2.6. Librería MPC de Simulink .....	71
d.3.2.6.1. Bloque Controlador MPC .....	72
e.- MATERIALES Y MÉTODOS.....	75
e.1. Materiales y Métodos.....	75
f. RESULTADOS .....	76
f.1. Descripción del proceso de la planta .....	76
f.2. Obtención del modelo matemático de la planta.....	77
f.2.1. Adquisición de datos.....	77
f.2.2. Obtención del Modelo Matemático .....	79

f.2.3. Validación del Modelo.....	82
f.3. Desarrollo del controlador avanzado .....	83
f.4. Simulación del MPC.....	88
f.5. Comparación del MPC con el controlador convencional.....	90
g.- DISCUSIÓN .....	93
h.- CONCLUSIONES .....	94
i.- RECOMENDACIONES .....	95
j.- BIBLIOGRAFÍA .....	96
k.- ANEXOS.....	97
k.1. Diagrama de control.....	97
k.2. Banco de Control de Presión .....	99
k.3. Diagrama del Controlador MPC .....	100
k.4. Gráfica del Controlador MPC.....	101
k.5. Diagrama Controladores MPC y PI.....	102
k.6. Gráfica comparación de Controladores .....	103
k.7. Proyecto de Tesis .....	104

## ÍNDICE DE FIGURAS

<b>Figura 1.</b> Esquema del procedimiento experimental de identificación de sistemas .....	19
<b>Figura 2.</b> Paso escalón de amplitud $\Delta r$ .....	21
<b>Figura 3.</b> Respuesta de un sistema de primer orden a un paso escalón .....	22
<b>Figura 4.</b> Determinación de los tiempos $T_A$ y $T_C$ para la aplicación del método de Oldenbourg-Sartorius .....	23
<b>Figura 5.</b> Curvas para la aplicación del método de Oldenbourg-Sartorius.....	24
<b>Figura 6.</b> Representación gráfica del método de Anderson.....	25
<b>Figura 7.</b> Determinación del tiempo de retardo $L$ .....	26
<b>Figura 8.</b> Determinación de los tiempos $T_A$ y $T_L$ para la aplicación del método de Strej .....	27
<b>Figura 9.</b> Función de autocorrelación .....	31
<b>Figura 10.</b> Estrategia del Control Predictivo .....	35
<b>Figura 11.</b> Estructura básica del MPC .....	36
<b>Figura 12.</b> Respuesta impulsional y ante escalón .....	38
<b>Figura 13.</b> Respuestas libre y forzada.....	41
<b>Figura 14.</b> Trayectoria de referencia .....	43
<b>Figura 15.</b> Icono de Matlab .....	46
<b>Figura 16.</b> Ventana de Identificación de Sistemas .....	47
<b>Figura 17.</b> Importar datos .....	47
<b>Figura 18.</b> Estimar múltiples modelos.....	48
<b>Figura 19.</b> Ventana de estimación de modelos de proceso.....	48
<b>Figura 20.</b> Validación del modelo .....	49
<b>Figura 21.</b> Estructura del modelo utilizado en el controlador MPC.....	49
<b>Figura 22.</b> Vista inicial MPC toolbox .....	65
<b>Figura 23.</b> Diálogo Importador Modelo de Planta.....	65
<b>Figura 24.</b> La GUI mpctool .....	66
<b>Figura 25.</b> Ventana general del controlador .....	67
<b>Figura 26.</b> Restricciones .....	68
<b>Figura 27.</b> Ajuste de pesos .....	69
<b>Figura 28.</b> Predicciones .....	70
<b>Figura 29.</b> Escenario de Simulación.....	70

<b>Figura 30.</b> Librería MPC de Simulink.....	72
<b>Figura 31.</b> Máscara Bloque Controlador MPC.....	72
<b>Figura 32.</b> Esquema banco de medición de presión .....	76
<b>Figura 33.</b> Interfaz de MAX .....	78
<b>Figura 34.</b> Pantalla de Adquisición de Datos .....	78
<b>Figura 35.</b> Creación de variables de entrada y salida .....	79
<b>Figura 36.</b> Interfaz de Identificación de Sistemas .....	79
<b>Figura 37.</b> Importar Datos .....	80
<b>Figura 38.</b> Estimación de datos .....	80
<b>Figura 39.</b> Modelos de proceso .....	81
<b>Figura 40.</b> Modelos obtenidos .....	81
<b>Figura 41.</b> Simulación salida de modelos.....	82
<b>Figura 42.</b> Análisis de Polos y Ceros .....	83
<b>Figura 43.</b> Interfaz de diseño de MPC.....	84
<b>Figura 44.</b> Definición del proceso .....	84
<b>Figura 45.</b> Importar modelo de planta .....	85
<b>Figura 46.</b> Definición de horizontes de control.....	85
<b>Figura 47.</b> Restricciones del MPC.....	86
<b>Figura 48.</b> Simulación de Escenarios del MPC.....	86
<b>Figura 49.</b> Gráfica de la entrada y salida del MPC.....	87
<b>Figura 50.</b> Exportar controlador MPC.....	87
<b>Figura 51.</b> Diagrama de bloques del controlador .....	88
<b>Figura 52.</b> Parámetros del controlador MPC.....	88
<b>Figura 53.</b> Parámetros bloque función de transferencia .....	89
<b>Figura 54.</b> Resultados del controlador MPC .....	89
<b>Figura 55.</b> Diagrama de bloque controlador PI implementado en el banco .....	90
<b>Figura 56.</b> Diagrama de bloque controladores MPC y PI .....	91
<b>Figura 57.</b> Comparación de modelos.....	91

**a.- TÍTULO**

Diseño de un sistema de control avanzado para el banco de presión del Laboratorio de Automatización Industrial.

## **b.- RESUMEN**

El presente proyecto aborda el diseño y evaluación mediante simulación de un sistema de Control Predictivo Basado en Modelo (MPC) para el Banco de Control de Presión del Laboratorio de Automatización Industrial del Área de la Energía, las Industrias y los Recursos Naturales no Renovables.

En primer lugar se hace una revisión general del estado del arte correspondiente al procedimiento de identificación experimental de procesos industriales, de los Sistemas de Control Predictivo Basado en Modelo y de las herramientas de Matlab-Simulink utilizadas para el diseño y simulación de controladores predictivos.

Luego se explica el proceso de diseño del sistema de control partiendo de la obtención del modelo matemático mediante identificación de procesos, para posteriormente obtener el controlador predictivo mediante herramientas de Matlab-Simulink.

Finalmente, se desarrolla un análisis comparativo del desempeño del sistema de control predictivo diseñado para la planta objeto de estudio con el controlador convencional. Se hace un análisis de los resultados mostrando las bondades del MPC diseñado.

## **SUMMARY**

This project is about the design and evaluation by simulation of a predictive control system - based in the model MPC for the pressure control bank of the industrial automation laboratory in the area of energy, industry and non-renewable natural resources.

First, general review of the state of the art corresponding the procedure for the experimental identification of industrial processes, systems of predictive control based in the model and Matlab-Simulink tools used for design and simulation of predictive controllers

Then explain the process control system design based on the mathematical model obtained by identification of processes, after that to obtain the predictive controller by Matlab-Simulink tools.

Finally, develops a comparative analysis of the performance of predictive control system designed for the plant which is the object of study with conventional controller. Makes an analysis of the results showing the benefits of MPC already designed.

### **c.- INTRODUCCIÓN**

Los controladores convencionales PID, se caracterizan por generar la señal de control de acuerdo a la señal de error que se produce entre la salida medida y la consigna de referencia, son el inicio de la Automatización Industrial.

Estos sistemas de control convencional han sido hasta hoy el estándar del control industrial. Sin embargo, necesidades como la mejora de la efectividad de los sistemas de control y controlar sistemas cuya dinámica es compleja y exigente, han provocado el desarrollo de nuevos métodos y herramientas para automatizar procesos, es decir controladores avanzados. Uno de estos métodos es el denominado Control Predictivo Basado en Modelo, que actualmente es una herramienta práctica para afrontar los problemas presentes en el control industrial, tanto por la estabilidad y precisión proporcionada como por su fácil uso.

El Laboratorio de Automatización cuenta con un Banco experimental para el Control de la Presión en un proceso, el cual tiene implementado un sistema de control convencional PI. Este proceso es de dinámica compleja lo que provoca que el controlador convencional no sea eficiente.

Es por ello que el objetivo principal de este trabajo es diseñar un sistema de Control Avanzado para el Banco de Presión del Laboratorio de Automatización que garantice un control efectivo del proceso. Este sistema avanzado está compuesto principalmente por un controlador predictivo basado en modelo.

Para cumplir este objetivo principal se han trazado los siguientes objetivos específicos:

- Obtener el modelo matemático del sistema mediante la técnica de identificación de sistemas.
- Diseñar el algoritmo de control avanzado basado en el modelo obtenido de la planta.
- Validar mediante simulación el desempeño del controlador avanzado desarrollado.



## **d.- REVISIÓN DE LITERATURA**

### **d.1. CAPÍTULO I: Modelado Matemático e Identificación de Sistemas**

Un modelo de un proceso es una forma de resumir el conocimiento que se tiene sobre su dinámica, y por tanto es una herramienta importante en el diseño y análisis de sistemas de control. Sin embargo, al construir modelos se está obteniendo representaciones simplificadas de la dinámica real del proceso. Un solo modelo no suele ser suficiente para describir un proceso. Por otra parte, según sea el uso destinado al modelo, éste debería ser más o menos detallado. Por tanto, se establece una jerarquía de modelos que describe al proceso con mayor o menor detalle.

Hay dos maneras de abordar la construcción de un modelo: obtenerlo mediante principios y leyes físicas que describan la dinámica del proceso, o bien obtenerlo mediante experimentación sobre el proceso que se quiere modelar. La primera opción requiere un conocimiento muy preciso del proceso que se quiere modelar. Por ejemplo, hay que elegir las variables que vayan a ser los estados del sistema, y esto puede ser un problema. Es en general un proceso complicado y muy arduo, excepto en casos muy simples. Normalmente, se debe combinar con la otra estrategia que es la denominada identificación de sistemas. (Rodríguez Ramírez, y otros, 2005)

#### **d.1.1. Modelado matemático de sistemas físicos.**

El uso de modelos matemáticos para resolver problemas reales (en oposición a los problemas artificiales – académicos, de libros) se ha generado en tiempos recientes. La metodología usada se puede resumir como sigue:

- Formulación del problema.
- Descripción matemática.
- Análisis matemático.
- Interpretación del análisis para obtener la solución.

La fase más crucial e importante es la traducción satisfactoria del problema desde el mundo físico real en una descripción matemática. Una vez que se hace esto, se pueden utilizar técnicas de análisis matemático para obtener una solución al problema. La validez de la solución dependerá en gran parte de cuan bien la descripción matemática propuesta modela el mundo real.

La importancia del análisis y del modelado ha aumentado en varias disciplinas como la economía, la biología, en medicina, la ecología, lógicamente en todas las ramas de la ingeniería y en el control y automatización de procesos o sistemas.

El diseño de modelos matemáticos es necesario para entender el comportamiento dinámico de los procesos y el modelado pretende entonces el desarrollo metodológico de los mismos. La resolución del modelado matemático de un sistema se realiza mediante algoritmos numéricos apropiados implementados en un computador digital. (Lischinsky, 2004)

#### **d.1.1.1. Modelado analítico.**

Procedimiento para la elaboración de modelos matemáticos

1. Dibujar un diagrama esquemático del sistema y definir las variables.
2. Utilizando las leyes de la física, escribir ecuaciones para cada componente, combinándolos de acuerdo con el diagrama del sistema y obtener un modelo matemático.
3. Para verificar la validez del modelo, la predicción acerca del funcionamiento obtenida al resolver las ecuaciones del modelo, se compara con resultados experimentales.

Si los resultados experimentales se alejan de la predicción en forma considerable, debe modificarse el modelo; hasta obtener una concordancia satisfactoria entre la predicción y los resultados experimentales.

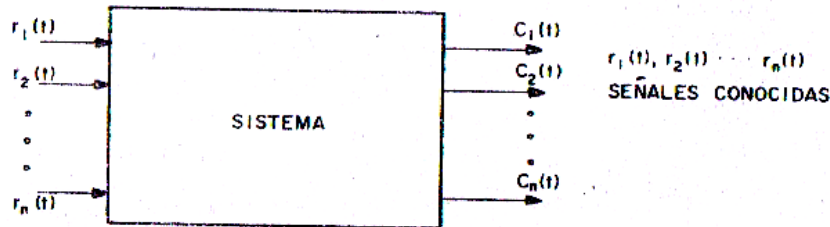
#### **Utilización de los modelos matemáticos.**

Los modelos matemáticos pueden ser utilizados para estudiar propiedades y/o predecir el comportamiento del sistema ante diferentes situaciones. Existen dos grandes grupos de técnicas para tal fin: (López, 1999)

- Análisis Teórico de los modelos matemáticos: Métodos matemáticos de análisis cualitativo (estabilidad, etc.) y cuantitativo (resolución de ecuaciones, etc.)
- Análisis Experimental de los modelos matemáticos: Estudio de propiedades cuantitativas y cualitativas del Modelo Matemático mediante experimentos en equipos de cómputo programables: Simulación o Matemática Experimental.

### d.1.2. Identificación de Sistemas

La identificación experimental de un sistema consiste en aplicar en las variables de entrada de este un conjunto de estímulos conocidos y a partir del análisis de las respuestas correspondientes, determinar el modelo matemático del mismo.



**Figura 1.** Esquema del procedimiento experimental de identificación de sistemas

**Fuente:** (Trejo, 1986)

La identificación experimental se aplica en aquellos casos en los cuales no es posible determinar el modelo matemático del sistema objeto de estudio, de forma fácil, a partir del planteamiento de las ecuaciones de balance dinámico. Esto ocurre, cuando se desconocen las relaciones físico-químicas existentes en el sistema o bien cuando éstas son muy complejas, al igual que cuando se desconocen los parámetros de los diferentes componentes del mismo o su determinación es muy complicada.

La identificación experimental de sistemas presenta algunas desventajas, que es conveniente señalar, tales como: (Trejo, 1986)

- Los resultados que se obtienen son rigurosamente válidos solo en las condiciones en los cuales se realiza el experimento; aunque desde el punto de vista práctico pueden ser utilizables en otras condiciones.
- La mayoría de los métodos exigen que aquellas variables de entrada que no son estimuladas se estabilicen completamente durante la realización del experimento, lo cual no es fácil de lograr.
- Se perturba el funcionamiento normal del sistema durante el tiempo de realización del experimento.
- Con frecuencia, se requiere la utilización de equipos de medición especiales.
- No son aplicables en el proyecto o diseño del sistema, pues es necesario que esté ya construido.

#### **d.1.2.1. Métodos de Identificación**

Se utiliza distintos métodos para la identificación, siendo los principales: Identificación Analítica, Identificación Experimental utilizando señales especiales de prueba e Identificación Recursiva. (Aguado, 2000)

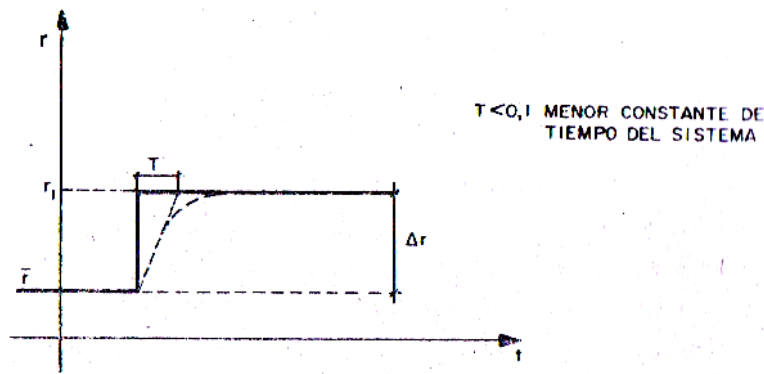
1) Identificación Analítica. Consiste en desarrollar un modelo basado en las relaciones físico-químicas del proceso por identificar planteando ecuaciones de balance de masa, de energía, entre otros. Este enfoque conduce generalmente a modelos complejos y no lineales que deben ser sometidos a un proceso de simplificación y linealización.

2) Identificación Experimental mediante Señales Especiales. Este enfoque resulta generalmente el más directo y el que puede producir resultados a más corto plazo. Las señales utilizadas con más frecuencia son los escalones y las llamadas secuencias binarias pseudo-aleatorias. En condiciones ideales, el experimento de identificación debería hacerse contando con una computadora conectada en línea con el proceso, pero en la práctica, en muchas ocasiones, es necesario basarse en observaciones periódicas de las variables y el procesamiento fuera de línea de datos.

3) Identificación Recursiva. Se basa generalmente en una u otra modificación del método de mínimos cuadrados. Se asume una estructura fija del modelo que consiste casi siempre en una ecuación de diferencias lineales. Lo característico de este enfoque es que se parte de una estimación inicial, generalmente arbitraria, de los parámetros del modelo que se va actualizando y mejorando con cada nueva información obtenida. Aunque este tipo de identificación está concebida fundamentalmente para usarse dentro de un esquema de control adaptable, también por razones de conveniencia práctica, se ha utilizado para obtener modelos para estudios de simulación. Como los datos que se utilizan tienen generalmente la forma de secuencia discreta de mediciones de las variables del proceso, los modelos que se identifican son también de naturaleza discreta.

#### **d.1.2.2. Identificación experimental utilizando la respuesta a la señal escalón.**

La señal de prueba de más fácil utilización es la función paso-escalón, la cual se muestra en la figura. (Trejo, 1986)



**Figura 2.** Paso escalón de amplitud  $\Delta r$

**Fuente:** (Trejo, 1986)

Esta señal se aproxima, por ejemplo, mediante la apertura o cierre súbitos de una válvula, un cambio rápido de voltaje o corriente que alimenta algún tipo de actuador eléctrico, entre otros. Un escalón ideal es una señal cuyo tiempo de crecimiento inicial es cero.

En la práctica una señal de este tipo solo puede lograrse de forma aproximada tal como aparece señalado en línea de puntos en la figura. Si el tiempo de crecimiento inicial es varias veces más pequeño que el periodo correspondiente a la frecuencia máxima de interés, el error que se introduce en la identificación puede considerarse despreciable.

La función escalón es la señal que más se ha aplicado en la práctica convencional del control automático y con ella se obtienen modelos sencillos suficientemente exactos, sobre todo en los casos de procesos monovariantes simples y poco perturbados. (Aguado, 2000)

El uso de la señal escalón como señal de prueba tiene la ventaja de la sencillez de la generación de dicha función y que el tiempo de experimentación es corto. Como desventaja puede mencionarse que, introduce una alteración relativamente grande en el comportamiento del sistema, lo cual no siempre es permisible en procesos industriales.

La forma de determinar los parámetros del modelo, cuando se aplica un paso-escalón, estará en dependencia de la propia forma del modelo y algo similar se realiza cuando se emplean otras señales como estímulo. En la Tabla 1 aparecen diferentes modelos que se utilizan, con las características más importantes de cada una de ellas. (Trejo, 1986)

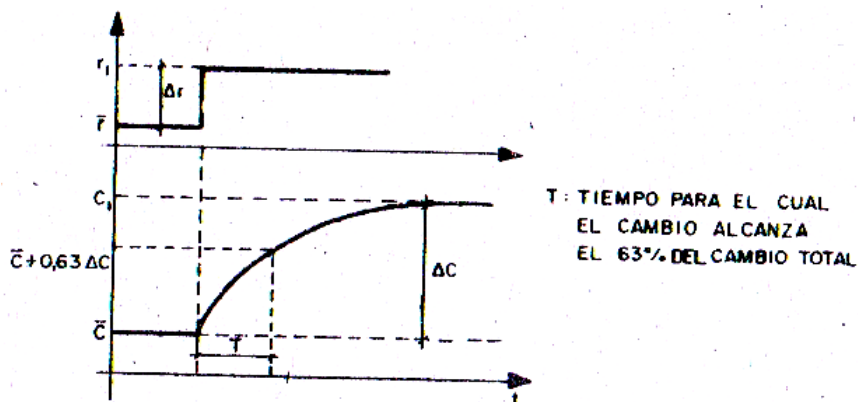
**Tabla 1.** Modelos utilizados en la identificación experimental de sistemas

Nombre	Función transferencial	Parámetros a determinar
Modelo de 1 <sup>er</sup> orden	$\frac{K}{sT + 1}$	$K, T$
Modelo de 2 <sup>do</sup> orden	$\frac{K}{(sT_1 + 1)(sT_2 + 1)}$	$K, T_1, T_2$
Modelo de orden $n$	$\frac{K}{(sT + 1)^n}$	$K, T, n$
Modelo de orden superior	$\frac{K}{(sT_1 + 1)(sT_2 + 1) \dots (sT_i + 1)}$	$K, T_1, T_2, \dots, T_i$
Modelo de primer orden con retraso puro	$\frac{Ke^{-sL}}{sT + 1}$	$K, T, L$
Modelo de segundo orden con retraso puro	$\frac{Ke^{-sL}}{(sT_1 + 1)(sT_2 + 1)}$	$K, T_1, T_2, L$
Modelo de orden $n$ con retraso puro	$\frac{Ke^{-sL}}{(sT + 1)^n}$	$K, T, n, L$
Modelo de orden superior con retraso puro	$\frac{Ke^{-sL}}{(sT_1 + 1)(sT_2 + 1) \dots (sT_i + 1)}$	$K, T_1, T_2, \dots, T_i, L$

Fuente: (Trejo, 1986)

#### d.1.2.2.1. Modelo de primer orden

Cuando se emplea un modelo de primer orden, los parámetros a determinar son dos: la ganancia  $K$  y la constante de tiempo  $T$ . Ambos valores se determinan a partir de las características de la respuesta de un sistema de primer orden a un paso-escalón (fig. 3).



**Figura 3.** Respuesta de un sistema de primer orden a un paso escalón

Fuente: (Trejo, 1986)

La ganancia estará dada por la relación entre el cambio total del valor de estado estacionario de la variable de salida y la amplitud del paso-escalón, o sea:

$$K = \frac{\Delta c}{\Delta r} \quad (1)$$

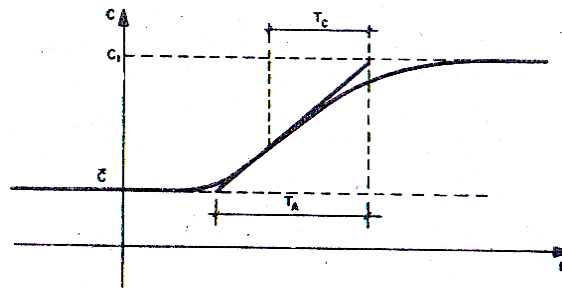
La constante de tiempo se corresponde con el tiempo, medida a partir del instante en que se perturba el sistema para el cual la variable de salida ha sufrido el 63,2% del cambio total. (Trejo, 1986)

#### d.1.2.2.2. Modelo de segundo orden: Método de Oldenbourg-Sartorius

Este método para determinar los parámetros de un modelo de segundo orden fue desarrollado en 1984 por los dos investigadores que le dieron nombre.

La determinación de la ganancia ocurre, al igual que para un modelo de primer orden, a partir de la relación entre el cambio total que sufre la salida del sistema y la amplitud del paso-escalón aplicado en la entrada.

Las constantes de tiempo se determinan a partir de las relaciones entre los tiempos  $T_A$  y  $T_C$ , definidos cuando se traza una tangente por el punto de inflexión de la curva que representa la respuesta de un sistema de segundo orden a un paso-escalón, según muestra la figura 4.



**Figura 4.** Determinación de los tiempos  $T_A$  y  $T_C$  para la aplicación del método de Oldenbourg-Sartorius

**Fuente:** (Trejo, 1986)

De la respuesta de un sistema de segundo orden a un paso escalón, se puede demostrar que los tiempos  $T_A$  y  $T_C$  se relacionan con las constantes de tiempo  $T_1$  y  $T_2$  por las expresiones siguientes: (Trejo, 1986)

$$T_A = T_1 \left( \frac{T_2}{T_1} \right)^n \quad \text{Donde } n = \frac{T_2}{T_1 - T_2} \quad (2)$$

$$T_C = T_1 + T_2 \quad (3)$$

El método de Oldenbourg-Sartorius consiste en la obtención de los valores de las constantes de tiempo  $T_1$  y  $T_2$  por medio de las expresiones 2 y 3, partiendo de los valores de los tiempos  $T_A$  y  $T_C$ . Estos valores se determinan gráficamente de la representación de la respuesta del sistema a un paso-escalón.

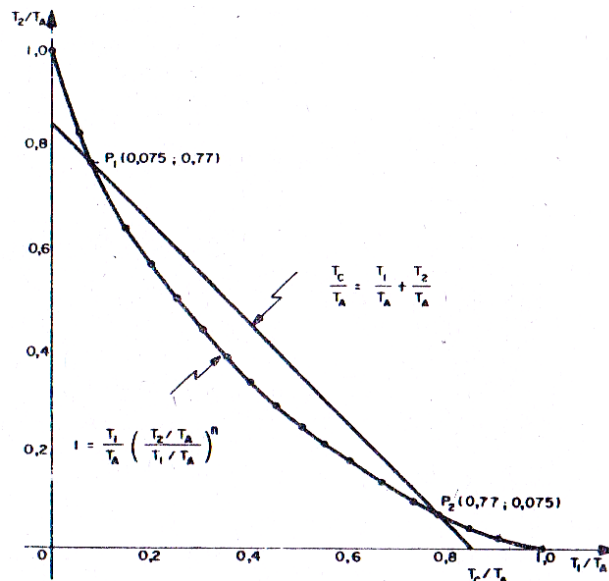
La solución analítica de las ecuaciones resultantes, al sustituir  $T_A$  y  $T_C$  en las expresiones 2 y 3, es muy compleja, siendo más conveniente el empleo de un método gráfico.

La expresión 2 puede escribirse:

$$1 = \frac{T_1}{T_A} \left[ \frac{T_2/T_A}{T_1/T_A} \right]^n \quad (4)$$

Cuya representación gráfica es la curva de la figura 5. Por otra parte, la expresión 3 se puede escribir dividiendo por  $T_A$ :

$$\frac{T_C}{T_A} = \frac{T_1}{T_A} + \frac{T_2}{T_A} \quad (5)$$



**Figura 5.** Curvas para la aplicación del método de Oldenbourg-Sartorius

**Fuente:** (Trejo, 1986)

#### **d.1.2.2.3. Modelo de segundo orden: Método de Anderson**

Es conocido que la respuesta de un sistema de segundo orden cuya función de transferencia es de la forma:

$$\frac{C(s)}{R(s)} = \frac{K}{(sT_1+1)(sT_2+1)} \quad (6)$$

Cuando se aplica un paso-escalón de amplitud  $A$ , está dada por:



$$C(t) = AK(1 + \frac{T_1 e^{-t/T_1} - T_2 e^{-t/T_2}}{T_2 - T_1}) \quad (7)$$

Que puede escribirse:

$$C(t) = AK + K_1 e^{-t/T_1} - K_2 e^{-t/T_2} \quad (8)$$

Donde  $K_1$  y  $K_2$  son positivos si  $T_2 > T_1$ . La diferencia entre el valor de estado estacionario y la respuesta transitoria será:

$$C_L(t) = AK - C(t) = -K_1 e^{-t/T_1} + K_2 e^{-t/T_2} \quad (9)$$

Y como  $T_1$  y  $T_2$  son reales, la respuesta  $C(t)$  será no oscilatoria y  $C_L(t)$  será positiva.

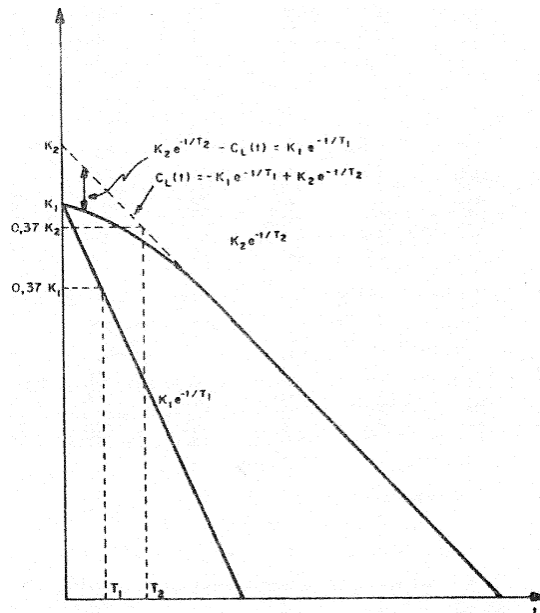
En las condiciones planteadas ( $T_2 > T_1$ ), el primer término de  $C_L(t)$  disminuye más rápidamente que el segundo, luego, para valores altos de  $t$ , se puede decir que:

$$C_L(t) \approx K_2 e^{-t/T_2} \quad (10)$$

Y tomando logaritmo en la expresión anterior, se tiene que:

$$\log_{10} C_L(t) \approx \log_{10} K_2 - (\frac{1}{T_2} \log_{10} e) t \quad (11)$$

O sea, si se construye un gráfico de  $C_L(t)$ , usando papel semilogarítmico, tomando el eje semilogarítmico para  $C_L(t)$ , y el eje lineal para  $t$ , se tiene que para valores altos de  $t$  se obtiene una línea recta (fig. 6). (Trejo, 1986)



**Figura 6.** Representación gráfica del método de Anderson

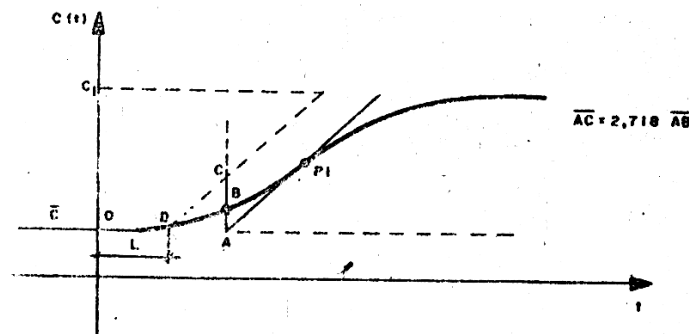
**Fuente:** (Trejo, 1986)

#### d.1.2.2.4. Modelos de primero y segundo orden con retraso puro

Cuando se utiliza un modelo de primero o segundo orden con retraso puro, se determina primeramente el valor del tiempo de retardo  $L$  y se procede en la forma vista anteriormente para modelos de primero y segundo orden, considerando en la respuesta que el origen del tiempo está en  $t = L$ .

Una forma de determinar el tiempo de retardo es tomar para el mismo el valor de  $t$  a partir del instante en el cual se aplica el paso-escalón y la respuesta sufre el 5% del cambio total del valor del estado estacionario.

Otra forma posible, que da buenos resultados, aunque no tiene una estricta fundamentación teórica, se muestra gráficamente en la figura 7; consiste en: (Trejo, 1986)



**Figura 7.** Determinación del tiempo de retardo  $L$

**Fuente:** (Trejo, 1986)

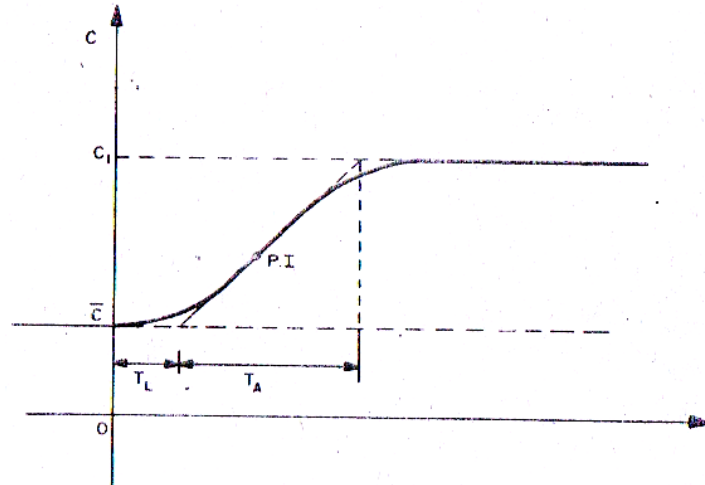
1. Se traza una tangente a la curva que representa la respuesta al paso-escalón aplicado por el punto de inflexión.
2. Se levanta una perpendicular por el punto A, que es la intersección de la tangente anterior con el valor de estado estacionario inicial.
3. Sobre la perpendicular anterior se marca el punto C, tal, que  $\overline{AC} = 2,718 \overline{AB}$
4. Se traza una paralela a la tangente por el punto de inflexión, que pase por el punto C.
5. El tiempo  $\overline{OD}$  corresponde al tiempo de retardo  $L$ .

#### d.1.2.2.5. Modelo de orden $n$ : Método de Strej

Este método se utiliza para determinar los parámetros  $n$  y  $T$  de la función transferencial de un sistema, con  $n$  constantes de tiempo iguales y de valor  $T$ ,  $[G(s) = K/(st + 1)^n]$ , a partir de su respuesta a un paso-escalón.

Al igual que en los casos anteriores, la ganancia  $K$  se determina por la relación entre el cambio del valor de estado estacionario de la repuesta, antes y después de aplicado, el paso y la amplitud de éste.

Considérese que en la curva que representa la respuesta de un sistema con las características mencionadas se traza una tangente por el punto de inflexión, determinándose los tiempos  $T_L$  y  $T_A$  (fig. 8).



**Figura 8.** Determinación de los tiempos  $T_A$  y  $T_L$  para la aplicación del método de Strej

**Fuente:** (Trejo, 1986)

La relación entre los tiempos  $T_L$  y  $T_A$  es una función creciente de  $n$  y  $T_A$  y  $T_L$  son proporcionales a  $T$ , y los factores de proporcionalidad dependen de  $n$  (Tabla 2).

**Tabla 2.** Relaciones entre los tiempos  $T_L$ ,  $T_A$  y  $T$  en función de  $n$

$n$	$T_L/T_A$	$T_A/T$	$T_L/T$
2	0,104	2,718	0,282
3	0,218	3,695	0,805
4	0,319	4,463	1,425
5	0,410	5,119	2,100
6	0,493	5,700	2,811

**Fuente:** (Trejo, 1986)

De acuerdo con lo anterior, el procedimiento para identificar un sistema de  $n$  constantes de tiempo iguales, por medio de un paso-escalón, es el siguiente:

1. A partir de la respuesta del sistema al paso aplicado representada gráficamente, se determinan los tiempos  $T_L$  y  $T_A$ .
2. Con la relación  $T_L/T_A$  se determina el valor de  $n$  por medio de la tabla 2.
3. Con el valor de  $T_A/T$ , de la tabla correspondiente al valor de  $n$ , determinado en 2 y el de  $T_A$ , se calcula  $T$ . Similarmente se puede hacer usando la relación  $T_L/T$  y el tiempo  $T_L$ .

Este método tiene la desventaja, de las imprecisiones en el trazado de la tangente por el punto de inflexión. El método es también aplicable cuando el sistema a identificar tiene un retraso puro, siempre que éste sea menor que el 30% de la constante de tiempo. (Trejo, 1986)

#### **d.1.2.3. Identificación con Secuencias binarias pseudo-aleatorias (SBSA)**

La señal aleatoria o pseudo-aleatoria que se utiliza para la identificación, puede superponerse sobre los valores normales de operación de la variable de entrada, sin que esto afecte sensiblemente al proceso de identificación. Consideremos, en efecto, que la entrada del sistema puede descomponerse en la forma:

$$u(t) = u_n(t) + r(t) \quad (12)$$

Donde  $u_n(t)$  es la señal correspondiente a la operación normal y  $r(t)$  es el ruido superpuesto. La salida  $y(t)$ , también puede descomponerse en la forma:

$$y(t) = y_u(t) + y_r(t) \quad (13)$$

Donde  $y_u(t)$  e  $y_r(t)$  son respectivamente las respuestas correspondientes a  $u_n(t)$  y  $r(t)$ .

Ocorre, sin embargo, que las componentes  $y_u(t)$  e  $y_r(t)$  no pueden distinguirse a partir de la medición de la salida  $y(t)$  y es necesario, por tanto, calcular la función de correlación cruzada utilizando la señal de ruido  $r(t)$  y la salida completa, es decir, se calcula la integral:

$$\phi_{yu}(\theta) = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T y(t)r(t - \theta)dt \quad (14)$$

Utilizando la ecuación 13, la ecuación 14 puede descomponerse de la forma:

$$\phi_{yu}(\theta) = \lim_{T \rightarrow \infty} \frac{1}{2T} \left[ \int_{-T}^T y_u(t)r(t - \theta)dt + \int_{-T}^T y_r(t)r(t - \theta)dt \right] \quad (15)$$

El primer término de 15 expresa la correlación entre la salida del sistema  $y_u(t)$  y la señal de ruido  $r(t)$ . Esta correlación, sin embargo, es generalmente débil ya que  $y_u(t)$  es una variable con un ancho de banda estrecho, en tanto que  $r(t)$  se aproxima al ruido blanco y por tanto es una señal que varía muy rápidamente. Si el período  $T$  se escoge suficientemente grande, puede suponerse entonces que la primera integral se acerca a cero y la función  $\phi_{yu}(\theta)$  calculada será aproximadamente:

$$\phi_{yu}(\theta) \approx \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T y_r(t) r(t - \theta) dt \quad (16)$$

Lo que constituye precisamente el resultado deseado. Nótese que la señal de ruido puede tener, dentro de límites razonables, una amplitud arbitrariamente pequeña, lo cual constituye una de las ventajas fundamentales de este tipo de perturbación.

Las SBSA son señales periódicas cuyas integrales de autocorrelación se aproximan satisfactoriamente a la función Delta de Dirac. Las SBSA se codifican mediante secuencias de ceros y unos que satisfacen una ecuación de diferencias del tipo

$$x(t) = a_m D^m x(t) (+) a_{m-1} D^{m-1} x(t) (+) \dots (+) a_1 dx(t) \quad (17)$$

Donde:

$D$ : Operador de retardo, de manera que  $D^m x(t) = x(t - m)$

$(+)$ : Operador de suma módulo 2

$x(t)$ : Variable binaria que asume valores 0 o 1

$a_i$ : Coeficientes binarios que asumen también valores 0 o 1

$m$ : Orden de la secuencia

Las reglas de adición módulo 2 u OR exclusivo son

$$0 (+) 0 = 0$$

$$0 (+) 1 = 1$$

$$1 (+) 0 = 1$$

$$1 (+) 1 = 0$$

Las SBSA son secuencias periódicas con un período igual a:

$$N = 2^m - 1 \quad (18)$$

Los coeficientes  $a_i$  de la expresión 17, sin embargo, no pueden escogerse arbitrariamente. Se puede demostrar que la elección de los coeficientes que aparecen en la tabla 3, generan SBSA de máxima longitud, es decir, cuyo período está dado por la ecuación 18.

**Tabla 3.** Coeficientes  $a_i$

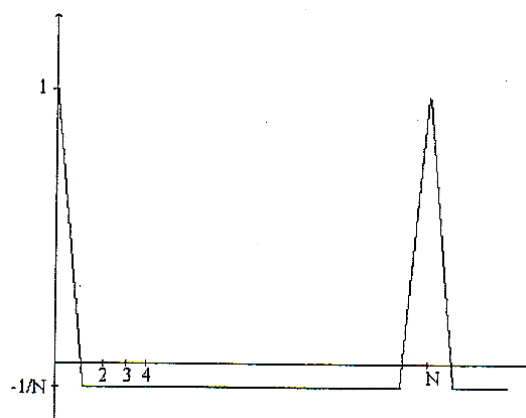
<b>m</b>	<b>Coeficientes iguales a 1</b>
2	a1, a2
3	a1, a3
4	a3, a4
5	a3, a5
6	a2, a3, a6
7	a4, a7
8	a2, a3, a4, a8
9	a5, a9
10	a7, a10
11	a9, a11

**Fuente:** (Aguado, 2000)

Las SBSA tienen una serie de propiedades interesantes:

- Su promedio es aproximadamente  $N/2$
- Se producen  $2^{m-1}$  cambios (de 0 a 1 ó de 1 a 0)
- Incluyen un grupo de  $m$  y otro de  $m-1$  unos o ceros consecutivos
- Incluyen  $2^i$  grupos de  $m-i-1$  ceros o unos con  $i = 1, 2, \dots, m-2$
- Su función de autocorrelación es aproximadamente como la mostrada en la figura 9. O sea, que a medida que la longitud de la secuencia es mayor, la función de autocorrelación se aproxima más a la Delta de Dirac.

Se sobreentiende que al valor “1” en la secuencia le corresponde un incremento positivo de la variable de control y al cero un incremento negativo con respecto al valor medio.



**Figura 9.** Función de autocorrelación

**Fuente:** (Aguado, 2000)

Aunque las SBSA fueron concebidas inicialmente para la identificación de la respuesta impulsiva del sistema, ellas pueden aplicarse también ventajosamente para la identificación de otros tipos de modelos. La utilización práctica de estas secuencias consiste en la asignación de 2 valores de control, por ejemplo, posiciones de una válvula, que corresponden al 0 y al 1. Estos 2 valores deben ser simétricos con respecto a la posición normal del elemento de control, de manera que el efecto neto o promedio de la perturbación introducida al final del experimento sea cero.

Como las SBSA, calculada previamente o generada en línea mediante una computadora consiste en una secuencia de unos y ceros, su aplicación práctica se concreta en una secuencia de aperturas y cierres de la válvula. El tiempo de conmutación de la señal binaria debe estimarse en cada caso, atendiendo a consideraciones prácticas y a la dinámica del proceso en cuestión. La regla práctica para escoger el período de conmutación de la señal binaria puede ser la misma que la recomendada para la elección del período de control en un regulador discreto y se expresa en la tabla 4.

**Tabla 4.** Periodos de Conmutación de la SBSA

Variable	Periodo de Conmutación de la SBSA
Flujo	1 segundo
Nivel, Presión	5 segundos
Temperatura	10 segundos

**Fuente:** (Aguado, 2000)

El orden de la secuencia se escoge generalmente 5 o 6 y se recomienda hacer varias réplicas de la secuencia (2 o 3 al menos). La selección de un orden mayor implicaría un experimento excesivamente largo, difícilmente realizable en la práctica. (Aguado, 2000)



## **d.2. CAPÍTULO II: Control Predictivo Basado en Modelo (MPC)**

### **d.2.1. Perspectiva histórica**

El Control Predictivo se desarrolló en base a dos líneas básicas. Por un lado, a finales de los años setenta surgieron diversos algoritmos que usaban explícitamente un modelo dinámico del proceso para predecir el efecto de las acciones de control futuras en la salida, las cuales eran determinadas minimizando el error predicho sujeto a restricciones de operación. La optimización se repetía en cada instante de muestreo con información actualizada del proceso. Estas formulaciones eran de naturaleza heurística y algorítmica e intentaban aprovechar el creciente potencial de los computadores digitales por aquella época.

Rápidamente el MPC adquirió gran popularidad en las industrias de procesos químicos principalmente debido a la simplicidad del algoritmo y al uso del modelo de respuesta impulsional o en escalón, que aunque posea muchos más parámetros que las formulaciones en el espacio de estados o función de transferencia suele ser preferido por ser intuitivo y necesitar menos información a priori para identificar. La mayoría de las aplicaciones fueron llevadas a cabo sobre sistemas multivariantes incluyendo restricciones. Los algoritmos utilizados fueron principalmente el IDCOM (Identification-Command) y el DMC (Control con Matriz Dinámica, Dynamic Matrix Control).

Independientemente fue surgiendo otra línea de trabajo en torno a las ideas del control adaptativo, desarrollando estrategias esencialmente para procesos monovariantes formuladas con modelos entrada/salida. En este contexto se extendieron las ideas del Controlador de Mínima Varianza y se desarrolló el Control Predictivo Generalizado (Generalized Predictive Control GPC) que es uno de los métodos más populares en la actualidad. (Rodríguez Ramírez, y otros, 2005)

### **d.2.2. Conceptos básicos de control predictivo**

El Control Predictivo Basado en Modelo, Model (Based) Predictive Control (MBPC o MPC) constituye un campo muy amplio de métodos de control desarrollados en torno a ciertas ideas comunes e integra diversas disciplinas como control óptimo, control estocástico, control de procesos con tiempos muertos, control multivariable o control con restricciones.

El Control Predictivo no es una estrategia de control específica, sino que se trata más bien de un campo muy amplio de métodos de control desarrollados en torno a ciertas ideas comunes. Estos métodos de diseño conducen a controladores lineales que poseen prácticamente la misma estructura y presentan suficientes grados de libertad. Las ideas que aparecen en mayor o menor medida en toda la familia de controladores predictivos son básicamente:

- Uso explícito de un modelo para predecir la salida del proceso en futuros instantes de tiempo (horizonte).
- Cálculo de las señales de control minimizando una cierta función objetivo.
- Estrategia deslizando, de forma que en cada instante el horizonte se va desplazando hacia el futuro, lo que implica aplicar la primera señal de control en cada instante y desechar el resto, repitiendo el cálculo en cada instante de muestreo.

Los distintos algoritmos de MPC difieren entre sí casi exclusivamente en el modelo usado para representar el proceso y los ruidos y en la función de coste a minimizar. Aunque las diferencias puedan parecer pequeñas a priori, pueden provocar distintos comportamientos en bucle cerrado, siendo críticas para el éxito de un determinado algoritmo en una determinada aplicación.

El Control Predictivo es un tipo de control de naturaleza abierta dentro del cual se han desarrollado muchas realizaciones, encontrando gran aceptación tanto en aplicaciones industriales como en el mundo académico. En la actualidad existen numerosas aplicaciones de controladores predictivos funcionando con éxito, tanto en la industria de procesos como en control de motores o Robótica. El buen funcionamiento de estas aplicaciones muestra la capacidad del MPC para conseguir sistemas de control de elevadas prestaciones capaces de operar sin apenas intervención durante largos períodos de tiempo.

El MPC presenta una serie de ventajas sobre otros métodos, entre las que destacan:

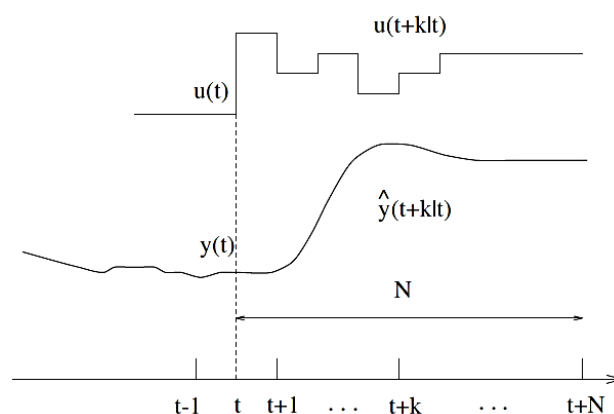
- Resulta particularmente atractivo para personal sin un conocimiento profundo de control, puesto que los conceptos resultan muy intuitivos, a la vez que la sintonización es relativamente fácil.

- Puede ser usado para controlar una gran variedad de procesos, desde aquéllos con dinámica relativamente simple hasta otros más complejos incluyendo sistemas con grandes retardos, de fase no mínima o inestable.
- Permite tratar con facilidad el caso multivariable.
- Posee intrínsecamente compensación del retardo.
- Resulta conceptualmente simple la extensión al tratamiento de restricciones, que pueden ser incluidas de forma sistemática durante el proceso de diseño.
- Es muy útil cuando se conocen las futuras referencias (robótica o procesos en batch).
- Es una metodología completamente abierta basada en algunos principios básicos que permite futuras extensiones.

Pero, lógicamente, también presenta inconvenientes. Uno de ellos es la carga de cálculo necesaria para la resolución de algunos algoritmos. Pero quizás el mayor inconveniente venga marcado por la necesidad de disponer de un modelo apropiado del proceso. El algoritmo de diseño está basado en el conocimiento previo del modelo y es independiente de éste, pero resulta evidente que las prestaciones obtenidas dependerán de las discrepancias existentes entre el proceso real y el modelo usado. (Rodríguez Ramírez, y otros, 2005)

#### **d.2.3. Estrategia de los controladores predictivos**

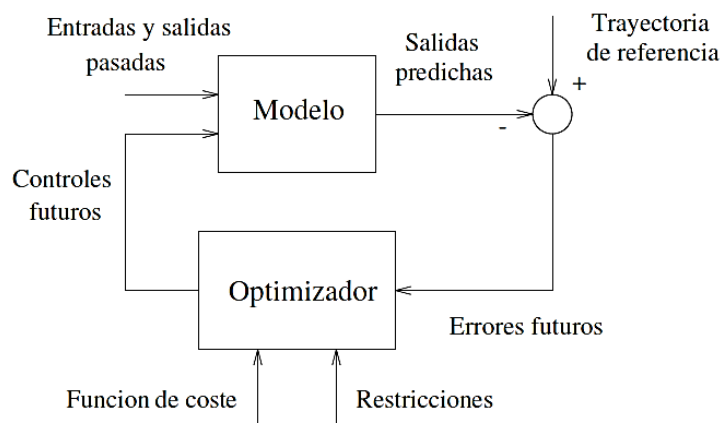
La metodología de todos los controladores pertenecientes a la familia del MPC se caracteriza por la estrategia siguiente, representada en la figura 10:



**Figura 10.** Estrategia del Control Predictivo

**Fuente:** (Rodríguez Ramírez, y otros, 2005)

1. En cada instante  $t$  y haciendo uso del modelo del proceso se predicen las futuras salidas para un determinado horizonte  $N$ , llamado horizonte de predicción. Estas salidas predichas,  $\hat{y}(t + k|t)$  para  $k = 1 \dots N$  dependen de los valores conocidos hasta el instante  $t$  (entradas y salidas pasadas) y de las señales de control futuras  $u(t + k | t)$ ,  $k = 0 \dots N - 1$  que se pretenden mandar al sistema y que son las que se quieren calcular.
2. El conjunto de señales de control futuras se calcula optimizando un determinado criterio en el que se pretende mantener el proceso lo más próximo posible a la trayectoria de referencia  $w(t + k)$  (que puede ser directamente el setpoint o una suave aproximación a éste). Este criterio suele tomar la forma de una función cuadrática de los errores entre la salida predicha y la trayectoria de referencia también predicha, incluyendo en muchos casos el esfuerzo de control. Si el criterio es cuadrático, el modelo lineal y no existen restricciones se puede obtener una solución explícita, en otro caso se debe usar un método iterativo de optimización. Adicionalmente se hace alguna suposición sobre la estructura de la ley de control futura, como por ejemplo que va a ser constante a partir de cierto instante.
3. La señal de control  $u(t / t)$  es enviada al proceso mientras que las siguientes señales de control calculadas son desechadas, puesto que en el siguiente instante de muestreo ya se conoce  $y(t + 1)$  y se repite el paso 1 con este nuevo valor y todas las secuencias son actualizadas. Se calcula por tanto  $u(t+1 / t+1)$  (que en principio será diferente al  $u(t+1 / t)$  al disponer de nueva información), haciendo uso del concepto de horizonte deslizante.



**Figura 11.** Estructura básica del MPC

**Fuente:** (Rodríguez Ramírez, y otros, 2005)

Para llevar a cabo esta estrategia, se usa una estructura como la mostrada en la figura 11. Se hace uso de un modelo para predecir las salidas futuras del proceso, basándose en las futuras señales de control propuestas. Estas señales son calculadas por el optimizador teniendo en cuenta la función de coste (donde aparece el futuro error de seguimiento) así como las restricciones. Por tanto el modelo juega un papel decisivo en el controlador. El modelo elegido debe ser capaz de capturar la dinámica del proceso para poder predecir las salidas futuras al mismo tiempo que debe ser sencillo de usar y de comprender.

El optimizador es otra parte fundamental de la estrategia pues proporciona las acciones de control. Si la función de coste es cuadrática, el mínimo se puede obtener como una función explícita de las entradas y salidas pasadas y de la trayectoria de referencia. Sin embargo, cuando existen restricciones de desigualdad la solución debe ser calculada por métodos numéricos con más carga de cálculo. (Rodríguez Ramírez, y otros, 2005)

#### **d.2.4. Elementos básicos**

Todos los controladores predictivos poseen elementos comunes y para cada uno de estos elementos se pueden elegir diversas opciones, dando lugar a distintos algoritmos.

Estos elementos son: (Rodríguez Ramírez, y otros, 2005)

- Modelo de predicción
- Función objetivo
- Obtención de la ley de control

##### **d.2.4.1. Modelo de predicción**

La piedra angular del MPC es el modelo; un diseño completo debe incluir los mecanismos necesarios para la obtención del mejor modelo posible, el cual debe ser lo suficientemente rico para capturar al máximo la dinámica del proceso y debe ser capaz de permitir el cálculo de las predicciones a la vez que sea intuitivo y permita un análisis teórico. El uso del modelo del proceso viene determinado por la necesidad del cálculo de la salida predicha en instantes futuros  $\hat{y}(t + k|t)$ . Las diferentes estrategias de MPC pueden usar distintos modelos para representar la relación de las salidas con las entradas medibles, algunas de las cuales serán variables manipuladas y otras se pueden considerar como perturbaciones medibles, que pueden ser compensadas por acción feedforward. (Rodríguez Ramírez, y otros, 2005)

La acción feedforward es un medio para corregir el efecto de las perturbaciones, el cual compensa el efecto de las mismas antes de que produzcan error en la variable controlada. (Acedo Sánchez, 2003)

Además se tendrá en cuenta un modelo de las perturbaciones, para intentar describir el comportamiento que no aparece reflejado en el modelo del proceso, englobándose aquí el efecto de las entradas no medibles, el ruido y los errores de modelado.

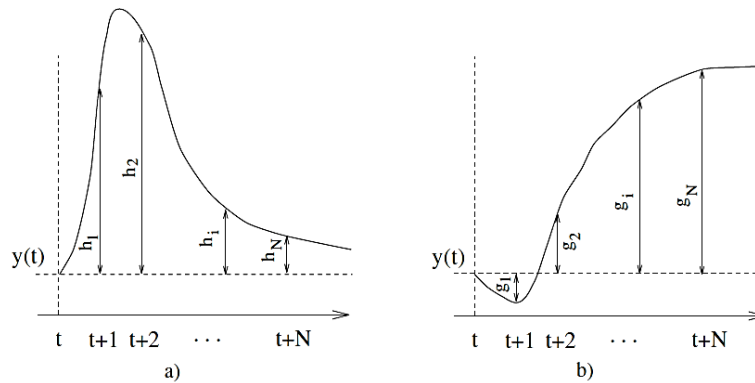
Para el estudio se puede separar el modelo en dos partes: el modelo del proceso propiamente dicho y el modelo de las perturbaciones. Cualquier método usará ambas partes para la predicción. (Rodríguez Ramírez, y otros, 2005)

### Modelo del Proceso

Casi todas las formas posibles de modelar un proceso aparecen en alguna formulación de MPC siendo las más usadas las siguientes:

- Respuesta impulsional. También conocida por secuencia de ponderación o modelo de convolución. La salida viene relacionada con la entrada por la ecuación

$$y(t) = \sum_{i=1}^{\infty} h_i u(t-i) \quad (19)$$



**Figura 12.** Respuesta impulsional y ante escalón

**Fuente:** (Rodríguez Ramírez, y otros, 2005)

donde  $h_i$  son los valores muestreados obtenidos al someter al proceso a un impulso unitario de amplitud igual al período de muestreo (ver figura 12.a). Esta suma es truncada y sólo se consideran  $N$  valores (por tanto sólo permite representar procesos estables y sin integradores), teniendo

$$y(t) = \sum_{i=1}^N h_i u(t-i) = H(z^{-1})u(t) \quad (20)$$

Donde  $H(z^{-1}) = h_1 z^{-1} + h_2 z^{-2} + \dots + h_N z^{-N}$ . Un inconveniente de este método es el gran número de parámetros que necesita, ya que N suele ser un valor elevado (del orden de 40-50). La predicción vendrá dada por:

$$\hat{y}(t+k|t) = \sum_{i=1}^N h_i u(t+k-i|t) = H(z^{-1})u(t+k|t) \quad (21)$$

Este método es ampliamente aceptado en la práctica industrial debido a que es muy intuitivo y no requiere información previa sobre el proceso, con lo que el procedimiento de identificación se simplifica, a la vez que permite describir fácilmente dinámicas complejas como fase no mínima o retardos.

- Respuesta ante escalón. Es muy similar al anterior sólo que ahora la señal de entrada es un escalón. Para sistemas estables se tiene la respuesta truncada que será

$$y(t) = y_0 + \sum_{i=1}^N g_i \Delta u(t-i) = y_0 + G(z^{-1})(1-z^{-1})u(t) \quad (22)$$

donde las  $g_i$  son los valores muestreados ante la entrada en escalón y  $\Delta u(t) = u(t) - u(t-1)$ , según se muestra en la figura 12.b. El valor de  $y_0$  puede tomarse 0 sin pérdida de generalidad, con lo cual el predictor será:

$$\hat{y}(t+k|t) = \sum_{i=1}^N g_i \Delta u(t+k-i|t) \quad (23)$$

Este método presenta las mismas ventajas e inconvenientes que el anterior.

- Función de transferencia. Se utiliza el concepto de función de transferencia  $G = B/A$  con lo que la salida viene dada por:

$$A(z^{-1})y(t) = B(z^{-1})u(t) \quad (24)$$

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{na} z^{-na} \quad (25)$$

$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + \dots + b_{nb} z^{-nb} \quad (26)$$

Por tanto la predicción vendrá dada por

$$\hat{y}(t+k|t) = \frac{B(z^{-1})}{A(z^{-1})} u(t+k|t) \quad (27)$$

Esta representación es válida también para procesos inestables y posee la ventaja de necesitar pocos parámetros, aunque es fundamental un conocimiento a priori del proceso sobre todo en cuanto al orden de los polinomios A y B.

- Espacio de estados. Tiene la siguiente representación:

$$x(t) = Ax(t-1) + Bu(t-1) \quad (28)$$

$$y(t) = Cx(t) \quad (29)$$

siendo  $x$  el estado y  $A$ ,  $B$  y  $C$  las matrices del sistema, de entrada y de salida respectivamente. Para este modelo la predicción viene dada por

$$\hat{y}(t+k|t) = C\hat{x}(t+k|t) = C[A^k x(t) + \sum_{i=1}^k A^{i-1} Bu(t+k-1|t)] \quad (30)$$

Posee la ventaja de que sirve también para sistemas multivariantes a la vez que permite analizar la estructura interna del proceso (aunque a veces los estados obtenidos al discretizar no tienen ningún significado físico). Los cálculos pueden ser complicados, con la necesidad adicional de incluir un observador si los estados no son accesibles. (Rodríguez Ramírez, y otros, 2005)

### Modelo de las perturbaciones

De tanta importancia como la elección de un determinado modelo del proceso es la elección del modelo utilizado para representar las perturbaciones. Un modelo bastante extendido es el Autorregresivo Integrado de Media Móvil (Auto-Regressive and Integrated Moving Average, ARIMA), en el que las perturbaciones, es decir, las diferencias entre la salida medida y la calculada por el modelo vienen dadas por

$$n(t) = \frac{C(z^{-1})e(t)}{D(z^{-1})} \quad (31)$$

donde el polinomio  $D(z^{-1})$  incluye explícitamente el integrador  $\Delta = 1 - z^{-1}$ ,  $e(t)$  es un ruido de media cero y normalmente el polinomio  $C$  se considera igual a uno. Este modelo se considera apropiado para dos tipos de perturbaciones: cambios aleatorios ocurridos en instantes aleatorios (por ejemplo cambio en la calidad del material) y movimiento browniano (en procesos con balance de energía) y es usado en varios métodos. Nótese que al incluir un integrador se consigue un control con error nulo en régimen permanente (offset-free). (Rodríguez Ramírez, y otros, 2005)

Como caso particular del ARIMA se puede incluir la perturbación constante

$$n(t) = \frac{e(t)}{1-z^{-1}} \quad (32)$$

cuya mejor predicción será  $\hat{n}(t+k|t) = n(t)$ .



#### d.2.4.1.1. Respuestas libre y forzada

Una característica típica de la mayoría de los controladores MPC es el empleo de los conceptos de repuesta libre y forzada. La idea es expresar la secuencia de acciones de control como la suma de dos señales:

$$u(t) = u_f(t) + u_c(t) \quad (33)$$

La señal  $u_f(t)$  corresponde a las entradas pasadas (anteriores al instante  $t$ ) y en el futuro se mantiene constante e igual al último valor de la variable manipulada. Es decir,

$$u_f(t - j) = u(t - j) \text{ para } j = 1, 2, \dots \quad (34)$$

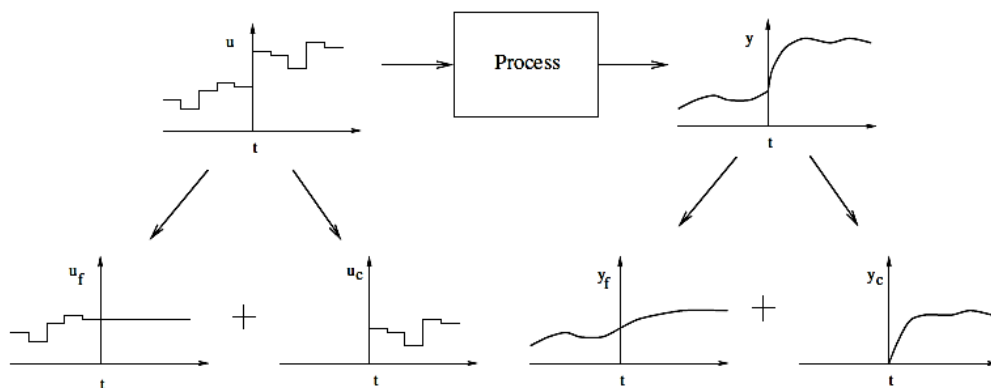
$$u_f(t + j) = u(t - 1) \text{ para } j = 0, 1, 2, \dots \quad (35)$$

La señal  $u_c(t)$  vale cero en el pasado y corresponde a las señales de control en los instantes futuros:

$$u_c(t - j) = 0 \text{ para } j = 1, 2, \dots \quad (36)$$

$$u_c(t + j) = u(t + j) - u(t - 1) \text{ para } j = 0, 1, 2, \dots \quad (37)$$

La predicción de la secuencia de salida se separa en dos partes, como se ve en la figura 13. Una de ellas ( $y_f(t)$ ), la respuesta libre, corresponde a la predicción de la salida cuando la variable manipulada se hace igual a  $u_f(t)$ , y la otra, la repuesta forzada ( $y_c(t)$ ), corresponde a la predicción de la salida cuando la señal de control es  $u_c(t)$ . La respuesta libre corresponde a la evolución del proceso debido a su estado actual (incluido por tanto el efecto de acciones pasadas) mientras que la respuesta forzada es la debida a las acciones de control futuras. (Rodríguez Ramírez, y otros, 2005)



**Figura 13.** Respuestas libre y forzada

**Fuente:** (Rodríguez Ramírez, y otros, 2005)

#### d.2.4.2. Función objetivo

Los diversos algoritmos de MPC proponen distintas funciones de coste para la obtención de la ley de control. En general se persigue que la salida futura en el horizonte considerado siga a una determinada señal de referencia al mismo tiempo que se puede penalizar el esfuerzo de control requerido para hacerlo. La expresión general de tal función objetivo será:

$$J(N_1, N_2, N_u) = \sum_{j=N_1}^{N_2} \delta(j) [\hat{y}(t+j|t) - w(t+j)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta u(t+j-1)]^2 \quad (38)$$

En algunos métodos el segundo sumando, que considera el esfuerzo de control, no se tiene en cuenta, mientras que en otros también aparecen directamente los valores de la señal de control (no sus incrementos). En la función de coste se pueden considerar:

- Parámetros:  $N_1$  y  $N_2$  son los horizontes mínimo y máximo de costo (o de predicción) y  $N_u$  es el horizonte de control, que no tiene por qué coincidir con el horizonte máximo. El significado de  $N_1$  y  $N_2$  resulta bastante intuitivo: marcan los límites de los instantes en que se desea que la salida siga a la referencia. Así, si se toma un valor grande de  $N_1$  es porque no importa que haya errores en los primeros instantes, lo cual provocará una respuesta suave del proceso. Nótese que para procesos con tiempo muerto  $d$  no tiene sentido que  $N_1$  sea menor que dicho valor puesto que la salida no empezará a evolucionar hasta el instante  $t + d$ . Además, si el proceso es de fase no mínima, este parámetro permite eliminar de la función objetivo los primeros instantes de respuesta inversa.

Los coeficientes  $\delta(j)$  y  $\lambda(j)$  son secuencias que ponderan el comportamiento futuro. Usualmente se consideran valores constantes o secuencias exponenciales. Por ejemplo se puede conseguir un peso exponencial de  $\delta(j)$  a lo largo del horizonte usando:

$$\delta(j) = \alpha^{N_2-j} \quad (39)$$

Si  $\alpha$  está comprendido entre 0 y 1 indica que se penaliza más a los errores más alejados del instante  $t$  que a los más próximos, dando lugar a un control más suave y con menor esfuerzo. Si, por el contrario,  $\alpha > 1$  es que se penalizan más los primeros errores, provocando un control más brusco.

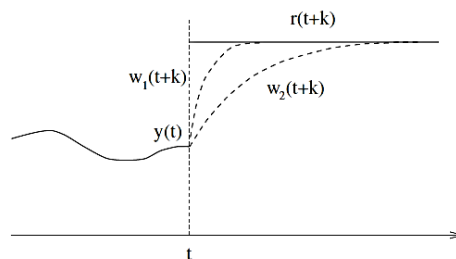
Todos estos valores pueden ser usados como parámetros de sintonización, obteniendo un abanico muy amplio de posibilidades con las que se puede cubrir una extensa gama de opciones, desde un control estándar hasta una estrategia diseñada a medida para un proceso en particular.

- **Trayectoria de referencia:** Una de las ventajas del control predictivo es que si se conoce a priori la evolución futura de la referencia, el sistema puede empezar a reaccionar antes de que el cambio se haya efectivamente realizado, evitando los efectos del retardo en la respuesta del proceso. En muchas aplicaciones la evolución futura de la referencia  $r(t+k)$  es conocida de antemano, como en Robótica, servos o procesos en *batch*; en otras aplicaciones aunque la referencia sea constante, se puede conseguir una sensible mejora de prestaciones simplemente conociendo el instante de cambio de valor y adelantándose a esa circunstancia.

En el criterio de minimización, la mayoría de los métodos suelen usar una trayectoria de referencia  $w(t+k)$  que no tiene por qué coincidir con la referencia real. Normalmente será una suave aproximación desde el valor actual de la salida  $y(t)$  a la referencia conocida mediante un sistema de primer orden:

$$w(t) = y(t) \quad w(t+k) = \alpha w(t+k-1) + (1-\alpha)r(t+k) \quad k = 1 \dots N \quad (40)$$

$\alpha$  es un parámetro comprendido entre 0 y 1 (mientras más próximo a 1 más suave será la aproximación) que constituye un valor ajustable que influirá en la respuesta dinámica del sistema. En la figura 14 se muestra la forma de la trayectoria cuando la referencia  $r(t+k)$  es constante y para dos valores distintos de  $\alpha$ ; para valores pequeños de este parámetro se tiene un seguimiento rápido ( $w_1$ ) mientras que si aumenta, la trayectoria de referencia será  $w_2$  dando lugar a una respuesta más suave.



**Figura 14.** Trayectoria de referencia

**Fuente:** (Rodríguez Ramírez, y otros, 2005)

- **Restricciones:** En la práctica, todos los procesos están sujetos a restricciones. Los actuadores tienen un campo limitado de acción así como una determinada velocidad de cambio (*slew rate*), como es el caso de las válvulas, limitadas por las posiciones de totalmente abierta o cerrada y por la velocidad de respuesta. Razones constructivas, de seguridad o medioambientales o bien los propios alcances de los sensores pueden causar límites en las variables de proceso, tales como niveles en depósitos, caudales en tuberías o temperaturas y presiones máximas. Además, normalmente las condiciones de operación vienen definidas por la intersección de ciertas restricciones por motivos fundamentalmente económicos, con lo que el sistema de control operará cerca de los límites. Todo lo expuesto anteriormente hace necesaria la introducción de restricciones en la función a minimizar.

Muchos algoritmos predictivos tienen en cuenta el tema de las restricciones por lo cual han tenido gran éxito en la industria. Normalmente se considerarán límites en la amplitud y el *slew rate* de la señal de control y límites en las salidas:

$$u_{min} \leq u(t) \leq u_{max} \quad \forall t$$

$$du_{min} \leq u(t) - u(t-1) \leq du_{max} \quad \forall t$$

$$y_{min} \leq y(t) \leq y_{max} \quad \forall t$$

con la adición de estas restricciones a la función objetivo, la minimización resulta más compleja, no pudiendo obtenerse la solución analíticamente como en el caso sin restringir. (Rodríguez Ramírez, y otros, 2005)

#### **d.2.4.3. Obtención de la ley de control**

Para obtener los valores  $u(t+k|t)$  será necesario minimizar la funcional  $J$  de la ecuación (38). Para ello se calculan los valores de las salidas predichas  $\hat{y}(t+k|t)$  en función de valores pasados de entradas y salidas y de señales de control futuras, haciendo uso del modelo que se haya elegido y se sustituyen en la función de coste, obteniendo una expresión cuya minimización conduce a los valores buscados. Para el criterio cuadrático si el modelo es lineal y no existen restricciones se puede obtener una solución analítica, en otro caso se debe usar un método iterativo de optimización.

De cualquiera de las maneras la obtención de la solución no resulta trivial pues existirán  $N_2 - N_1 + 1$  variables independientes, valor que puede ser elevado (del orden de 10 a 30). Con la idea de reducir estos grados de libertad se puede proponer cierta estructura a la ley de control. Además se ha encontrado que esta estructuración de la ley de control produce una mejora en la robustez y en el comportamiento general del sistema, debido fundamentalmente a que el hecho de permitir la libre evolución de las variables manipuladas (sin estructurar) puede conducir a señales de control de alta frecuencia no deseables y que en el peor de los casos podrían conducir a la inestabilidad.

Esta estructura de la ley de control se plasma en el uso del concepto de horizonte de control ( $N_u$ ), que consiste en considerar que tras un cierto intervalo  $N_u < N_2$  no hay variación en las señales de control propuestas, es decir:

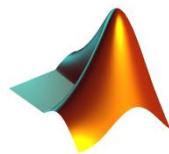
$$\Delta u(t + j - 1) = 0 \quad j > N_u \quad (41)$$

lo cual es equivalente a dar pesos infinitos a los cambios en el control a partir de cierto instante. El caso límite sería considerar  $N_u$  igual a 1 con lo que todas las acciones futuras serían iguales a  $u(t)$ . (Rodríguez Ramírez, y otros, 2005)

### **d.3. CAPÍTULO III: Herramientas de Matlab aplicadas al control del proceso**

Matlab es un Laboratorio matemático, que ofrece un entorno de desarrollo integrado, empleando programación en el Lenguaje M.

Las funciones básicas son: Manejar matrices, representación de datos y funciones, implementar algoritmos, crear interfaces gráficas de usuario. A parte de éstas funciones, Matlab permite la instalación de herramientas que el usuario necesite, tales como Simulink, y GUIDE y la instalación de diferentes Toolboxes.



**Figura 15.** Icono de Matlab

**Fuente:** (The MathWorks Inc., 2015)

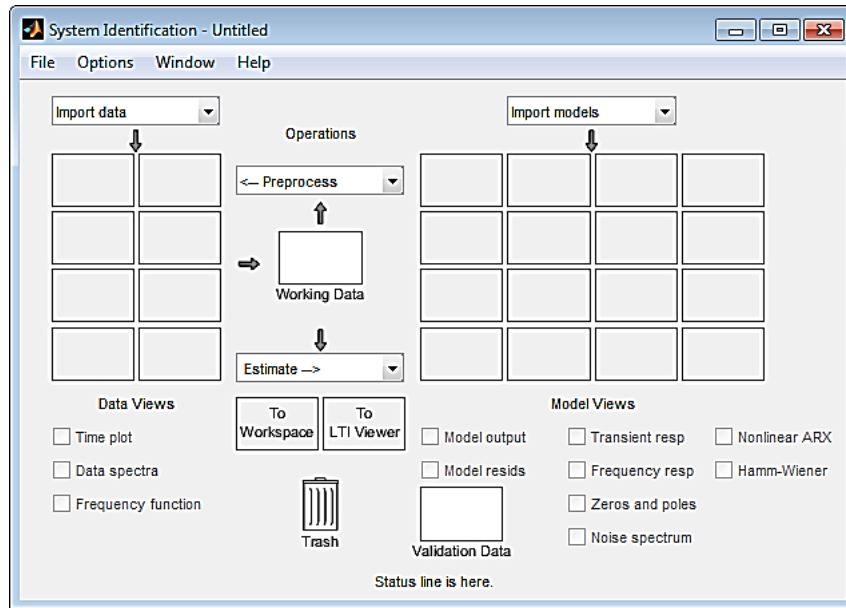
#### **d.3.1. Toolbox de Identificación de Sistemas (ident)**

La Toolbox de Identificación de Sistemas es para la construcción de precisión, simplificado de modelos de sistemas complejos a partir de los datos de series de tiempo ruidosos. Proporciona herramientas para la creación de modelos matemáticos de sistemas dinámicos basado en los datos de entrada / salida observados. La Toolbox cuenta con un sistema flexible interfaz gráfica de usuario que ayuda en la organización de los datos y modelos.

Las técnicas de identificación proporcionadas con esta Toolbox son útiles para aplicaciones que van desde el diseño de sistema de control y procesamiento de señales para análisis de series de tiempo y análisis de vibraciones.

Un flujo de trabajo típico en la aplicación de identificación de sistema incluye los siguientes pasos: (The MathWorks, Inc, 2000)

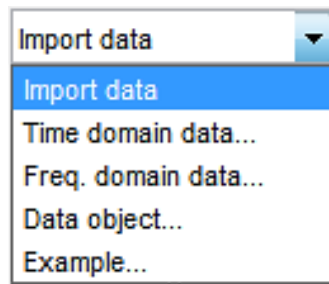
1. Importar sus datos en el espacio de trabajo de MATLAB
2. Iniciar una nueva sesión en la aplicación de Identificación de Sistemas, o abrir una sesión guardada.



**Figura 16.** Ventana de Identificación de Sistemas

**Fuente:** (The MathWorks, Inc, 2000)

3. Importar datos en la aplicación desde el espacio de trabajo de MATLAB.

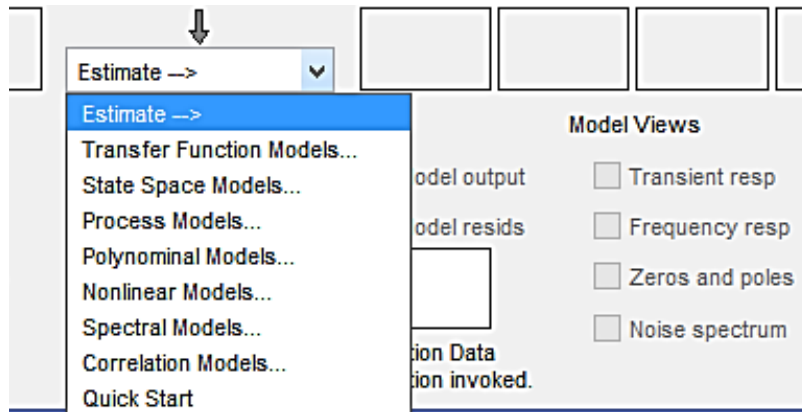


**Figura 17.** Importar datos

**Fuente:** (The MathWorks, Inc, 2000)

4. Estimación de parámetros del modelo.

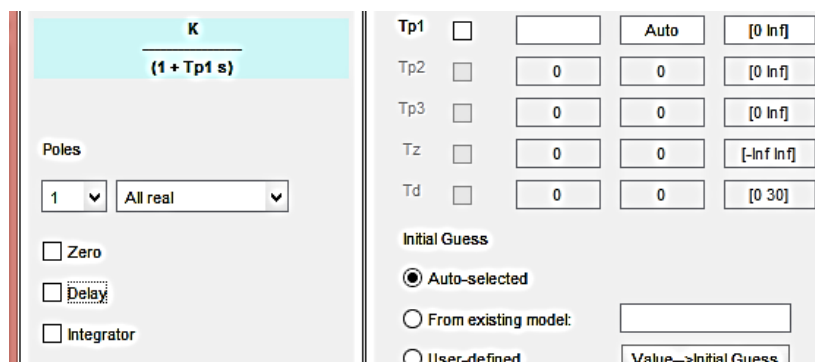
Los modelos paramétricos, tales como funciones de transferencia o modelos de espacio de estado, utilizan un pequeño número de parámetros para capturar la dinámica del sistema. La Toolbox de Identificación de Sistemas estima los parámetros del modelo y sus incertidumbres de tiempo de respuesta y los datos de respuesta en frecuencia. Puede analizar estos modelos utilizando el tiempo de respuesta y parcelas de respuesta en frecuencia, como paso, impulso, diagramas de Bode, y mapas de polos y ceros.



**Figura 18.** Estimar múltiples modelos

**Fuente:** (The MathWorks, Inc, 2000)

La Toolbox de Identificación de Sistemas permite estimar con múltiples entradas y múltiples salidas de transferencia de tiempo discreto o continuo funciones con un número determinado de polos y ceros. Se puede especificar el retardo de transporte o dejar que la Toolbox lo determine de forma automática. En los casos en que se necesita un modelo de tiempo continuo de orden inferior en polos y ceros, la Toolbox permite estimar modelos de procesos, que son funciones de transferencia simples que involucran tres o menos polos, y opcionalmente, un cero, un tiempo de retardo, y un integrador.



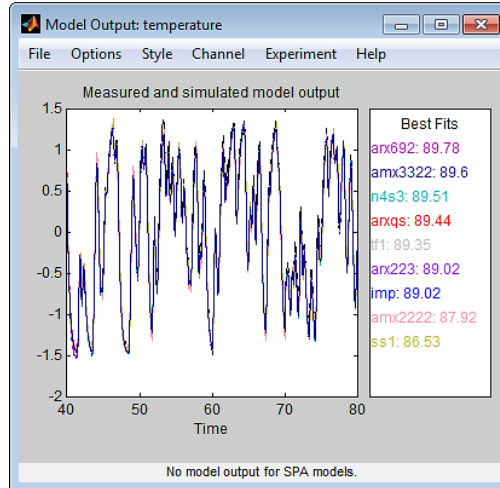
**Figura 19.** Ventana de estimación de modelos de proceso

**Fuente:** (The MathWorks, Inc, 2000)

## 5. Validar modelos.

Después de estimar un modelo, se puede validar si se reproduce el comportamiento del sistema dentro de límites aceptables. Iterar entre el modelo de perfeccionamiento y validación hasta encontrar el modelo más simple que capte mejor la dinámica del sistema.





**Figura 20.** Validación del modelo

**Fuente:** (The MathWorks, Inc, 2000)

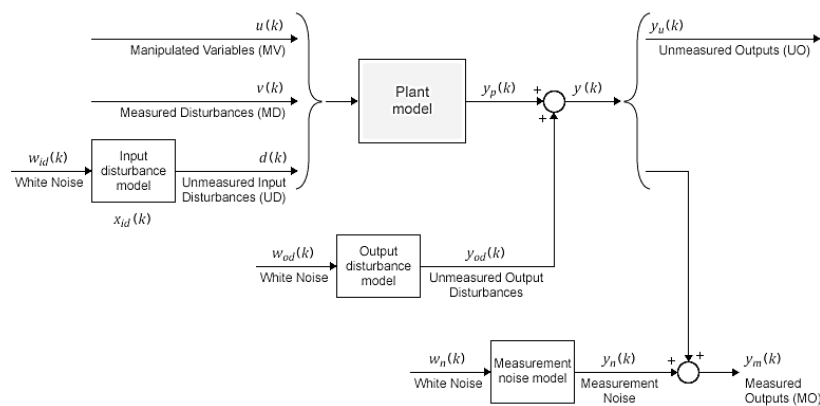
6. Exportar modelos al área de trabajo de MATLAB para su posterior análisis. (The MathWorks, Inc, 2000)

### d.3.2. Toolbox Model Predictive Controller (MPC)

Matlab dispone un conjunto de herramienta para el diseño de controladores predictivos. Estas herramientas se basan en modelos en el espacio de estados y admite el tratamiento de restricciones y optimizaciones de forma transparente. (The MathWorks, Inc, 2014)

#### d.3.2.1. Modelado MPC

La estructura del modelo utilizado en el controlador MPC aparece en la siguiente figura. En donde se explica cómo los modelos se conectan para la predicción y estimación de estado. (The MathWorks Inc., 2015)



**Figura 21.** Estructura del modelo utilizado en el controlador MPC

**Fuente:** (The MathWorks Inc., 2015)

## Modelo de Planta

El modelo de la planta puede estar en uno de los formatos (LTI) siguiente invariante en tiempo lineal:

- Modelos numéricos LTI: Función de transferencia (TF), espacio de estado (ss), cero-polo-ganancia (ZPK).
- Modelos Identificados: IDSS, IDTF, idproc y idpoly.

El controlador MPC realiza todos los cálculos de estimación y optimización mediante un tiempo discreto, sistema de espacio de estado sin retrasos, con variables de entrada y salida sin dimensiones. Por lo tanto, cuando se especifica un modelo de planta en el controlador MPC, los siguientes pasos del modelo de conversión se realizan automáticamente (si es necesario): (The MathWorks Inc., 2015)

1. La conversión en espacio de estado. El comando `ss` apropiado para el formato modelo suministrado genera un modelo de estado-espacio LTI.
2. Discretización o remuestreo. Si el tiempo de muestreo del modelo se diferencia de tiempo de muestreo del controlador MPC (definido en la propiedad `Ts`), una de lo siguiente ocurre:
  - Si el modelo es en tiempo continuo, el comando `c2d` la convierte en un objeto de LTI en tiempo discreto usando el tiempo de muestreo del controlador.
  - Si el modelo es de tiempo discreto, el comando `d2d` vuelve a muestrear para generar un objeto en tiempo discreto LTI usando el tiempo de muestreo del controlador.
3. Eliminación de retardo. Si el modelo de tiempo discreto incluye cualquier entrada, salida o retrasos internos, el comando `absorbDelay` los reemplaza con el número apropiado de polos en  $z = 0$ , aumentando el número total de estados discretos. Las propiedades `InputDelay`, `OutputDelay` e `InternalDelay` del modelo de espacio de estado resultante son todos cero.
4. Conversión de variables de entrada y salida sin dimensiones. El controlador MPC le permite especificar un factor de escala para cada entrada de la planta y la variable de salida. Si no especifica los factores de escala, que por defecto es la unidad. El software convierte las variables de entrada y salida de la planta de forma adimensional de la siguiente manera:

$$x_p(k + 1) = A_p x_p(k) + B S_i u_p(k) \quad (42)$$

$$y_p(k) = S_o^{-1} C x_p(k) + S_o^{-1} D S_i u_p(k) \quad (43)$$

Aquí, donde  $A_p$ ,  $B$ ,  $C$ , y  $D$  son matrices de espacio de estado constante determinada en el paso 3, y:

- $S_i$  - matriz diagonal de los factores de escala de entrada en unidades de ingeniería.
- $S_o$  - matriz diagonal de los factores de escala de salida en unidades de ingeniería.
- $x_p$  - vector Estado desde el paso 3 en unidades de ingeniería. No escalado se realiza sobre variables de estado.
- $u_p$  - las variables de entrada de la planta sin dimensiones.
- $u_p$  - las variables de salida de la planta sin dimensiones.

El modelo de la planta resultante tiene la siguiente forma equivalente:

$$x_p(k + 1) = A_p x_p(k) + B_{pu} u(k) + B_{pv} v(k) + B_{pd} d(k) \quad (44)$$

$$y_p(k) = C_p x_p(k) + D_{pu} u(k) + D_{pv} v(k) + D_{pd} d(k) \quad (45)$$

Aquí,  $C_p = S_o^{-1} C$ ,  $B_{pu}$ ,  $B_{pv}$  y  $B_{pd}$  son las columnas correspondientes de la  $B S_i$ . También,  $D_{pu}$ ,  $D_{pv}$  y el  $D_{pd}$  son las columnas correspondientes de  $S_o^{-1} D S_i$ . Finalmente,  $u(k)$ ,  $v(k)$  y  $d(k)$  son las variables adimensionales manipuladas, perturbaciones medidas y perturbaciones de entrada no medidas, respectivamente.

Controlador MPC impone la restricción de la  $D_{pu} = 0$ , lo que significa que el controlador MPC no permite conexión de interfaz directa de cualquier variable manipulada a cualquier salida de la planta. (The MathWorks Inc., 2015)

### Modelo de Perturbación de entrada

Si el modelo de planta incluye perturbaciones de entrada no medidas, etiquetadas como  $d(k)$  en el diagrama del sistema MPC, el modelo de perturbación de entrada indica como  $d(k)$  evoluciona con el tiempo. Es decir, el modelo de perturbación de entrada especifica qué tipo de señal se espera de  $d(k)$ . Si no se proporciona un modelo de perturbación de entrada, el controlador utiliza un modelo de perturbación de entrada por defecto. El comando *getindist* proporciona acceso al modelo que se utiliza.

El modelo de perturbación de entrada es un factor clave que influye en los siguientes comportamientos de atributos del controlador: (The MathWorks Inc., 2015)

- Respuesta dinámica a las perturbaciones aparentes, es decir, el carácter de la respuesta del controlador cuando la producción de la planta medida se desvía de su trayectoria prevista (debido a una perturbación desconocida o error de modelado).
- Rechazo asintótica de las perturbaciones sufridas. Si el modelo predice una perturbación sostenida, los ajustes del controlador continúan hasta que la salida de la planta vuelve a su trayectoria deseada. Este emula un modo integral en un controlador de retroalimentación clásica.

Puede proporcionar el modelo de perturbación de entrada como un espacio de estado LTI (ss), la función de transferencia (TF), o cero-polo-ganancia (ZPK).

El controlador MPC convierte el modelo de perturbación de entrada a un tiempo discreto, retraso- libre, sistema LTI de estado-espacio con los mismos pasos que se utilizan para convertir el modelo de la planta. El resultado es:

$$x_{id}(k + 1) = A_{id}x_{id}(k) + B_{id}w_{id}(k) \quad (46)$$

$$d(k) = C_{id}x_{id}(k) + D_{id}w_{id}(k) \quad (47)$$

Aquí,  $A_{id}$ ,  $B_{id}$ ,  $C_{id}$  y  $D_{id}$  son matrices de espacio de estado constantes y:

$x_{id}(k) - n_{xid} \geq 0$  estados perturbación de entrada del modelo.

$d_k(k) - n_d$  perturbaciones adimensionales de entrada no medidos.

$w_{id}(k) - n_{id} \geq 1$  entradas de ruido blanco sin dimensiones, supone que tiene media cero y varianza unidad.

### **Modelo de Perturbación de salida**

El modelo de perturbación de salida es un caso especial del modelo más general que la perturbación de entrada. Su salida se añade directamente a la salida de la planta en lugar de afectar los estados de la central. El diagrama muestra su ubicación en la jerarquía del modelo MPC. El modelo de perturbación de salida indica cómo la perturbación evolucionará con el tiempo y se utiliza a menudo en la práctica.

Si no se proporciona un modelo de perturbación de salida, el controlador MPC creará uno por defecto cuando no existe un modelo de perturbación de entrada y aumentando el modelo de planta no viola la observabilidad de estado. El comando *getoutdist* proporciona acceso al modelo que se utiliza.

Utilizando los mismos pasos que para el modelo de planta, el controlador MPC convierte el modelo de perturbación de salida a un tiempo discreto, sistema LTI espacio de estado sin retrasos. El resultado es: (The MathWorks Inc., 2015)

$$x_{od}(k+1) = A_{od}x_{od}(k) + B_{od}w_{od}(k) \quad (48)$$

$$y_{od}(k) = C_{od}x_{od}(k) + D_{od}w_{od}(k) \quad (49)$$

Aquí,  $A_{od}$ ,  $B_{od}$ ,  $C_{od}$  y  $D_{od}$  son matrices de espacio de estado constantes y:

$x_{od}(k) - n_{xod} \geq 1$  estados perturbación de salida del modelo.

$y_{od}(k) - n_y$  perturbaciones de salida adimensionales que se añaden a las salidas de plantas adimensionales.

$w_{od}(k) - n_{od}$  entradas de ruido blanco sin dimensiones, supone que tiene media cero, varianza unidad.

### Medición de ruido del modelo

Uno de los objetivos de diseño del controlador es distinguir los disturbios, que requieren una respuesta, a partir de la medición de ruido, lo que debería ser ignorado. La medición de ruido del modelo tiene este propósito. El diagrama muestra su ubicación en la jerarquía del modelo MPC. La medición de ruido del modelo indica cómo el ruido va a evolucionar con el tiempo (es decir, qué tipo de ruido de la señal se puede esperar).

Utilizando los mismos pasos que para el modelo de planta, el controlador MPC convierte el modelo de medición de ruido a un tiempo discreto, retraso-libre, sistema LTI espacio de estado sin retrasos. El resultado es: (The MathWorks Inc., 2015)

$$x_n(k+1) = A_nx_n(k) + B_nw_n(k) \quad (50)$$

$$y_n(k) = C_nx_n(k) + D_nw_n(k) \quad (51)$$

Aquí,  $A_n$ ,  $B_n$ ,  $C_n$  y  $D_n$  son matrices de espacio de estado constantes y:

$x_n(k) - n_{xn} \geq 0$  estados de ruido del modelo.

$y_n(k) - n_{ym}$  señales de ruido adimensionales que se añaden a las salidas de planta medida adimensionales.

$w_n(k) - n_n \geq 1$  entradas de ruido blanco sin dimensiones, supone que tiene media cero, varianza unidad.

Si no se proporciona un modelo de ruido, el valor predeterminado es una unidad de ganancia estática:  $n_{xn} = 0$ ,  $D_n$  es una matriz identidad  $n_{ym} \times n_{ym}$ , y  $A_n$ ,  $B_n$  y  $C_n$  están vacías.

#### **d.3.2.2. Estimación del estado del controlador**

Cómo funciona el controlador, que utiliza su estado actual,  $x_c$ , como base para predicciones. Por definición, el vector de estado es el siguiente:

$$x_c^T(k) = [x_c^T(k) \quad x_{id}^T(k) \quad x_{od}^T(k) \quad x_n^T(k)] \quad (52)$$

Donde,

- $x_c$  es el estado del controlador, que comprende las variables de estado  $n_{xp} + n_{xid} + n_{xod} + n_{xn}$ .
- $x_p$  es el vector de estado modelo de la planta, de longitud  $n_{xp}$ .
- $x_{id}$  es el vector de estado modelo de perturbación de entrada, de longitud  $n_{xid}$ .
- $x_{od}$  es el vector de estado modelo de perturbación de salida, de longitud  $n_{xod}$ .
- $x_n$  es el vector de estado modelo de medición de ruido, de longitud  $n_{xn}$ .

Por defecto, el controlador actualiza su estado de forma automática utilizando las últimas mediciones de la planta. Por otra parte, la función de estimación de estado personalizada le permite actualizar el estado del controlador usando un procedimiento externo y, a continuación, proporciona estos valores en el controlador. (The MathWorks Inc., 2015)

#### **Observador de Estado**

La combinación de los modelos que se muestran en el diagrama se obtiene el observador de estado:

$$x_c(k+1) = Ax_c(k) + Bu_o(k) \quad (53)$$

$$y(k) = Cx_c(k) + Du_o(k) \quad (54)$$

El Controlador MPC utiliza el observador de estado de las siguientes maneras:

- Para estimar los valores de los estados no medidos necesarios como base para predicciones.
- Para predecir cómo las variables manipuladas propuestas por el controlador (MV) afectarán a los valores de salida futuros de la planta.

Las señales de entrada del observador son adimensional manipuladas de la planta y entradas de perturbaciones medidas, y las entradas de ruido blanco a los modelos de perturbación y de ruido:

$$u_o^T(k) = [u^T(k) \ v^T(k) \ w_{id}^T(k) \ w_{od}^T(k) \ w_n^T(k)] \quad (55)$$

Las salidas del observador son las  $n_y$  salidas de la planta sin dimensiones.

En términos de los parámetros que definen los cuatro modelos que se muestran en el diagrama, los parámetros del observador son:

$$\begin{aligned} A &= \begin{bmatrix} A_p & B_{pd}C_{id} & 0 & 0 \\ 0 & A_{id} & 0 & 0 \\ 0 & 0 & A_{od} & 0 \\ 0 & 0 & 0 & A_n \end{bmatrix}, \quad B = \begin{bmatrix} B_{pu} & B_{pv} & B_{pd}D_{id} & 0 & 0 \\ 0 & 0 & B_{id} & 0 & 0 \\ 0 & 0 & 0 & B_{od} & 0 \\ 0 & 0 & 0 & 0 & B_n \end{bmatrix} \\ C &= \begin{bmatrix} C_p & D_{pd}C_{id} & C_{od} & \begin{bmatrix} C_n \end{bmatrix} \\ 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & D_{pv} & D_{pd}D_{id} & D_{od} & \begin{bmatrix} D_n \end{bmatrix} \\ 0 \end{bmatrix} \end{aligned} \quad (56)$$

Aquí, los modelos de perturbación de la planta y la producción se resecuencia para que las salidas medidas precedan a las salidas no medidas. (The MathWorks Inc., 2015)

### Estimación de Estado

En general, los estados del controlador son no medidos y deben ser estimados. Por defecto, el controlador utiliza un filtro de Kalman estado estacionario que se deriva del observador de estado.

Al comienzo del  $k$ -ésimo intervalo de control, el estado del controlador se calcula con los siguientes pasos: (The MathWorks Inc., 2015)

1. Obtener los datos siguientes:

- $x_c(k | k-1)$  - Controlador de estimación de estado de intervalo de control anterior,  $k-1$
- $u^{act}(k-1)$  - de hecho se utiliza en la planta de la  $k-1$  a  $k$  Variable manipulada (MV) (que se supone constante)
- $u^{opt}(k-1)$  - Optima MV recomendado por MPC y se asume que se utiliza en la planta de la  $k-1$  a  $k$
- $v(k)$  - Actual medida disturbios
- $y_m(k)$  - Actual medida salidas de plantas
- $B_u, B_v$  - Columnas del parámetro observador B correspondiente a entradas  $u(k)$  y  $v(k)$
- $C_m$  - Filas de parámetro observador C correspondiente a las salidas de plantas medidas
- $D_{mv}$  - Filas y columnas de parámetro observador D correspondientes a las salidas de plantas medidas e insumos de perturbación medidos
- $L, M$  - Constantes matrices de ganancia de Kalman

Las señales de entrada y salida de la planta se escalan para ser adimensional antes de su uso en los cálculos.

2. Revisar  $x_c(k|k-1)$  cuando  $u^{act}(k-1)$  y  $u^{opt}(k-1)$  son diferentes:

$$x_c^{rev}(k|k-1) = x_c(k|k-1) + B_u[u^{act}(k-1) - u^{opt}(k-1)] \quad (57)$$

3. Calcular la innovación:

$$e(k) = y_m(k) - [C_m x_c^{rev}(k|k-1) + D_{mv}v(k)] \quad (58)$$

4. Actualización de la estimación de estado del controlador para dar cuenta de las últimas mediciones.

$$x_c(k|k) = x_c^{rev}(k|k-1) + Me(k) \quad (59)$$

A continuación, el software utiliza la estimación de estado  $x_c(k|k)$  para resolver el programa de segundo grado en  $k$  intervalos. La solución es  $u^{opt}(k)$ , valor de la variable manipulada recomendado por el MPC para ser utilizado entre los intervalos de control  $k$  y  $k+1$ .

Finalmente, el software se prepara para el siguiente intervalo de control suponiendo que las entradas desconocidas,  $w_{id}(k)$ ,  $w_{od}(k)$ , y  $w_n(k)$  asumen su valor medio (cero)



entre los tiempos de  $k$  y  $k + 1$ . El software predice el impacto de las entradas conocidas y la innovación de la siguiente manera:

$$x_c(k + 1|k) = Ax_c(k|k) + B_u u^{opt}(k) + B_v v(k) + Le(k) \quad (60)$$

### Predicción Variable de salida

El Modelo de Control Predictivo requiere la predicción de futuras salidas de plantas libres de ruido utilizadas en la optimización. Esta es una aplicación clave del observador de estado.

En el control de intervalo  $k$ , los datos necesarios son los siguientes:

- $p$  - Horizonte de predicción (número de intervalos de control, que es mayor o igual a 1)
- $x_c(k|k)$  - Estimaciones del estado del controlador
- $v(k)$  - Actual perturbación de entrada medida (MD)
- $v(k + i|k)$  - Futuro proyectado MD, donde  $i = 1: p-1$ . Si no está utilizando la vista previa MD, entonces  $v(k + i|k) = v(k)$ .
- $A, B_u, B_v, C, D_v$  - Constantes de observador de estado, donde  $B_u, B_v$ , y  $D_v$  denotan columnas de las matrices  $B$  y  $D$  correspondientes a las entradas  $u$  y  $v$ .  $D_u$  es una matriz nula por falta de alimentación de paso directo

Las predicciones asumen que las entradas de ruido blanco desconocidos son cero (su expectativa). Además, las salidas de la planta previstas son libres de ruido. De este modo, todos los términos medición de ruido de estados que participan desaparecen de las ecuaciones de observación del estado. Esto es equivalente a la reducción a cero de los últimos elementos de  $n \times n$  de  $x_c(k|k)$ .

Teniendo en cuenta los datos y las simplificaciones anteriores, por el primer paso del observador de estado predice:

$$x_c(k + 1|k) = Ax_c(k|k) + B_u u(k|k) + B_v v(k) \quad (61)$$

Continuando por etapas sucesivas,  $i = 2: p$ , el observador de estado predice:

$$x_c(k + i|k) = Ax_c(k + i - 1|k) + B_u u(k + i - 1|k) + B_v v(k + i - 1|k) \quad (62)$$

En cualquier paso,  $i = 1: p$ , las salidas de plantas libres de ruido previstos son:

$$y(k + i|k) = Cx_c(k + i|k) + D_v v(k + i|k) \quad (63)$$

Todas estas ecuaciones emplean variables de entrada y de salida de la planta adimensionales. Las ecuaciones también asumen los traslados de origen. La inclusión de las compensaciones no nulas es sencillo.

Para cálculos más rápidos, el controlador MPC utiliza una forma alternativa de las ecuaciones anteriores en las que términos constantes se calculan y almacenan durante la inicialización del controlador. (The MathWorks Inc., 2015)

### **d.3.2.3. Optimización del problema**

#### **Visión de conjunto**

El Modelo de Control Predictivo resuelve un problema de optimización, específicamente, un programa cuadrático (QP), en cada intervalo de control. La solución determina las variables manipuladas (MV) para ser utilizado en la planta hasta el siguiente intervalo de control. (The MathWorks Inc., 2015)

Este problema QP incluye las siguientes características:

- La función objetivo (o función de costo) - Un escalar, medida no negativa de rendimiento del controlador que se reduce al mínimo.
- Restricciones - Condiciones que la solución debe satisfacer, como los límites físicos en MV y variables de salida de la planta.
- Decisión - Los ajustes de MV que minimiza la función de costo al tiempo que satisface las restricciones.

#### **Función de costos estándar**

La función de costo estándar es la suma de cuatro términos, cada uno centrado en un aspecto concreto de la actuación del controlador, de la siguiente manera:

$$J(z_k) = J_y(z_k) + J_u(z_k) + J_{\Delta u}(z_k) + J(z_k) \quad (64)$$

Aquí,  $z_k$  es la decisión QP. Como se describe a continuación, cada término incluye pesos que ayudan a equilibrar los objetivos perseguidos. El Controlador MPC ofrece pesos por defecto pero en general tendrá que ajustarlos al ajustar el controlador para su aplicación. (The MathWorks Inc., 2015)

### ***Seguimiento Referencia de salida***

En la mayoría de las aplicaciones, el controlador debe mantener ciertas salidas de planta en o cerca de los valores de referencia especificados. Controlador MPC utiliza la siguiente medida de desempeño escalar:

$$J_y(z_k) = \sum_{j=1}^{n_y} \sum_{i=1}^p \left\{ \frac{w_{i,j}^y}{s_j^y} [r_j(k+i|k) - y_j(k+i|k)] \right\}^2 \quad (65)$$

Donde,

- $k$  - Intervalo de control actual.
- $p$  - Horizonte de predicción (número de intervalos).
- $n_y$  - Número de variables de salida de la planta.
- $z_k$  - Decisión QP, dada por :

$$z_k^T = [u(k|k)^T \ u(k+1|k)^T \dots u(k+p-1|k)^T \quad k] \quad (66)$$

- $y_j(k+i|k)$  - Valor pronosticado de la producción de la planta  $j$  al  $i$ -ésimo paso horizonte de predicción, en unidades de ingeniería.
- $r_j(k+i|k)$  - Valor de referencia para la salida de la planta  $j$  en  $i$ -ésimo paso horizonte de predicción, en unidades de ingeniería.
- $s_j^y$  - Factor de escala para la producción de la planta  $j$ , en unidades de ingeniería.
- $w_{i,j}^y$  - Peso de sintonización para la salida de la planta  $j$  en  $i$ -ésimo paso horizonte de predicción (sin dimensiones).

Los valores de  $n_y$ ,  $p$ ,  $s_j^y$ , y  $w_{i,j}^y$  son especificaciones del controlador, y son constantes. El controlador recibe los valores  $r_j(k+i|k)$  para todo el horizonte de predicción. El controlador utiliza el observador de estado para predecir las salidas de las plantas. Al intervalo  $k$ , las estimaciones del estado del controlador y los valores de MD están disponibles. Por lo tanto,  $J_y$  es una función de sólo  $z_k$ . (The MathWorks Inc., 2015)

### ***Seguimiento variable manipulada***

En algunas aplicaciones, es decir, cuando hay más variables manipuladas que las salidas de la planta, el controlador debe mantener variables manipuladas seleccionadas (MV) en o cerca de los valores objetivo especificados. El Controlador MPC utiliza la siguiente medida de desempeño escalar:

$$J_u(z_k) = \sum_{j=1}^{n_{cc}} \sum_{i=0}^{p-1} \left\{ \frac{w_{i,j}^u}{s_j^u} [u_j(k+i|k) - u_{j,target}(k+i|k)] \right\}^2 \quad (67)$$

Donde,

- $k$  - Intervalo de control actual.
- $p$  - Horizonte de predicción (número de intervalos).
- $n_u$  - Número de variables manipuladas.
- $z_k$  - Decisión QP, dada por (66)
- $u_{j, objetivo}(k+i|k)$  - Valor objetivo para  $j$  MV en la predicción  $i$ -ésimo paso horizonte, en unidades de ingeniería.
- $s_j^y$  -Factor de escala para  $j$  MV, en unidades de ingeniería.
- $w_{i,j}^y$  -Peso de sintonización para  $j$  MV en  $i$ -ésimo paso del horizonte de predicción (sin dimensiones).

Los valores  $n_u$ ,  $p$ ,  $s_j^y$  y  $w_{i,j}^y$  son especificaciones del controlador, y son constantes. El controlador recibe los valores  $u_{j, objetivo}(i+k|k)$  para todo el horizonte. El controlador utiliza el observador de estado para predecir las salidas de las plantas. Por lo tanto,  $J_u$  es una función de sólo  $z_k$ . (The MathWorks Inc., 2015)

### ***Supresión de Variable manipulada***

La mayoría de las aplicaciones prefieren pequeños ajustes de MV (se mueve). MPC utiliza la siguiente medida de desempeño escalar:

$$J_{\Delta u}(z_k) = \sum_{j=1}^{n_{cc}} \sum_{i=0}^{p-1} \left\{ \frac{w_{i,j}^{\Delta u}}{s_j^u} [u_j(k+i|k) - u_j(k+i-1|k)] \right\}^2 \quad (68)$$

Donde,

- $k$  - Intervalo de control actual.
- $p$  - Horizonte de predicción (número de intervalos).
- $n_u$  - Número de variables manipuladas.
- $z_k$  - Decisión QP, dada por (66)
- $s_j^y$  -Factor de escala para  $j$  MV, en unidades de ingeniería.
- $w_{i,j}^{\Delta u}$  -Peso de sintonización para  $j$  MV en  $i$ -ésimo paso del horizonte de predicción (sin dimensiones).

Los valores  $n_u$ ,  $p$ ,  $s_j^y$  y  $w_{i,j}^{\Delta u}$  son especificaciones del controlador, y son constantes.  $u(k-1|k) = u(k-1)$ , que son las MVs conocidos a partir del intervalo de control previo.  $J_{\Delta u}$  es una función de sólo  $z_k$ .

Además, un horizonte de control  $m \leq p$  (o el bloqueo MV) limita cierta MV se mueve a ser cero. (The MathWorks Inc., 2015)

### ***Violación de restricciones***

En la práctica, la violación de restricción podría ser inevitable. Las Restricciones blandas permiten una solución QP posible bajo tales condiciones. El Controlador MPC emplea una dimensión, variable de holgura no negativo  $\epsilon_k$ , que cuantifica el peor de los casos de violación de restricción. La medida de rendimiento correspondiente es: (The MathWorks Inc., 2015)

$$J(z_k) = p \frac{2}{k} \quad (69)$$

Donde,

- $z_k$  - Decisión QP, dada por (66)
- $\epsilon_k$  - Variable de holgura en el control de intervalo  $k$  (sin dimensiones).
- $\rho\epsilon$  - Peso de penalización de violación de restricción (sin dimensiones).

### **Función de Costo Alternativa**

Se puede optar por utilizar la siguiente alternativa a la función de costo estándar:

$$J(z_k) = \sum_{i=0}^{p-1} \{ [e_y^T(k+i)Qe_y(k+i)] + [e_u^T(k+i)R_ue_u(k+i)] + [\Delta u^T(k+i)R_{\Delta u}\Delta u(k+i)] \} + p_k^2 \quad (70)$$

Aquí,  $Q$  ( $n_y$ -por- $n_y$ ),  $R_u$ , y  $R_{\Delta u}$  ( $n_u$ -por- $n_u$ ) son matrices de peso positivo semi definidos, y:

$$e_y(i+k) = S_y^{-1}[r(k+i+1|k) - y(k+i+1|k)] \quad (71)$$

$$e_u(i+k) = S_u^{-1}[u_{target}(k+i|k) - u(k+i|k)] \quad (72)$$

$$\Delta u(k+i) = S_u^{-1}[u(k+i|k) - u(k+i-1|k)] \quad (73)$$

también,

- $S_y$  - Matriz diagonal de salida de la planta factores de escala variable en unidades de ingeniería.
- $S_u$  - Matriz diagonal de los factores de escala MV en unidades de ingeniería.
- $r(k+1 | k)$  - Los valores de referencia de salida de planta de  $n_y$  en el paso horizonte de predicción  $i$ , en unidades de ingeniería.
- $y(k+1 | k)$  - Salidas de plantas  $n_y$  en el paso horizonte de predicción  $i$ , en unidades de ingeniería.
- $z_k$  - Decisión QP, dada por (66)
- $u_{\text{target}}(k+i | k)$  -  $n_u$  valores objetivo MV correspondientes a  $u(k+i | k)$ , en unidades de ingeniería.

Predicciones de salida utilizan el observador de estado, como en la función de costos estándar.

La función de costo alternativa permite ponderación off-diagonal, pero requiere los pesos que son idénticos en cada paso del horizonte de predicción.

Las funciones de costos alternativos y estándar son idénticos si cumplen las siguientes condiciones: (The MathWorks Inc., 2015)

- Las funciones de coste estándar emplea pesos, y que son constantes con respecto al índice,  $i = 1:p$ .
- Las matrices  $Q$ ,  $R_u$ , y  $R_{\Delta u}$  son diagonales con los cuadrados de los pesos como los elementos de la diagonal.

## Restricciones

Ciertas limitaciones están implícitos. Por ejemplo, un horizonte de control  $m < p$  (o el bloqueo MV) obliga a algunos incrementos MV a ser cero, y el observador de estado utilizado para la predicción de producción de la planta es un conjunto de restricciones de igualdad implícitos. Limitaciones explícitas que se pueden configurar se describen a continuación. (The MathWorks Inc., 2015)

### *Límites en las salidas de las plantas, MV, e Incrementos MV*

Las limitaciones MPC más comunes son los límites, de la siguiente manera.

$$\frac{y_{j,min}(i)}{s_j^y} - k V_{j,min}^y(i) \leq \frac{y_j(k+i|k)}{s_j^y} \leq \frac{y_{j,max}(i)}{s_j^y} + k V_{j,max}^y(i), \quad i = 1:p, \quad j = 1:n_y \quad (74)$$

$$\frac{u_{j,min}(i)}{s_j^u} - k V_{j,min}^u(i) \leq \frac{u_j(k+i|k)}{s_j^u} \leq \frac{u_{j,max}(i)}{s_j^u} + k V_{j,max}^u(i), \quad i = 1:p, \quad j = 1:n_u \quad (75)$$

$$\frac{\Delta u_{j,min}(i)}{s_j^u} - k V_{j,min}^{\Delta u}(i) \leq \frac{\Delta u_j(k+i|k)}{s_j^u} \leq \frac{\Delta u_{j,max}(i)}{s_j^u} + k V_{j,max}^{\Delta u}(i), \quad i = 1:p, \quad j = 1:n_u \quad (76)$$

Aquí, los parámetros  $V$  (valores ECR) son constantes adimensionales de los controladores análogos a los pesos de la función de costo pero utilizados para la restricción de ablandamiento. También,

- $\epsilon_k$ - Escalar QP variable de holgura (sin dimensiones) que se utiliza para suavizar la restricción.
- $s_j^y$ -Factor de escala para la salida de la planta  $j$ , en unidades de ingeniería.
- $s_j^u$ -Factor de escala para  $j$  MV, en unidades de ingeniería.
- $y_{j, \min}(i)$ ,  $y_{j, \max}(i)$  - los límites superiores de producción de la planta  $j$  en el  $i$ -ésimo paso del horizonte de predicción, en unidades de ingeniería e inferior.
- $u_{j, \min}(i)$ ,  $u_{j, \max}(i)$  - los límites inferior y superior para  $j$  MV en la predicción  $i$ -ésimo paso horizonte, en unidades de ingeniería.
- $\Delta u_{j, \min}(i)$ ,  $\Delta u_{j, \max}(i)$  - los límites inferior y superior para  $j$  incremento MV en el  $i$ -ésimo paso del horizonte de predicción, en unidades de ingeniería.

Excepto para la condición de no negatividad variable de holgura, todas las restricciones anteriores son opcionales y se inactiva por defecto (es decir, inicializado con valores límite infinito). Para incluir una restricción de cota, debe especificar un límite finito cuando se diseña el controlador. (The MathWorks Inc., 2015)

#### d.3.2.4. Solucionador QP

El solucionador QP del controlador predictivo basado en modelo convierte un problema de optimización MPC a la forma general QP

$$\text{Min}(f^T x + \frac{1}{2} x^T H x) \quad (77)$$

de tal manera que

$$Ax \leq b \quad (78)$$

donde  $x^T = [z^T \ \varepsilon]$  son las decisiones,  $H$  es la matriz de Hesse,  $A$  es una matriz de coeficientes de restricción lineales, y  $b$  y  $f$  son vectores. Las matrices  $H$  y  $A$  son constantes. El controlador calcula éstos durante la inicialización y los recupera de la memoria del ordenador cuando sea necesario. Calcula los vectores  $b$  y  $f$  variables en el tiempo al comienzo de cada instante de control.

La Toolbox utiliza el algoritmo KWIK para resolver el problema QP, que requiere la Hesse ser definida positiva. En la primera etapa de control, KWIK utiliza un arranque en frío, en el que la estimación inicial es la solución sin restricciones se describe en el Modelo de Control Predictivo sin restricciones. Si esta  $x$  satisface las restricciones, es la solución óptima QP,  $x^*$ , y el algoritmo termina. De lo contrario, esto significa que al menos una de las restricciones de desigualdad lineales debe tener constancia de una igualdad. En este caso, KWIK utiliza una estrategia eficiente, robusto numéricamente para determinar el conjunto de restricciones activa que cumpla las condiciones de optimalidad estándar. En los siguientes pasos de control, KWIK utiliza un arranque en caliente. En este caso, el conjunto de restricciones activo determinado en la etapa de control anterior se convierte en la estimación inicial para el siguiente.

Aunque KWIK es robusto, se debe considerar lo siguiente:

- Uno o más restricciones lineales pueden ser violados ligeramente debido a errores de redondeo numéricos. La Toolbox emplea una relativa tolerancia no ajustable. Esta tolerancia permite que una restricción sea violada por 6.10 veces la magnitud de cada término. Tales violaciones se consideran normales y no generan mensajes de advertencia.
- La Toolbox también utiliza una tolerancia no ajustable cuando se pone a prueba una solución para el óptimo.
- La búsqueda del conjunto de restricciones activo es un proceso iterativo. Si las iteraciones alcanzan un máximo dependiente de problema, el algoritmo termina.
- Si el problema incluye restricciones duras, estas limitaciones podrían ser inviable (imposible de satisfacer). Si el algoritmo detecta inviabilidad, termina inmediatamente.

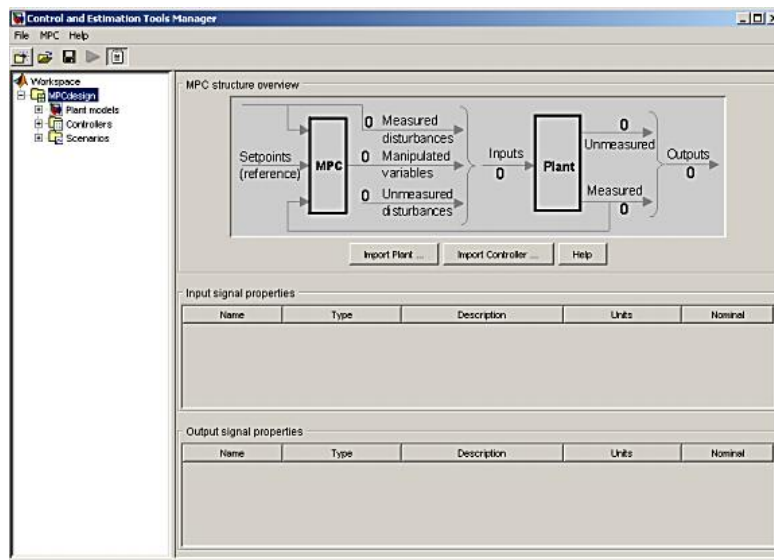
En los dos últimos casos, con un resultado anormal de la búsqueda, el controlador mantendrá la última salida de control exitoso. Puede detectar un resultado anormal y



reemplazar el comportamiento predeterminado como mejor le parezca. (The MathWorks Inc., 2015)

#### d.3.2.5. GUI mpctool

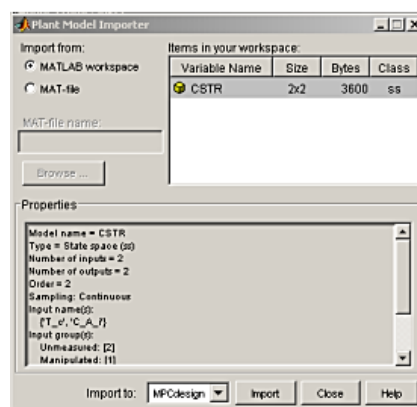
El primer paso en el diseño es cargar un modelo de planta. Sus dimensiones y señal características establecen el contexto para el resto de pasos.



**Figura 22.** Vista inicial MPC toolbox

**Fuente:** (The MathWorks, Inc, 2014)

Hacer clic en el botón Importar Planta en la vista inicial de la herramienta de diseño (Figura 22). Aparece el diálogo Importador Modelo Planta.



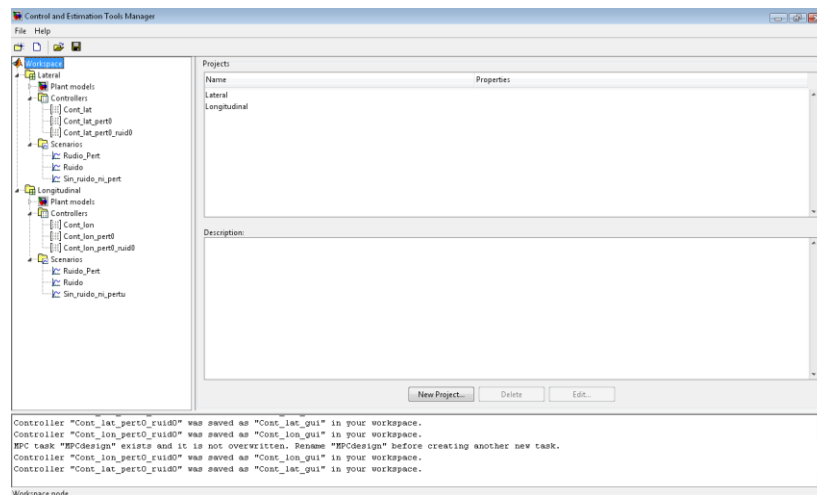
**Figura 23.** Diálogo Importador Modelo de Planta

**Fuente:** (The MathWorks, Inc, 2014)

La importación desde MATLAB del área de trabajo debe ser seleccionado por defecto, como se muestra. La sección del diálogo etiqueta artículos en su lista de espacio de trabajo.

Una vez que aparezca, se selecciona. Las propiedades de la sección de diálogos etiquetados después muestra el número de entradas y salidas, sus nombres y tipos de señal, etc.

Se hace clic en el botón Importar. Esto carga en la herramienta de diseño. Luego se hace clic en el Botón de cierre (de lo contrario el diálogo permanece visible en caso de que se desee importar otro modelo). La herramienta de diseño debería aparecer como en la Figura 23.



**Figura 24.** La GUI *mpctool*

**Fuente:** (The MathWorks, Inc, 2014)

La Figura 24 muestra la ventana de la interfaz gráfica de diseños de controladores predictivos de Matlab. Puede verse a la izquierda una estructura de árbol. La raíz de ese árbol es el *Workspace* de Matlab. Dentro del mismo hay diferentes ramas por cada MPC importado. Dentro de cada MPC hay tres entradas: (The MathWorks, Inc, 2014)

- Modelo, que especifica el modelo sobre el que se basa el controlador, incluyendo tiempo de muestreo.
- Controlador, que contiene todas las opciones de ajuste del mismo, como funciones de coste, restricciones, predicciones. Es la parte más importante.
- Escenarios, que permite realizar simulaciones con los MPC calculados.

Dentro de la sección de ajustes de los controladores hay varias pestañas

- *Model and horizon*: especifica los tiempos de muestreo y de predicción del controlador, así como el modelo al que está asociado.

Model and Horizons | Constraints | Weight Tuning | Estimation (Advanced)

Plant model: DCp

Horizons

Control interval (time units): 1.0

Prediction horizon (intervals): 15

Control horizon (intervals): 2

☐ Blocking

Blocking

Blocking allocation within prediction horizon: Beginning

Number of moves computed per step: 3

Custom move allocation vector: [2 3 5]

Help

**Figura 25.** Ventana general del controlador

**Fuente:** (The MathWorks, Inc, 2014)

Esta lista desplegable le permite especificar el modelo de la planta que el controlador utiliza para sus predicciones. Se puede elegir cualquiera de los modelos de planta que ha importado.

La opción **intervalo de Control** establece el tiempo transcurrido entre sucesivos movimientos del controlador. Debe ser un número real positivo, finito. Los cálculos asumen un asimiento de orden cero en las variables manipuladas (las señales ajustadas por el regulador). Por lo tanto, estas señales son constantes entre movimientos.

La opción de **horizonte de predicción** establece el número de intervalos de control sobre el cual el controlador predice sus salidas. Debe ser un entero positivo, finito.

La opción de **horizonte de Control** establece el número de movimientos calculados. Debe ser un entero positivo, finito y no debe superar el horizonte de predicción. Si es menor que el horizonte de predicción, el último movimiento calculado llena el resto del horizonte de predicción.

Por defecto, la opción de bloqueo está desactivada (*off*). Cuando se selecciona, la herramienta de diseño sustituye a la especificación horizonte de control con un patrón de movimiento. (The MathWorks, Inc, 2014)

- *Constraints*: se ajustan los valores máximos y mínimos tanto de los actuadores como de sus tasas de cambio. Asimismo se indican los valores extremos de la salida admisibles. Un elemento importante es el la opción *Softening constraints*, que permite relajar restricciones, tanto individualmente como de forma colectiva, evitando así llegar a sistemas irresolubles.

Constraints on manipulated variables					
Name	Units	Minimum	Maximum	Max Down Rate	Max Up Rate
Reflux Rate					
Steam Rate					

Constraints on output variables			
Name	Units	Minimum	Maximum
Distillate Purity			
Bottoms Purity			

**Figura 26.** Restricciones

**Fuente:** (The MathWorks, Inc, 2014)

Cada entrada de la tabla puede ser un escalar o un vector. Una entrada escalar define una restricción que es constante para todo el horizonte de predicción. A la entrada de vectores define una restricción variable en el tiempo.

Las columnas **nombre** y **unidades** son no editable. Para cambiarlos, utilizar la vista de la definición de señal. Las entradas restantes son editables. Si se deja en blanco un espacio, el controlador ignora esa restricción. Puede lograr el mismo efecto introduciendo *-Inf* o *Inf* (por un mínimo o máximo, respectivamente).

Los valores **mínimo** y **máximo** establecen rango de cada MV.

Los valores de restricción deben ser coherentes con los valores nominales. En otras palabras, el valor nominal de cada MV debe satisfacer las restricciones.

Los valores de restricción también deben ser uniformes. Por ejemplo, el límite inferior de una MV no debe exceder su límite superior. (The MathWorks, Inc, 2014)

### Ablandamiento de restricciones

Una restricción dura no puede ser violada. Las restricciones duras son arriesgadas, sobre todo para las salidas, ya que el controlador ignorará sus otros objetivos con el fin de satisfacerlas. Además, las restricciones podrían ser imposibles de satisfacer en ciertas situaciones, en cuyo caso los cálculos son matemáticamente inviable. (The MathWorks, Inc, 2014)

El Software Model Predictive Control Toolbox le permite especificar restricciones blandas. Estos pueden ser violados, pero se especifica una tolerancia para cada violación.

- *Weight Tuning*: permite ajustar los valores de ponderación de la función de coste. Una opción muy interesante es el ajuste de un coeficiente de compromiso entre sistemas de respuesta más rápida y controladores más robustos. El ajuste de este parámetro resulta clave en el caso de calcular controladores para sistemas no lineales con sistemas linealizados. De hecho, valores más bajos de ese valor (que teóricamente mejoran la robustez penalizando la respuesta del sistema), proporcionan respuestas más rápidas, ya que al encontrarse con las incertidumbres, el sistema más robusto es capaz de resolverlas más rápidamente. (The MathWorks, Inc, 2014)

Name	Description	Units	Weight	Rate Weight
da			0	0.1
dr			0	0.1

Name	Description	Units	Weight
phi			1
psi			1

**Figura 27.** Ajuste de pesos

**Fuente:** (The MathWorks, Inc, 2014)

- *Estimation*: permite establecer características avanzadas de predicción que tengan en cuenta tanto perturbaciones a la salida como ruidos. Estos ajustes pueden ser muy útiles

para modelar efectos del viento (primer caso) como de errores de los sensores (segundo caso). (The MathWorks, Inc, 2014)

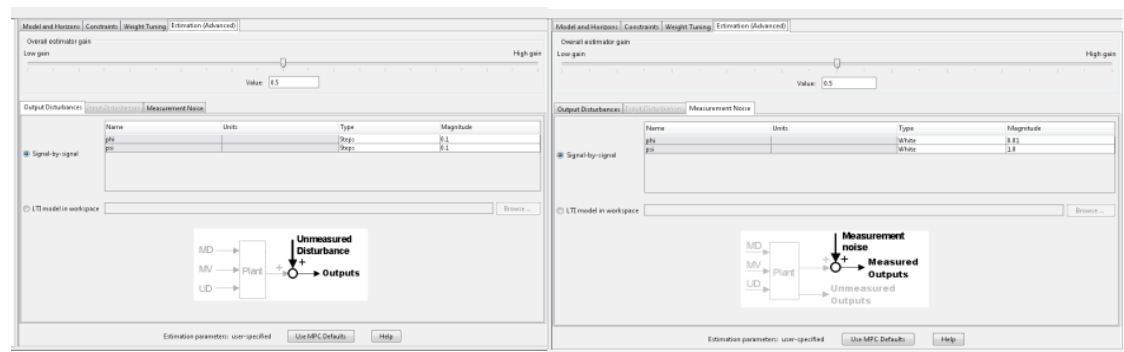


Figura 28. Predicciones

Fuente: (The MathWorks, Inc, 2014)

- *Simulation Scenario*: Esta vista aparece cuando se selecciona uno de los nodos de especificación del escenario. Permite especificar la configuración de la simulación y las variables independientes. Todos tienen valores predeterminados, pero se tendrá que cambiar por lo menos algunos de ellos (de lo contrario todas las variables independientes será constantes).

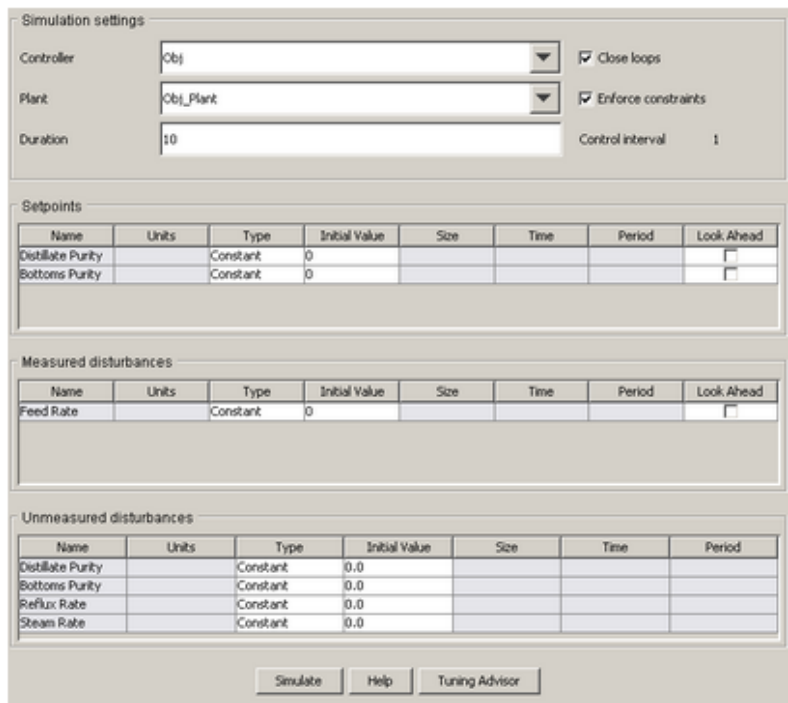


Figura 29. Escenario de Simulación

Fuente: (The MathWorks, Inc, 2014)

**Controlador** - Seleccionar uno de los controladores.

**Planta** - Seleccionar el modelo de la planta que actuará como la planta "real" en la simulación, es decir, no necesita ser el mismo que el utilizado para las predicciones del controlador.

**Duración** - La duración de simulación en unidades de tiempo.

**Cerrar bucles** - Si se desactiva, la simulación será de bucle abierto.

**Hacer cumplir las restricciones** - Si se desactiva, se ignorarán todas las limitaciones del controlador.

El campo **intervalo de control** es de sólo visualización, y refleja el ajuste en su selección Controller.

### **Puntos de ajuste**

Utilizar esta tabla para especificar el punto de referencia para cada salida.

Las columnas **Nombre** y **Unidades** son sólo de visualización.

La columna **Tipo** especifica la variación de consigna. Para cambiar esto, hacer clic en la celda y seleccionar una opción en el menú resultante.

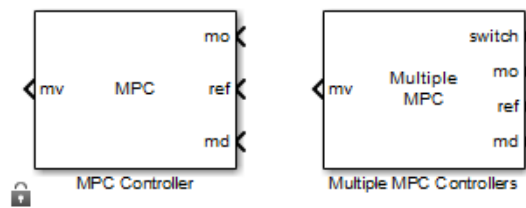
El significado de las columnas iniciales del **valor, tamaño, tiempo, y período** depende del tipo. Si una celda es gris (no editable), no se aplica al tipo elegido.

Si se selecciona la opción *Look Ahead*, el controlador utilizará los valores futuros de los puntos de ajuste en sus cálculos. Esto mejora el seguimiento de consigna, pero el conocimiento de los futuros cambios de consigna es inusual en la práctica. (The MathWorks, Inc, 2014)

### **d.3.2.6. Librería MPC de Simulink**

La librería MPC de Simulink ofrece dos bloques que se puede utilizar para implementar el control MPC en Simulink, MPC Controller, y Multiple MPC Controllers.

Acceder a la biblioteca utilizando el Navegador Simulink Library o escribiendo *mpcplib* en la ventana de comandos. La siguiente figura muestra el contenido de la biblioteca.



**Figura 30.** Librería MPC de Simulink

**Fuente:** (The MathWorks, Inc, 2014)

Una vez que se tenga acceso a la biblioteca, se puede agregar uno de sus bloques al modelo Simulink haciendo clic y arrastrando o copiar y pegar. (The MathWorks, Inc, 2014)

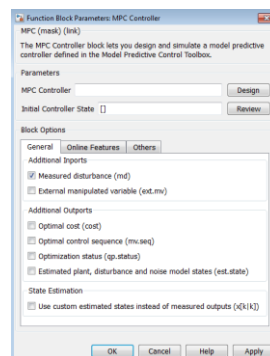
#### **d.3.2.6.1. Bloque Controlador MPC**

El bloque controlador MPC representa un solo controlador MPC. Puede ajustar la planta de entrada para el control de salidas de las plantas, lo que representa limitaciones, como los límites del actuador. (The MathWorks, Inc, 2014)

#### **Máscara Bloque Controlador MPC**

Insertar un bloque controlador MPC en su modelo de Simulink y especificar sus propiedades. Hacer doble clic en el bloque para abrir su máscara. La siguiente figura muestra la máscara ajustes por defecto. Esta sección muestra cómo configurar el controlador a través de: (The MathWorks, Inc, 2014)

- Especificación de los parámetros requeridos.
- Habilitación de puertos de entrada y salidas opcionales.
- Conexión de las señales apropiadas.



**Figura 31.** Máscara Bloque Controlador MPC

**Fuente:** (The MathWorks, Inc, 2014)



## Parámetros Controlador MPC

Especificar un objeto MPC válido en el campo Controlador MPC en una de dos maneras:

- Escribir el nombre de un objeto MPC guardado en el espacio de trabajo. Para revisar su configuración antes de ejecutar una simulación, hacer clic en Diseño para cargar el objeto con nombre en la herramienta de diseño MPC.
- Si la instalación incluye el software Simulink Control Design <sup>TM</sup>, conectar el Bloque Controlador MPC de la planta que controla. Si se deja el campo controlador MPC vacía, hacer clic en Diseño. El bloque construye un objeto predeterminado MPC linealizando la planta en el punto de funcionamiento por defecto y exporta este controlador por defecto al espacio de trabajo de base. Si el punto de funcionamiento por defecto no es el que se desea, consultar "Importación de un modelo de planta" para ayudar con la importación de un nuevo modelo de planta.

## Estado inicial del Controlador

Si no se especifica un estado inicial del controlador, el controlador utiliza una condición inicial por defecto en simulaciones. Para cambiar el estado inicial, especificar un objeto `mpcstate`. (The MathWorks, Inc, 2014)

## Señales

El controlador MPC requiere al menos dos entradas y genera al menos una salida. También hay entradas y salidas opcionales. En la mayoría de los casos, las señales son vectores compuestos de variables de la planta. Comprobar que las dimensiones de señal y la secuencia de elementos dentro de cada señal vector son consistentes con su definición de controlador. (The MathWorks, Inc, 2014)

## Entradas Requeridas

Conectar lo siguiente para los puertos de entrada del controlador como se indican:

- Variables de salida medidos (*mo*). Conecte las variables de salida de la planta medidos al puerto *mo* del controlador MPC como señal vector, longitud  $n_{ym} \geq 1$ . Esto proporciona la retroalimentación del controlador.
- Referencias (*ref*). El modelo de predicción de su controlador contiene variables de salida  $n_y \geq n_{ym}$ . Cada uno debe tener un objetivo de referencia o el valor de consigna. Conecte este vector 1-por- $n_y$  al puerto *ref* del controlador. Este puerto define los valores del primer paso de referencia en el horizonte de

predicción del controlador. Por defecto, el controlador asume estos valores válidos para todo el horizonte.

### **Puerto Requerido**

Durante la operación, el controlador actualiza el puerto de salida variable manipulada (mv) en cada intervalo de control. El puerto de salida mv define ajustes para plantar variables de entrada. Conectar este vector de señal a la planta. (The MathWorks, Inc, 2014)

### **Señales de entrada**

Se debe conectar señales de Simulink adecuadas a los puertos de entrada del bloque controlador MPC. La salida medida (mo) y la referencia (ref) se requieren. Se puede agregar puertos opcionales mediante la selección de casillas de verificación en la parte inferior de la máscara.

Como se muestra en la figura 31, habilitar perturbaciones medidas es una selección por defecto y el puerto correspondiente (md) aparece en la figura MPC Simulink de la Biblioteca. Esto proporciona una compensación de alimentación hacia delante para perturbaciones medidas. (The MathWorks, Inc, 2014)

### **Señales de salida**

El bloque actualiza su salida (s) de forma periódica. El objeto MPC nombrado en el bloque de campo controlador MPC contiene el intervalo de control (su propiedad Ts). El objeto también especifica el número de variables manipuladas,  $n_u$ , que se calcula y se envía a la planta en cada instante de control. El puerto de salida predeterminado (mv marcado) ofrece estos como una señal vector de  $n_u$  dimensión. (The MathWorks, Inc, 2014)

## e.- MATERIALES Y MÉTODOS

### e.1. Materiales y Métodos

La metodología seguida para lograr los objetivos propuestos fue realizar la descripción del proceso que realiza la planta y determinar cuáles son los elementos que la constituyen.

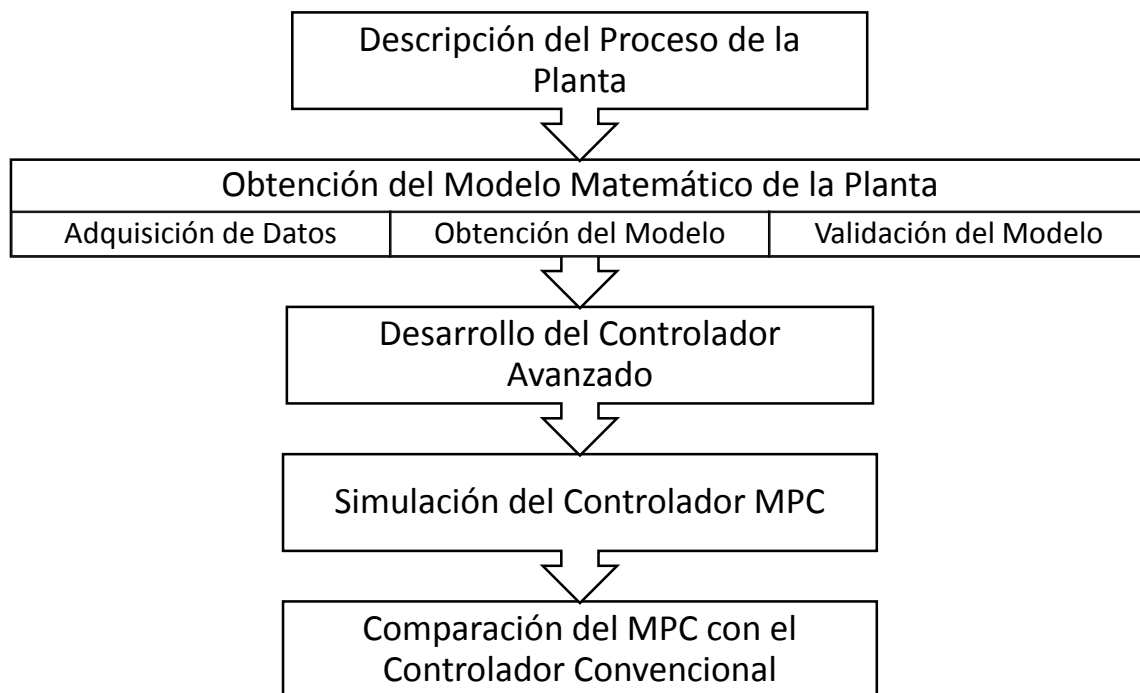
Luego se realizó la obtención del modelo matemático de la planta, para esto primero fue necesario la adquisición de datos haciendo uso del VI de Labview diseñado para el efecto y del equipo FieldPoint 2010, después se obtuvo la función de transferencia de la planta mediante la herramienta de Identificación de Sistemas *ident* de Matlab, por último se realizó la validación del modelo que representa mejor el proceso de la planta.

Al contar con el modelo de la planta se desarrolló el Controlador Avanzado utilizando la herramienta *mpctool* de Matlab que permitió el desarrollo del Controlador Predictivo Basado en Modelo (MPC).

Posteriormente se efectuó la simulación del Controlador MPC en Simulink.

Finalmente se hizo la comparación de desempeño del Controlador MPC diseñado en relación con el Controlador Convencional PI implementado en el Banco de Control de Presión.

El siguiente esquema detalla lo antes mencionado:



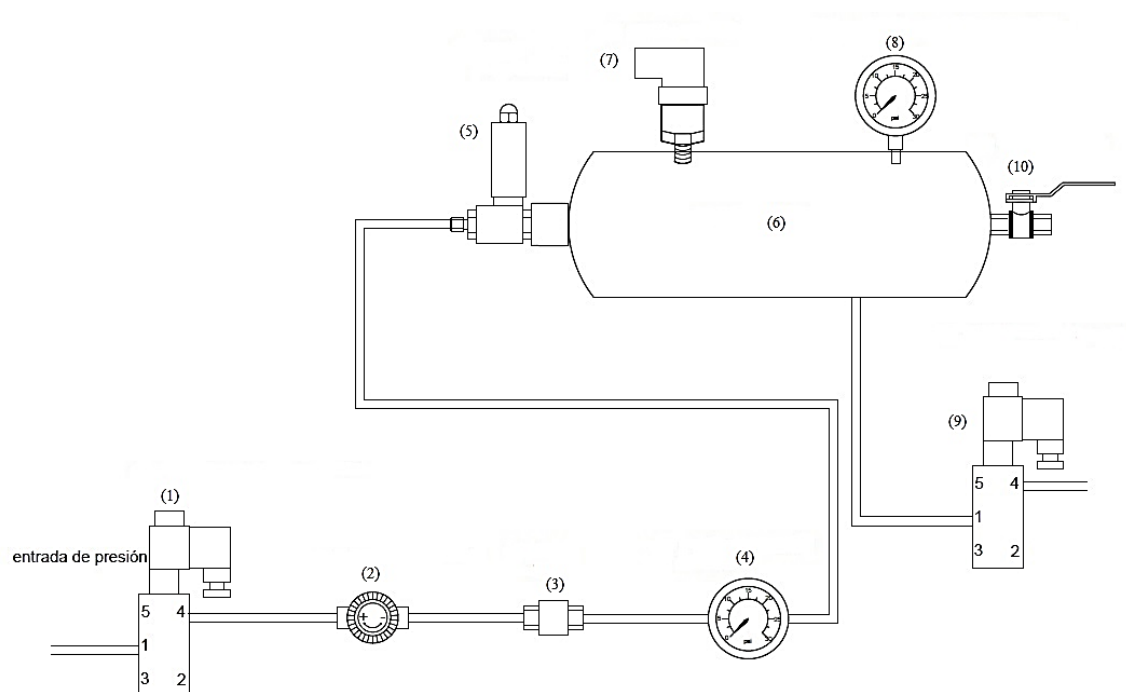
## f. RESULTADOS

### f.1. Descripción del proceso de la planta

La primera fase fue el reconocimiento del banco para la medición y control de presión del Laboratorio de Automatización de la U.N.L. y se determinó de qué elementos está constituido para su funcionamiento y cuál es el proceso que realiza.

Los elementos que componen el banco son los siguientes:

- (1) Electroválvula de accionamiento
- (2) Regulador de presión
- (3) Válvula anti retorno
- (4) Manómetro de presión de entrada
- (5) Válvula proporcional
- (6) Acumulador
- (7) Sensor de presión
- (8) Manómetro de presión del acumulador
- (9) Electroválvula de purga
- (10) Válvula manual



**Figura 32.** Esquema banco de medición de presión

**Fuente:** (Tandazo, 2014)

El aire comprimido ingresa por la electroválvula de accionamiento (1), pasa por el regulador de presión (2), el cual regula la presión de entrada al acumulador (6) y no sobrepase los 15 PSI, figuran también una válvula anti retorno (3) y dos manómetros de presión (4) y (8) los cuales permiten visualizar la presión en la tubería y en el acumulador respectivamente, la válvula proporcional (5) permite variar la cantidad de aire en el tanque acumulador, la válvula manual (10) permite simular perturbaciones, la electroválvula de purga (9) permite evacuar el aire contenido en el tanque en caso de un exceso de presión.

El sistema consta de un FieldPoint 2010 al cual se debe conectar una computadora a través de una comunicación Ethernet para lo que se utiliza un cable UTP cat. 6 cruzado.

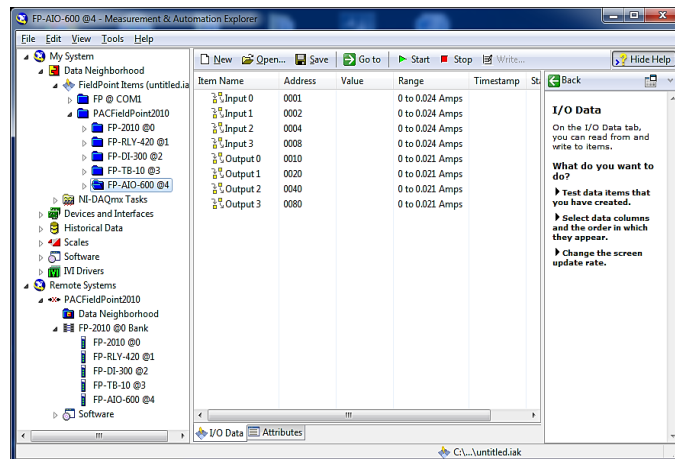
Lo que se quiere controlar es que la presión en el acumulador sea constante, para esto se asigna el valor de presión que se desea mantener en el acumulador que debido al consumo del aire puede variar, por lo que dependiendo de este valor establecido la válvula proporcional se debe abrir o cerrar para conservar la presión dentro del acumulador en el nivel asignado.

El experimento que se realizó fue conectar el FieldPoint 2010 con la computadora para a través del software manipular la válvula proporcional del banco, permitiendo el paso del aire comprimido hasta el tanque acumulador aplicándole una señal de escalón desde el estado cerrado hasta el estado completamente abierto de la válvula proporcional. Y de esta manera se adquirió los datos proporcionados por el sensor de presión ubicado en el tanque acumulador que luego fueron utilizados en la obtención del modelo de la planta.

## **f.2. Obtención del modelo matemático de la planta**

### **f.2.1. Adquisición de datos**

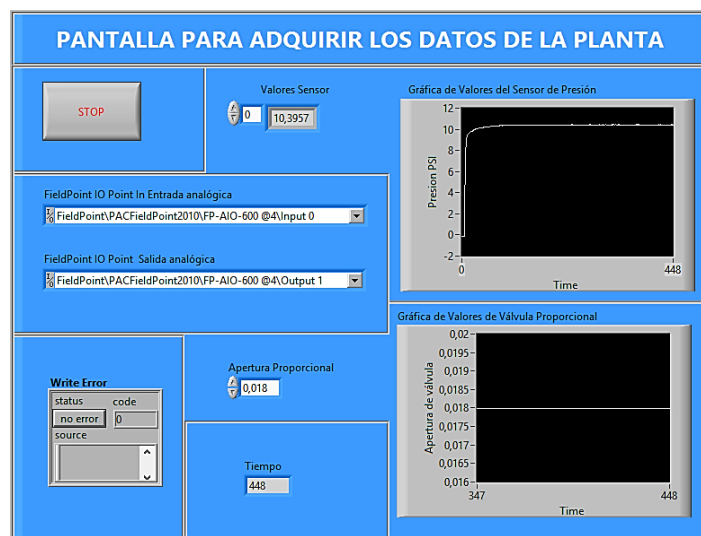
Para la adquisición de datos se inició con lograr conexión entre el FieldPoint2010 y la PC a través del programa *Measurement & Automation Explorer* (MAX) siendo indispensable la instalación de controladores NIDAQ1410f1 y nifpsw1310 los cuales sirven para adquisición de datos y acondicionamiento de señal de dispositivos y para el uso de FieldPoint con entornos de aplicaciones respectivamente. Establecida esta comunicación se realizó la configuración del *hardware* y *software* de *National Instruments* que se utilizaron para la obtención de datos, que consistió en asignar un rango de acción a los canales de entrada y de salida del equipo FieldPoint2010 utilizados.



**Figura 33.** Interfaz de MAX

**Fuente:** (El Autor)

Terminada la configuración se procedió a utilizar un VI (Instrumento Virtual) de LabView diseñado para la adquisición de datos de la planta, aquí se asignó los canales del FieldPoint 2010 que fueron utilizados tanto para la entrada como para la salida y el valor de la válvula proporcional, como se muestra en la figura 34. Este VI por medio de un *write to measurement* recoge los datos en un archivo para luego utilizarlos en la obtención de la función de transferencia de la planta, además se puede observar la gráfica descrita debido a los valores que el sensor de presión entregó durante el periodo de prueba conjuntamente con la gráfica generada por los valores de apertura de la válvula proporcional al aplicar la señal escalón.

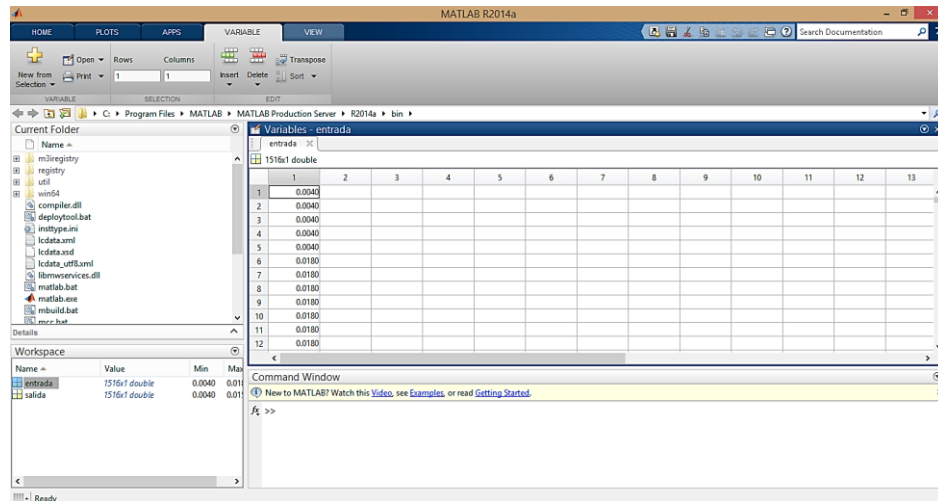


**Figura 34.** Pantalla de Adquisición de Datos

**Fuente:** (El Autor)

### f.2.2. Obtención del Modelo Matemático

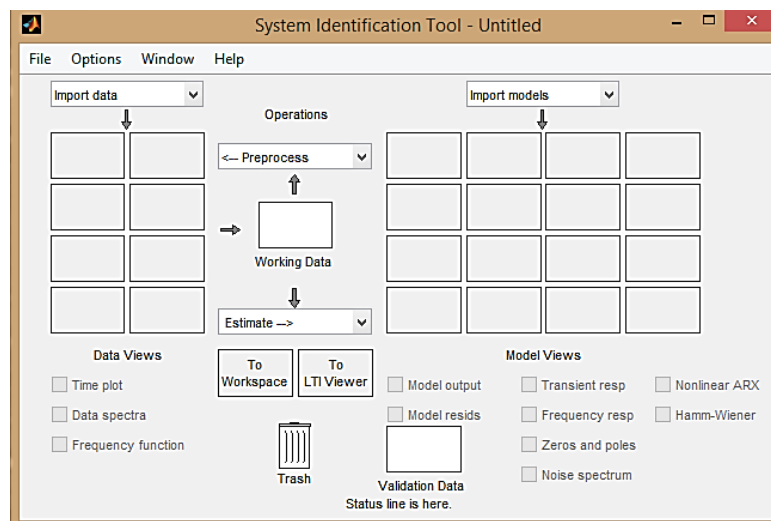
Con los datos conseguidos de la planta corresponde obtener la función de transferencia, para esto se utilizó el programa Matlab, se inició creando dos nuevas variables, entrada y salida, en el *workspace* con los datos que se obtuvieron de la planta.



**Figura 35.** Creación de variables de entrada y salida

**Fuente:** (El Autor)

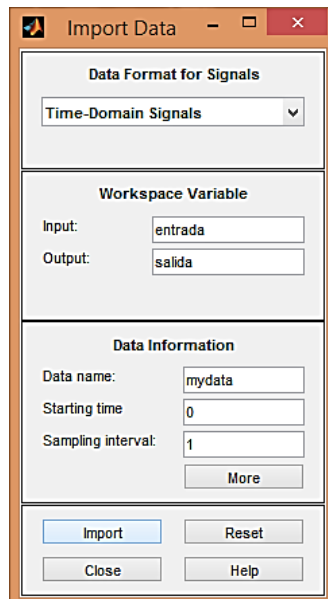
El siguiente paso fue utilizar la herramienta *ident* de Matlab, para ello se escribió en la ventana de comandos la palabra *ident* y apareció la interfaz de identificación de sistemas, una vez aquí se importó los datos para ello se abre la pestaña *import data* y se selecciona la opción *Time domain data*, desplegándose una nueva ventana llamada *import data*.



**Figura 36.** Interfaz de Identificación de Sistemas

**Fuente:** (El Autor)

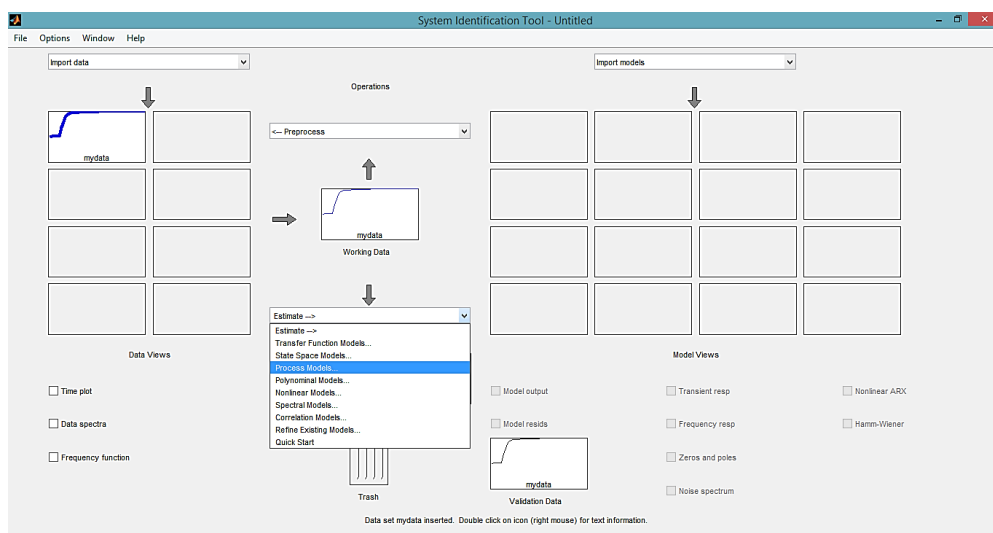
En esta ventana figura 37, se colocó las variables creadas anteriormente en el *workspace* denominadas entrada y salida, y se presiona el boton *import*.



**Figura 37.** Importar Datos

**Fuente:** (El Autor)

A continuación se desplegó nuevamente la ventana de identificación de sistemas pero ahora ya con los datos ingresados y se abrió la pestaña *estimate* y se eligió la opción *Process model* como se muestra en la figura 38.

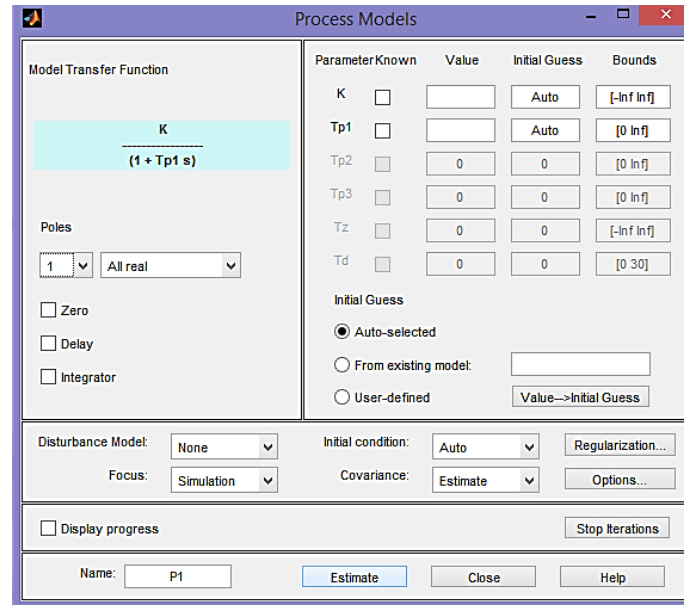


**Figura 38.** Estimación de datos

**Fuente:** (El Autor)



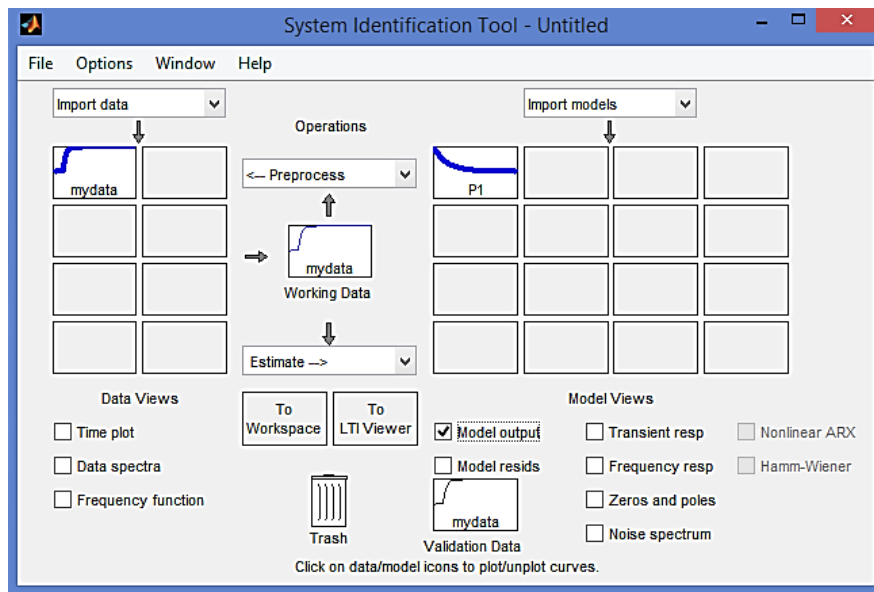
La ventana *Process Models* aparece y en ésta se pudo estimar algunos modelos de función de transferencia variando el número de polos o eligiendo utilizar las opciones *zero*, *delay* o *integrator*, y luego se presionó *estimate*.



**Figura 39.** Modelos de proceso

**Fuente:** (El Autor)

Finalmente apareció la ventana de identificación de sistemas pero esta vez con un modelo obtenido que se pudo observar eligiendo la opción *Model output*.



**Figura 40.** Modelos obtenidos

**Fuente:** (El Autor)

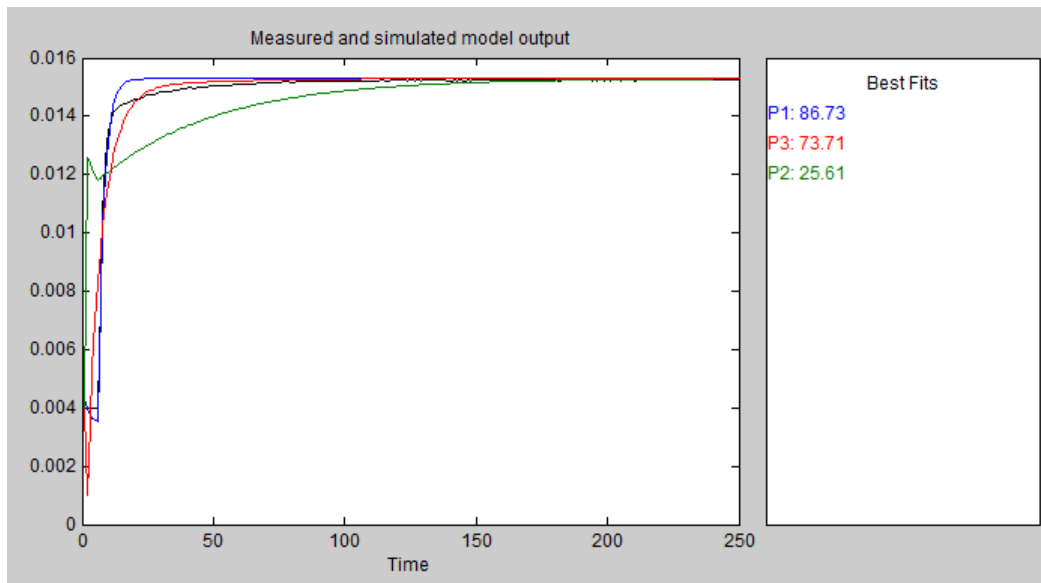
### f.2.3. Validación del Modelo

Tras obtener varios modelos se mostraron las gráficas de los resultados, estos modelos son alimentados con la aportación del conjunto de datos cuya salida se representa en color negro, luego se verificó cuál de todos ellos representa mejor el proceso de la planta mediante el porcentaje de ajuste que es calculado por la herramienta *ident* utilizando la siguiente ecuación:

$$Best\ Fit = \left(1 - \frac{|y - \hat{y}|}{|y - \bar{y}|}\right) \times 100 \quad (79)$$

Donde  $y$  es la salida medida,  $\hat{y}$  es el resultado del modelo simulado,  $\bar{y}$  es la media de  $y$ .

Los gráficos muestran los resultados simulados de los modelos seleccionados, un número más alto de porcentaje muestra un mejor modelo, por lo que para éste caso el modelo válido fue una función de primer orden que presenta un porcentaje de ajuste del 86,73%.



**Figura 41.** Simulación salida de modelos

**Fuente:** (El Autor)

Por último se presentó la ecuación de la función de transferencia que luego fue utilizada para realizar el sistema de control avanzado.

$$G(s) = \frac{0.8492}{2.3378s+1} \quad (80)$$

Posteriormente se realizó un análisis de estabilidad, para ello se aplicó a la función de transferencia obtenida un análisis de polos y ceros, que consiste en encontrar la raíces de

la ecuación característica, los ceros corresponden al polinomio del numerador de la función que para este caso no hay mientras que los polos corresponden al denominador de la función para lo que de la expresión 80 se tomó el denominador y se realizó lo siguiente:

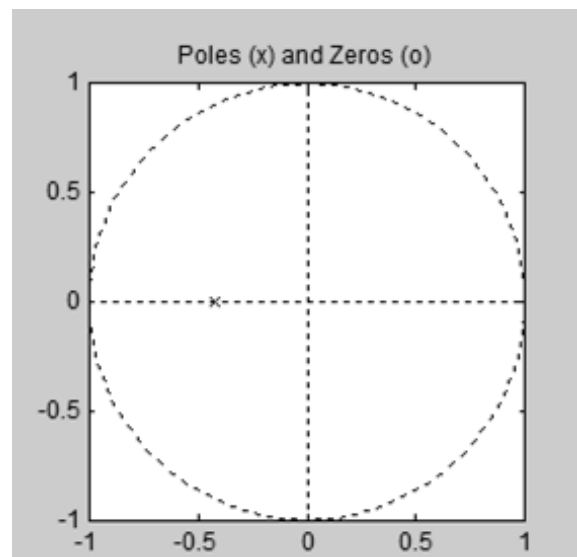
$$2,3378s + 1 = 0$$

$$s = -\frac{1}{2,3378}$$

$$s = -0,4277$$

El resultado mostró (figura 43) que la planta posee un polo en el semiplano izquierdo del diagrama por lo que es estable.

Con el índice de ajuste y el análisis de estabilidad realizado se determinó que el modelo obtenido es apropiado para el desarrollo del Controlador MPC.



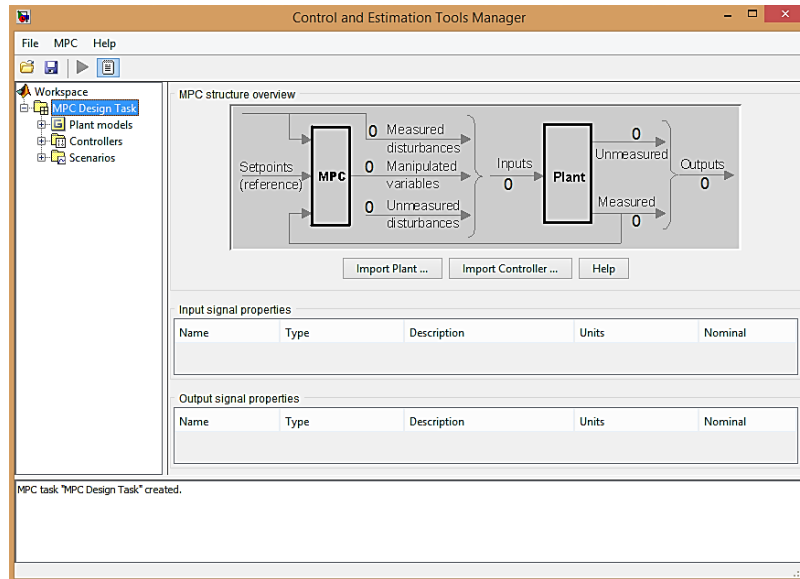
**Figura 42.** Análisis de Polos y Ceros

**Fuente:** (El Autor)

### **f.3. Desarrollo del controlador avanzado**

Para desarrollar el controlador avanzado se utilizó la herramienta *mpctool* de Matlab que permitió el desarrollo de un Controlador Predictivo Basado en Modelo (MPC).

Se inició escribiendo *mpctool* en la ventana de comandos que abre una interfaz que se observa en la figura 43, a la izquierda están tres elementos a completar que son el modelo de la planta, controlador y escenario que sería la configuración de la simulación.



**Figura 43.** Interfaz de diseño de MPC

**Fuente:** (El Autor)

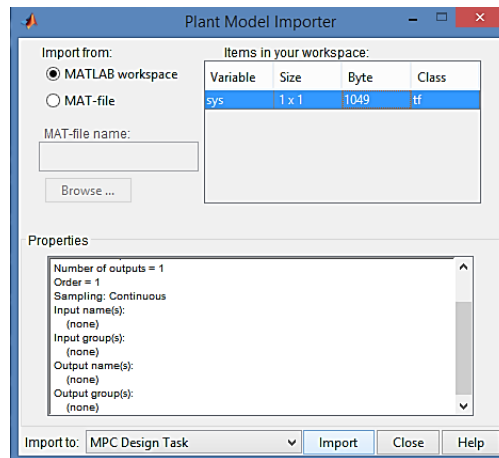
Lo primero que se debió hacer es definir un proceso en Matlab, que no es otra cosa que la función de transferencia obtenida anteriormente, esta sirve como planta para el controlador predictivo.

**Figura 44.** Definición del proceso

**Fuente:** (El Autor)

Una vez hecho esto se regresó al entorno del controlador predictivo (figura 43), y se seleccionó la opción *Import Plant* y apareció una nueva ventana en donde estaba

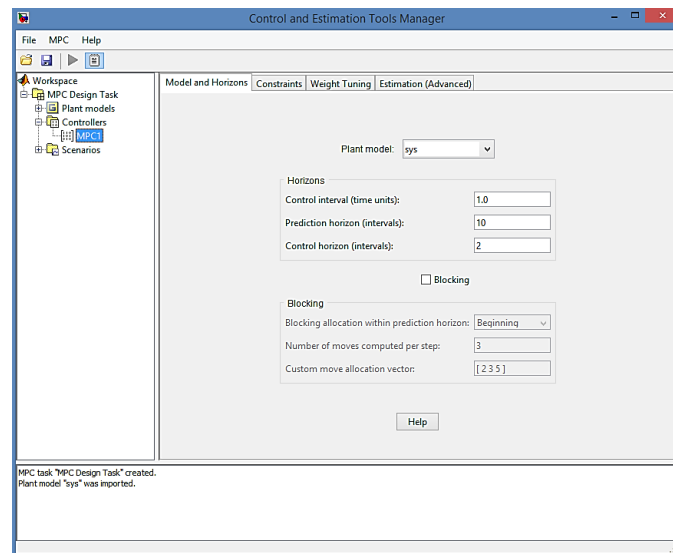
disponible la variable antes creada y lo que se hizo fue seleccionarla y oprimir *import* contando ya con una planta definida.



**Figura 45.** Importar modelo de planta

**Fuente:** (El Autor)

Lo que se debió hacer luego es ir a la parte del controlador que por defecto se denomina MPC1, en donde hay varias pestañas que se fue configurando, en la primera pestaña se definió el sistema y el intervalo de control en unidades de tiempo.



**Figura 46.** Definición de horizontes de control

**Fuente:** (El Autor)

Después se fue a la pestaña *constraints* en donde se configuró las restricciones a la entrada y a la salida del controlador, que en este caso serían la cantidad de corriente a los

que la válvula proporcional está cerrada (0.004 A) y completamente abierta (0.018 A). Con esto se concluyó la configuración del controlador.

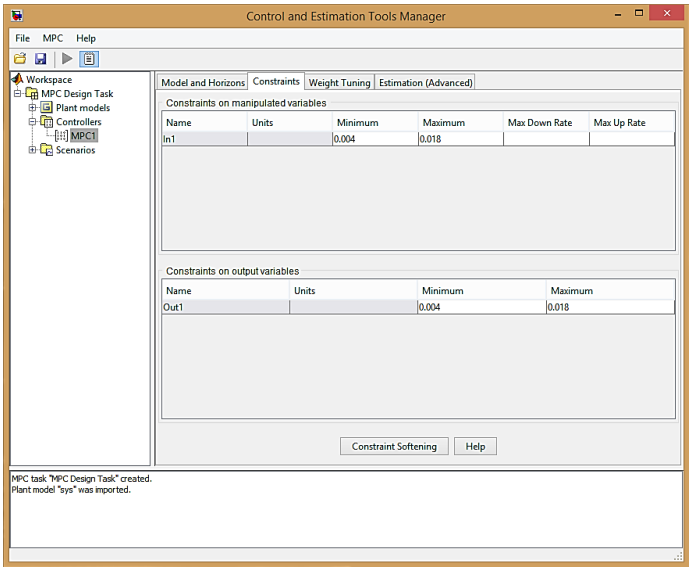


Figura 47. Restricciones del MPC

Fuente: (El Autor)

Ahora lo que se definió es el escenario, en donde se fue seleccionando elementos como el controlador MPC1 que se configuró antes, la planta que se está utilizando, la duración del tiempo de simulación, además se indicó que se usen las restricciones y por último la consigna que en este caso es la cantidad de corriente (0.015 A) a la que la válvula trabajará y logrará estabilidad estando completamente abierta.

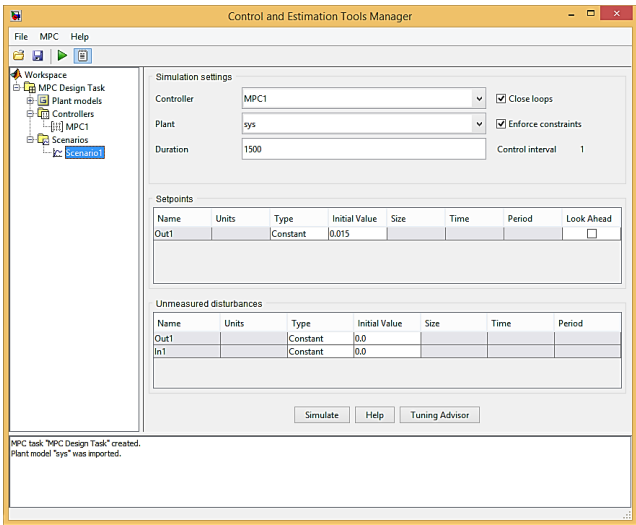
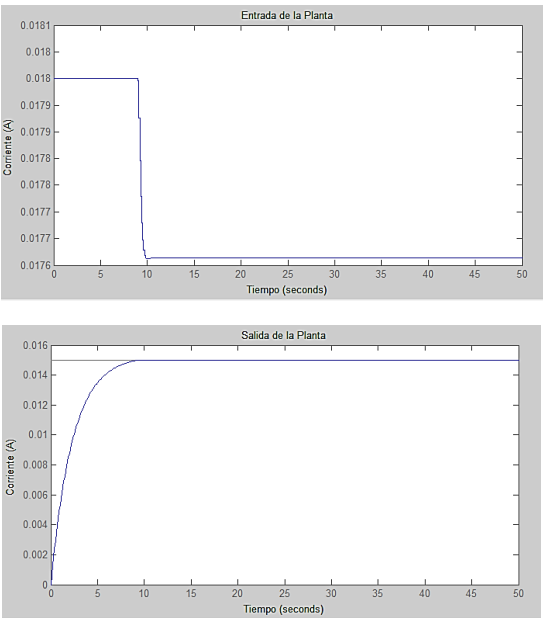


Figura 48. Simulación de Escenarios del MPC

Fuente: (El Autor)

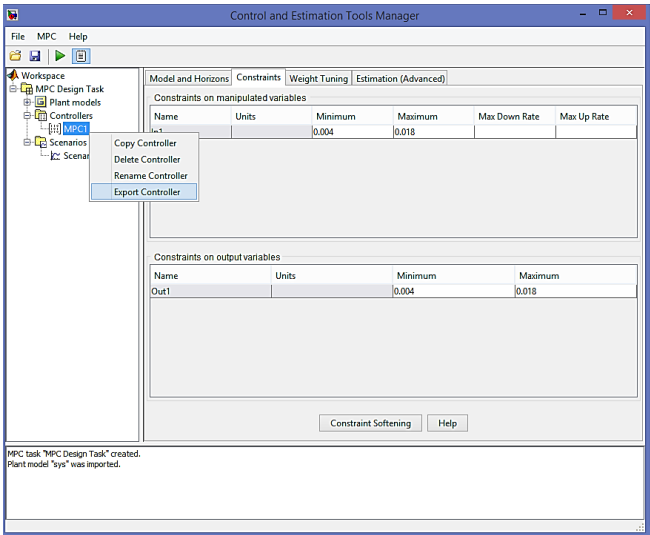
Posteriormente se presionó *simulate* y lo que se mostró como resultado fue la gráfica de la evolución tanto de la entrada como de la salida del controlador elaborado.



**Figura 49.** Gráfica de la entrada y salida del MPC

**Fuente:** (El Autor)

Finalmente se exportó el controlador desarrollado al *workspace* de Matlab para utilizarlo luego en la simulación.



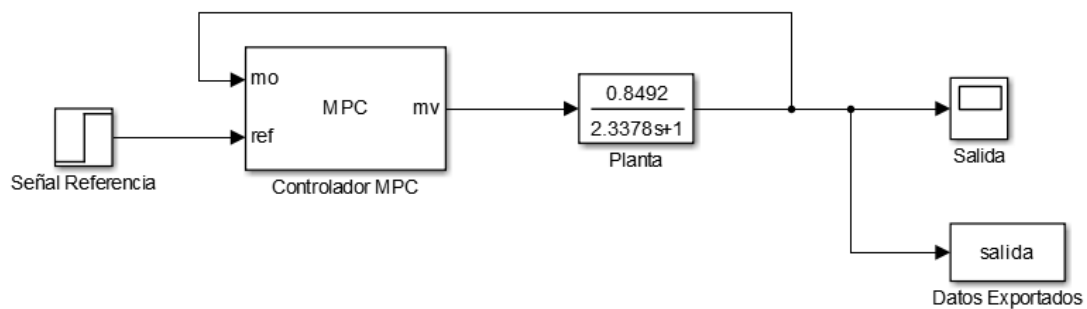
**Figura 50.** Exportar controlador MPC

**Fuente:** (El Autor)

#### f.4. Simulación del MPC

En lugar de utilizar la simulación con el entorno *mpctool* se puede plantear la simulación en Simulink y para ello existe una librería de control predictivo.

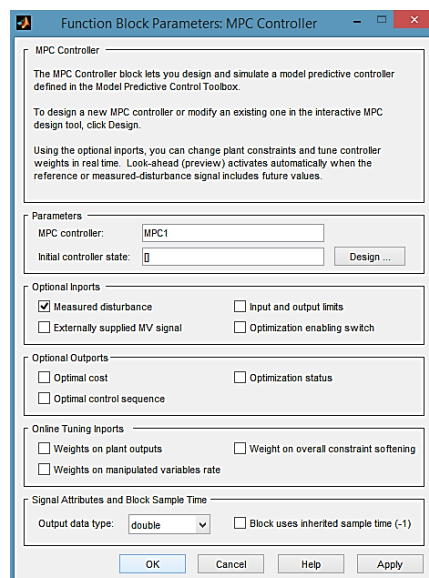
Los bloques que se utilizaron para este diagrama son un bloque *Step*, controlador MPC, bloque de función de transferencia y un *scope* y se los dispuso de la manera indicada en la figura 51.



**Figura 51.** Diagrama de bloques del controlador

**Fuente:** (El Autor)

Posteriormente se parametrizó el controlador MPC que iba a ser utilizado para ello con doble clic sobre el bloque Controlador MPC se desplegó la ventana donde se definió la variable MPC1 que es el controlador desarrollado anteriormente.

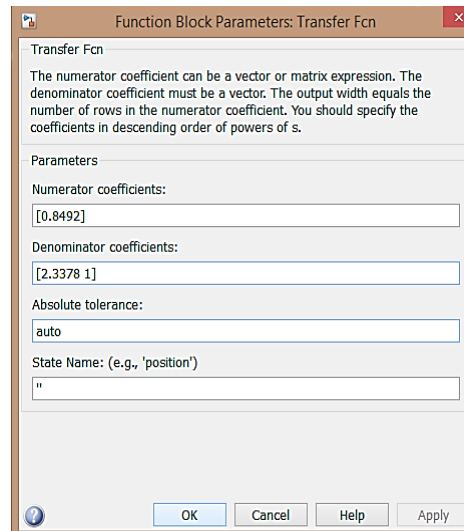


**Figura 52.** Parámetros del controlador MPC

**Fuente:** (El Autor)



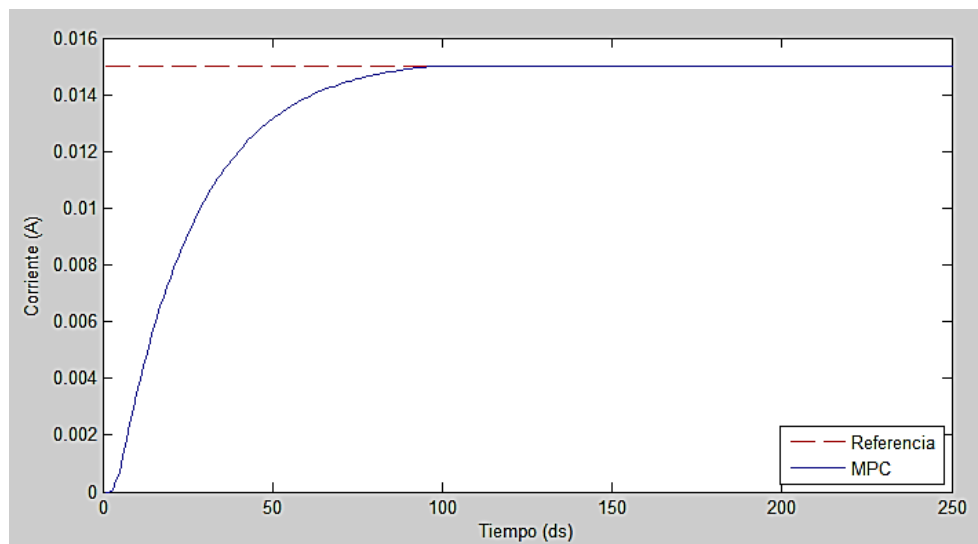
Luego se colocó el proceso, para esto se implementó un bloque función de transferencia donde se tuvo que colocar los coeficientes del numerador y del denominador correspondiente a la función de transferencia del modelo matemático obtenido, tal como se indica en la figura 53.



**Figura 53.** Parámetros bloque función de transferencia

**Fuente:** (El Autor)

El bloque *Step* genera un escalón entre dos niveles establecidos que en este caso son de 0 a 0.015 y el bloque *Scope* representa gráficamente el resultado del controlador sobre el proceso.



**Figura 54.** Resultados del controlador MPC

**Fuente:** (El Autor)

En la Figura 55 se puede observar como el controlador MPC realizó una respuesta amortiguada hasta la señal de referencia y se estabilizó de forma casi inmediata, presentando las características de desempeño mostradas en la Tabla 5.

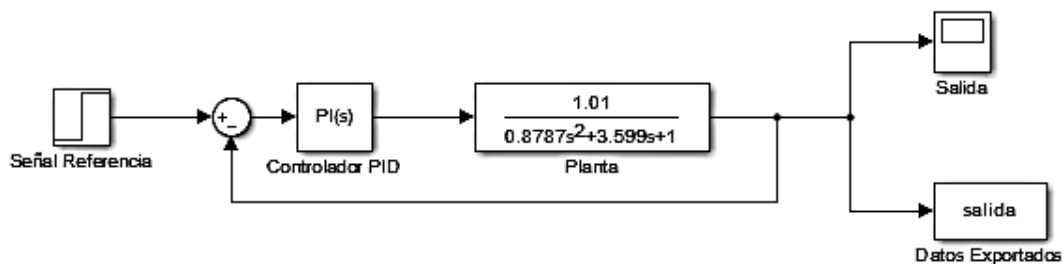
**Tabla 5.** Características de desempeño del Controlador MPC

Tiempo de subida (s)	Tiempo de estabilización (s)	Overshoot (%)
4,77	8,02	0

Fuente: (El Autor)

### f.5. Comparación del MPC con el controlador convencional

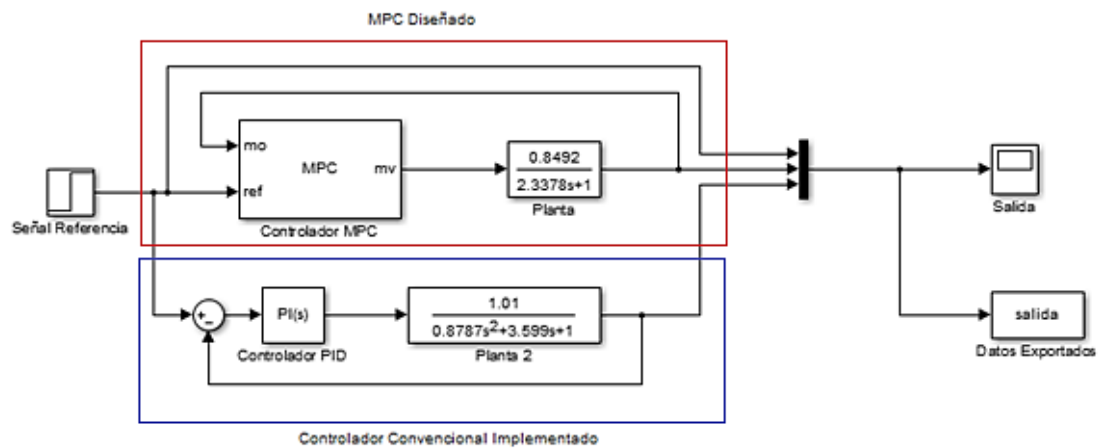
Para comparar el controlador avanzado desarrollado que en este caso es un Controlador Predictivo Basado en Modelo (MPC) con el controlador convencional implementado para controlar la presión en este banco, el cual es un controlador PI, se realizó una simulación del proceso en Simulink para lo que se configuró un bloque PID y se utilizó la función de transferencia que posee el banco.



**Figura 55.** Diagrama de bloque controlador PI implementado en el banco

Fuente: (El Autor)

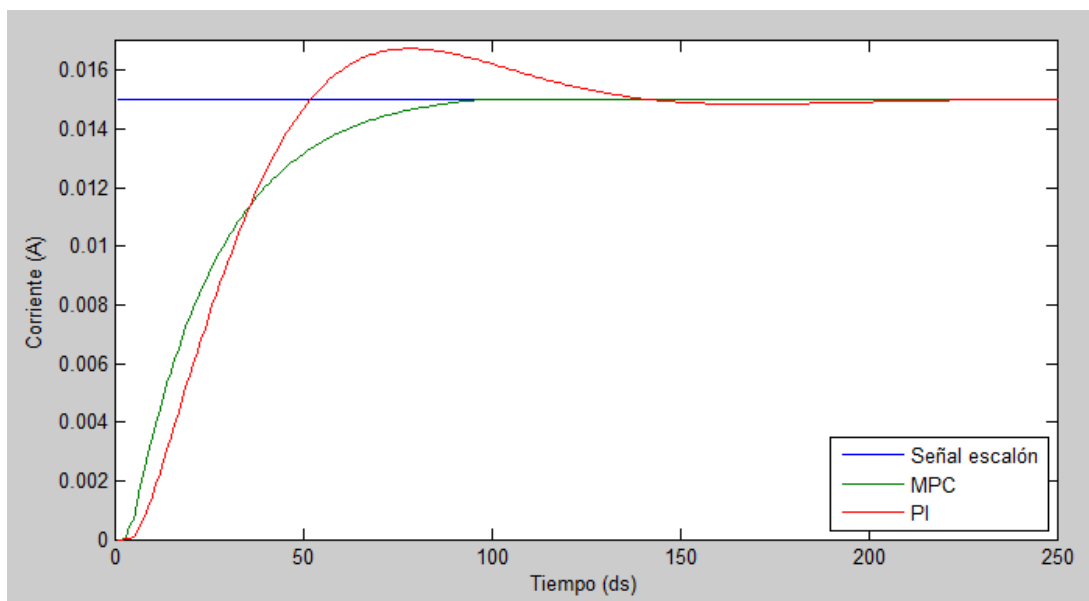
Luego se realizó un diagrama en conjunto de los dos controladores, se utilizó una señal de referencia, para este caso una señal de escalón cuyo valor máximo es de 0.015 y en el que la válvula proporcional estará completamente abierta, luego está el controlador MPC y la función de transferencia de la planta correspondiente y por debajo está el controlador PI con la planta que se ha utilizado para implementarlo y finalmente el bloque *Scope* en el que se mostraran las gráficas de la señal de salida de los dos controladores además de la señal de referencia a la que se quería estabilizar el proceso, pudiendo así observar cuál de los dos controladores muestra mejores características de desempeño.



**Figura 56.** Diagrama de bloques controladores MPC y PI

**Fuente:** (El Autor)

En la comparación de desempeño del sistema diseñado versus el sistema que actualmente está implementado, se pudo observar que el controlador MPC ofrece mejores características para el proceso debido a que tiene una respuesta más suave y logra estabilizarse en tiempo más corto que el controlador PI como se puede observar en la gráfica y en los datos mostrados en la Tabla 6.



**Figura 57.** Comparación de modelos

**Fuente:** (El Autor)

**Tabla 6.** Características de desempeño de los controladores

<b>Controlador</b>	<b>Tiempo de subida (s)</b>	<b>Tiempo de estabilización (s)</b>	<b>Overshoot (%)</b>	<b>Pico (A)</b>	<b>Tiempo de pico (s)</b>
<b>MPC</b>	4,77	8,02	0	0,015	10,3
<b>PI</b>	3,39	12,7	11,6	0,0167	7,8

Fuente: (El Autor)

## **g.- DISCUSIÓN**

Se planteó desarrollar para el Banco de Control de Presión un controlador avanzado debido a la dinámica compleja que posee el proceso y luego realizar un análisis comparativo y determinar las ventajas de este controlador con respecto al implementado.

Se efectuó el procedimiento de identificación de sistemas para obtener el modelo matemático del proceso, para esto se realizó pruebas de recolección de datos utilizando un sistema de adquisición de datos implementado con LabView y un equipo FieldPoint 2010, con estos datos se obtuvo el modelo matemático mediante la herramienta *ident* de Matlab porque proporciona varias estimaciones de modelos de los cuales se validó el que mejor se ajustó a los datos de la planta.

Al poder obtener el modelo matemático del proceso fue posible el desarrollo del controlador avanzado MPC utilizando la herramienta *mpctool* de Matlab, para lo que se definió la planta y las restricciones del proceso, luego se realizó la simulación en Simulink obteniendo así los resultados del controlador que se muestra de manera gráfica.

Al contar con el controlador MPC se realizó el análisis comparativo con el controlador convencional implementado en el banco, para esto se efectuó la simulación de los controladores utilizando Simulink, obteniendo los resultados y comprobando que el controlador avanzado presenta mejores características de desempeño al proceso.

## **h.- CONCLUSIONES**

- Se obtuvo el modelo matemático de la dinámica del proceso el mismo que resultó como una función de transferencia de primer orden estable, presentando un índice de ajuste del 86.73% con relación a los datos reales obtenidos de la planta.
- Se diseñó un control avanzado para el sistema, específicamente un Controlador Predictivo Basado en Modelo (MPC), para lo que se utilizó como planta el modelo matemático obtenido, se usó la herramienta *mpctool* de Matlab para desarrollar este controlador y se implementó en Simulink para comprobar su desempeño mediante simulación, obteniendo un tiempo de subida de 4,77 s y un tiempo de estabilización de 8,02 s.
- Al comparar el controlador diseñado con el ya existente en el Banco se obtuvo mejores resultados con el controlador MPC, que aunque es 1,38 s más lento en llegar al nivel de referencia, logra estabilizarse 4,68 s antes que el controlador PI, además el controlador MPC no presenta overshoot, por lo que será una mejor opción para el funcionamiento del banco de control de presión.

## **i.- RECOMENDACIONES**

- Es posible la realización de un nuevo experimento utilizando una señal de secuencias binarias pseudoaleatorias (señales de unos y ceros que dependen de una función pseudoaleatoria), lo cual puede permitir obtener un modelo más preciso.
- Lo realizado en este trabajo es el estudio del comportamiento del controlador MPC y se ha comparado con el controlador PID, se podría realizar la implementación del controlador MPC para observar su desempeño en la práctica.
- El estudio de los controladores predictivos MPC es escaso se recomienda se siga impulsando es estudio de este tipo de controladores por ejemplo para ser implementado en aplicaciones industriales.

## **j.- BIBLIOGRAFÍA**

**Acedo Sánchez, José. 2003.** *Control Avanzado de Procesos*. Madrid : Ediciones Díaz de Santos S.A., 2003. ISBN: 84-7978-545-4.

**Aguado, Alberto. 2000.** *Temas de Identificación y Control Adaptable*. La Habana : ICIMAF, 2000.

**Lischinsky, Pablo. 2004.** *Introducción al modelado de sistemas físicos*. Mérida, Venezuela : s.n., 2004.

**López, Elena. 1999.** *Identificación de sistemas*. 1999.

**Martínez, Angel. 2011.** *Identificación experimental de sistemas*. 2011.

**Rodríguez Ramírez, Daniel y Bordóns Alba, Carlos. 2005.** *Apuntes de Ingeniería de Control*. 2005.

**Salcedo, Robin y Correa, Rodrigo. 2008.** *Simulador para Control Predictivo Basado en Modelos*. Santander : Universidad Industrial de Santander, 2008.

**The MathWorks Inc. 2015.** [En línea] 2015. [Citado el: 9 de Febrero de 2015.] <http://es.mathworks.com/help/mpc/getting-started-with-model-predictive-control-toolbox.html>.

**The MathWorks, Inc. 2000.** *Guía de Usuarios Toolbox de Identificación de Sistemas*. 2000.

**The MathWorks, Inc. 2014.** *Guía de Usuarios Toolbox Control Predictivo basado en Modelo*. 2014.

**Trejo, Victor. 1986.** *Identificación experimental de sistemas*. s.l. : ISPJAE, 1986.

**Tandazo S., Luis. 2014.** *Diseño y Construcción de un Banco para la Medición y Control de la Presión en un Proceso Experimental*. (Tesis Ing. Electromecánico) Loja, Ecuador, Universidad Nacional de Loja. Ingeniería Electromecánica.



## **k.- ANEXOS**

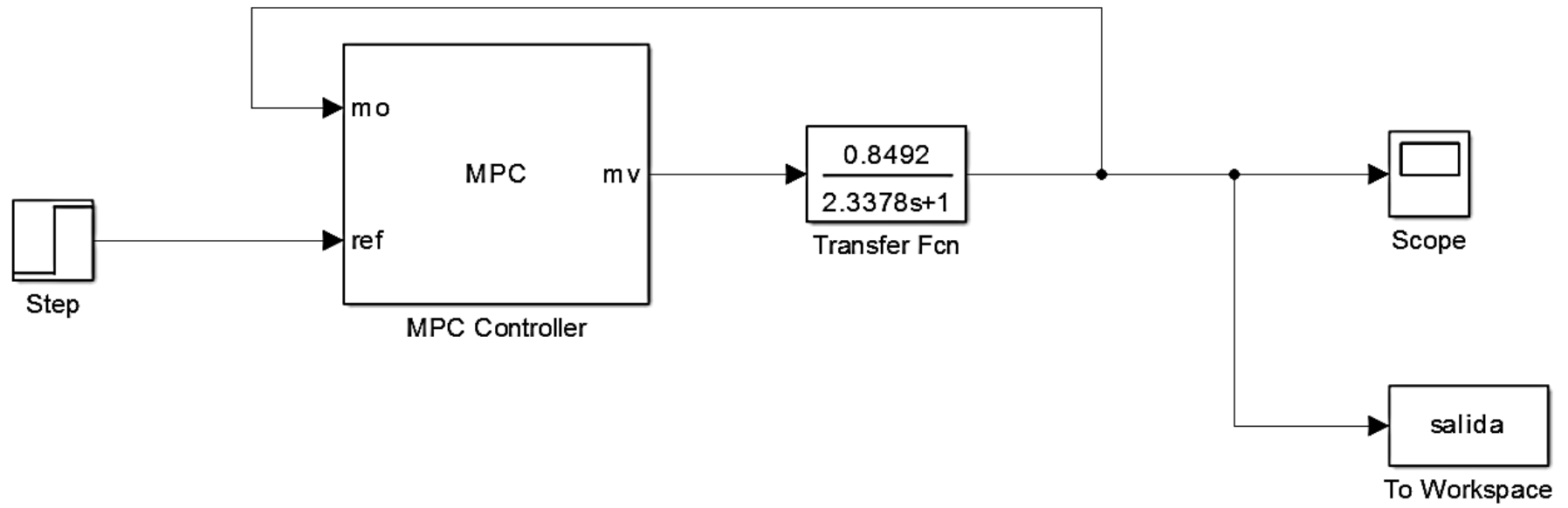
### **k.1. Diagrama de control**



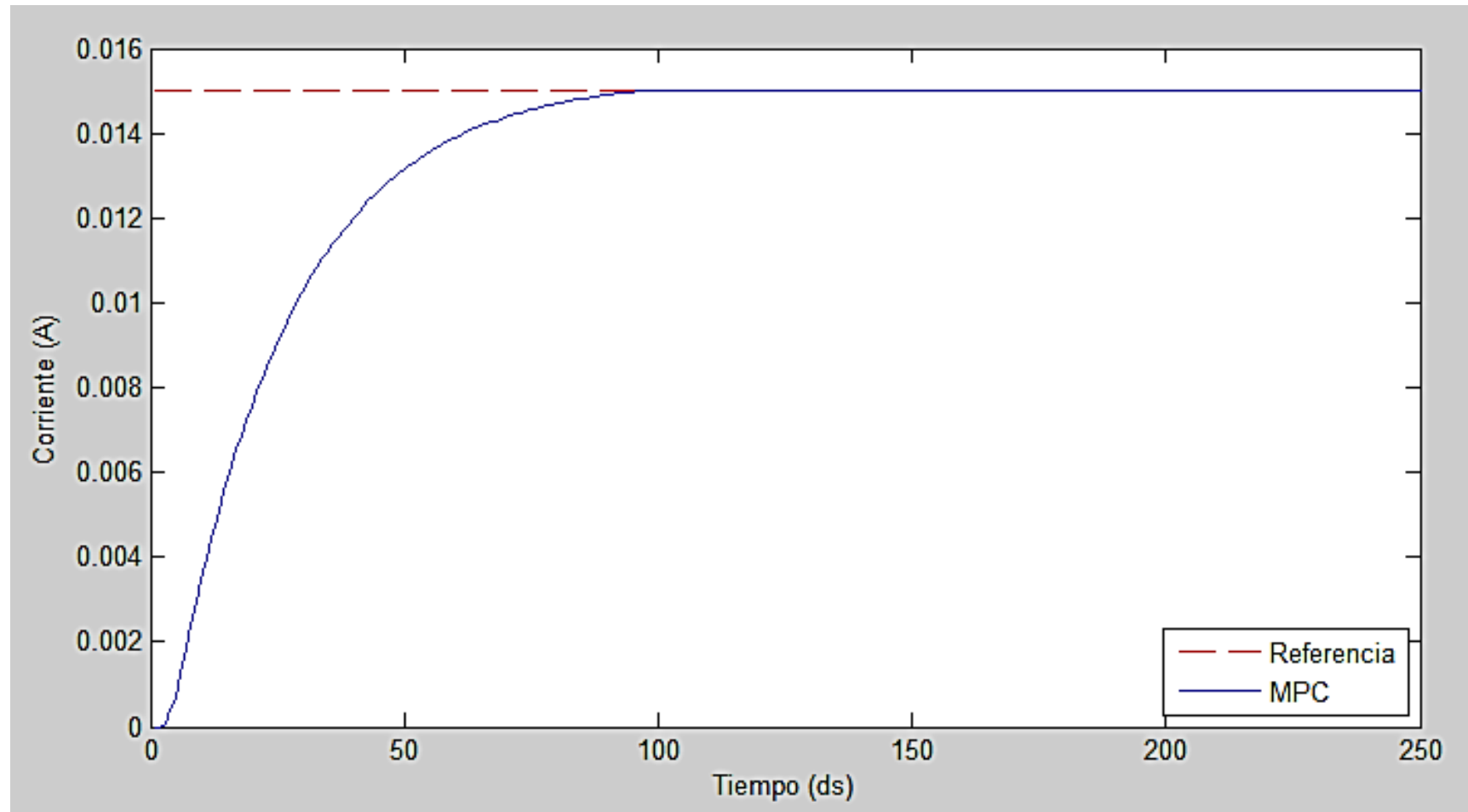
## k.2. Banco de Control de Presión



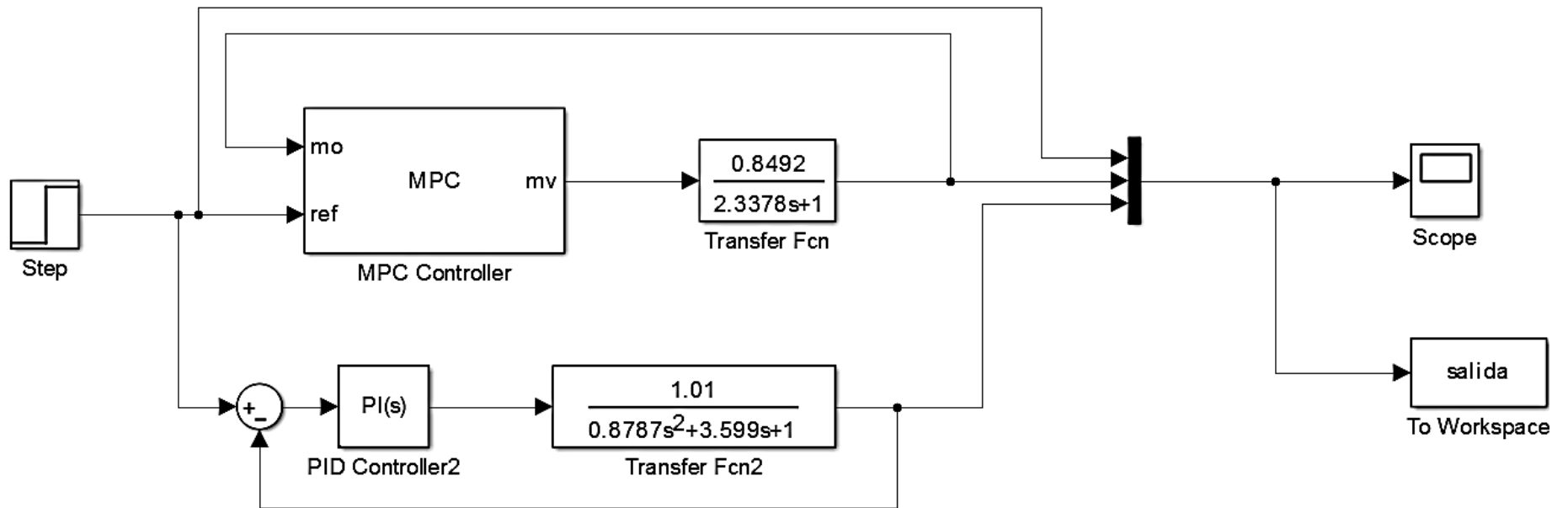
### k.3. Diagrama del Controlador MPC



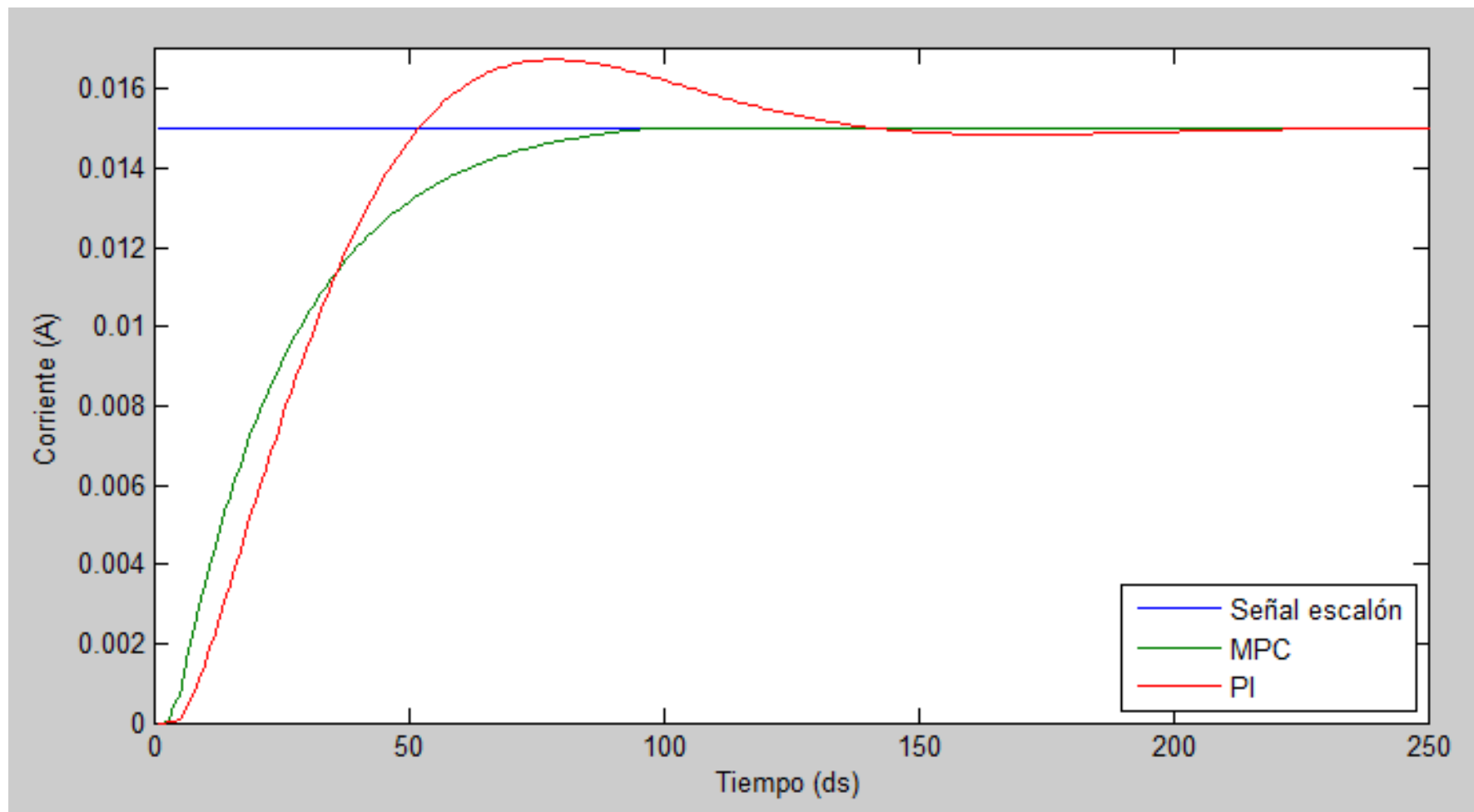
#### k.4. Gráfica del Controlador MPC



### k.5. Diagrama Controladores MPC y PI



### k.6. Gráfica comparación de Controladores



#### **k.7. Proyecto de Tesis**