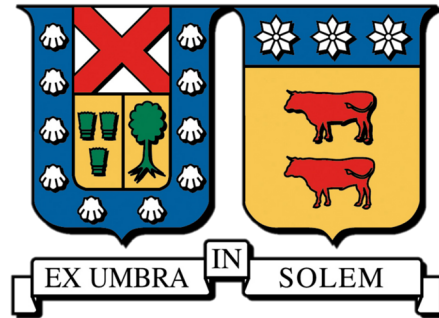


UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE ELECTRÓNICA
VALPARAÍSO - CHILE



**“VISUALIZACIÓN Y ANÁLISIS AUTOMÁTICO DE
DATOS DE PROCESOS DE PLANTA
CONCENTRADORA”**

IGNACIA ARAYA CARREÑO

**MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERA CIVIL
ELECTRÓNICA**

PROFESOR GUÍA:

MANUEL OLIVARES.

PROFESOR CORREFERENTE:

IVÁN RIVERO.

INGENIERO CORREFERENTE:

JORGE POBLETE.



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título; ☐ Tesis de Postgrado;

Título del trabajo: Visualización y Análisis Automático de Datos de Procesos de Planta Concentradora

Nombre del candidato(a): Ignacia Araya Carreño

Carrera / Grado: Ingeniera Civil Electrónica

Campus: Casa Central Valparaíso ; **Departamento:** Electrónica

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Manuel Olivares Salinas
Yo, , en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL

☒ El trabajo **NO contiene información que amerite confidencialidad** y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (embargo) por:

☐ 6 meses; ☐ 12 meses; ☐ 2 años; ☐ 3 años; ☐ 5 años; ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 26-09-25 **Firma:** Manuel Olivares S.

Estudiante o Candidato(a):

Fecha: 28-7-25; **Firma:** Araya

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

Agradecimientos

Quiero agradecerle a mis papás. Gracias mamá por siempre ponernos primero, por estar en cada presentación del colegio y por educarme para ser una persona independiente. Por repetirme que soy capaz de todo y por cuidarme en cada resfrío. Gracias papá porque nunca me faltó nada, siempre tuve comida, una cama donde dormir y alguien que me fuera a buscar a los cumpleaños o venir a dejar a la universidad. Gracias a los dos por estar orgullosos de mí.

Quiero también agradecer a mi hermano, por ser mi modelo a seguir, por estar conmigo en mi primer año universitario, por darme consejos para la vida adulta y por darme las mejores historias para contar. Gracias por ser mi enemigo y mi mejor amigo.

Gracias a mi pareja. Partimos esta etapa universitaria juntos y no pensé que la terminaríamos así. Has sido un pilar en mi vida, gracias por estar conmigo cada vez que te necesité, porque creíste en mí cuando ni yo lo hacía, por acompañarme hasta el último momento y porque cada día contigo soy mejor persona.

Quiero agradecer a mi abuela, que aunque ya no está, gracias a ella tuve donde vivir mis primeros años de universidad. Gracias por comprarme helados cuando era pequeña y por enseñarme a hacer pantrucas. Siempre su sabor me recuerdan a ti.

Finalizando, gracias a todos los que me acompañaron en este proceso. Gracias a mis tutores por darme la oportunidad de trabajar con ellos y aprender. Gracias a mis amigos por desvelarnos estudiando. Gracias a mi gato, porque a veces su compañía era lo único que necesitaba.

Por último, me agradezco a mí, por nunca rendirme. Después de todo, si se pudo sacar la carrera en 6 años e invicta.

Para mi familia que está en la tierra y en el cielo.

Para mí.

Resumen

El presente trabajo desarrolla una herramienta de monitoreo y análisis de desempeño para lazos de control en una planta concentradora. En primer lugar, se lleva a cabo un estudio del estado del arte sobre los sistemas de Monitoreo de desempeño de Lazos de Control (CLPM) disponibles comercialmente, identificando sus características, capacidades y limitaciones. Posteriormente, se establecen los indicadores clave de desempeño utilizados en este estudio, incluyendo el tiempo de subida, tiempo de asentamiento, razón de amortiguamiento, sobrepaso y error absoluto medio, seleccionados por su relevancia en la evaluación de la estabilidad y eficiencia del control. Para garantizar la fiabilidad de los indicadores calculados, se consideran aspectos como la estandarización de datos y la correcta selección de ventanas de tiempo.

A continuación, se implementa un programa computacional en Python para el cálculo automatizado de estos indicadores. Dicho programa permite la importación y procesamiento de datos operacionales, así como la generación de métricas clave, integrando los resultados con el software LabVIEW para su visualización. La interfaz gráfica en LabVIEW proporciona una representación clara de los datos, facilitando el monitoreo y la evaluación del rendimiento de los lazos de control. La automatización del análisis reduce la necesidad de interpretación manual y permite detectar problemas operacionales en los lazos de control, favoreciendo la toma de decisiones basada en datos y la mejora continua del proceso industrial.

La validación del sistema se realiza en dos etapas: en primer lugar, a través de pruebas en plantas simuladas, con el objetivo de verificar la correcta implementación y funcionamiento del programa. Posteriormente, se aplica el análisis a datos históricos a dos procesos de la planta concentradora, permitiendo obtener información sobre la eficiencia de los lazos de control en condiciones reales de operación. Los resultados obtenidos facilitan la identificación de oportunidades de mejora y la optimización de la sintonización de los controladores, contribuyendo a una mayor estabilidad y eficiencia en el proceso productivo.

Abstract

This work develops a monitoring and performance analysis tool for control loops in a concentrator plant. First, a state-of-the-art study on commercially available Control Loop Performance Monitoring (CLPM) systems is conducted, identifying their characteristics, capabilities, and limitations. Subsequently, the key performance indicators used in this study are established, including rise time, settling time, damping ratio, overshoot, and mean absolute error, selected for their relevance in assessing control stability and efficiency. To ensure the reliability of the calculated indicators, aspects such as data standardization and the appropriate selection of time windows are considered.

Next, a computational program is implemented in Python for the automated calculation of these indicators. This program enables the import and processing of operational data, as well as the generation of key metrics, integrating the results with LabVIEW software for visualization. The graphical interface in LabVIEW provides a clear representation of the data, facilitating the monitoring and evaluation of control loop performance. The automation of the analysis reduces the need for manual interpretation and allows detection of control loop issues, supporting data-driven decision-making and continuous process improvement.

The system validation is carried out in two stages: first, through testing in pilot plants to verify the correct implementation and functionality of the program. Subsequently, the analysis is applied to historical data from the concentrator plant, providing insights into the efficiency of control loops under real operating conditions. The obtained results facilitate the identification of improvement opportunities and the optimization of controller tuning, contributing to greater process stability and efficiency.

Nomenclatura

MV	: Variable manipulada o actuación de control (Manipulated Variable)
PV	: Variable de Proceso (Process Variable)
SP	: Referencia de Control (Set Point)
$e(t)$	: Error de control
K_p	: Constante proporcional de controlador PID
K_i	: Constante integrativa de controlador PID
K_d	: Constante derivativa de controlador PID
t_r	: Tiempo de subida
$t_{PV10\%}$	: Tiempo en que PV alcanza el 10 % de SP
$t_{PV90\%}$	: Tiempo en que PV alcanza el 90 % de SP
t_s	: Tiempo de asentamiento
τ	: Constante de tiempo de un sistema
ME	: Margen de error de oscilación aceptable
d	: Razón de amortiguamiento
E_{peak1}	: Error entre el primer peak máximo de PV y SP
E_{peak2}	: Error entre el segundo peak máximo de PV y SP
α	: Porcentaje de sobrepaso de PV
E_{peak}	: Error máximo entre PV y SP
σ	: Desviación estándar
$\bar{e}$	: Valor promedio del error entre PV y SP
$\rho(k)$	: Función de autocorrelación muestral
R	: Índice de decaimiento de método de Miao & Seborg
a	: Distancia mínima desde el primer máximo a la línea recta que conecta los primeros dos mínimos en respuesta escalón para método de Miao & Seborg
b	: Distancia mínima desde el primer mínimo a la línea recta que conecta el coeficiente de correlación con retraso 0 y el primer máximo en respuesta escalón para método de Miao & Seborg.

η_{mv}	: Índice de Harris
σ_{mv}	: Desviación de mínima varianza
σ_y	: Desviación de salida
IAE	: Integral del error absoluto
ISE	: Error cuadrático integrado
MAE_n	: Promedio muestral de error absoluto
MAE_t	: Promedio temporal de error absoluto
MAE	: Error absoluto medio

Abreviaciones

ACF	:	Función de Autocorrelación Muestral (Auto-Correlation Function)
ARIMA	:	Modelo Autorregresivo Integrado de Media Movil (Autoregressive Integrated Moving Average model)
CPM	:	Monitoreo de rendimiento de control (Control Performance Monitoring)
CLPM	:	Monitoreo de rendimiento de lazos de control (Control Loop Performance Monitoring)
DCS	:	Sistema de Control Distribuido (Distributed Control System)
FIC	:	Lazo de control de flujo con indicador (Flux Indicator Controller)
HMI	:	Interfaz Humano-Máquina (Human-Machine Interface)
KPI	:	Indicador Clave de Rendimiento (Key Performance Index)
LIC	:	Lazo de control de nivel con indicador (Level Indicator Controller)
MPC	:	Control Predictivo basado en Modelo (Model Predictive Control)
MVC	:	Control de Mínima Varianza (Minimum Variance Control)
PID	:	Control Proporcional Integrativo Derivativo (Proportional-Integral-Derivative)
PLC	:	Controlador Lógico Programable (Programmable Logic Controller)
P&ID	:	Diagrama de cañerías e instrumentación (Piping and Instrumentation Diagram)
SCADA	:	Control Supervisor y Adquisición de Datos (Supervisory Control and Data Acquisition)
WIC	:	Lazo de control de peso con indicador (Weight Indicator Controller)

Índice general

1..	Introducción	XIV
1.1.	Problema a resolver	XIV
1.2.	Objetivos	XV
1.3.	Datos disponibles	XV
1.4.	Conceptos fundamentales	1
1.4.1.	Tipos de lazos de Control	1
1.4.2.	Componentes Industriales de un lazo de control cerrado	2
1.4.3.	Ejemplo de lazo de control realimentado industrial	3
1.4.4.	Ventajas en Automatización Industrial	4
1.4.5.	Tipos de lazos de control realimentado	5
1.4.6.	Controlador PID	6
2..	Estado del arte	7
2.1.	Soluciones Comerciales	7
2.1.1.	Valmet PlantTriage - <i>Valmet</i>	7
2.1.2.	Loop Performance Manager - <i>ABB</i>	8
2.1.3.	Ecostruxure Control Advisor - <i>Schneider Electric</i>	10
2.1.4.	Control Performance Analytics - <i>Siemens</i>	11
2.2.	Trabajos relacionados	12
2.3.	Indicadores de desempeño	13
2.3.1.	Tiempo de Subida	14
2.3.2.	Tiempo de Asentamiento	15
2.3.3.	Razón de Amortiguamiento	16
2.3.4.	Sobrepaso	17
2.3.5.	Desviación Estándar	18
2.3.6.	Método Miao y Seborg	19
2.3.7.	Índice de Harris	21
2.3.8.	Porcentaje de tiempo de controlador en modo automático o manual	22

2.3.9.	Error Cuadrático Integrado	22
2.3.10.	Error Absoluto Integrado	22
2.3.11.	Error Absoluto Medio	23
2.4.	Procedimiento previo al análisis de datos históricos	24
2.4.1.	Estudio de la planta industrial	24
2.4.2.	Análisis de diagramas P&ID	25
2.4.3.	Determinación de lazos críticos	25
2.4.4.	Adquisición de Datos de operación	25
3..	<i>Diseño de la Interfaz de Visualización y Análisis Automático de Desempeño . . .</i>	27
3.1.	Indicadores seleccionados	28
3.1.1.	Tabla Resumen	31
3.2.	Programa de cálculo de indicadores	33
3.2.1.	Estandarización de los datos	33
3.2.2.	Código	34
3.2.3.	Diagrama de Flujo	37
3.3.	Visualización de Proceso	38
3.3.1.	Descripción de Proceso y pasos previos	38
3.3.2.	Interfaz en LabVIEW	40
3.4.	Software CLPM Integrado	42
3.5.	Guía de usuario	44
3.5.1.	Flujo de trabajo general	45
3.5.2.	Instrucciones de Uso	45
4..	<i>Validación del análisis automatizado mediante procesos simulados</i>	47
4.1.	Sistema de primer orden	48
4.1.1.	Resultados Esperados	48
4.1.2.	Resultados Obtenidos en CLPM	50
4.1.3.	Cálculo de parámetros dinámicos	50
4.2.	Sistema de segundo orden 1	51
4.2.1.	Resultados Esperados	51

4.2.2.	Resultados Obtenidos en CLPM	52
4.2.3.	Cálculo de parámetros dinámicos	53
4.3.	Sistema de segundo orden 2	53
4.3.1.	Resultados Esperados	53
4.3.2.	Resultados Obtenidos en CLPM	55
4.3.3.	Cálculo de parámetros dinámicos	55
4.4.	Análisis de resultados	56
4.4.1.	Tabla Comparativa	57
5..	<i>Análisis automatizado de datos de procesos de la planta concentradora</i>	58
5.1.	Proceso 1: Alimentación del Molino SAG	58
5.2.	Proceso 2: Flujo y Nivel en Celdas de Flotación Rougher	62
5.2.1.	Análisis del proceso general	62
5.2.2.	Desempeño del lazo de control de Flujo	63
5.2.3.	Desempeño de los lazos de control de nivel	67
6..	<i>Análisis de resultados y conclusiones</i>	
6.1.	Resultados y Conclusiones	
6.2.	Trabajo futuro	
7..	<i>Anexos</i>	
7.1.	Código	

Índice de figuras

1.1. Diagrama de Control a Lazo Abierto	1
1.2. Diagrama de Control a Lazo Cerrado	2
1.3. Lazo de Control Regulatorio de Temperatura en un intercambiador de calor [9]	4
2.1. Valmet PlantTriage [13]	8
2.2. Loop Performance Manager [15]	10
2.3. Ecostruxure Control Advisor [16]	11
2.4. Tiempo de subida	14
2.5. Tiempo de asentamiento con un margen de error del 2 %	15
2.6. Razón de Amortiguamiento	17
2.7. Sobrepaso	18
2.8. Distribución de probabilidad a diferentes desviaciones estándar	19
2.9. Método Miao & Seborg	20
3.1. Diagrama de flujo para diseño de programa CLPM	28
3.2. Ejemplo de formato estandarizado de datos en Excel.	33
3.3. Diagrama de flujo de programa	37
3.4. Diagrama de Planta Concentradora	39
3.5. Diagrama P&ID simplificado de Proceso Molino SAG	40
3.6. Diagrama P&ID simplificado de Proceso Flotación Rougher	41
3.7. Diagrama de Bloques de programación para obtener Indicadores	43
3.8. Diagrama de Bloques de programación para obtener Gráficos de respuesta	43
3.9. Interfaz final Molino SAG	44
3.10. Interfaz final Flotación Rougher	44
4.1. Gráfica de respuesta a escalón de sistema de primer orden en MATLAB	49
4.2. Resultados de Indicadores	49
4.3. Gráfica de respuesta a escalón de sistema de primer orden con programa CLPM	50
4.4. Resultados de Indicadores	50
4.5. Gráfica de respuesta de sistema de segundo orden 1 en MATLAB	51

4.6. Resultados de Indicadores	52
4.7. Gráfica de respuesta de sistema de segundo orden 1 con programa CLPM . . .	52
4.8. Resultados de Indicadores	53
4.9. Gráfica de respuesta de sistema de segundo orden 2 en MATLAB	54
4.10. Resultados de Indicadores	54
4.11. Gráfica de respuesta de sistema de segundo orden 2 en programa CLPM . . .	55
4.12. Resultados de Indicadores	55
5.1. Resultados sección Molino SAG	59
5.2. Gráfica señal de actuación lazo WIC	60
5.3. Gráfica señal de actuación lazo FIC	61
5.4. Resultados sección flotación Rougher	63
5.5. Lazo FIC 151 con $SP=265\left[\frac{m^3}{h}\right]$	64
5.6. Lazo FIC 151 con $SP=274\left[\frac{m^3}{h}\right]$	65
5.7. Lazo FIC 151 con $SP=292\left[\frac{m^3}{h}\right]$	65
5.8. Lazos de Nivel en el mismo periodo de tiempo	67

Índice de tablas

1.1. Cantidad de datos disponibles del proceso de concentrado por línea de procesamiento	XVI
3.1. Indicadores de lazos de control a usar	32
4.1. Error porcentual entre resultados esperados y obtenidos	57

1. INTRODUCCIÓN

La continua evolución y progreso de la industrialización ha incrementado la necesidad de automatización de plantas a nivel mundial. Cada vez se requieren sistemas de control automático más robustos, rápidos y precisos que puedan satisfacer las demandas de producción.

Estos sistemas de control, están basados en diversos algoritmos. Uno de los algoritmos más utilizados es el PID (Proportional Integrative Derivative), que ha demostrado sus ventajas y desventajas en distintos escenarios [1][2].

En plantas con alta integración tecnológica, como refinerías, plantas de energía y mineras, se incluyen cientos o incluso miles de lazos de control. Un lazo de control mal sintonizado o con fallas puede resultar en la detención de la producción, disminución de la calidad del producto o mayor consumo de material o energía, lo que disminuye la rentabilidad de la planta. Por ello, los lazos de control son cada vez más reconocidos como un factor importante que debe ser monitoreado continuamente. Tanto el rendimiento de los controladores como otros componentes de los lazos pueden ser monitoreados y mejorados a través de softwares industriales de monitoreo y auto-sintonización de lazos, conocidos como software CPM (Control Performance Monitoring) [3][4][5][6][7].

1.1. *Problema a resolver*

El análisis de desempeño de lazos de control es un proceso que actualmente puede consumir mucho tiempo, ya que los datos obtenidos de cada planta son valores crudos que requieren la interpretación de un experto. Se busca sistematizar este proceso para hacerlo menos laborioso y más eficiente, automatizando la obtención de indicadores necesarios para evaluar la eficiencia del lazo de control.

1.2. *Objetivos*

Objetivo general

Desarrollar una interfaz personalizada de visualización y análisis automático de datos históricos de una planta concentradora, utilizando Python y LabVIEW, para evaluar el desempeño de los lazos de control de los procesos que la componen.

Objetivos específicos

- Desarrollar un software CLPM que identifique los indicadores mínimos necesarios para conocer el comportamiento de un lazo de control, estableciendo una línea base de desempeño actual a partir del análisis de datos históricos.
- Comprender el proceso detrás de un CLPM y cómo abordar este tipo de proyectos en diferentes plantas, unificando la metodología para su repetición futura.
- Entregar información mínima necesaria para evaluar el rendimiento de las operaciones del proceso, con el fin de identificar posibles mejoras en los lazos de control existentes en la planta.

1.3. *Datos disponibles*

Para la implementación y evaluación del software CLPM diseñado se utilizan datos reales tomados de una planta concentradora de cobre. Los datos son recopilados a partir de sistemas de control y adquisición de datos de la planta. Esto se explicará con mayor detalle en la sub-sección 2.4.4. El formato en el que se reciben los datos y cómo deben estandarizarse para poder ser procesados por el programa CLPM diseñado se explica en la sub-sección 3.2.1. A continuación se presenta un resumen de los datos disponibles del proceso:

Tab. 1.1: Cantidad de datos disponibles del proceso de concentrado por línea de procesamiento

Tipo de Lazo	Línea de procesamiento 1	Línea de procesamiento 2
FIC	12	12
HIC	1	1
LIC	14	14
WIC	1	1

1.4. Conceptos fundamentales

1.4.1. Tipos de lazos de Control

Los sistemas de control se clasifican en dos categorías principales: lazo abierto y lazo cerrado.

Un semáforo es un ejemplo de un sistema de control de lazo abierto representado en la figura 1.1, donde la señal de control (cambio de luces) se ejecuta según una secuencia temporal predeterminada, sin considerar variables del proceso como el flujo vehicular, densidad del tráfico o condiciones específicas de la intersección. En términos técnicos, no existe realimentación entre la salida del sistema (estado del tráfico) y la entrada de control (secuencia de luces).

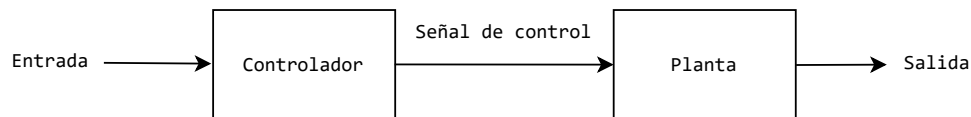


Fig. 1.1: Diagrama de Control a Lazo Abierto

Un lazo de control realimentado es un sistema utilizado en ingeniería para mantener una variable de proceso (Process Value: PV) en un valor deseado (Set Point: SP) modificando la variable manipulada (Manipulated Variable: MV). Este sistema es fundamental en la automatización industrial, ya que permite mantener la estabilidad y continuidad de los procesos. Un ejemplo de lazo de control realimentado es la regulación de temperatura en una ducha, donde el usuario actúa como controlador dentro del sistema de retroalimentación. En este caso, existe una referencia que es la temperatura deseada, y el usuario continuamente mide la salida del proceso (temperatura actual del agua), compara con la referencia deseada (error), ajusta la variable manipulada (posición de la válvula) y repite el ciclo hasta alcanzar y mantener la temperatura objetivo.

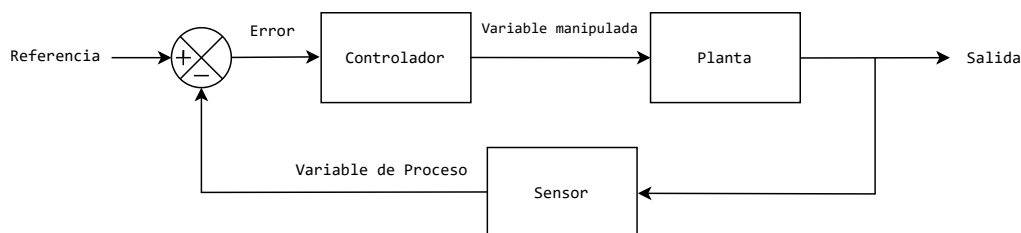


Fig. 1.2: Diagrama de Control a Lazo Cerrado

El control realimentado o de lazo cerrado visto en la figura 1.2 sigue los siguientes pasos de forma estándar:

1. Realimentación: Se mide la variable del proceso.
2. Comparación: Se compara la medición con la señal de referencia.
3. Acción de Control: Si existe desviación, el controlador modifica una señal de control a la planta para corregir el error, típicamente calculada por un controlador PID.
4. Proceso: La acción de control debe producir un efecto en la planta tal que su salida se aproxime a la referencia.

Este principio de control con retroalimentación es fundamental en la automatización industrial, donde se implementa en forma de controladores PID y sistemas más avanzados para regular variables críticas como temperatura, presión, flujo, nivel y composición en procesos complejos, entre otras. La retroalimentación permite compensar perturbaciones y mantener las variables de proceso en sus valores óptimos de operación.[8]

1.4.2. Componentes Industriales de un lazo de control cerrado

Para la implementación de un lazo de control cerrado se requieren al menos los siguientes componentes:

- Sensores y Transmisores: Los sensores son los dispositivos de medición primarios. Miden la variable de proceso y la convierten en señales que pueden ser convertidas en

señales electrónicas. Los transmisores por su parte, son dispositivos de conversión a señales normalizadas para sistemas de control. Generalmente incluyen sensores en su estructura, pero su función principal es convertir una forma de señal a otra para que pueda ser leída y entendida por el controlador.

- Controladores: Dispositivo que se encarga de interpretar las mediciones realizadas por los sensores. Mediante esta interpretación, calcula y determina la acción que se debe tomar para controlar el proceso, transmitiendo una acción de control al actuador.
- Elementos Finales de Control y Actuadores: Reciben la señal de acción de control del controlador y la ejecutan para modificar la variable de proceso. Estos pueden ser válvulas, interruptores o motores que modifican directamente la salida del proceso.

1.4.3. Ejemplo de lazo de control realimentado industrial

Un ejemplo común es el control de temperatura en un intercambiador de calor, como el descrito a continuación obtenido de [9]:

Un sensor electrónico de temperatura instalado en la salida de agua caliente envía continuamente una señal de entrada al controlador.

A intervalos establecidos, el controlador compara esta señal con un punto de ajuste predefinido (SP). Si la señal de entrada se desvía del punto de ajuste, el controlador envía una señal de salida eléctrica correctiva al elemento de control.

Esta señal eléctrica debe convertirse en una señal neumática cuando se utiliza con una válvula operada por aire. Usando un transductor I/P, que convierte una señal eléctrica de 4 a 20 [mA] en una señal de aire de 3 a 15 [psi], se envía el suministro de aire respectivo al posicionador de la válvula de control.

El posicionador de la válvula ajusta la posición del vástago de la misma y regula el flujo de vapor a través de esta, controlando así la temperatura del agua. Este ciclo se repite, logrando que el controlador alcance el punto de ajuste.

Lo descrito anteriormente se puede ver en la figura 1.3.

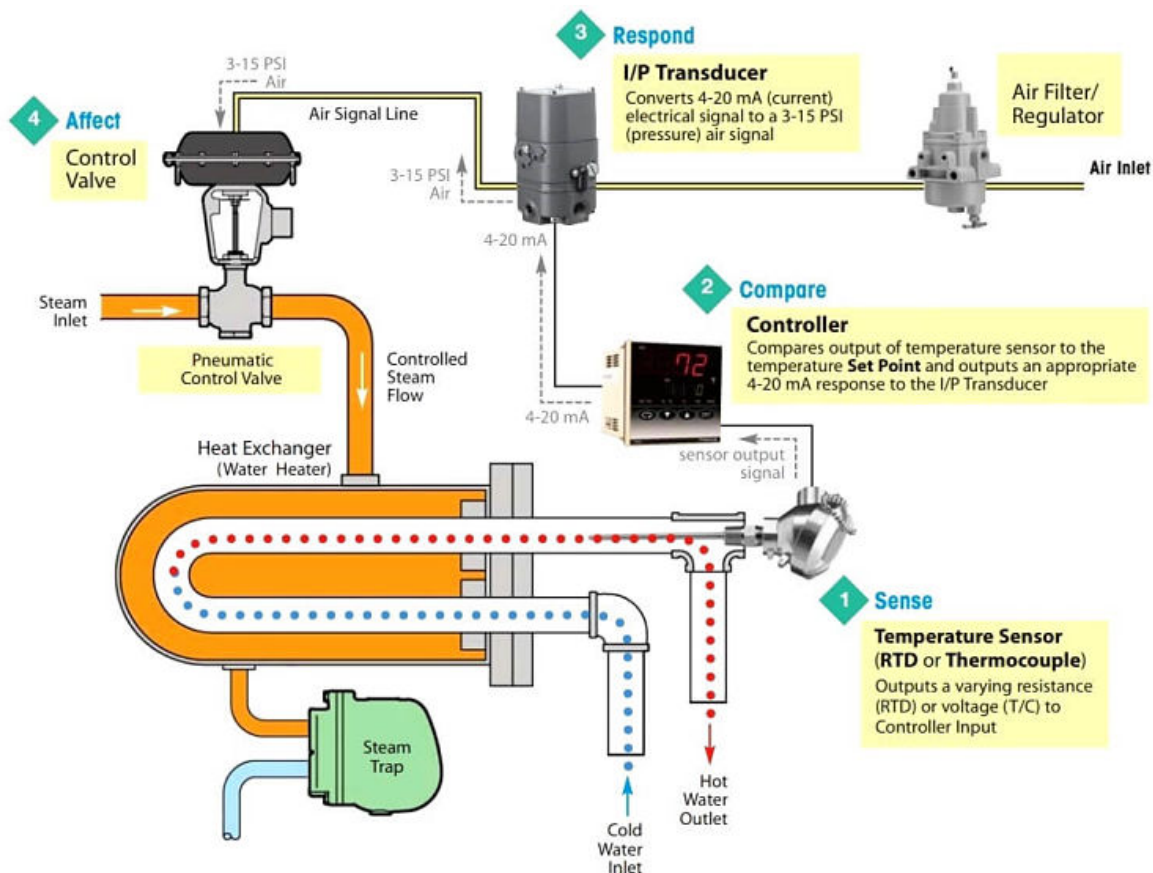


Fig. 1.3: Lazo de Control Regulatorio de Temperatura en un intercambiador de calor [9]

1.4.4. Ventajas en Automatización Industrial

Los lazos de control ofrecen numerosas ventajas en la automatización industrial [10], entre las que destacan:

- **Mejora en la Eficiencia de los Procesos Industriales:** Los lazos de control permiten mantener las variables de proceso dentro de un rango específico, lo que se traduce en una mayor eficiencia operativa. Al regular parámetros como la temperatura, presión y flujo, se optimizan los recursos y se minimizan los desperdicios. Además, el tener los sistemas automatizados minimiza la interacción humana, ofreciendo una mayor rapidez y exactitud que un operador manual.
- **Mayor Seguridad:** Estos sistemas permiten un monitoreo constante del funcionamiento

de los procesos industriales, reduciendo significativamente el riesgo de fallos o accidentes. La capacidad de detectar y corregir desviaciones en tiempo real contribuye a un entorno de trabajo más seguro.

- **Menor Costo de Mantenimiento:** Al regular los procesos industriales de manera automática, los lazos de control disminuyen la necesidad de intervención humana. Esto no solo reduce el costo de mantenimiento, sino que también prolonga la vida útil de los equipos al evitar ajustes manuales frecuentes.
- **Mejora en la Calidad de los Productos:** Mantener las variables de proceso dentro de un rango específico asegura una mayor consistencia y calidad en la producción. Esto es crucial para cumplir con los estándares de calidad y satisfacer las expectativas del cliente.
- **Aumento de la Productividad:** La optimización de los procesos industriales mediante el uso de lazos de control realimentado resulta en una mayor eficiencia y productividad. Al reducir tiempos de inactividad y mejorar la precisión de los procesos, se incrementa la capacidad de producción y se maximiza el rendimiento.

1.4.5. Tipos de lazos de control realimentado

Existen diferentes tipos de lazos de control, siendo los más conocidos:

- **Lazo de Control ON/OFF:** Se activa si el valor de proceso está por debajo del SP y la desactiva si está por encima.

Dependiendo de la naturaleza del proceso bajo control, es posible que haya que introducir un valor de histéresis para evitar la constante activación y desactivación del dispositivo de control.

- **Lazo de Control Regulatorio:** Es uno de los más comunes. El control ajusta de forma continua la demanda de salida según una ecuación matemática con el fin de mantener el proceso lo más cerca posible de las condiciones exigidas. Se pueden utilizar lazos de control de lógica difusa, MPC, redes neuronales, y el más común siendo PID.

Estudios muestran que dentro de los diferentes esquemas de control regulatorio, el más utilizado en diversos rubros y aplicaciones industriales son los lazos de control PID [1][2]. El control PID, a diferencia del control ON/OFF, tiene una configuración más compleja, ya que la elección de los parámetros (K_p , K_i , K_d) deben relacionarse con las características dinámicas del proceso bajo control.

1.4.6. Controlador PID

Un controlador PID es un mecanismo de control retroalimentado que se construye utilizando tres términos principales: Proporcional, Integral y Derivativo. La suma ponderada de los términos forma la salida del controlador, siendo la ecuación que lo describe la siguiente:

$$u(t) = K_p \cdot e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \quad (1.1)$$

El Término Proporcional (K_p), como dice su nombre, es proporcional a la magnitud de la señal del error $e(t)$. Mientras mayor sea el error, mayor será el ajuste aportado a la salida de control.

Por su parte, el término Integral (K_i) ajusta la salida del controlador basado en el error de control acumulado. Se requiere para conseguir un error cero en estado estacionario.

Finalmente, el término derivativo (K_d) genera un cambio en la salida que depende de la tasa de cambio del error. Resulta útil para corregir perturbaciones en sistemas con inercia.

Al diseñar este tipo de controladores, es posible desactivar los términos integral y/o derivativo y aplicar control exclusivamente proporcional (P), proporcional e integral (PI) o proporcional y derivativo (PD).

2. ESTADO DEL ARTE

Para iniciar el desarrollo de un software CLPM, es primordial primero llevar a cabo un estudio del estado del arte acerca de trabajos que hayan hecho algo similar, así como de soluciones comerciales disponibles en el mercado por las marcas más reconocidas en el mundo de la automatización industrial. En particular, se estudian ofertas de productos de Valmet, ABB, Schneider Electric y Siemens. Posteriormente, se indaga sobre los indicadores de desempeño de lazos de control más utilizados en el estado del arte, destacando las ventajas y desventajas de cada uno en términos de relevancia de la información que entregan y facilidad de cálculo. Finalmente, este capítulo termina con un estudio de la planta industrial en la que se orienta la aplicación del software diseñado.

2.1. *Soluciones Comerciales*

En el mercado existen distintos sistemas de Control Loop Performance Monitoring (CLPM) disponibles. Resulta crucial analizar sus características para determinar puntos claves que permiten sus capacidades analíticas y beneficios operacionales.

2.1.1. *Valmet PlantTriage - Valmet*

Valmet PlantTriage se presenta como una solución integral que permite el procesamiento de datos en tiempo real las 24 horas del día, durante todo el año.

Su diseño está orientado a satisfacer las necesidades de diversos profesionales dentro de la organización, desde ingenieros de control y automatización hasta gerentes de operaciones, proporcionando una plataforma versátil que facilita la toma de decisiones basada en datos y utilizando algoritmos de priorización, diagnóstico y resolución.[11]

Su sistema de priorización se basa en identificar el impacto económico, productivo y la

calidad de los productos. Sus reportes facilitan la visualización de métricas clave permitiendo focalizar recursos en las áreas más críticas.

Por otro lado, su algoritmo de detección de problemas utiliza la información almacenada para identificar rápidamente la causa raíz y ajustar a la sintonización correcta, corrigiendo el error en minutos.

El software recopila y analiza toda la información en un solo lugar, lo que permite optimizar los procesos, minimizar errores, incrementar la producción y mejorar la eficiencia energética. En este contexto, la información recopilada históricamente y las herramientas digitales adecuadas son esenciales para seguir mejorando la producción.[12]

Es ideal para procesos continuos y de gran escala, como los de las industrias de pulpa y papel, química, minería, petróleo y gas, y generación de energía. Lo anterior destaca la importancia de comprender el proceso y la necesidad de una planta de gran magnitud para manejar más datos, obteniendo así suposiciones más precisas.[13]

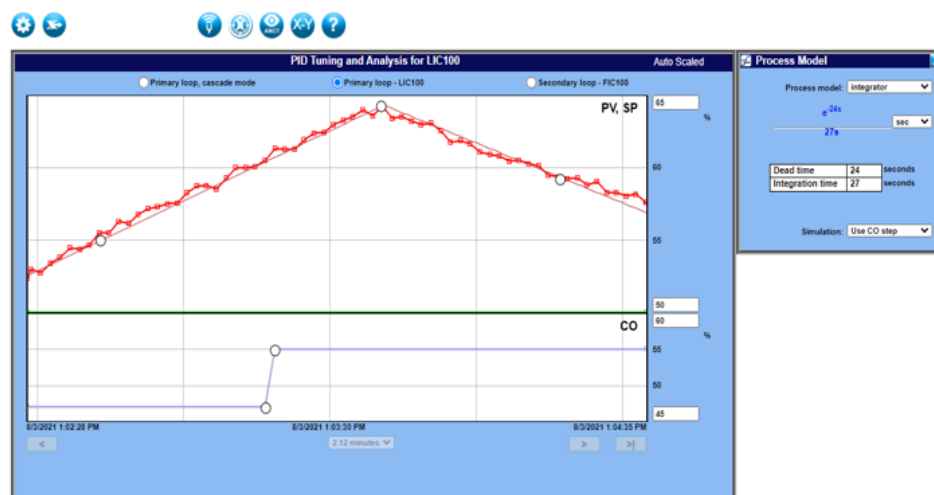


Fig. 2.1: Valmet PlantTriage [13]

2.1.2. Loop Performance Manager - ABB

La empresa de tecnología ABB desarrolló Loop Performance Manager, una aplicación diseñada para mantener la instrumentación de cada planta en un estado óptimo. El sistema se compone de dos secciones principales para la mejora de procesos: una herramienta de sintonización y otra de monitoreo.

La sintonización va siguiendo las siguientes etapas: Recopilación de datos, procesamiento y almacenamiento, identificación del modelo de proceso, evaluación del modelo, cálculo de parámetros de ajuste del PID, y evaluación y registro de la sintonización del controlador. Este proceso asegura que los controladores PID estén ajustados de manera óptima, lo que mejora la estabilidad y eficiencia del sistema.

El monitoreo por otro lado, sigue una estructura de: recopilación periódica de datos, cálculo de indicadores de rendimiento, almacenamiento de los índices y diagnóstico de problemas. Esto permite una supervisión continua y proactiva de los lazos de control, identificando y resolviendo problemas antes de que afecten la producción.

Entre los beneficios destacados del monitoreo se encuentran la prolongación de la vida útil de los equipos, la mejora de la eficiencia energética, el aumento de la producción, la reducción de viajes a la planta y el mantenimiento de la calidad del producto.[14]

Para calcular los índices obtenidos durante el monitoreo, Loop Performance Manager utiliza datos recopilados de la salida del controlador (MV), la variable medida (PV) y el valor deseado (SP). Estos datos son fundamentales para evaluar el rendimiento de los lazos de control.

Entre los índices de desempeño utilizados se incluyen áreas de estadística básica, modos de lazos de control e índices de oscilación, entre otros. Los indicadores y métricas más destacados son el sobrepaso, la desviación estándar, el IAE, el tiempo de subida y el tiempo de asentamiento.

Además de estos indicadores, Loop Performance Manager también puede utilizar otros índices avanzados para un análisis más detallado, como el análisis de la frecuencia de oscilación y la detección de patrones de comportamiento anómalos. Estos análisis permiten una identificación temprana de problemas y la implementación de soluciones.[15]

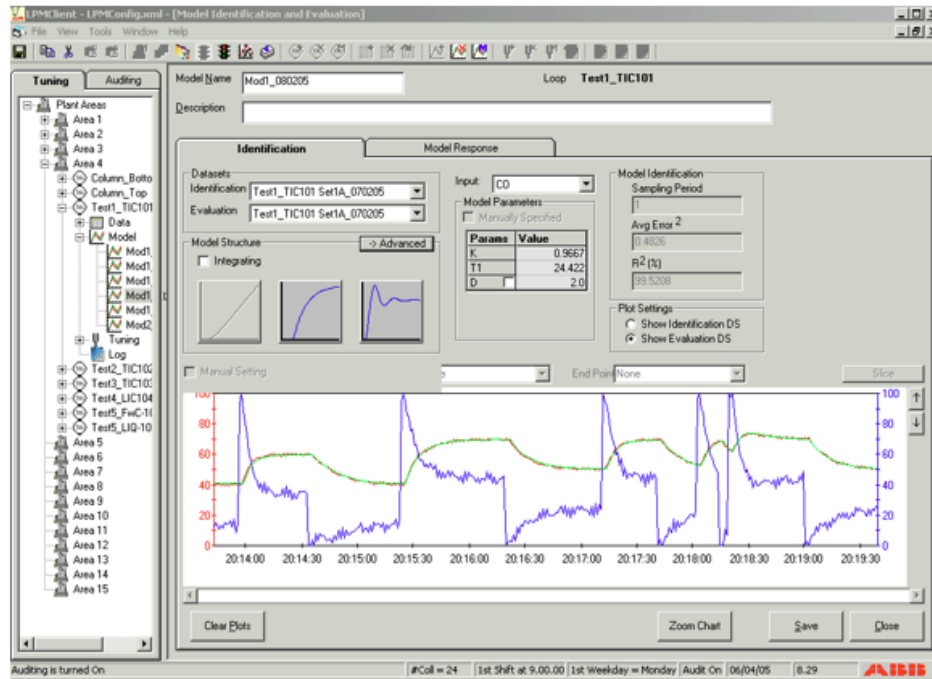


Fig. 2.2: Loop Performance Manager [15]

2.1.3. Ecostruxure Control Advisor - Schneider Electric

Schneider Electric aborda el monitoreo y optimización del desempeño de lazos de control a través de EcoStruxure Control Advisor, una solución integrada dentro de su plataforma EcoStruxure diseñada para optimizar la eficiencia operacional en entornos industriales.

La plataforma ofrece un diagnóstico detallado de las oportunidades de mejora, enfocándose en los problemas más críticos y de mayor impacto económico. Identifica la causa de las oscilaciones o alteraciones en el proceso y proporciona recomendaciones de ajuste para mejorar el rendimiento del controlador.[16]

Esta plataforma es aplicable a una amplia variedad de industrias, incluyendo plantas de energía, químicas, de minerales y metales, de tratamiento de aguas, e incluso para la automatización de procesos.

Entre los indicadores utilizados se encuentran el tiempo de asentamiento, la oscilación del lazo, IAE, el histograma de la variable medida (PV), entre otros. [17]

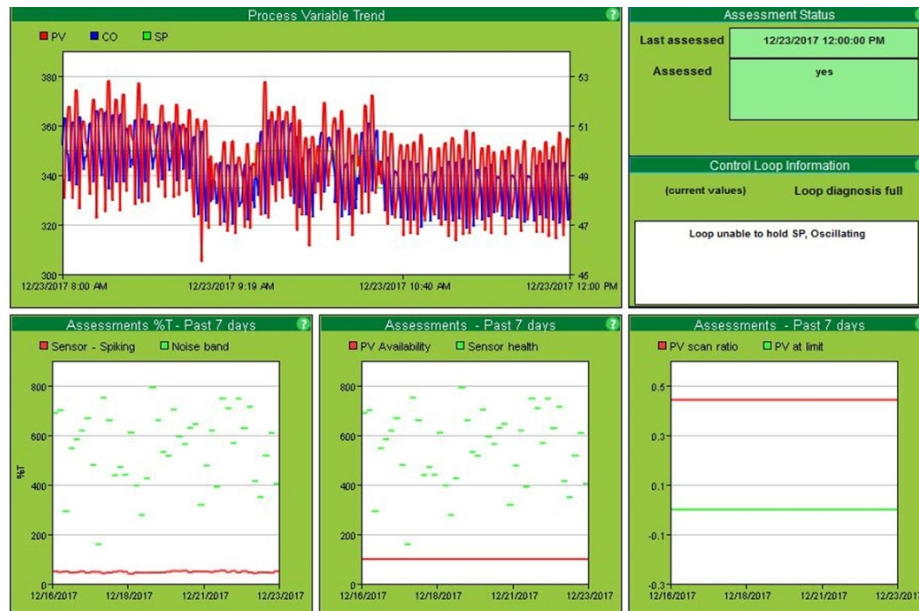


Fig. 2.3: Ecostruxure Control Advisor [16]

2.1.4. Control Performance Analytics - Siemens

Control Performance Analytics es un servicio que mejora la transparencia de los datos de proceso y optimiza los lazos de control. La identificación del estado del lazo de control se basa en el cálculo de KPI, que son capaces de identificar el seguimiento a la referencia, los problemas en estado estacionario y los problemas en las válvulas de control.

Además, permite realizar cálculos automatizados de nuevos conjuntos de parámetros para optimizar y ajustar los lazos de control sin afectar el funcionamiento de la planta. Este servicio promete incrementar la productividad, maximizar la vida útil de los equipos y reducir la carga de trabajo del operador al disminuir las alarmas, entre otros beneficios.[3]

Entre los KPIs que ofrece se encuentran el factor de servicio, la precisión del seguimiento a la referencia y el error en estado estacionario. Estos indicadores permiten una evaluación continua y precisa del rendimiento del sistema, facilitando la identificación y corrección de problemas para optimizar la eficiencia operativa.[4]

2.2. Trabajos relacionados

La importancia de mantener un control preciso de la información en la industria es crucial para asegurar la eficiencia y la calidad de los productos. En este contexto, se ha estudiado cuáles y cuántos KPIs son necesarios para obtener una buena respuesta del desempeño de los lazos de control. Los índices pueden variar desde los más simples hasta los más complejos, permitiendo medir el rendimiento de acuerdo con las metas de la empresa y optimizar los procesos acordemente.[5]

Entre los índices de estadística simples que se utilizan se encuentran:

- Promedio del Error de Control: Entendida como la diferencia entre la referencia y la variable de proceso, normalmente debería ser 0 en estado estacionario, por lo que se busca minimizar su existencia.
- Desviación Estándar del Error de Control: Este parámetro, al igual que el anterior, a menor desviación estándar se tiene una mejor estabilidad del lazo ante perturbaciones en el horizonte de tiempo de estudio.

Estos índices simples permiten evaluar la estabilidad y el rendimiento básico del lazo de control.[5][6][7]

Complementariamente, existen indicadores avanzados como el Índice de Harris, que es popular en varios documentos técnicos.

Este índice es indicativo del rendimiento óptimo y se utiliza para diseñar controladores que minimicen la variación del proceso tanto como sea posible (MVC). Al analizar el MVC, se debe ajustar un modelo de media móvil autorregresiva (ARIMA) a los datos del proceso. Se trata de un modelo predictivo que representa la acción que tomaría un controlador de varianza mínima.[5][6][7][18][19]

Otro indicador importante es el modo de control, el cual ve si el controlador está en modo automático o manual.[6][20]

Por otro lado, existen indicadores basados en la curva de respuesta ante cambios de SP e índices basados en la curva de respuesta total:

Índices basados en la curva de respuesta ante cambios de SP

- Tiempo de subida: Tiempo de cambio de la variable de procesos desde su punto anterior al nuevo.
- Tiempo de asentamiento: Tiempo que tarda la variable de proceso en llegar a estado estacionario.
- Razón de amortiguamiento: Razón entre dos máximos de error consecutivos.
- Sobrepaso: Razón entre la diferencia del peak máximo de la variable de procesos y el nuevo SP.

Índices basados en la curva de respuesta total

- Integral del error absoluto (IAE): Identifica la variabilidad promedio de la curva de respuesta de control, en una ventana de tiempo definida.
- Promedio muestral del error absoluto (MAE_n): Funciona igual que IAE pero divide en un total de muestras, en un tiempo dado.
- Promedio temporal del error absoluto (MAE_t): Se obtiene si en lugar de muestras se considera un tiempo determinado.

Estos índices permiten entender el comportamiento del lazo al momento de querer cambiar su referencia y cómo actúan ante esto.[18]

Los indicadores permiten una evaluación continua y precisa del rendimiento del sistema, facilitando la identificación y corrección de problemas para optimizar la eficiencia operativa. Utilizar estos KPIs ayuda a mantener los procesos bajo control, mejorar la calidad del producto y aumentar la eficiencia energética.

El análisis de desempeño de lazos de control ha sido objeto de estudio en otros trabajos de memoria, tal como se documenta en la referencia [18].

2.3. Indicadores de desempeño

Para tener un buen entendimiento de los indicadores utilizados tanto comercialmente como en trabajos de investigación, se hará un estudio detallado de cada uno de estos y cómo impactan en un software CLPM.[18]

2.3.1. Tiempo de Subida

El tiempo de subida es el intervalo de tiempo que tarda la respuesta del sistema en ir desde el 10 % al 90 % de su valor en estado estacionario después de aplicar un escalón. Este índice es importante en aplicaciones en las que se necesita una respuesta rápida del sistema. Un tiempo de subida muy alto, evidencia problemas en la sintonización del lazo ya que está directamente relacionado con el tiempo de respuesta.[21]

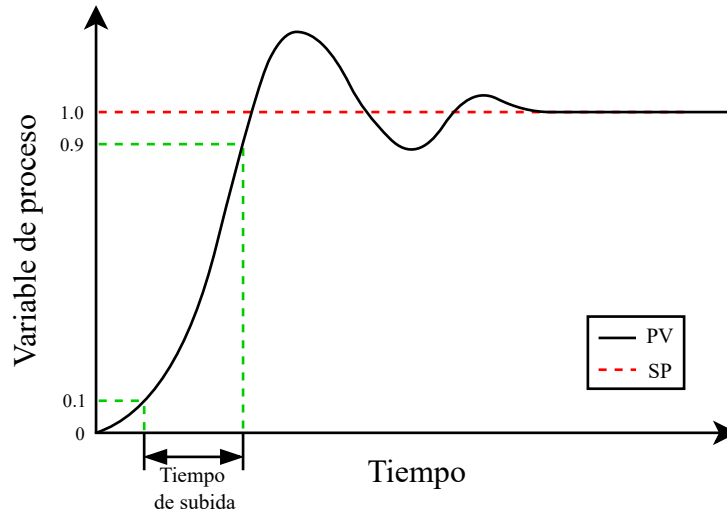


Fig. 2.4: Tiempo de subida

La ecuación matemática que lo describe es :

$$t_r = PV(t_{PV_{90\%}}) - PV(t_{PV_{10\%}}) \quad (2.1)$$

Donde,

$PV(t)$: Variable de proceso del lazo de control analizado

$t_{PV_{10\%}}$: Tiempo en que PV alcanza el 10 % de la referencia

$t_{PV_{90\%}}$: Tiempo en que PV alcanza el 90 % de la referencia

Para interpretar este indicador, hay que entender que decir que un sistema es rápido depende de la aplicación del lazo. Resulta importante tener en cuenta el tipo de proceso con el que se trabaja y que un usuario establezca qué límites le parecen apropiados.

2.3.2. Tiempo de Asentamiento

El tiempo de asentamiento mide cuánto tiempo le toma al sistema llegar a su valor en estado estacionario ante una entrada de tipo escalón. En el control de procesos industriales, el tiempo de asentamiento es un factor crítico para garantizar la calidad y la eficiencia del proceso. A mayor tiempo en un estado inestable, mayor el tiempo que el proceso no estará funcionando en las condiciones deseadas.[22]

La ecuación matemática que lo describe es :

$$t_s = PV(4\tau) \quad (2.2)$$

Donde,

$PV(t)$: Variable de proceso del lazo de control analizado

τ : Constante de tiempo del sistema

Otra manera de obtenerlo es estableciendo un criterio de asentamiento. Se genera un margen de error del 2 % al 5 %, y se ve cuándo la variable de proceso oscila en ese rango.

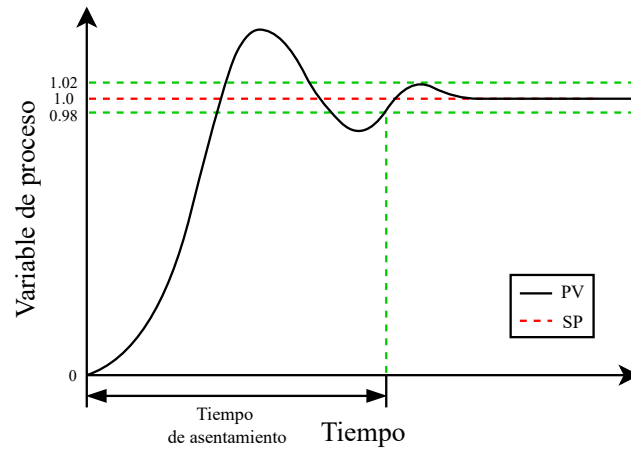


Fig. 2.5: Tiempo de asentamiento con un margen de error del 2 %

La ecuación matemática que lo describe es :

$$|PV(t) - SP| \leq ME * SP \quad (2.3)$$

Donde,

PV(t): Variable de proceso del lazo de control analizado

SP: Referencia del sistema

ME: Margen de error que oscila entre [0.02-0.05]

2.3.3. Razón de Amortiguamiento

Este cociente indica qué tan oscilante es el sistema midiendo la proporción de dos errores máximos consecutivos entre la variable de procesos y su referencia.

Un valor muy alto, alude a notorias oscilaciones en la respuesta ante cambios de SP. En el caso contrario, se observará una respuesta con oscilaciones amortiguadas rápidamente.

En la literatura, es aceptable una razón de amortiguamiento de valor 0.25, sin embargo, su aceptación depende de la robustez requerida por la planta de estudio[23].

La ecuación matemática que lo describe es :

$$d = \frac{E_{peak2}}{E_{peak1}} \quad (2.4)$$

Donde,

E_{peak1}: Error entre el primer peak máximo de PV y SP

E_{peak2}: Error entre el segundo peak máximo de PV y SP

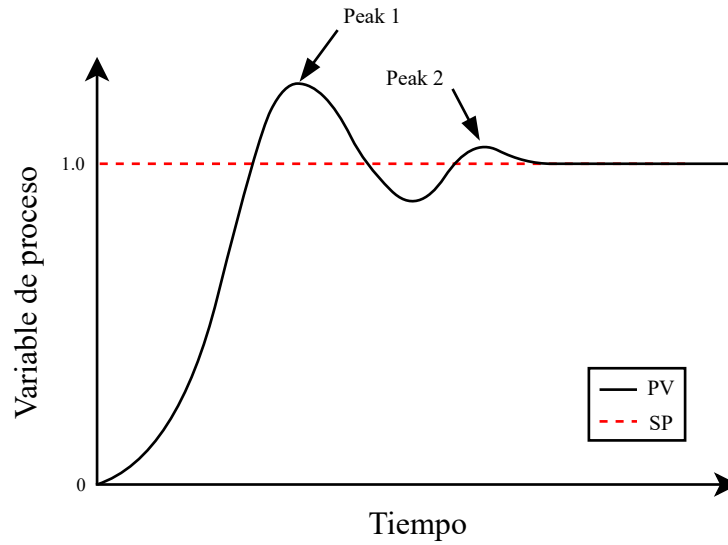


Fig. 2.6: Razón de Amortiguamiento

2.3.4. Sobrepaso

Valor definido por la diferencia entre el primer máximo de la variable de proceso y su valor estacionario.

La ecuación matemática que lo describe porcentualmente es :

$$\alpha = \frac{E_{peak}}{SP} * 100 \% \quad (2.5)$$

Donde,

E_{peak} : Valor del error máximo, ante cambio de referencia escalón

SP : Referencia del sistema

El sobrepaso aceptado depende del proceso. Es importante entender que a la larga producen un desgaste en los equipos por lo que se recomienda que su valor ronde en el 10 % o menos.

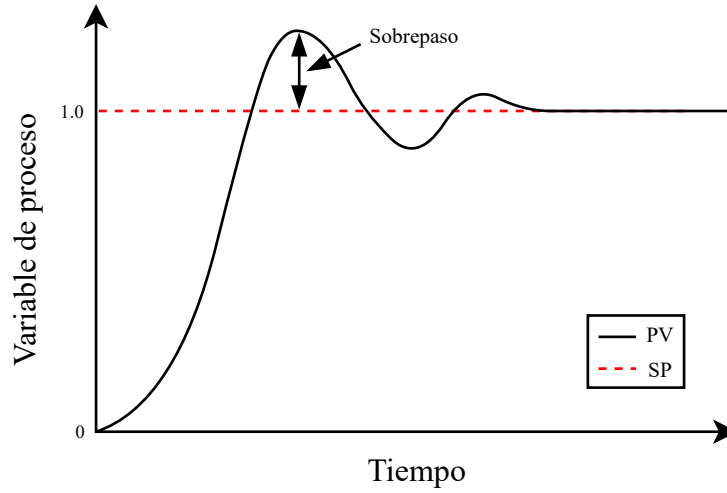


Fig. 2.7: Sobrepaso

2.3.5. Desviación Estándar

Es utilizada para medir la dispersión que poseen los datos del proceso con respecto a su valor promedio. En el ámbito de control industrial, una desviación baja indica que la variable de proceso mantiene sus valores cerca de la referencia, mientras que una desviación alta muestra que el proceso no tiene un rendimiento consistente.[24]

La ecuación matemática que la describe es :

$$\sigma = \sqrt{\frac{\sum_{t=0}^N ((SP - PV(t)) - Media(SP - PV(t)))^2}{N - 1}} \quad (2.6)$$

Donde,

$PV(t)$: Variable de proceso del lazo de control analizado

SP : Referencia del sistema

N : Cantidad total de datos

El análisis puede ser realizado tanto en estado estacionario como en la curva total de respuesta, siendo la primera forma útil para detectar oscilaciones constantes y errores en estado estacionario.

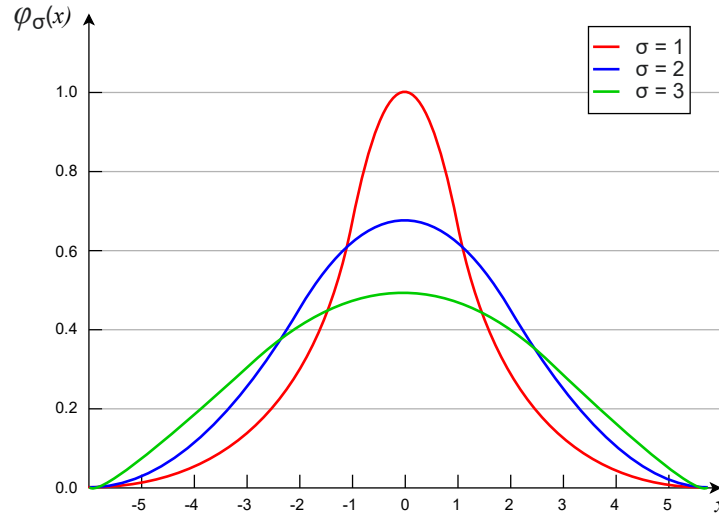


Fig. 2.8: Distribución de probabilidad a diferentes desviaciones estándar

2.3.6. Método Miao y Seborg

Método basado en el análisis de la función de autocorrelación muestral (ACF) para determinar si los lazos de control son excesivamente oscilatorios.

La ACF es un estimador estadístico que permite determinar la dependencia estadística de una serie de datos en el tiempo, en este contexto, se debe evaluar sobre datos de operación normal de la variable de error de control. Se obtienen los coeficientes de autocorrelación, los cuales oscilan entre $[-1, 1]$. [25]

La ecuación matemática que describe la ACF es :

$$\rho(k) = \frac{\sum_{t=1}^{N-k} (e(t) - \bar{e})(e(t+k) - \bar{e})}{\sum_{t=1}^N (e(t) - \bar{e})^2} \quad (2.7)$$

Donde,

$e(t)$: Valor de error medido en el instante t entre PV y SP

$\bar{e}$: Valor promedio del error entre PV y SP

N : Cantidad total de datos

k : Posición de coeficiente de auto-correlación

El método utiliza el índice de decaimiento R , el cual provee información de qué tan oscilatoria es la serie de datos en el tiempo.

La ecuación matemática que lo describe es:

$$R = \frac{a}{b} \quad (2.8)$$

Donde,

a : Distancia mínima desde el primer máximo a la línea recta que conecta los primeros dos mínimos

b : Distancia mínima desde el primer mínimo a la línea recta que conecta el coeficiente de correlación con retraso 0 y el primer máximo.

El cálculo de este índice se aplica sobre la señal construida por los distintos $\rho(k)$. Se puede ver de forma gráfica en la figura 2.9.

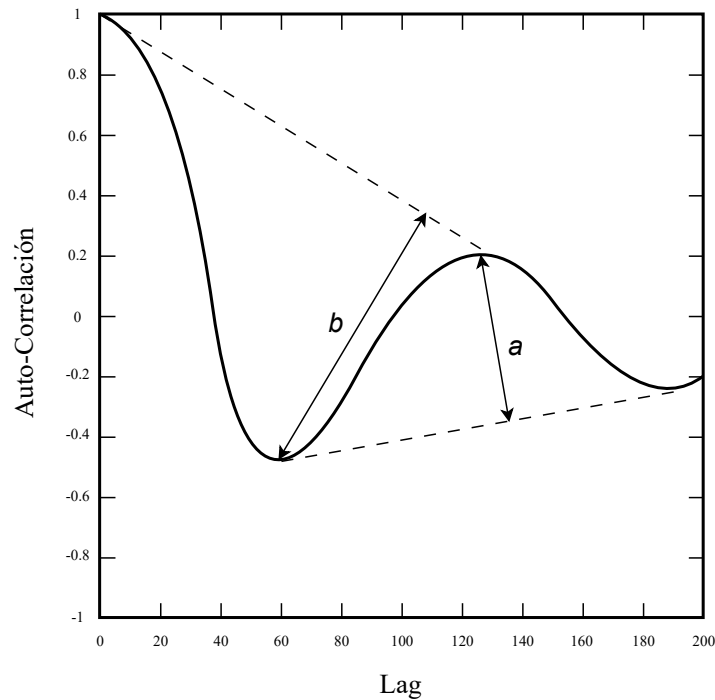


Fig. 2.9: Método Miao & Seborg

Es importante destacar que el ángulo formado entre la intersección de a con la unión de los mínimos forma un ángulo recto, al igual que la intersección respectiva de b .

Un valor de $R \geq 0,5$ implica un excesivo grado de oscilación, pues un valor menor corresponde a una razón de amortiguamiento menor a 0.25, lo cuál es aceptado en la literatura.

La metodología para obtener R no es trivial, es necesario tener por lo menos dos mínimos y un máximo, lo cuál se traduce en un mínimo de ciclos en la ACF. Esta información se aborda de mejor manera en el documento [25].

2.3.7. Índice de Harris

Es un método que compara el rendimiento real del controlador con el rendimiento teórico óptimo a partir de los datos de funcionamiento normal. Para su implementación, se necesita en primera instancia obtener el rendimiento teórico óptimo, el cual se logra operando bajo un control de mínima varianza (MVC). [26]

El MVC busca reducir el efecto de perturbaciones sobre la salida. Resulta de utilidad cuando la salida del proceso se encuentra contaminada por una perturbación estocástica que no puede eliminarse completamente, pero si reducir su varianza. El proceso es representado por un modelo ARIMA.[27][28]

El modelo ARIMA se utiliza para hacer predicciones sobre el futuro, determinando un modelo que permita hacer pronósticos de la variable basándose en valores pasados.[29]

Se propone entonces el Índice de Harris, basado en un MVC.

La ecuación matemática que lo describe es:

$$\eta_{mv} = \frac{\sigma_{mv}^2}{\sigma_y^2} \quad (2.9)$$

Donde,

σ_{mv} : Desviación de mínima varianza respecto de la referencia

σ_y : Desviación de salida respecto de la referencia

El índice oscila en un rango de [0,1], donde el mínimo es el peor desempeño y a medida que aumenta, indica un mejor desempeño.[30]

2.3.8. Porcentaje de tiempo de controlador en modo automático o manual

Este indicador nos señala cuánto porcentaje de tiempo ha estado el lazo operando en modo manual o automático, es decir, si está siendo controlado por un operador o por un sistema de control, como un PID, MPC, entre otros.

La estimación de este valor puede realizarse bajo ciertos supuestos, tales como que en modo manual la variable manipulada (MV) se mantiene constante, o bien, que en modo automático la referencia (SP) permanece fija.

En general, los DCS o SCADA disponibles en las plantas concentradoras registran el modo de operación de manera continua.

2.3.9. Error Cuadrático Integrado

El Error Cuadrático Integrado (ISE) evalúa la acumulación del error cuadrático a lo largo de todo el rango de datos.

La ecuación matemática que lo describe es:

$$ISE = \int_0^t e^2(t)dt \quad (2.10)$$

Donde,

$e(t)$: Valor de error medido en el instante t entre PV y SP

Debido a la elevación al cuadrado de los errores, se tiende a priorizar la reducción de los errores grandes, mientras que los errores pequeños tienen menor impacto.[31]

2.3.10. Error Absoluto Integrado

El Error Absoluto Integrado (IAE) se define como el promedio de las diferencias absolutas entre la referencia y los valores reales.

La ecuación matemática que lo describe es:

$$IAE = \int_0^t |e(t)|dt \quad (2.11)$$

Donde,

e(t): Valor de error medido en el instante t entre PV y SP

El IAE otorga la misma importancia a todos los errores. Debido a que utiliza la función valor absoluto, los errores positivos y negativos tienen el mismo impacto sobre la métrica.[31]

2.3.11. Error Absoluto Medio

Esta métrica estadística, llamada también MAE se define como el promedio del indicador IAE.

Una característica clave es su sensibilidad, pues al igual que el IAE, trata por igual todos los errores tomándolos en valor absoluto. Esta propiedad hace que no pueda reflejar adecuadamente el impacto de los grandes errores, sin embargo, entrega una visión equilibrada. [32]

La ecuación matemática que lo describe es:

$$MAE = \frac{1}{t} \cdot \int_0^t |e(t)| dt \quad (2.12)$$

Donde,

e(t): Valor de error medido en el instante t entre PV y SP

t: Periodo de observación

Es esperable cierto tipo de variabilidad, pero los aumentos notables deben ser al menos investigados. Se considera modesto un incremento del 5-10 %. Un número mayor es indicador de un problema.[33]

Para verlo de forma porcentual y normalizado, la ecuación matemática que lo describe es:

$$MAE = \frac{1}{t} \cdot \int_0^t |e(t)| \frac{100}{PV_{max} - PV_{min}} \quad (2.13)$$

2.4. *Procedimiento previo al análisis de datos históricos*

Para lograr un correcto análisis de datos históricos mediante un CLPM, debe existir conocimiento previo de distintos aspectos existentes.

2.4.1. *Estudio de la planta industrial*

El análisis del desempeño de lazos de control debe abordarse de manera específica según el tipo de planta o industria, ya que cada proceso industrial tiene características y necesidades únicas que determinan los criterios de evaluación y optimización. Por ejemplo, en una refinería, donde se manejan procesos químicos complejos y materiales peligrosos, el control debe ser extremadamente preciso y rápido, pues las desviaciones, incluso pequeñas, pueden resultar en pérdidas económicas significativas o, peor aún, en riesgos de seguridad considerables. En contraste, en una operación minera, ciertos procesos como el control de nivel en tanques pueden tolerar mayor variabilidad sin comprometer significativamente la operación o la seguridad del personal.

La dinámica del proceso también juega un papel fundamental en cómo se debe abordar el control. Los procesos térmicos, característicos de muchas operaciones industriales, suelen presentar dinámicas lentas con retardos significativos, mientras que los procesos de flujo tienden a responder más rápidamente. Esta distinción es crucial al momento de sintonizar los controladores y establecer métricas de desempeño. Además, la interacción entre variables varía significativamente según el tipo de planta. En una refinería, por ejemplo, existe una alta interrelación entre diferentes variables del proceso, lo que hace que el control sea más complejo y requiera un enfoque más sofisticado. En cambio, en procesos más simples, los lazos de control pueden operarse de manera más independiente.

Los criterios de evaluación deben ajustarse a la realidad de cada planta. Por esto, antes de implementar cualquier sistema CLPM, es fundamental realizar un estudio profundo del proceso específico, comprender sus requerimientos particulares y establecer criterios de desempeño apropiados. Esto permite identificar qué lazos son verdaderamente críticos y requieren mayor atención, además de establecer objetivos de mejora que sean realistas y significativos para la operación específica.

2.4.2. *Análisis de diagramas P&ID*

Ya teniendo conocimiento del proceso, se procede a entender cómo funciona la planta industrial y la automatización de la misma con los diagramas de cañerías e instrumentación (P&ID). Estos diagramas muestran la cantidad y los distintos tipos de lazos de control existentes en la planta, y es fundamental saber dónde está cada uno en cada parte del proceso.

Identificar los lazos permite entender cuáles son los que afectan de manera significativa y cómo se relacionan unos con otros, para así en caso de problemas, identificar si existe una relación causa-efecto.

En algunos casos, puede ser necesario actualizar la información contenida en los diagramas P&IDs debido a cambios en la instrumentación del proceso.

2.4.3. *Determinación de lazos críticos*

Una vez identificados los lazos de control, es necesario determinar las partes cruciales para el éxito del sistema. Se debe examinar todas las secciones de la planta y determinar cuáles de ellas, en caso de fallar, serían críticas.

Este análisis permite definir los márgenes de error del usuario frente a la información entregada por la plataforma de CLPM.

2.4.4. *Adquisición de Datos de operación*

Seleccionados los lazos más importantes, se debe solicitar una base de datos histórica de éstos.

Un aspecto crítico que frecuentemente se pasa por alto es la necesidad de mantener consistencia en el formato de los datos recolectados. Cuando los datos provienen de diferentes fuentes, como diversos sistemas de control distribuido (DCS), PLCs, o sistemas SCADA, pueden presentarse en formatos distintos, con diferentes períodos de muestreo, o incluso con unidades de medición no estandarizadas.

La sincronización temporal de los datos también es crucial. Por ejemplo, si se analiza la interacción entre una variable de flujo y una de temperatura, se necesita que ambas mediciones estén alineadas en el tiempo y con la misma frecuencia de muestreo. Un desajuste en la

temporalidad de los datos puede llevar a conclusiones erróneas sobre la dinámica del proceso y las relaciones causa-efecto entre variables.

El período de muestreo debe seleccionarse cuidadosamente según la dinámica del proceso que se está monitoreando. Un proceso térmico lento podría requerir muestras cada minuto, mientras que un control de flujo rápido podría necesitar muestras cada segundo o incluso más frecuentes. Sin embargo, todos estos datos deben poder integrarse en un formato común para su análisis.

La estandarización de unidades es otro aspecto fundamental. Es común encontrar que diferentes sistemas expresan las mismas variables en unidades distintas; por ejemplo, temperaturas en Celsius y Fahrenheit, o presiones en psi y bar. La conversión y normalización de estas unidades debe realizarse de manera sistemática y documentada para evitar errores de interpretación.

También es esencial considerar la calidad de los datos recolectados. Los datos deben pasar por un proceso de validación que identifique y maneje apropiadamente valores atípicos, períodos de mantenimiento, paradas de planta, o fallas en sensores. Estos eventos deben marcarse adecuadamente en los datos para no contaminar el análisis posterior del rendimiento del control.

3. DISEÑO DE LA INTERFAZ DE VISUALIZACIÓN Y ANÁLISIS AUTOMÁTICO DE DESEMPEÑO

Este capítulo describe el proceso de diseño de la interfaz de visualización y análisis automático de desempeño de lazos de control. En primer lugar, se muestran los indicadores de desempeño que se seleccionan para utilizar en el análisis de desempeño, explicando las razones por las que ciertos indicadores se vuelven más relevantes que otros en esta aplicación. Posteriormente, se explica en profundidad el diseño del programa de cálculo de indicadores, describiendo todas las secciones del código y su funcionamiento. Seguido, se habla de la visualización de los resultados del programa en una interfaz gráfica creada en el software LabVIEW. Finalmente, se crea una pequeña guía de usuario, detallando la forma correcta de utilizar el software CLPM integrado.

Para el diseño del programa, el proceso inicia con la recopilación de los datos de la planta. Estos datos se importan al programa, donde se lleva a cabo su procesamiento y análisis. En esta etapa, se aplica el acondicionamiento de datos y el cálculo de métricas específicas que permiten extraer información relevante sobre el funcionamiento de la planta.

Una vez procesados los datos, se obtienen los índices de desempeño calculados. Estos índices se exportan en reportes para posteriormente ser leídos por la interfaz. En la interfaz, los índices y la data de la planta son leídos y utilizados para generar gráficos e indicadores visuales que permiten al usuario monitorear el sistema de forma clara.

Este flujo de trabajo combina la capacidad analítica del programa con las herramientas gráficas de la interfaz, logrando una integración eficiente entre el análisis de datos y su visualización interactiva.

Este modelo se puede ver simplificado en el diagrama de flujo de la figura 3.1.

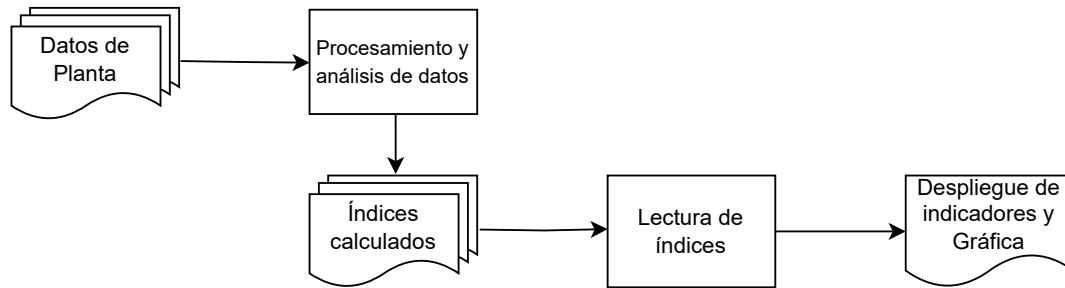


Fig. 3.1: Diagrama de flujo para diseño de programa CLPM

3.1. Indicadores seleccionados

Para evaluar el rendimiento de un lazo de control, generalmente se utilizan índices calculados a partir de su respuesta a un cambio de referencia o de su respuesta total a lo largo del tiempo. Estos indicadores son útiles para la futura sintonización, ya que permiten identificar posibles errores.

En la sección 2.3 se estudiaron los indicadores utilizados en las distintas empresas y trabajos para un CLPM ideal, sin embargo, no todos son esenciales para que un operador sepa que el lazo de control requiere corrección.

A continuación, se hará un estudio de cada uno de estos indicadores para determinar cuáles son más relevantes para el análisis de desempeño de lazos de control de una planta concentradora, y que sean factibles de implementar en este caso.

Tiempo de Subida

El tiempo de subida es un valor a interpretar, ya que lo que se considera un sistema rápido depende del proceso que se esté evaluando. Ejemplificando, no es lo mismo tardar dos segundos en alcanzar una velocidad de un motor que en activar una alarma de seguridad.

Sin embargo, al sintonizar un lazo de control se busca lograr la máxima rapidez alcanzable en la respuesta del sistema. Un tiempo de transición más largo generalmente indica pérdidas, ya que no se está trabajando al nivel que es requerido.

Por tanto, este indicador es necesario.

Tiempo de Asentamiento

Nuevamente, su interpretación depende de la industria. En una empresa de alimentos es vital reducir este tiempo, ya que se pueden producir productos que no cumplan con estándares legales o de calidad.

El tiempo de asentamiento nos dice el periodo de tiempo donde el sistema se mantuvo produciendo un producto que no cumple con lo solicitado, resultando en pérdidas.

Por tanto, este indicador es necesario.

Razón de Amortiguamiento

Este coeficiente habla de la oscilación del lazo. Un sistema oscilante nunca es bueno en términos de eficiencia energética y de cuidado de maquinaria.

Para entender lo anterior, en un sistema oscilante se puede actuar sobre una válvula de modo que se abra o cierre cada 0.1 segundos, dañando la maquinaria al largo plazo.

Por tanto, este indicador es necesario.

Sobrepaso

El sobrepaso es el exceso de energía utilizada cuando el sistema supera temporalmente el valor deseado al cambiar la referencia. Este fenómeno ocurre cuando la respuesta transitoria del sistema excede el valor final antes de estabilizarse.

Se busca minimizar el sobrepaso, ya que implica que los equipos operan a niveles máximos innecesarios, lo que no solo puede afectar la precisión del sistema, sino también su eficiencia. El exceso de energía utilizado durante el sobrepaso no es necesario para mantener el sistema en su referencia estable, y puede conducir a un mayor desgaste de los componentes, un consumo energético innecesario y una disminución de la vida útil del equipo.

Por tanto, este indicador es necesario.

Desviación Estándar

La desviación estándar es un indicador para evaluar la dispersión de los datos alrededor de su media. Nos proporciona una medida de la variabilidad dentro de un conjunto de datos.

Su evaluación es especialmente importante cuando se considera solo el promedio para realizar comparaciones, ya que un valor máximo y un valor mínimo podrían cancelarse entre sí en el cálculo del promedio. Sin embargo, estos valores extremos se reflejarán claramente en la desviación estándar, lo que permite identificar el grado de dispersión y tener una visión más precisa sobre la estabilidad de los datos.

Por tanto, este indicador es necesario.

Método Miao y Seborg

Este método, al igual que la razón de amortiguamiento, tiene como objetivo determinar si los lazos de control presentan un comportamiento excesivamente oscilatorio.

El indicador se basa en matemáticas avanzadas, ya que requiere un estimador estadístico para obtener coeficientes, de los cuales se genera un indicador final. Este indicador, a su vez, puede relacionarse con la razón de amortiguamiento del sistema.

El indicador final ofrece una mayor precisión pero no ofrece información sustancialmente diferente a la que se obtiene con la razón de amortiguamiento, y su cálculo resulta más complejo.

Su utilización no cambia la información proporcionada por la razón de amortiguamiento, pero añade una capa de complejidad al análisis.

Por tanto, este indicador no es necesario.

Índice de Harris

El índice de Harris fue diseñado con el objetivo de reducir la varianza en el lazo de control. Utiliza principios matemáticos derivados de un control de mínima varianza, buscando atenuar las perturbaciones estocásticas que, aunque no se pueden eliminar por completo, sí pueden reducirse. Este índice evalúa el desempeño del lazo frente a tales perturbaciones.

Aunque proporciona información valiosa, precisa y útil, su complejidad es considerable. No solo implica un modelo matemático avanzado para predecir valores, sino que también presenta desafíos en términos de cálculo.

Al implementarlo en código, se observa que el tiempo de ejecución para calcular el modelo ARIMA puede superar los 5 minutos, lo que resulta excesivo y poco práctico cuando se manejan grandes volúmenes de datos y lazos, ya que representa un tiempo valioso en el análisis.

Además, existen estudios que sugieren que la complejidad de este índice no siempre está justificada, especialmente en contextos como el mencionado, donde su implementación puede no ofrecer suficientes beneficios en relación con el tiempo y los recursos invertidos.[19]

Por tanto, este indicador no es necesario

Porcentaje de tiempo controlador en modo automático o manual

Es fundamental determinar si el lazo de control está operando en modo automático bajo un controlador PID, o si está siendo operado por un usuario de forma manual. Esta distinción es crucial, ya que la interpretación de los datos depende del tipo de control utilizado.

Si un sistema permanece en modo manual durante largos períodos, puede ser un indicio de que la sintonización del lazo no es adecuada, lo que requiere la intervención de un operador. También puede señalar que el sistema de control no está respondiendo correctamente, sugiriendo un problema en la maquinaria que debe ser atendido.

Además, para evaluar la correcta sintonización del controlador, es crucial hacerlo únicamente cuando esté en funcionamiento. Evaluarlo en otros momentos podría distorsionar los resultados y llevar a una interpretación incorrecta de la información.

Por tanto, este indicador es necesario.

ISE - IAE - MAE

Las tres medidas se utilizan para obtener información sobre la acumulación del error a lo largo de todo el rango de datos.

Si se tiene una respuesta rápida y un tiempo de estabilización corto, el índice ISE puede ser más adecuado, ya que refleja el efecto de grandes errores. En cambio, si se tiene una respuesta más suave y un menor esfuerzo de control, el índice IAE podría ser la mejor opción, ya que otorga la misma importancia a todos los errores sin importar su magnitud, pero puede dejar de lado problemas graves.

Por otro lado, el índice MAE entrega la información del IAE pero tiene la ventaja de poder ser estandarizado, lo que permite comparar su valor entre lazos de control que utilicen distintos métodos de muestreo o ventanas de tiempo, facilitando una evaluación más flexible y versátil.

Dado que la desviación estándar refleja los cambios bruscos en la señal, se opta por utilizar el Error Absoluto Medio como el indicador más adecuado para analizar la situación. Esto se debe a que el MAE ofrece una visión general y estandarizada del error, lo que lo convierte en una herramienta apropiada para evaluar el comportamiento del sistema.

3.1.1. Tabla Resumen

Finalmente, la selección de indicadores se muestra en la tabla 3.1:

Tab. 3.1: Indicadores de lazos de control a usar

Indicadores	¿Es requerido para el análisis?
Tiempo de Subida	Sí
Tiempo de Asentamiento	Sí
Razón de Amortiguamiento	Sí
Sobrepaso	Sí
Desviación Estándar	Sí
Método Miao & Seborg	No
Índice de Harris	No
% Tiempo controlador en modo automático o manual	Sí
ISE	No
IAE	No
MAE	Sí

3.2. Programa de cálculo de indicadores

Para la creación de un programa CLPM, el lenguaje de programación Python destaca por su capacidad de procesamiento masivo de datos y su amplia variedad de bibliotecas especializadas en análisis estadístico, manipulación de datos y aprendizaje automático.

En este contexto, su uso permite optimizar el tratamiento de los datos provenientes de una planta, facilitando la obtención de resultados precisos y eficientes para la toma de decisiones.

3.2.1. Estandarización de los datos

Es importante tener en cuenta que los datos que se ingresen al programa deben cumplir con un formato estandarizado para que el sistema pueda procesar correctamente los archivos. Para garantizar que los datos sean legibles, se realiza una revisión previa de la información proporcionada.

Como estándar, se ha definido que la información debe estar contenida en un archivo de formato .xlsx (Excel) y debe tener cuatro columnas clave: Fecha, PV, SP, y MV o CO. En la figura 3.2 se muestra un ejemplo de formato estandarizado de datos en Excel que puede ser procesado por el programa CLPM.

fechas	PV	CO	SP	Time
7/31/24 11:47 AM	63,53	46,00	952,93	0,00
7/31/24 11:47 AM	63,04	46,00	945,46	1,00
7/31/24 11:47 AM	62,55	46,00	938,00	2,00
7/31/24 11:47 AM	61,95	46,00	930,53	3,00
7/31/24 11:47 AM	61,43	46,00	923,07	4,00
7/31/24 11:47 AM	60,90	46,00	916,27	5,00

Fig. 3.2: Ejemplo de formato estandarizado de datos en Excel.

Cada empresa puede utilizar nombres distintos para sus columnas en los archivos. Por lo tanto, manualmente se debe editar el nombre de cada columna junto a la ubicación de los archivos. Además, se debe definir el tiempo de muestreo.

3.2.2. Código

El programa inicia importando librerías esenciales para la manipulación de datos, cálculos matemáticos y generación de gráficos. Se configuran las variables de entrada que el usuario debe personalizar según la estructura del archivo Excel a analizar, incluyendo los nombres de las columnas correspondientes al tiempo, PV, SP y MV, así como el tiempo de muestreo.

El usuario debe especificar la ubicación del archivo Excel que contiene los datos. Este archivo se carga en un DataFrame, permitiendo un manejo eficiente y flexible de la información. Un DataFrame es una estructura de datos bidimensional que organiza datos en filas y columnas, similar a un objeto o variable matricial de MATLAB.

Posteriormente, el programa identifica tramos donde la referencia permanece constante durante un número mínimo de muestras consecutivas. Este número fue elegido experimentalmente, notando cuántos valores eran necesarios dependiendo del tiempo de muestreo del archivo.

El núcleo del análisis se basa en la función 'indicadores()', que calcula los indicadores de desempeño del lazo mencionados en la sección 3.1 y que se detalla su cálculo a continuación :

- **Tiempo de Subida** Definido entre las líneas 28 y 41 del Anexo 7.1. Se calcula identificando los valores de la variable de proceso (PV) correspondientes al 10 % y 90 % de la ganancia del escalón aplicado. Para determinar el instante en que se alcanzan estos valores, se utiliza la función 'np.interp()', que permite interpolar el tiempo asociado a un valor específico de PV. En este caso, se invierten los ejes: se toma como eje X la PV y como eje Y el tiempo, ya que se busca estimar el instante en que se alcanza un determinado valor de la señal. Esta interpolación es válida porque 'np.interp()' entrega el primer cruce con el valor objetivo, lo cual es coherente con la definición del tiempo de subida. Este enfoque no es aplicable al cálculo del tiempo de asentamiento, como se explica más adelante.
- **Tiempo de Asentamiento** Definido entre las líneas 43 y 58 del Anexo 7.1. Este indicador se obtiene estableciendo un margen de tolerancia en torno a la referencia, considerando el ruido inherente al sistema. En este caso, se define un margen del $\pm 2\%$,

representado por las variables ' PV_{98} ' y ' PV_{102} ', aunque este valor puede ser modificado por el usuario. A diferencia del tiempo de subida, aquí no se utiliza interpolación, ya que no se busca el primer cruce con el margen sino que el momento en que la señal permanece dentro de él durante un periodo sostenido. Por esta razón, el cálculo se realiza sobre datos muestreados.

- **Razón de Amortiguamiento** Definido entre las líneas 61 y 122 del Anexo 7.1. Requiere primero determinar si el cambio de referencia es positivo o negativo. Posteriormente, se identifican máximos y mínimos locales en la señal, y se calcula una serie de promedios móviles entre ellos. Esta técnica permite suavizar la señal y obtener una tendencia general sobre la cual se superpone el ruido. Una vez identificada esta tendencia, se verifica si existen al menos dos picos consecutivos, y se calcula la razón entre ellos. Este procedimiento permite estimar la amortiguación del sistema, incluso en presencia de ruido.
- **Sobrepaso** Definido entre las líneas 125 y 132 del Anexo 7.1. Se considera tanto el overshoot como el undershoot, dependiendo de la dirección del cambio. La dirección ya ha sido determinada en el cálculo de la razón de amortiguamiento, y se utiliza para identificar el máximo o mínimo relevante. El sobrepaso se expresa como un porcentaje.
- **Desviación Estándar** Definido entre las líneas 135 y 136 del Anexo 7.1. Se calcula directamente utilizando la función '`std()`' de Python.
- **Error Absoluto Medio (MAE)** Definido entre las líneas 139 y 145 del Anexo 7.1. Este indicador se basa en la fórmula presentada en la sección 2.3.11 de esta memoria. Dado que se trabaja con datos muestreados, la integral se reemplaza por una sumatoria discreta. El resultado se normaliza respecto al rango de la PV, permitiendo comparar resultados entre distintos lazos.
- **Modos de Operación** Definido entre las líneas 196 y 198 del Anexo 7.1. Se determina el porcentaje de tiempo en que el sistema operó en modo automático o manual. Para ello, se calcula la proporción de muestras en las que el SP se mantuvo constante, entendiendo esos periodos donde el controlador estuvo activo en modo automático.

Una característica destacada del código es su capacidad para diferenciar entre los modos de operación manual y automático. El análisis determina el porcentaje de tiempo que el sistema estuvo en cada modo, permitiendo una comprensión completa del contexto operativo. Si el sistema estuvo en modo manual durante todo el análisis, el programa notifica que no es posible calcular los indicadores. El programa no detecta modos tracking y apagado.

En cuanto a la visualización de resultados, el programa genera gráficos detallados que muestran la evolución temporal de la PV, el SP y una curva representativa de los datos basada en promedios móviles para que el ruido no afecte el entendimiento de la respuesta del sistema ante diferentes estímulos. La selección del ajuste de curva no es una decisión trivial, por eso debe entenderse y estudiarse el comportamiento de los datos.[34]

Estas gráficas permiten al usuario interpretar visualmente el comportamiento del sistema y detectar anomalías o áreas de mejora.

Todos los resultados del análisis se almacenan en archivos Excel generados automáticamente. Estos reportes facilitan la documentación, permitiendo realizar análisis comparativos y evaluar la efectividad de cambios en la sintonización o en la operación.

El código también contempla escenarios donde no existen cambios mayores o iguales a un escalón unitario en el SP, situación en la cual notifica que no es posible realizar un análisis adecuado. Esto asegura que los resultados generados sean siempre relevantes y basados en datos representativos del comportamiento real del sistema.

Para mayor detalle, a continuación se muestra un diagrama de flujo del funcionamiento del programa y el código se encuentra en la sección Anexos.

3.2.3. Diagrama de Flujo

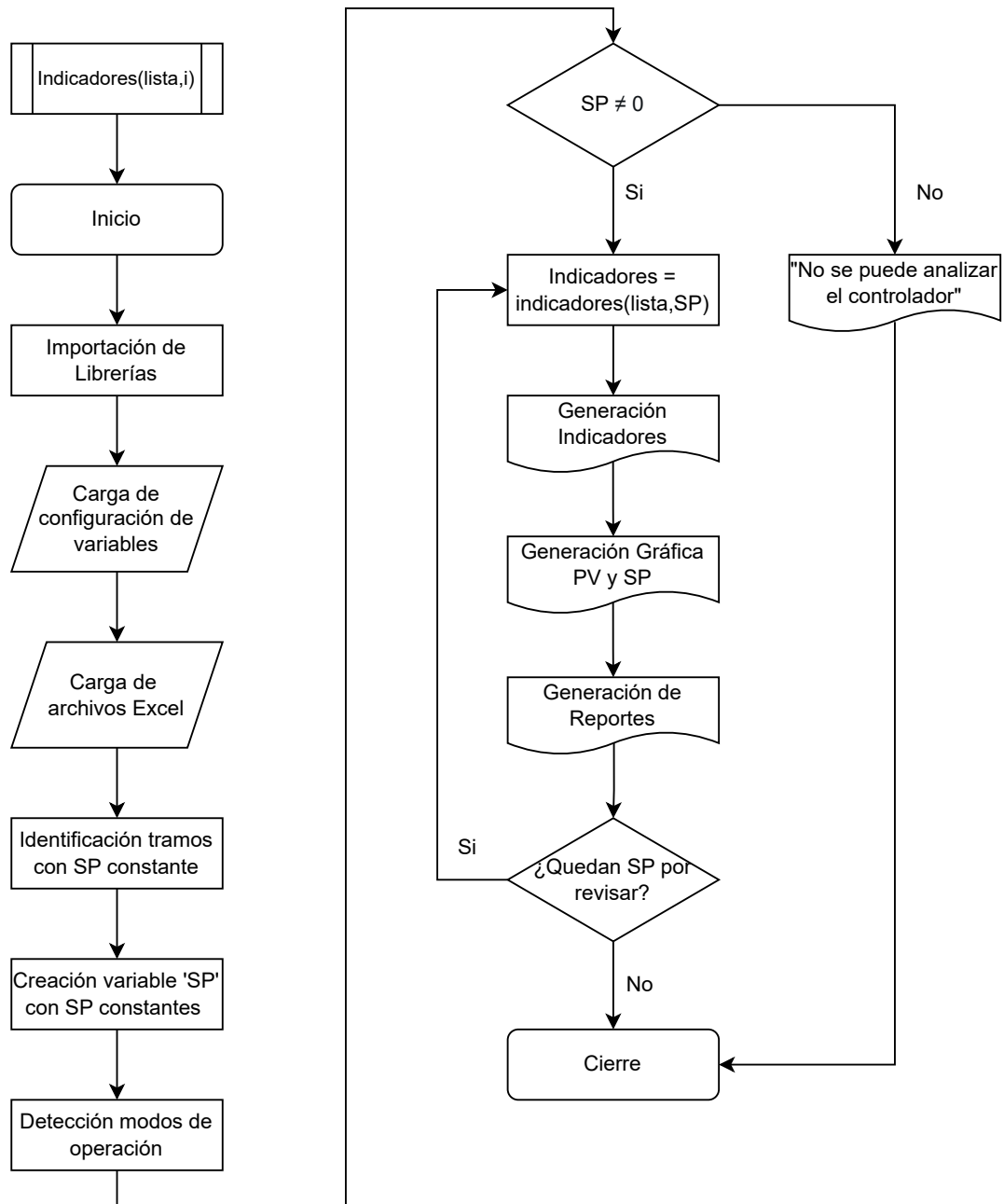


Fig. 3.3: Diagrama de flujo de programa

3.3. Visualización de Proceso

Para la visualización del proceso, el comportamiento de los lazos de control y los indicadores calculados se utilizará el software LabVIEW.

A diferencia de Python, LabVIEW es una plataforma dinámica que se especializa en ofrecer soluciones diversas para cada proceso. Para este proyecto se desarrolla la interfaz de dos procesos de una planta concentradora.

3.3.1. Descripción de Proceso y pasos previos

Planta Concentradora

En una planta típica de procesamiento de minerales que pueden ser recuperados mediante el proceso de flotación, (por ejemplo, una mina de cobre), el proceso comienza con la extracción de la roca desde la mina, y posteriormente, inicia su tratamiento en la planta.

A continuación se detallan las etapas principales del proceso:

- Chancado Primario: En esta etapa, la roca es triturada utilizando un chancador para reducir su tamaño. Posteriormente, es llevada a una pila (Stockpile) mediante correas transportadoras para seguir su proceso.
- Molienda SAG: La roca es sometida a molienda mediante impactos y fricción, utilizando las rocas y bolas de acero dentro del molino SAG.
- Molienda Convencional: El material se muele nuevamente utilizando bolas de acero o cerámica, lo que permite obtener una molienda más fina. En esta etapa se emplea un Molino de Bolas.
- Chancado de Pebbles: Las piedras más duras provenientes del molino SAG se trituran para mejorar la eficiencia de la molienda. Posteriormente, se reincorpora el material al proceso de Molienda Convencional.
- Flotación Rougher: En esta etapa, se separan los minerales valiosos de la roca. Se añaden reactivos químicos que hacen que los minerales valiosos se adhieran a las burbujas de aire inyectadas en la pulpa. Estas burbujas, con los minerales adheridos, flotan hacia

la superficie, formando una pulpa de concentrado del mineral, en este caso, un 20 % de cobre.

- Remolienda: El material que no alcanzó el tamaño deseado en las etapas anteriores se vuelve a moler, y se reincorpora al proceso en la etapa de Flotación Cleaner.
- Flotación Cleaner: Esta etapa tiene como objetivo mejorar la pureza del concentrado obtenido en la flotación Rougher, eliminando impurezas adicionales.
- Flotación Scavenger: Se recuperan los minerales valiosos que no fueron capturados en la etapa anterior, maximizando la recuperación antes de que los residuos finales sean descartados.

En la figura 3.4 presentada a continuación, se muestra lo descrito anteriormente:

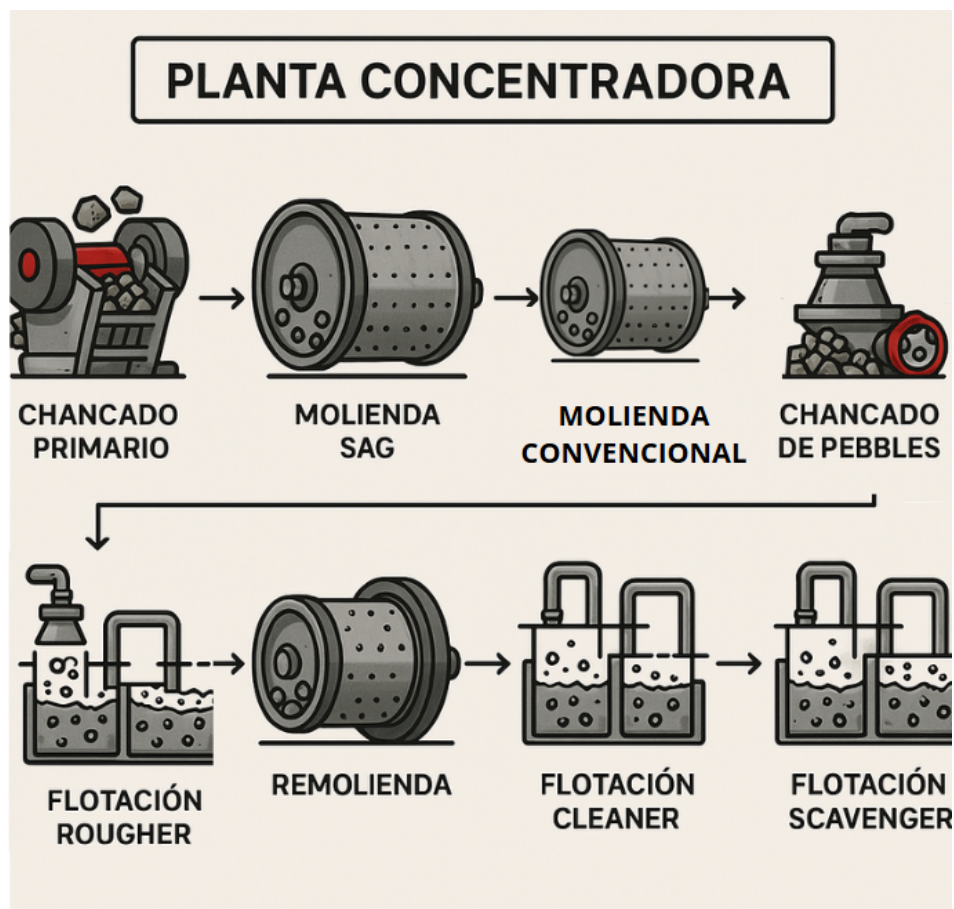


Fig. 3.4: Diagrama de Planta Concentradora

Se realiza un análisis de los P&ID de la planta concentradora en cuestión. Luego, se definen los procesos más relevantes, basándose en su impacto en el resto de la planta. Es importante determinar si el funcionamiento de ciertos procesos es más crítico que otros, y a partir de ahí, establecer los lazos de control a estudiar.

Se recogen datos operacionales favoreciendo un tiempo de muestreo pequeño para los datos más relevantes, con el objetivo de no perder información crucial para el análisis.

Finalmente, este proyecto se enfoca en los procesos de Molienda SAG y flotación Rougher. Sin embargo, el análisis de los datos de los demás procesos es análogo al que se muestra a continuación.

3.3.2. Interfaz en LabVIEW

Una vez realizado el estudio exhaustivo de la planta, se procede a diseñar la interfaz, buscando replicarla lo más fielmente posible en sus dos procesos.

El primer proceso a analizar es el del molino SAG.

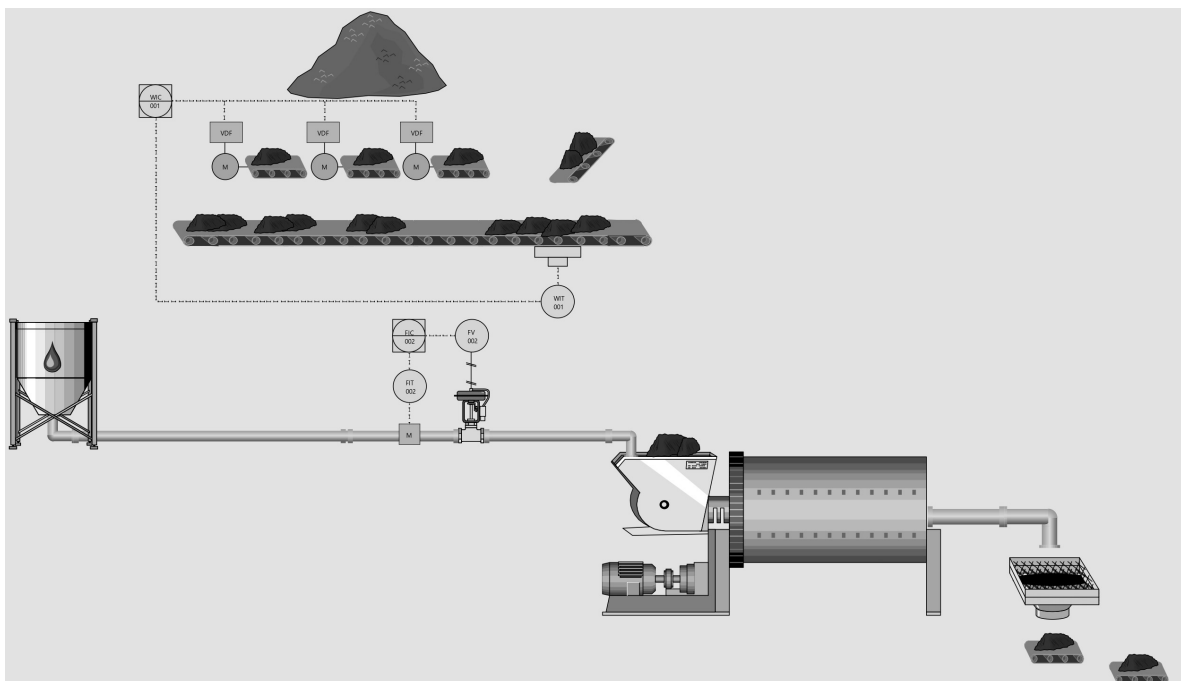


Fig. 3.5: Diagrama P&ID simplificado de Proceso Molino SAG

Este proceso cuenta con dos lazos que se analizan, uno de peso (WIC) y otro de flujo (FIC).

El lazo WIC mide el peso en una correa y envía una señal de control de velocidad al motor de los feeders ubicados debajo del stockpile. Este sistema está compuesto por un transmisor e indicador de peso, un controlador de peso, actuadores (en este caso, los variadores de frecuencia) y el motor de los feeders. El transmisor actúa como la variable de proceso, que se envía al controlador. A partir de esto, se obtiene una señal de control que se envía al motor del feeder, ajustando su velocidad con el fin de controlar el flujo másico de entrada al molino SAG.

Por otro lado, el lazo de flujo regula la cantidad de agua que ingresa al molino SAG a través de una válvula de control. Este sistema incluye un transmisor e indicador de flujo, un controlador de flujo y la válvula de control. El transmisor mide el flujo y envía la PV al controlador, que entrega una señal de control. Esta señal es enviada a la válvula, modificando el flujo del agua en el proceso.

El segundo proceso a analizar es el de flotación Rougher.

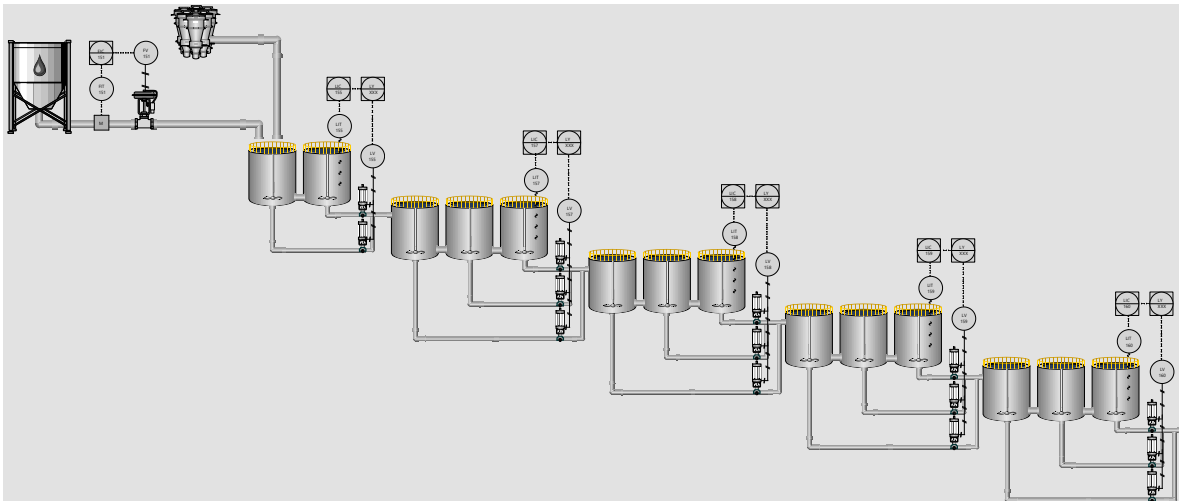


Fig. 3.6: Diagrama P&ID simplificado de Proceso Flotación Rougher

Este proceso consta de seis lazos, uno de flujo y cinco de nivel.

El lazo de flujo regula la cantidad de agua que ingresa a las celdas de flotación Rougher a través de una válvula de control. Este sistema incluye un transmisor e indicador de flujo, un

controlador de flujo y la válvula de control.

Por su lado, los cinco lazos de nivel dispuestos de manera secuencial controlan las válvulas dando para permitir el paso del flujo de material. Cada uno de estos sistemas está compuesto por un sensor de nivel, un controlador de nivel y un actuador del tipo cilindro neumático que actúa sobre la válvula dando de nivel. Este actuador incluye un posicionador.

El transmisor mide el nivel del estanque, que actúa como la variable de proceso y se envía al controlador. Este, a su vez, genera una señal de control que es enviada a la válvula dando, la cual se convierte en la variable manipulada. Cada uno de los lazos de nivel funciona de manera independiente, pero con el mismo principio de operación.

3.4. *Software CLPM Integrado*

Una vez que cada programa está funcionando de manera independiente, el siguiente paso es trabajar en la conectividad entre el programa y LabVIEW.

Es importante resaltar que se utiliza la versión LabVIEW 2019 junto con Python 3.6 para garantizar su compatibilidad. Este fue uno de los desafíos que se presentaron durante el proceso de integración.

Se logra configurar LabVIEW para leer archivos .xlsx, lo que permite que el programa CLPM genere reportes con la información a entregar y LabVIEW sea capaz de leerlos. Además, se lee en LabVIEW los archivos con la data utilizada. Esta información es luego presentada en una interfaz visual que muestra tablas y gráficos.

A continuación, se presenta el diagrama de bloques de la programación de LabVIEW que ilustra la conexión entre los sistemas.

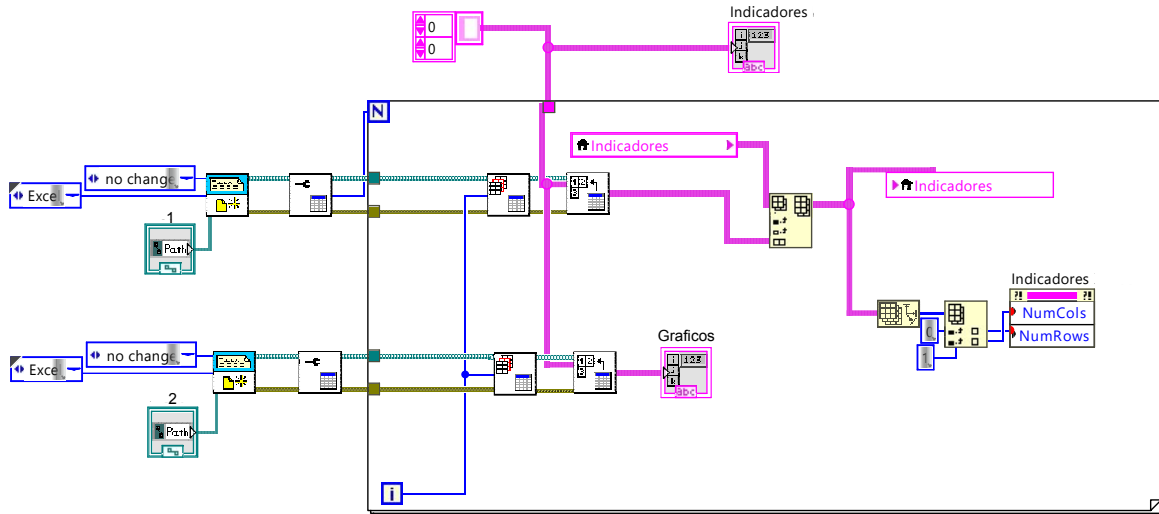


Fig. 3.7: Diagrama de Bloques de programación para obtener Indicadores

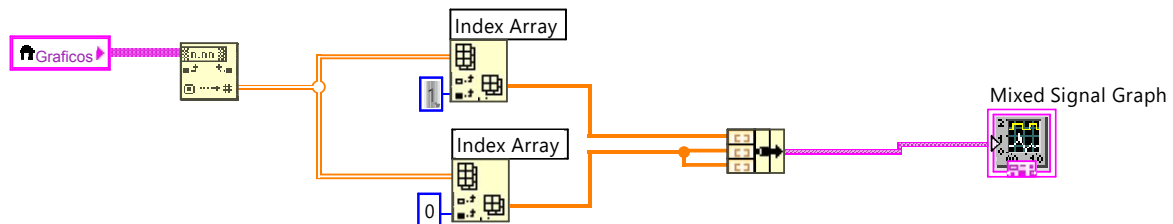


Fig. 3.8: Diagrama de Bloques de programación para obtener Gráficos de respuesta

En la figura 3.7 se muestra la lectura de dos archivos tipo .xlsx (1 y 2). El primero es el archivo que contiene los reportes generados por el programa CLPM y el segundo es la data del lazo a graficar.

Ambas lecturas iteran sobre los datos y producen tablas en LabVIEW con la información, siendo 'Indicadores' la tabla mostrada en la interfaz y 'Gráficos' la tabla de datos a graficar. Todo esto se muestra en la figura 3.8.

La HMI creada se puede observar en la figura 3.9 y 3.10.

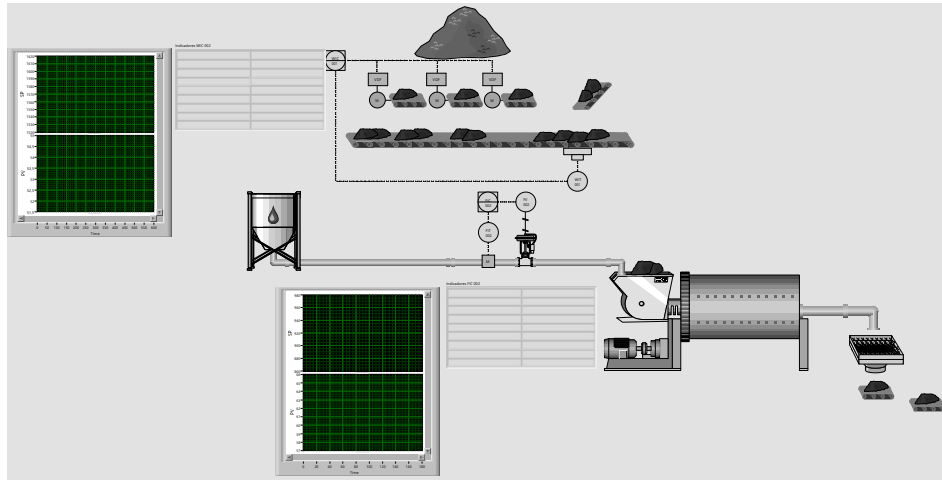


Fig. 3.9: Interfaz final Molino SAG

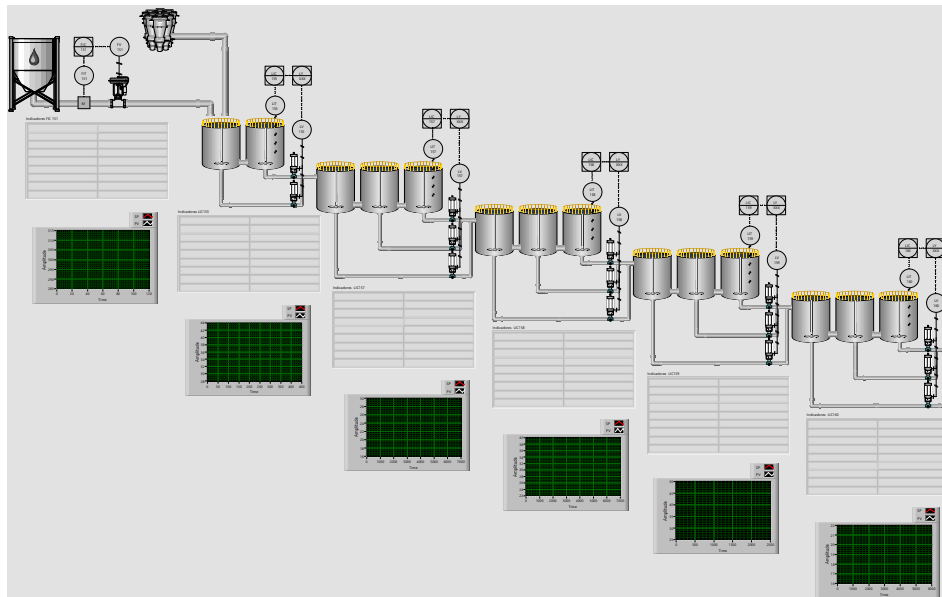


Fig. 3.10: Interfaz final Flotación Rougher

Para mayor información de la conexión y uso de Python en conjunto con LabVIEW, revisar [35][36][37].

3.5. Guía de usuario

La plataforma CLPM está compuesta por dos módulos principales:

- **Módulo de análisis en Python:** Encargado de importar, procesar y analizar los datos históricos. Calcula automáticamente los indicadores de desempeño y genera reportes en formato Excel y gráficos en PNG.
- **Interfaz gráfica en LabVIEW:** Permite visualizar los resultados de forma clara e interactiva, facilitando el monitoreo de los lazos de control.

3.5.1. Flujo de trabajo general

La plataforma sigue el siguiente flujo de trabajo:

1. Importación de datos desde archivos Excel con datos provenientes de planta.
2. Ejecución del módulo de análisis en Python.
3. Selección de tramo más significativo según criterio de usuario.
4. Creación archivo excel complementario con datos del tramo seleccionado.
5. Ejecución del módulo de análisis en Python que crea archivo con indicadores correspondientes.
6. Importación datos generados a LabVIEW
7. Ejecución de interfaz en LabVIEW.

3.5.2. Instrucciones de Uso

Paso 1: Preparación de los datos

El archivo de entrada debe estar en formato `.xlsx` y la estandarización de los datos se ve a mayor detalle en la subsección 3.2.1.

Paso 2: Ejecución del programa

Al iniciar el script de Python, se solicitará que el usuario revise la concordancia de los siguientes ítems:

- Nombres de las columnas

- Tiempo de muestreo
- Ruta del archivo Excel

Cumplíéndose lo anterior, se simula y se generan gráficos de PV y SP con sus respectivos indicadores.

Paso 3: Selección tramo y creación archivo complementario

Debido a la naturaleza de la conexión LabVIEW y Python (desarrollada en 3.4), se debe seleccionar de todos los tramos, uno característico que se desee presentar en la interfaz.

Se genera un nuevo archivo con los datos del tramo seleccionado. Este nuevo archivo se vuelve a simular en python, el cual crea un archivo con la data correspondiente. El usuario también debe definir la ubicación de este nuevo archivo.

Paso 4: Ejecución LabVIEW

Con todo estandarizado, desde LabVIEW se configuran los archivos que debe leer. Ya con lo anterior listo, el programa se simula y entrega la gráfica con los indicadores del tramo seleccionado de los distintos archivos que se desean visualizar.

Consideraciones Técnicas

- El análisis solo se realiza si hay cambios de referencia significativos.
- Si el lazo está completamente en modo manual, no se calculan indicadores.
- El sistema puede analizar múltiples tramos de un mismo SP si están separados por tiempo.
- El código está preparado para trabajar con grandes volúmenes de datos.

4. VALIDACIÓN DEL ANÁLISIS AUTOMATIZADO MEDIANTE PROCESOS SIMULADOS

Con el fin de garantizar el correcto funcionamiento del programa CLPM, se utilizan sets de datos con la respuesta a escalón de plantas simuladas en lazo abierto. Se emplea MATLAB para la obtención de modelos de primer y segundo orden a partir de parámetros específicos. Se utiliza la función 'stepinfo' para obtener la respuesta escalón de las plantas. Posteriormente, se lleva a cabo la simulación y la visualización de las características de la respuesta a escalón.

El propósito fundamental es validar la información proporcionada por el programa CLPM, comparándola con los parámetros originales de la planta simulada. Dado que MATLAB no genera ciertos indicadores de manera directa, estos se calculan en Excel mediante configuraciones internas específicas, permitiendo así una verificación precisa de los resultados obtenidos.

Es importante mencionar que algunos de los indicadores calculados sólo tienen sentido físico en lazos de control realimentado, como la desviación estándar y el MAE. Sin embargo, se mostrará igualmente el cálculo de los indicadores tanto por el CLPM como por MATLAB/Excel para verificar igualmente su correcto cálculo.

Además del cálculo de los indicadores de desempeño de lazo, se calculan parámetros dinámicos de las plantas ideales a partir de ciertos indicadores para reforzar la efectividad del programa creado. En el caso de la planta de primer orden se calculan la ganancia continua K y la constante de tiempo τ . En el caso de las plantas de segundo orden se calculan el coeficiente de amortiguamiento ξ y la frecuencia natural ω_n . Estos parámetros son conocidos de antemano para todas las plantas de prueba, de modo que servirán como confirmación del funcionamiento del programa.

Para calcular estos parámetros se siguen las relaciones obtenidas de [38] [39]:

Sistema de primer orden:

- K : Valor en estado estacionario del sistema ante entrada unitaria.

$$\tau = \frac{t_s}{4} \quad (4.1)$$

Sistemas de segundo orden:

- α : Porcentaje de overshoot

$$\omega_n = \frac{4}{\xi t_s} \quad (4.2)$$

$$\xi = \frac{|\ln \frac{\alpha}{100}|}{\sqrt{\pi^2 + \ln \left(\frac{\alpha}{100}\right)^2}} \quad (4.3)$$

4.1. Sistema de primer orden

El primer sistema de prueba es un sistema de primer orden con ganancia continua de $K = 2$ y constante de tiempo $\tau = 1$.

$$G_1(s) = \frac{2}{s + 1} \quad (4.4)$$

4.1.1. Resultados Esperados

Simulando el sistema en Matlab se obtiene:

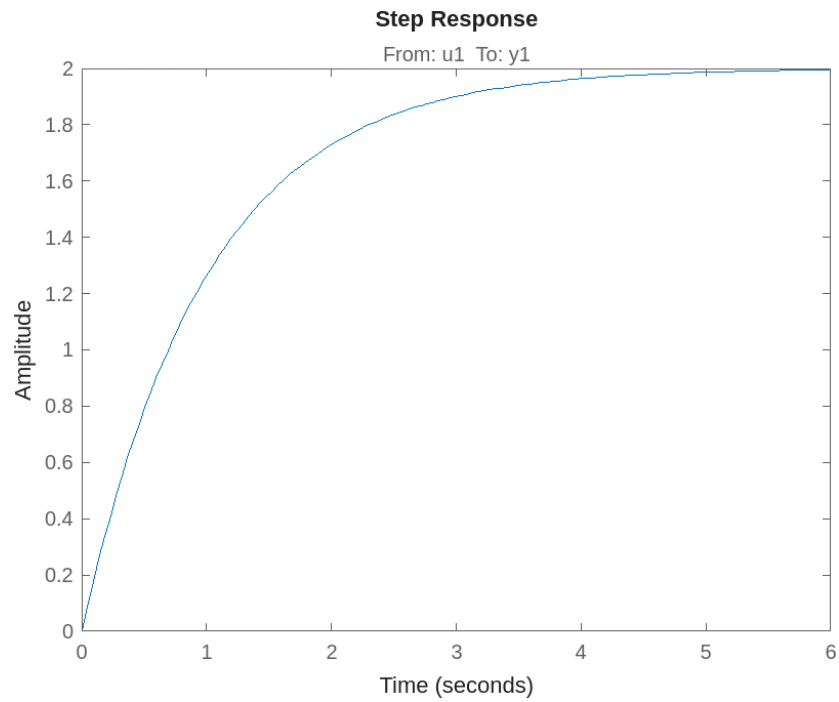


Fig. 4.1: Gráfica de respuesta a escalón de sistema de primer orden en MATLAB

```

RiseTime: 2.1970
TransientTime: 3.9121
SettlingTime: 3.9121
SettlingMin: 1.8090
SettlingMax: 1.9987
Overshoot: 0
Undershoot: 0
Peak: 1.9987
PeakTime: 7.3222

```

Desviación Estándar	0,456309556
MAE	11,39448742

(a) Indicadores obtenidos en MATLAB (b) Indicadores obtenidos en Excel

Fig. 4.2: Resultados de Indicadores

4.1.2. Resultados Obtenidos en CLPM

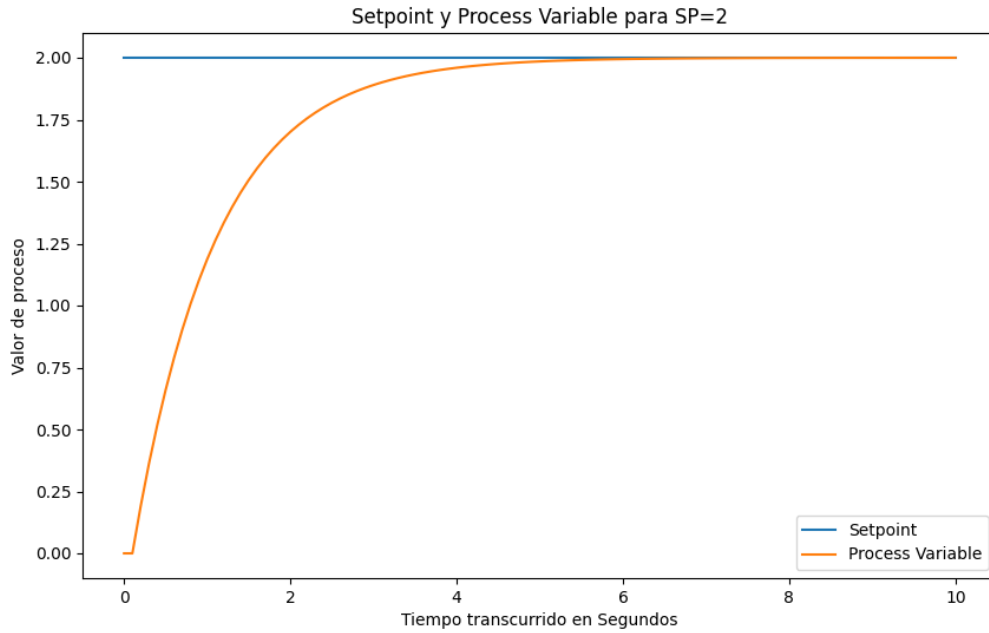


Fig. 4.3: Gráfica de respuesta a escalón de sistema de primer orden con programa CLPM

```
Tiempo de Subida [s] : 2.1970950565271252
Tiempo de Asentamiento [s] : 4.1000000000000005
Razón de Amortiguamiento : No posee dos peaks consecutivos
Sobrepaso [%] : No tiene
Desviación Estándar : 0.45630955594364914
Mean absolute error [%] : 11.394487424925208
```

Fig. 4.4: Resultados de Indicadores

4.1.3. Cálculo de parámetros dinámicos

De la gráfica obtenida se puede observar que la ganancia continua K es igual a 2. Por otro lado, del tiempo de asentamiento obtenido se puede calcular la constante de tiempo τ :

$$\tau = \frac{t_s}{4} = \frac{4,1}{4} = 1,025[s]$$

4.2. Sistema de segundo orden 1

El segundo sistema de prueba es uno de segundo orden con respuesta subamortiguada. Su coeficiente de amortiguamiento es $\xi = 0,75$ y su frecuencia natural es $\omega_n = 1,33[\text{rad/s}]$.

$$G_2(s) = \frac{3,5}{s^2 + 2s + 1,75} \quad (4.5)$$

4.2.1. Resultados Esperados

Se simula en MATLAB y se obtiene lo siguiente:

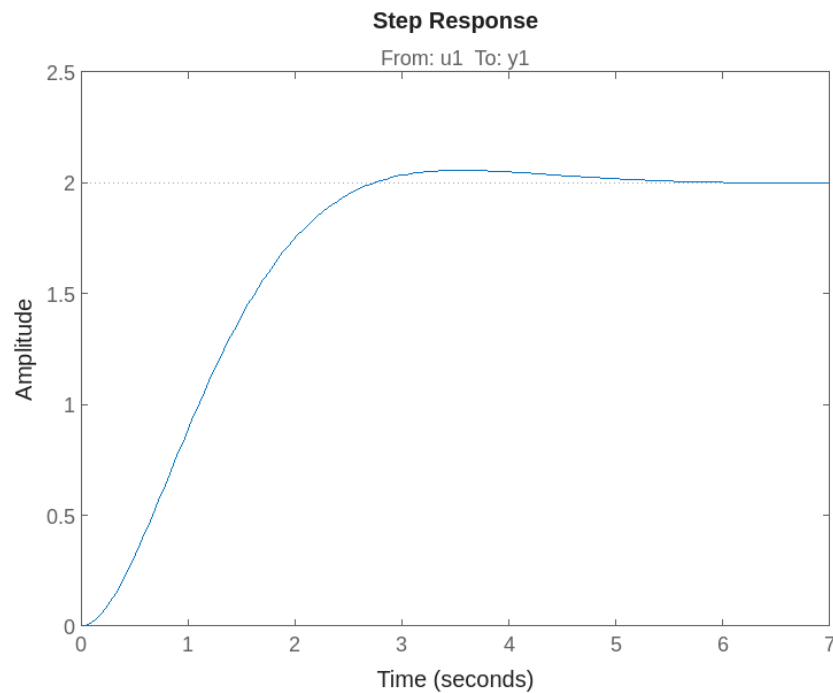


Fig. 4.5: Gráfica de respuesta de sistema de segundo orden 1 en MATLAB

```

RiseTime: 1.7163
TransientTime: 4.3069
SettlingTime: 4.3069
SettlingMin: 1.8098
SettlingMax: 2.0567
Overshoot: 2.8369
Undershoot: 0
Peak: 2.0567
PeakTime: 3.5460

```

Desviación Estándar	0,55929161
MAE	13,0560685

(a) Indicadores obtenidos en MATLAB (b) Indicadores obtenidos en Excel

Fig. 4.6: Resultados de Indicadores

4.2.2. Resultados Obtenidos en CLPM

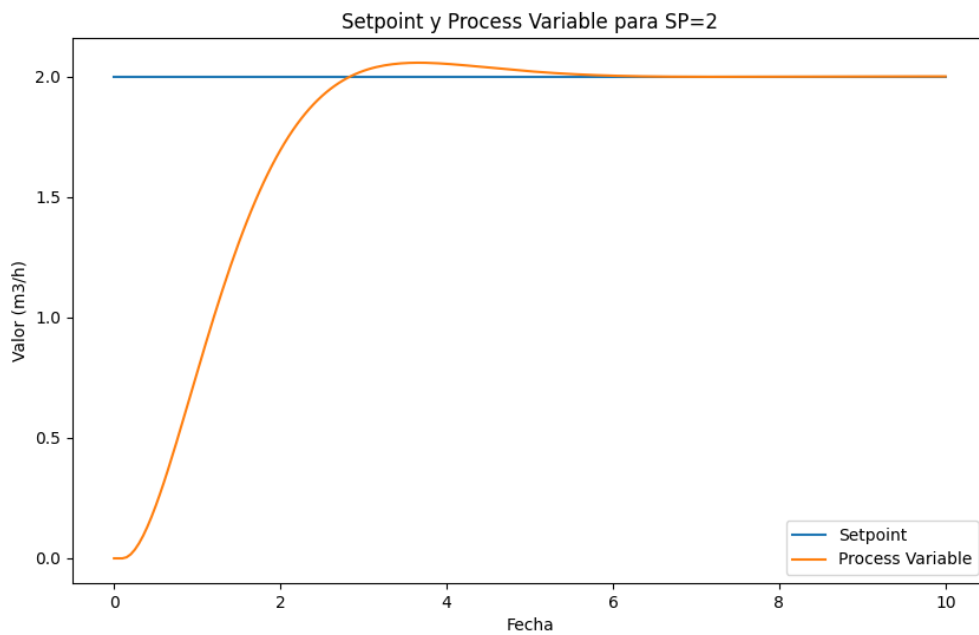


Fig. 4.7: Gráfica de respuesta de sistema de segundo orden 1 con programa CLPM

Tiempo de Subida [s] : 1.7161792179439024
 Tiempo de Asentamiento [s] : 4.45
 Razón de Amortiguamiento : No posee dos peaks consecutivos
 Sobrepaso [%] : 2.8371637490605117
 Desviación Estándar : 0.559291612437857
 Mean absolute error [%] : 13.056068520148829

Fig. 4.8: Resultados de Indicadores

4.2.3. Cálculo de parámetros dinámicos

Utilizando las relaciones entre los parámetros ξ y ω_n con los indicadores calculados se obtiene lo siguiente:

$$\xi = \frac{|\ln \frac{\alpha}{100}|}{\sqrt{\pi^2 + \ln \left(\frac{\alpha}{100}\right)^2}} = \frac{|\ln \frac{2.8371}{100}|}{\sqrt{\pi^2 + \ln \left(\frac{2.8371}{100}\right)^2}} = 0,75$$

$$\omega_n = \frac{4}{\xi t_s} = \frac{4}{0,75 \cdot 4,45} = 1,2$$

4.3. Sistema de segundo orden 2

El tercer sistema de prueba es también de segundo orden, pero presente mayor oscilación que el ejemplo anterior. Su coeficiente de amortiguamiento es $\xi = 0,125$ y su frecuencia natural es $\omega_n = 2[rad/s]$.

$$G_3(s) \frac{15}{s^2 + 0,5s + 4} \quad (4.6)$$

4.3.1. Resultados Esperados

Se simula en MATLAB y se obtiene lo siguiente:

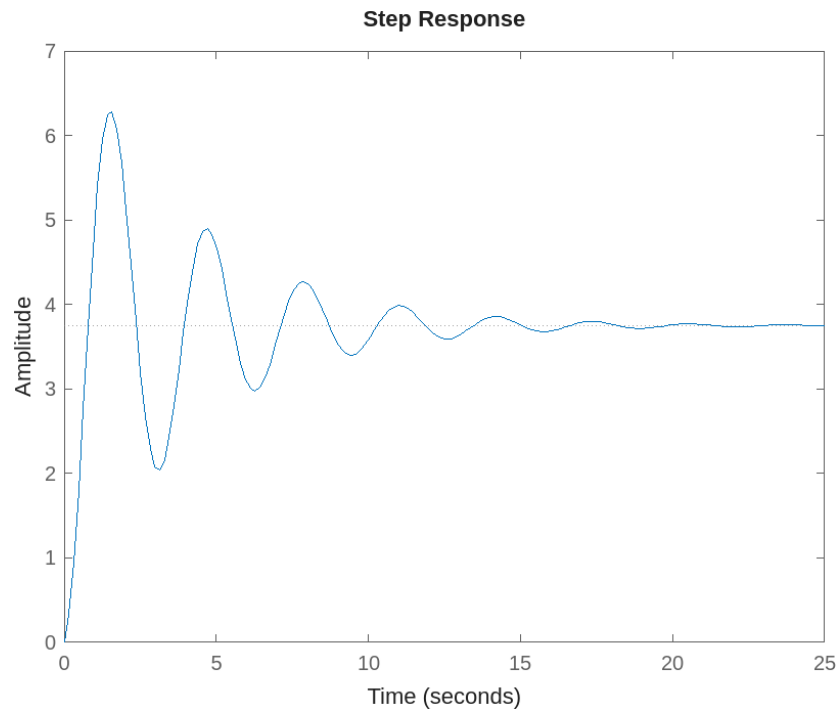


Fig. 4.9: Gráfica de respuesta de sistema de segundo orden 2 en MATLAB

```

RiseTime: 0.5584
TransientTime: 14.5926
SettlingTime: 14.5926
SettlingMin: 2.0416
SettlingMax: 6.2819
Overshoot: 67.5167
Undershoot: 0
Peak: 6.2819
PeakTime: 1.5708

```

Desviación Estándar	0,78395044
MAE	6,21106313

(a) Indicadores obtenidos en MATLAB (b) Indicadores obtenidos en Excel

Fig. 4.10: Resultados de Indicadores

4.3.2. Resultados Obtenidos en CLPM

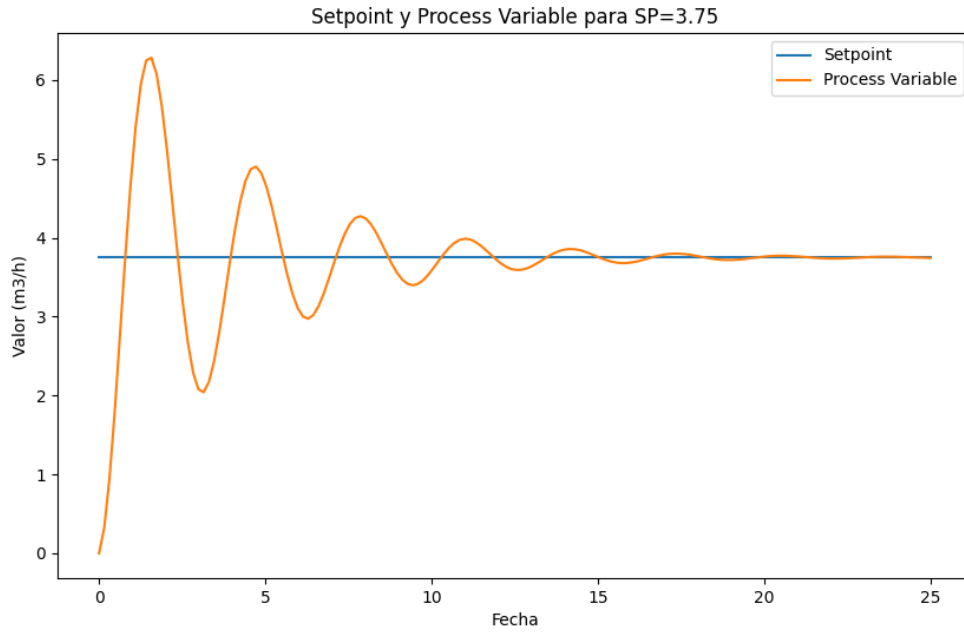


Fig. 4.11: Gráfica de respuesta de sistema de segundo orden 2 en programa CLPM

```
Tiempo de Subida [s] : 0.558220218894552
Tiempo de Asentamiento [s] : 14.6196
Razón de Amortiguamiento : 0.4558591378263513
Sobrepaso [%] : 67.49333333333333
Desviación Estándar : 0.7839504435163895
Mean absolute error [%] : 6.211063126890623
```

Fig. 4.12: Resultados de Indicadores

4.3.3. Cálculo de parámetros dinámicos

Al igual que en el caso anterior, se utilizan los indicadores calculados para encontrar ξ y

ω_n .

$$\xi = \frac{|\ln \frac{\alpha}{100}|}{\sqrt{\pi^2 + \ln \left(\frac{\alpha}{100}\right)^2}} = \frac{|\ln \frac{67,49}{100}|}{\sqrt{\pi^2 + \ln \left(\frac{67,49}{100}\right)^2}} = 0,125$$

$$\omega_n = \frac{4}{\xi t_s} = \frac{4}{0,125 \cdot 14,6196} = 2,1$$

4.4. *Análisis de resultados*

En los sistemas analizados se puede observar que el programa obtiene, en general, muy buenos resultados a la hora de compararlos con los indicadores calculados en MATLAB. Se observa una diferencia en el tiempo de asentamiento, ya que el sistema utiliza un método de muestreo y no realiza interpolación entre los datos, a diferencia de MATLAB, que sí realiza esta interpolación. En cuanto al tiempo de subida, el programa CLPM también interpola los datos, lo que explica por qué sus resultados fueron idénticos hasta el tercer decimal.

En un único caso se calcula la razón de amortiguamiento, dado que los primeros dos sistemas no presentan más de una o ninguna oscilación. En este caso, el resultado fue consistente con las expectativas.

El MAE se calcula tomando en cuenta los valores muestreados. Los resultados obtenidos coinciden con los esperados.

Por último, la desviación estándar y el sobrepaso son consistentes hasta el segundo decimal en todos los casos analizados.

En cuanto a los cálculos de los parámetros dinámicos, se puede observar en las subsecciones 4.1.3, 4.2.3 y 4.3.3 que, a partir de los resultados obtenidos del programa CLPM se pueden calcular la constante de tiempo, los coeficientes de amortiguamiento y las frecuencias naturales de los sistemas de prueba, resultando casi iguales a los esperados y probando así la efectividad del programa CLPM creado.

El MAE y la desviación estándar son indicadores para sistemas de lazo cerrado; como las pruebas fueron en lazo abierto, no serán estudiados.

4.4.1. Tabla Comparativa

Tab. 4.1: Error porcentual entre resultados esperados y obtenidos

Indicador	Error porcentual G1	Error porcentual G2	Error porcentual G3
Rise Time	0	$7,05 \times 10^{-5}$	$3,58 \times 10^{-4}$
SettlingTime	0,048	0,033	$1,85 \times 10^{-3}$
Overshoot	No aplica	$7,05 \times 10^{-5}$	$3,47 \times 10^{-4}$

5. ANÁLISIS AUTOMATIZADO DE DATOS DE PROCESOS DE LA PLANTA CONCENTRADORA

A continuación se procede a validar la interfaz de visualización y análisis automático de desempeño de lazos de control con datos de una planta concentradora real. En específico, se seleccionan dos procesos de la planta: la alimentación del molino SAG y el control de flujo y nivel en celdas de flotación rougher. En particular se estudia con mayor profundidad el segundo proceso, pues sus lazos de control se encuentran en modo automático la mayor parte del tiempo, por lo que su estudio es más relevante. En este capítulo se muestra el correcto funcionamiento del programa creado en esta aplicación, sin embargo, es importante destacar que el programa puede ser utilizado para estudiar el desempeño de cualquier lazo de control, cumpliéndose la estandarización de los datos disponibles.

5.1. Proceso 1: Alimentación del Molino SAG

Para el análisis del Proceso 1, se utiliza data obtenida directamente en terreno, donde las variables WI (Weight Indicator) y FIC (Flow Indicator Controller) fueron muestreadas cada 1 segundo, permitiendo un monitoreo detallado del comportamiento del sistema. Esta metodología de muestreo de alta frecuencia asegura que las variaciones rápidas en el proceso sean capturadas de manera precisa, facilitando un análisis más profundo del desempeño del lazo de control.

Los resultados obtenidos se ven reflejados en la figura 5.1.

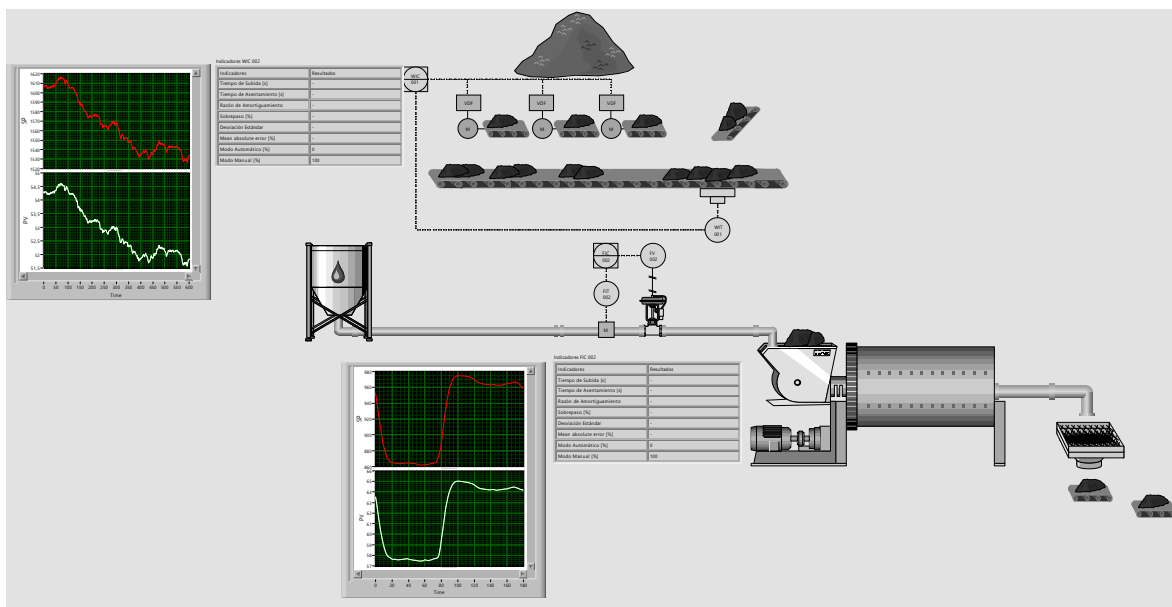


Fig. 5.1: Resultados sección Molino SAG

Analizando el proceso, se observa que no se obtienen indicadores de desempeño convencionales debido a que los datos analizados corresponden a una operación completamente en modo manual. En este modo, la MV es fijada directamente por el operador, lo que impide que el controlador actúe sobre el sistema y, por lo tanto, no se generan las métricas habituales de control.

Sin embargo, este escenario revela una técnica interesante conocido como traspaso suave. Esto ocurre cuando la referencia se ajusta para seguir la variable de proceso de forma gradual. La principal finalidad de esta técnica es evitar cambios bruscos al momento de reactivar el modo automático, ya que la referencia queda alineada con la condición actual del proceso. Al minimizar los saltos abruptos en la transición entre los modos manual y automático, se reduce el riesgo de generar sobrecargas o fluctuaciones indeseadas en los actuadores. Esto contribuye directamente a la durabilidad de los equipos y a la estabilidad operativa.

En el caso específico del lazo *FIC 002*, resulta inevitable interpretar la variación observada en la PV como si fuera una respuesta escalón. No obstante, al estar en modo manual se sabe que el cambio se debe exclusivamente a los ajustes realizados en la MV, por lo que la respuesta obtenida refleja el comportamiento inherente de la planta en lazo abierto y no el de

un sistema de control en lazo cerrado.

Este tipo de información es valiosa, ya que permite obtener modelos representativos del proceso real. Estos modelos pueden ser utilizados en futuras implementaciones tecnológicas, como la creación de gemelos digitales. Estos gemelos son réplicas virtuales exactas del proceso físico que, aunque no intervienen directamente en la operación real, permiten realizar simulaciones, pruebas de optimización y estrategias de control avanzadas sin poner en riesgo la planta[40]. Con estos modelos, se pueden explorar nuevas sintonizaciones, evaluar el impacto de cambios operativos y diseñar innovaciones que luego pueden ser implementadas en la planta real de forma segura y eficiente.

De manera complementaria, es posible visualizar la evolución de la variable manipulada (MV) para ambos lazos, obteniendo las siguientes gráficas:

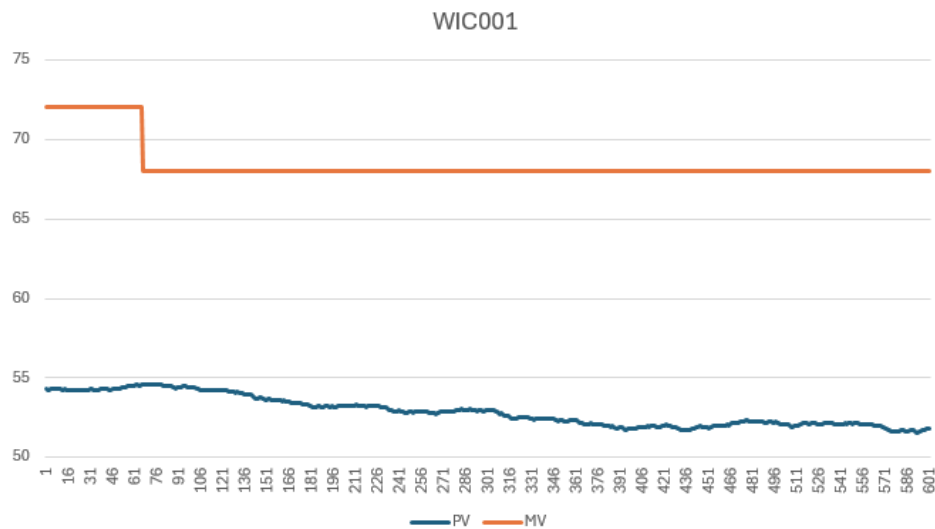


Fig. 5.2: Gráfica señal de actuación lazo WIC

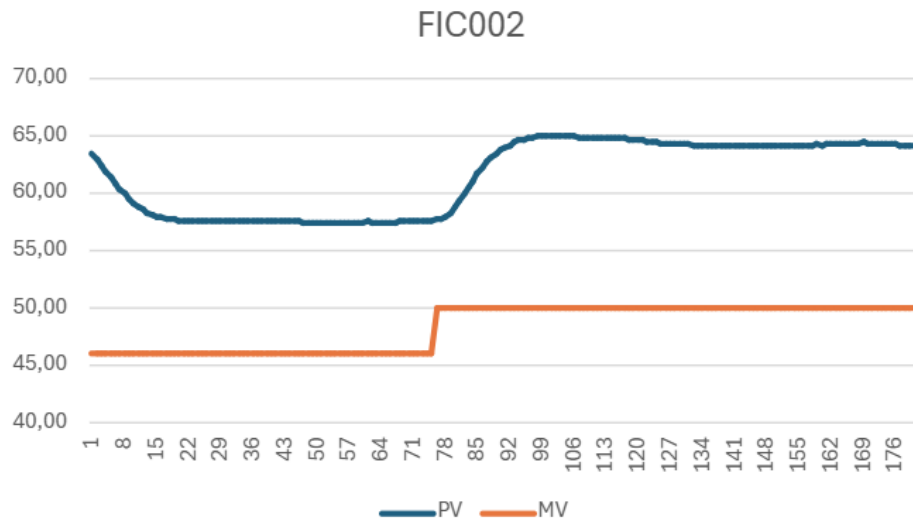


Fig. 5.3: Gráfica señal de actuación lazo FIC

Se nota visualmente que la variable manipulada se mantiene constante, lo que refuerza la afirmación de que el lazo estuvo el 100 % del tiempo en modo manual, por lo que no se pueden calcular indicadores.

5.2. *Proceso 2: Flujo y Nivel en Celdas de Flotación Rougher*

5.2.1. *Análisis del proceso general*

En este proceso se trabaja con información proveniente de dos tipos de lazos: Un lazo de nivel LIC y uno de flujo FIC.

La data utilizada para los LIC fue recolectada durante un período de dos meses, con un muestreo realizado cada 5 segundos. De toda la data, se seleccionan ciertos períodos de tiempo donde los lazos operan en modo automático para mostrar los indicadores de desempeño calculados y los gráficos de la respuesta de los lazos.

En el caso mostrado, los lazos no están necesariamente sincronizados, sólo se tomaron casos interesantes de estudio. En la sección 5.2.3 se analizará un sistema sincronizado.

En contraste, la información del lazo FIC 151 proviene de una prueba de sintonización realizada en terreno, aplicada mediante un cambio de escalón.

En la figura 5.4 se muestra la HMI creada en LabVIEW, mostrando los gráficos mencionados y los indicadores de desempeño calculados para cada lazo.

Se identifican diversos aspectos clave relacionados con el desempeño de los lazos de control involucrados. En primer lugar, analizando los lazos de nivel, uno de los factores más destacados es la variabilidad en los tiempos de asentamiento observados entre los diferentes lazos, existiendo incluso lazos donde no se logra cumplir el criterio de estabilidad del 2 %. Esto se debe a que los sistemas reales cuentan con un ruido inherente, ya sea de medición como por el mismo proceso. Para profundizar en el análisis de este lazo, el porcentaje de error aceptable del criterio puede modificarse para cada lazo para un estudio individual más detallado.

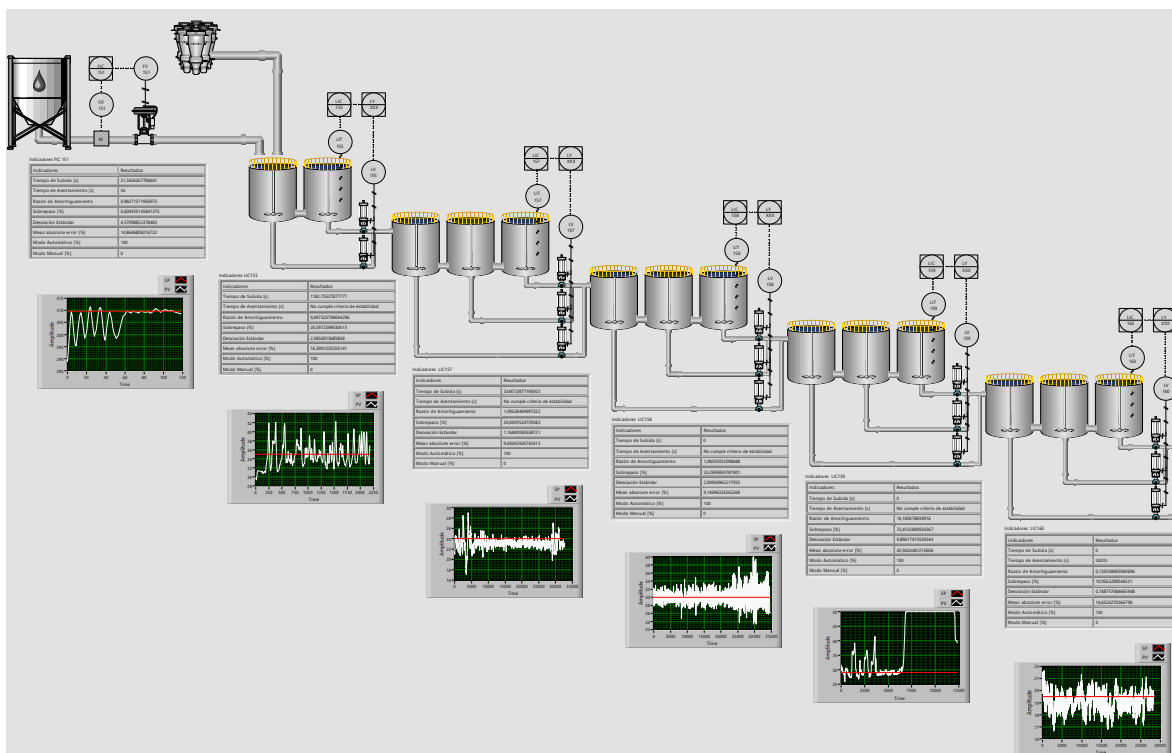


Fig. 5.4: Resultados sección flotación Rougher

En cuanto al lazo de flujo, se observa que tiene un largo transiente, presentando grandes oscilaciones al inicio que impiden un asentamiento rápido y eficiente. Este tipo de respuesta puede afectar la estabilidad del proceso global, ya que las oscilaciones en el flujo generan variaciones en otras variables dependientes aguas abajo en el proceso. Sin embargo, al lograr la estabilidad, tiene un muy buen seguimiento de referencia.

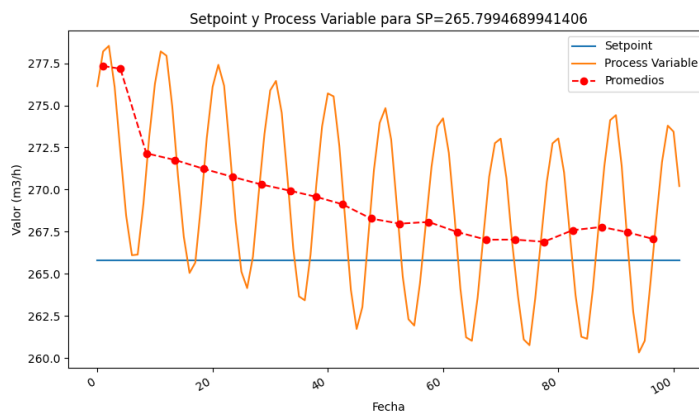
5.2.2. Desempeño del lazo de control de Flujo

Para la interfaz de LabVIEW se seleccionó un periodo de tiempo específico del proceso para mostrar el análisis. No obstante, es destacable que el programa CLPM no está limitado exclusivamente a un solo tramo. Este programa está diseñado para trabajar con la data cruda del sistema, identificando de manera automática las referencias constantes y segmentando la información según los distintos tramos operativos presentes en los datos.

El programa analiza cada tramo de manera individual, calculando los principales indica-

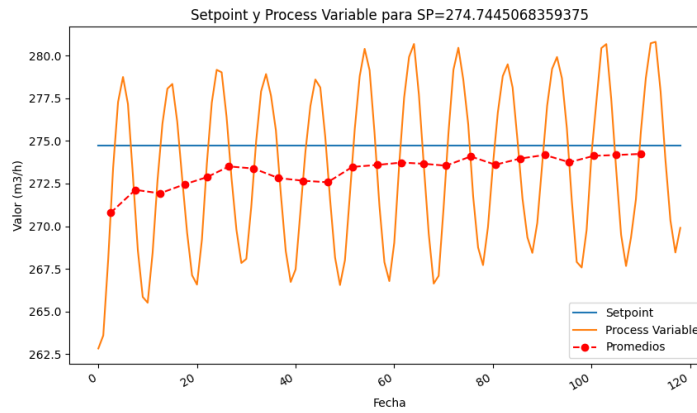
dores de desempeño, permitiendo así, una evaluación precisa y general del lazo de control. Además, informa el porcentaje de tiempo que el lazo operó en modo automático o modo manual tomando en cuenta toda la data.

A continuación, se presentan los indicadores de todos los tramos de tiempo del lazo que el programa logra identificar y evaluar con la información proporcionada. Esto amplía la perspectiva del análisis, permitiendo obtener una comprensión más integral del comportamiento del sistema de control en distintas condiciones operativas.



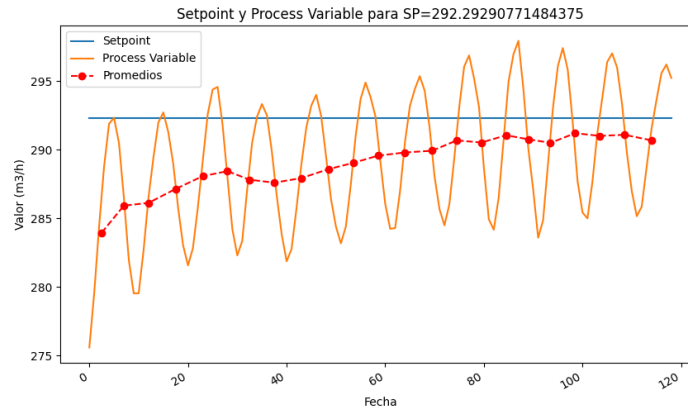
Tiempo de Subida [s] : 101.0
 Tiempo de Asentamiento [s] : No cumple criterio de estabilidad en tiempo muestreado
 Razón de Amortiguamiento : No posee dos peaks consecutivos
 Sobrepasso [%] : 2.0593436043903606
 Desviación Estándar : 5.023015541929483
 Mean absolute error [%] : 27.89039660811158
 Modo Automático [%] : 99.26062846580406
 Modo Manual [%] : 0.7393715341959393

Fig. 5.5: Lazo FIC 151 con $SP=265\left[\frac{m^3}{h}\right]$



Tiempo de Subida [s] : 116.9045614993343
 Tiempo de Asentamiento [s] : No cumple criterio de estabilidad en tiempo muestreado
 Razón de Amortiguamiento : No posee dos peaks consecutivos
 Sobrepasso [%] : 2.205518088316249
 Desviación Estándar : 4.681305270145168
 Mean absolute error [%] : 23.36053050780816
 Modo Automático [%] : 99.26062846580406
 Modo Manual [%] : 0.7393715341959393

Fig. 5.6: Lazo FIC 151 con $SP=274[\frac{m^3}{h}]$



Tiempo de Subida [s] : 92.99082633560177
 Tiempo de Asentamiento [s] : 113
 Razón de Amortiguamiento : No posee dos peaks consecutivos
 Sobrepasso [%] : 1.933595980895094
 Desviación Estándar : 4.797699470195417
 Mean absolute error [%] : 20.07930539029251
 Modo Automático [%] : 99.26062846580406
 Modo Manual [%] : 0.7393715341959393

Fig. 5.7: Lazo FIC 151 con $SP=292[\frac{m^3}{h}]$

El análisis del comportamiento del lazo de control de flujo para distintas referencias revela la presencia de oscilaciones persistentes alrededor del valor deseado, lo que evidencia una falta de amortiguamiento adecuado.

En los escenarios evaluados, el tiempo de asentamiento no cumple con el criterio de estabilidad dentro del tiempo de muestreo, impidiendo que el lazo alcance un estado estacionario del 2 % de error. No obstante, estas oscilaciones se interpretan como inherentes al lazo, y al analizar la curva de promedios móviles, se observa que logra mantenerse en un 2 % de error, aunque el ruido impide que se refleje claramente. Si se flexibiliza el criterio, es posible calcular el tiempo de asentamiento de los distintos tramos en algún momento durante la ejecución del programa.

El MAE se mantiene elevado, superando el 20 % en todas las pruebas a excepción de la mostrada por LabVIEW. Además, también es notado que la tendencia del lazo no se estaciona en SP, sino en algún valor mayor o menor dependiendo de si la ganancia del escalón de referencia es positiva o negativa.

La razón de amortiguamiento solo se calcula en un segmento, ya que su cálculo depende de la línea de tendencia generada mediante la media móvil de los valores máximos y mínimos del PV, omitiendo así el impacto de las oscilaciones del programa. Esto permite interpretar que el sistema responde de manera similar a un sistema de primer orden ante cambios escalonados.

En cuanto al tiempo de subida, se observa un rango entre 92 y 116 segundos. El sobrepaso se mantiene bajo, alrededor del 2 %, lo que sugiere que está más relacionado con las oscilaciones propias del sistema que con deficiencias en el control.

La desviación estándar oscila entre 4 y 5 en los cuatro tramos analizados, indicando que el sistema mantiene un control estable sin desviarse significativamente de su referencia.

Finalmente, se observa que el lazo de control opera la mayor parte del tiempo en modo automático, lo que sugiere que la sintonización actual ha sido efectiva a lo largo del tiempo evaluado.

Como resumen de los resultados observados, se observa que las distintas referencias presentan indicadores similares, lo que sugiere que el lazo de control mantiene un desempeño consistente frente a los distintos SP, basado en los promedios móviles calculados, logrando

una respuesta uniforme y adecuada en cada caso. Es importante mencionar que las oscilaciones observables en las imágenes 5.5, 5.6 y 5.7 son inherentes al sistema, por lo que el lazo de control no las amortigua. Sin embargo, el desempeño se mide de acuerdo a la tendencia promedio de la señal.

5.2.3. Desempeño de los lazos de control de nivel

En la figura 5.4 se observa un cambio drástico en la variable de proceso del Lazo de Nivel 159, a pesar de que su referencia se mantiene constante.

Esta irregularidad plantea la pregunta sobre su causa y abre la posibilidad de que se deba a factores externos o a la influencia de los estanques anteriores.

Para analizar esta hipótesis, se sincronizan todos los lazos del proceso 2 en un mismo periodo de tiempo para buscar posibles similitudes en su comportamiento. Los resultados se muestran en la figura 5.8.

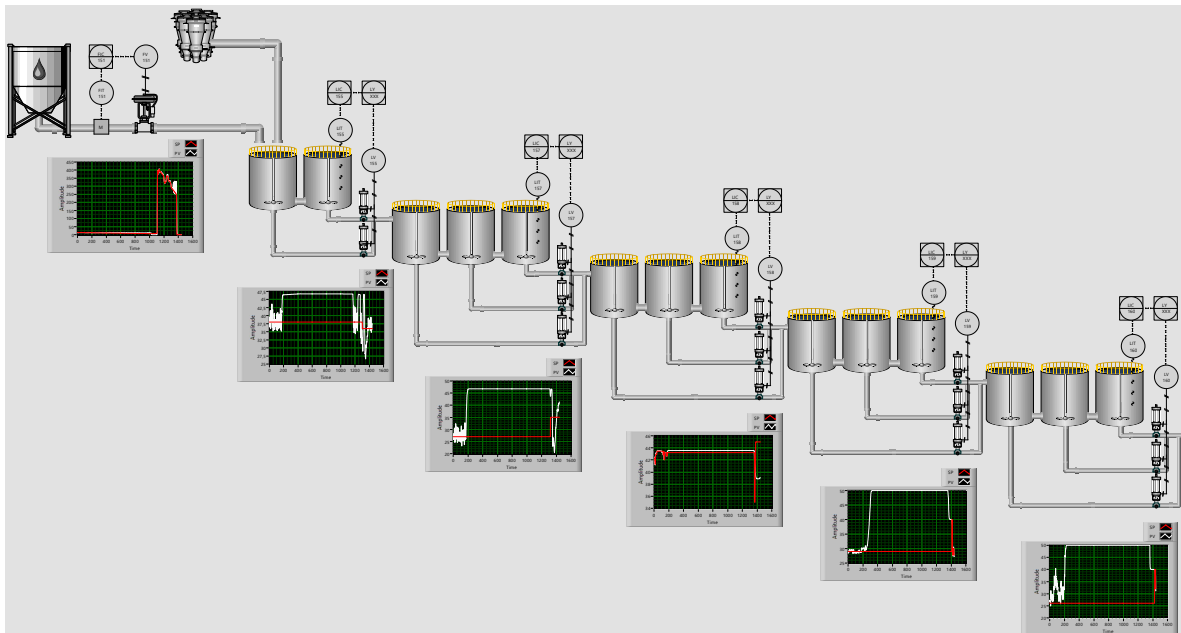


Fig. 5.8: Lazos de Nivel en el mismo periodo de tiempo

El propósito de cada lazo de control LIC es mantener estable el nivel de su estanque en un punto específico. Sin embargo, debido al acoplamiento entre los mismos, la acción de un

lazo para controlar el nivel de su estanque puede influir en el comportamiento de los demás.

En particular, se ha detectado un salto inesperado que impacta simultáneamente a los cinco lazos de nivel. Dado que el lazo de flujo permanece con su actuación en cero durante la perturbación, es poco probable que el origen provenga de ahí. En su lugar, el análisis sugiere que la causa está asociada con el flujo proveniente de los hidrociclones o una perturbación externa del proceso.

Además, el análisis general indica que los controladores están operando correctamente, lo que permite descartar fallas internas y atribuir el evento a una perturbación externa.

Este último ejemplo enfatiza la importancia de que la información sea leída por un operador que tenga acceso a las bitácoras de funcionamiento del proceso.

6. ANÁLISIS DE RESULTADOS Y CONCLUSIONES

6.1. *Resultados y Conclusiones*

El presente trabajo desarrolla una herramienta efectiva para el monitoreo y análisis del desempeño de lazos de control en una planta concentradora, mediante la implementación de un sistema computacional que automatiza el cálculo de indicadores clave de rendimiento. Se lleva a cabo un estudio exhaustivo del estado del arte sobre los sistemas de Monitoreo de Rendimiento de Lazos de Control (CLPM), identificando sus características, capacidades y limitaciones. Este análisis ha permitido establecer un marco de referencia sólido para evaluar la eficiencia de los lazos de control en entornos industriales.

Se han definido los indicadores clave de desempeño, tales como el tiempo de subida, tiempo de asentamiento, razón de amortiguamiento, sobrepaso y error absoluto medio, seleccionados por su relevancia en la evaluación de la estabilidad y eficiencia del control. La estandarización de datos y la selección adecuada de ventanas de tiempo aseguran la fiabilidad de los indicadores calculados. Se construye, además, un programa en Python que facilita la importación, procesamiento de datos operacionales y la generación de métricas clave, integrando los resultados con el software LabVIEW para ofrecer una representación visual clara y funcional de los datos, favoreciendo así el monitoreo y evaluación del rendimiento de los lazos de control.

Asimismo, se valida el sistema en dos etapas: a través de pruebas en plantas piloto y mediante el análisis de datos históricos de la planta concentradora. Estos pasos permiten corroborar la correcta implementación y funcionalidad del programa, así como evaluar el estado real de los lazos de control en condiciones operativas. Los resultados obtenidos no solo identifican oportunidades de mejora en la sintonización de los controladores, sino que también contribuyen a una mayor estabilidad y eficiencia del proceso productivo.

Respecto al cumplimiento de los objetivos planteados al inicio del trabajo, se puede afirmar que se logra sistematizar el análisis de los datos históricos de los lazos de control, logrando una reducción significativa en el tiempo y esfuerzo requeridos para la interpretación de esos datos. El desarrollo del software CLPM cumple con la necesidad de identificar los indicadores mínimos necesarios para comprender el comportamiento de un lazo de control, permitiendo establecer una línea base de desempeño a partir del análisis de datos históricos.

Además, se logra entender el proceso detrás de un sistema de monitoreo de rendimiento, sentando las bases para abordar futuros proyectos en diferentes plantas mediante una metodología unificada. Esto facilita la replicación del proceso en otros entornos industriales, optimizando de este modo el rendimiento de la operación. Por último, se entrega la información mínima necesaria para que un operador pueda mejorar el rendimiento de las operaciones del proceso al identificar y abordar posibles deficiencias en los lazos de control existentes, facilitando a su vez un control rutinario para evaluar el progreso en la optimización del rendimiento.

En conclusión, el trabajo realizado ha demostrado ser un importante aporte a la automatización y mejora del desempeño en los sistemas industriales de control. La integración de aplicaciones computacionales para la visualización y análisis de datos facilita no solo la toma de decisiones informadas, sino también la implementación de estrategias de mejora continua en procesos altamente tecnológicos. La capacidad de monitorear y ajustar de manera eficiente los lazos de control repercute directamente en la rentabilidad, seguridad y calidad de las operaciones industriales, estableciendo un camino claro hacia la optimización de procesos en el futuro.

6.2. *Trabajo futuro*

- Ajustar el programa python para trabajar de forma Online conectada a un sistema SCADA en una planta. Así, poder calcular los indicadores de rendimiento de los lazos de control en tiempo real.
- Configurar alarmas, tanto visuales como sonoras en la HMI que indiquen cuando algún indicador de desempeño de un lazo de control se salga de los rangos admisibles.
- Continuar desarrollando interfaces gráficas de los demás procesos de la planta concentradora o incluso implementar el trabajo a otras plantas y otros tipos de procesos.

BIBLIOGRAFÍA

- [1] R. P. Borase, D. K. Maghade, S. Y. Sondkar y S. N. Pawar, “A review of PID control, tuning methods and applications,” *International Journal of Dynamics and Control*, vol. 9, págs. 818-827, 2020. dirección: <https://api.semanticscholar.org/CorpusID:225560341>.
- [2] G. K. McMillan, “PID Control in the Third Millennium: Lessons Learned and New Approaches,” en Springer London, 2012, cap. Industrial Applications of PID Control.
- [3] S. X. Global, *Control Performance Analytics*, Visitado en 2024. dirección: <https://xcelerator.siemens.com/global/en/all-offerings/products/p/picpa00000.html>.
- [4] S. AG, *Control Performance Analytics (CPA)*, Brochure, Visitado en 2024, 2018. dirección: <https://assets.new.siemens.com/siemens/assets/api/uuid:409a9482-e7ec-4fb8-8fe2-72888dd144a3/version:1583831139/pd-pa004469f1-cpaderz01o2eenus-72.pdf>.
- [5] A. Horch y F. Heiber, “On Evaluating Control Performance on Large Data Sets,” *IFAC Proceedings Volumes*, vol. 37, n.º 9, págs. 535-540, 2004, 7th IFAC Symposium on Dynamics and Control of Process Systems 2004 (DYCOPS -7), Cambridge, USA, 5-7 July, 2004, ISSN: 1474-6670. DOI: [https://doi.org/10.1016/S1474-6670\(17\)31864-5](https://doi.org/10.1016/S1474-6670(17)31864-5). dirección: <https://www.sciencedirect.com/science/article/pii/S1474667017318645>.
- [6] M. Yumer, *Automating Control Loop Performance Assessment with MATLAB*, Entrenamiento, 2024.
- [7] Hatch-Team, *Control Loop Performance Metrics*, Entrenamiento Consultora, 2023.

- [8] B. Noguera, *¿Qué es un lazo de control?* Visitado en 2024, 2021. dirección: <https://www.ingenieriaquimicareviews.com/2021/09/lazo-de-control.html>.
- [9] Editorial-Staff, *What is a Control Loop?* Visitado en 2024. dirección: <https://instrumentationtools.com/what-is-control-loop/>.
- [10] M. Brisk, "Process control: potential benefits and wasted opportunities," *Australian Journal of Electrical and Electronics Engineering*, vol. 2, págs. 41-48, ene. de 2005. DOI: 10.1080/1448837X.2005.11464113.
- [11] Valmet, *Valmet PlantTriage*, Visitado en 2024. dirección: <https://www.valmet.com/automation/services/process-performance/pid-tuning-process-control/planttrriage/>.
- [12] Valmet, *Optimizing processes based on valve performance data*, Visitado en 2024. dirección: <https://new.valmet.com/insights/blogs/blog-23-digitalization-in-pulp--paper---optimizing-processes-based-on-valve-performance-data/>.
- [13] Valmet, *What's new in the latest version of Valmet PlantTriage software?* Visitado en 2024. dirección: <https://www.valmet.com/insights/articles/flow-control/the-latest-version-of-expertune-planttrriage-software/>.
- [14] ABB, *OptimizeIT Loop Performance Manager (LPM)*, Brochure, Visitado en 2024, 2004. dirección: <https://search.abb.com/library/Download.aspx?DocumentID=9AKK101130D1118>.
- [15] ABB, *Introduction to LPM*, Presentación empresa, Visitado en 2024. dirección: <https://view.officeapps.live.com/op/view.aspx?src=https%3A%2F%2Fsearch.abb.com%2Flibrary%2FDownload.aspx%3FDocumentID%3D9AKK101130D1100>.
- [16] S. Electric, *EcoStruxure Control Advisor*, Visitado en 2024. dirección: <https://www.se.com/us/en/product-range/107489819-ecostruxure-control-advisor/#overview>.

- [17] *EcoStruxure Control Advisor*, PSS 41S-4CTRLADV, Visitado en 2024, Schneider Electric, jul. de 2021.
- [18] D. A. J. Cárdenas, “Desarrollo de software para inspección técnica de una aplicación CPM,” Visitado en 2024, Memoria de Pregrado, Universidad de Concepción, Diciembre de 2017.
- [19] cyrus@timberbrookfamily.com, *Is the Harris Index Right for Your Loop Assessment Needs?* Visitado en 2024, 2016. dirección: <https://controlstation.com/harris-index/>.
- [20] G. P. G. Aclan, “GNPK Control Loop Performance Monitoring Technical Principles and Guidelines,” GNPK, inf. téc., Visitado en 2024.
- [21] J. C. Berríos, *Sistema de control de primer orden: ¿Qué es? (Tiempo de subida, tiempo de asentamiento y función de transferencia)*, Visitado en 2024, 2023. dirección: <https://telcomplus.org/sistema-de-control-de-primer-orden#:~:text=El%20tiempo%20de%20subida%20es%20el%20tiempo%20que,T%20es%20la%20constante%20de%20tiempo%20del%20sistema.>
- [22] J. C. Berríos, *Tiempo de asentamiento: ¿Qué es? (Fórmula y cómo encontrarla en MATLAB)*, Visitado en 2024, 2023. dirección: <https://telcomplus.org/tiempo-de-asentamiento/#:~:text=El%20tiempo%20de%20asentamiento%20es%20un%20concepto%20muy,alcanzar%20su%20estado%20estable%20despu%C3%A9s%20de%20una%20perturbaci%C3%B3n.>
- [23] K. Ogata, “Modern Control Engineering,” en Pearson, 2009, cap. 5, págs. 164-179.
- [24] C. Ortega, *Desviación estándar: Qué es, usos y cómo obtenerla*, Visitado en 2024, 2023. dirección: <https://www.questionpro.com/blog/es/desviacion-estandar/>.
- [25] G. Malina y D. Flamenco, “Investigación y desarrollo de herramientas para optimización de lazos de control (CASSA-UCA),” *CONIA, Congreso de Ingeniería y Arquitectura 2013. “Territorio - Agua - Tecnología”*, págs. 287-302, 2014, ISSN: 2308-409X. DOI: <http://hdl.handle.net/11674/6151>.

- [26] A. Horch, “Bucles de control, ¿un placer o una plaga?: vista de planta en la industria de procesos,” *Revista ABB*, n.º 4, págs. 55-59, ene. de 2005, ISSN: 1013-3135.
- [27] S. Castaño, *Control por Mínima Varianza*, Visitado en 2024, 2018. dirección: <https://controlautomaticoeducacion.com/control-de-procesos/control-por-minima-varianza/>.
- [28] T. J. Harris, “Assessment of control loop performance,” *The Canadian Journal of Chemical Engineering*, vol. 67, n.º 5, págs. 856-861, 1989. DOI: <https://doi.org/10.1002/cjce.5450670519>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/cjce.5450670519>. dirección: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cjce.5450670519>.
- [29] Probabilidad-Estadística-Team, *Modelo ARIMA*, Visitado en 2024, 2024. dirección: <https://www.probabilidadyestadistica.net/modelo-arima/>.
- [30] Y. L. Zheng O’Neill y K. Williams, “HVAC control loop performance assessment: A critical review (1587-RP),” *Science and Technology for the Built Environment*, vol. 23, n.º 4, págs. 619-636, 2017. DOI: [10.1080/23744731.2016.1239466](https://doi.org/10.1080/23744731.2016.1239466). eprint: <https://doi.org/10.1080/23744731.2016.1239466>. dirección: <https://doi.org/10.1080/23744731.2016.1239466>.
- [31] J. Mozaryn, *How can we know which criteria (ISE, IAE, ITSE, and ITAE) will be the best for my work, especially for power system applications?* Jun. de 2022.
- [32] Anónimo, *Qué es: error absoluto medio (MAE)*, Visitado en 2024, 2020. dirección: <https://es.statisticseasily.com/glosario/%C2%BFQu%C3%A9-significa-error-absoluto-mae%3F/>.
- [33] cyrus@timberbrookfamily.com, *Using Average Absolute Error to Uncover Tuning Issues*, Visitado en 2024, 2016. dirección: <https://controlstation.com/average-absolute-error/>.
- [34] Soporte-Técnico-Microsoft, *Elegir la mejor línea de tendencia para los datos*, Visitado en 2025, 2025. dirección: <https://support.microsoft.com/es-es/>

topic/elegir-la-mejor-l%C3%ADnea-de-tendencia-para-los-datos-1bb3c9e7-0280-45b5-9ab0-d0c93161daa8.

- [35] R. Cruz, *Manejo de LabVIEW: Leer worksheets de Excel*, Visitado en 2024, oct. de 2023. dirección: <https://www.youtube.com/watch?v=puTfI21M1F8>.
- [36] Tzili, *Gráficas en LabVIEW*, Visitado en 2024, jun. de 2020. dirección: <https://www.youtube.com/watch?v=7IrT8PrmIqc>.
- [37] E. D, *Bandpass modulation: Set up a "Mixed Signal Graph" to display both analog and digital signals*, Visitado en 2024, jun. de 2020. dirección: <https://www.youtube.com/watch?v=iD21gs05zsw>.
- [38] S. Castaño, *Sistemas de Segundo Orden*, Visitado en 2024, 2020. dirección: <https://controlautomaticoeducacion.com/control-realimentado/sistemas-de-segundo-orden/>.
- [39] A. Tijjani, M. Shehu, A. Alsabari, Y. Sambo y N. Tanko, "Performance Analysis for Coupled-Tank System Liquid Level Control Using MPC, PI and PI-plus-Feedforward Control Scheme," *Journal of Robotics and Automation*, vol. 2017, págs. 42-53, jun. de 2017. DOI: 10.36959/673/354.
- [40] N. Elbazi, H. E. Hadraoui, O. Laayati, A. E. Maghraoui, A. Chebak y M. Mabrouki, "Digital Twin in Mining Industry: A Study on Automation Commissioning Efficiency and Safety Implementation of a Stacker Machine in an Open-Pit Mine," en *2023 5th Global Power, Energy and Communication Conference (GPECOM)*, 2023, págs. 548-553. DOI: 10.1109/GPECOM58364.2023.10175788.

7. ANEXOS

7.1. Código

```
1 #Importar librerías necesarias
2 import datetime
3 import numpy as np
4 import pandas as pd
5 import matplotlib.pyplot as plt
6 import matplotlib.dates as mdates
7 import matplotlib.ticker as ticker
8
9 #Entradas. El programa solicita la nomenclatura utilizada
   por el usuario.
10 date_name = input("Nombre columna FECHA : ")
11 pv_name = input("Nombre de Variable de Procesos : ")
12 sp_name = input("Nombre de Referencia : ")
13 out_name = input("Nombre de Variable Manipulada : ")
14 sampling = float(input("Tiempo de Muestreo en [s] : "))
15
16 #Ubicación del archivo por leer. Esto debe modificarlo el
   usuario antes de utilizar.
17 ruta = r"C:\Users\xxxx\xxxxx\xxxxx\xxxx.xlsx"
18
19 #El archivo con la información se convierte en data frame
20 df = pd.read_excel(ruta, engine='openpyxl')
```

```

21
22 #Función de cálculo de indicadores y gráfica
23 def indicadores(lista,i): #Se define (lista,i) =
    (lista_SP_fijo , SP)
24     PV = lista[pv_name] # Lista con todos los valores de PV
25     tiempo = [w * sampling for w in range(len(lista))] #Se
    obtiene el tiempo total en segundos considerando la
    cantidad de muestras
26
27
28     #Tiempo de Subida
29     valor_inicial = lista.loc[PV.index[0],pv_name] #Se
    obtiene el primer valor de PV
30     span = i - valor_inicial # Se obtiene la ganancia del
    escalón aplicado
31     PV_90 = 0.9 * span + valor_inicial # Se obtiene el valor
    de PV al alcanzar el 90% de la ganancia del escalón
32     PV_10 = 0.1 * span + valor_inicial # Se obtiene el valor
    de PV al alcanzar el 10% de la ganancia del escalón
33     x_ej = [k*sampling for k in range(len(lista))] #Se
    define un eje X
34     y_ej = lista[pv_name] #Se define un eje y
35     t_90 = np.interp (PV_90,y_ej,x_ej) #Se interpola en los
    ejes para determinar el tiempo donde alcanza PV 90%
36     t_10 = np.interp (PV_10,y_ej,x_ej) #Se interpola en los
    ejes para determinar el tiempo donde alcanza PV 10%
37     t_r = abs(t_90 - t_10 )#El tiempo de subida es la
    diferencia entre estos tiempos, se obtiene un tiempo
    continuo
38

```

```

39     #Control
40     if abs(span) < 1 : #Revisar si el cambio de referencia
es de mínimo 1, para poder ver alteraciones significativas
41         return None #Si no cumple, no se retornan indicadores
42
43     #Tiempo de Asentamiento
44     PV_98 = float(0.98*i) # Se obtiene el 2% inferior del
valor de asentamiento
45     PV_102 = float(1.02*i) # Se obtiene el 2% superior del
valor de asentamiento
46     t_set = "" #Se define variable vacía
47     cont_set = len(lista)-1 #Contador
48     for j in range((len(lista) + PV.index[0]-1) ,
(PV.index[0]+4), -1): #Se revisan desde el último valor
de PV en la lista hasta el sexto, siendo "j" un valor de
ubicación
49         if PV_98 <= PV[j] <= PV_102 and PV_98 <= PV[j-1] <=
PV_102 and PV_98 <= PV[j-2] <= PV_102 and PV_98 <=
PV[j-3] <= PV_102 and PV_98 <= PV[j-4] <= PV_102 and
PV_98 <= PV[j-5] <= PV_102:
50             #Se revisa si los 5 valores anteriores salen del
porcentaje establecido superior e inferior
51             t_set = tiempo[cont_set-5] #Guarda el tiempo
muestreado del último valor revisado, teniendo un tiempo
discreto
52             else :
53                 break #Cuando ya se sale del rango, se termina
la función
54             cont_set = cont_set - 1 #Disminución de contador
55     if t_set == "" : #Si la lista sigue vacía

```

```

56         t_s = "No cumple criterio de estabilidad en tiempo
muestreado" #Nunca se estabiliza entre el rango
porcentual establecido
57     else : #Si no está vacía la lista
58         t_s = t_set #El tiempo de asentamiento es el último
valor almacenado antes de que se saliera del rango
59
60
61     #Razón de Amortiguamiento
62     if span >= 0 : #Si el span es positivo o negativo
63         orientacion = 1 #Crece
64     else :
65         orientacion = 0 #Decrece
66     # Variables para registrar el estado de crecimiento o
decrecimiento
67     direccion = None
68     maximos = []
69     minimos = []
70     tiem = []
71     #Obtener máximos y mínimos
72     for l in range(PV.index[0]+1,(len(lista) + PV.index[0]))
: #Se recorre los valores de PV
73         if PV[l] > PV[l - 1] and direccion != "crecer": #Si
el valor actual es mayor que el anterior y la dirección
No era crecer, significa que es un mínimo
74             direccion = "crecer" #Se cambia la dirección a
crecer
75             minimos.append(PV[l - 1]) #Se guarda el mínimo en
la lista

```

```

76         tiem.append(tiempo[(l-PV.index[0]-1)]) #Se guarda
el tiempo discreto de la muestra mínima
77         elif PV[l] < PV[l - 1] and direccion != "decrecer":
78             direccion = "decrecer" #Se cambia la dirección a
decrecer
79             maximos.append(PV[l-1]) #Se guarda el máximo en
la lista
80             tiem.append(tiempo[(l-PV.index[0]-1)]) #Se guarda
el tiempo discreto de la muestra máxima
81             prom = [] # Variable para obtener PV característicos
82             min_len = min(len(maximos), len(minimos)) # Determinar
el tamaño mínimo entre las dos listas
83             # Calcular los promedios intercalados según orientación
84             if orientación == 0: # Si orientación es 0, alternamos
de cambios_superiores a cambios_inferiores
85                 for z in range(min_len):
86                     if z < len(maximos):
87                         promedio = (maximos[z] + minimos[z]) / 2 #
Promedio de cambios_superiores[i] con
cambios_inferiores[i]
88                         prom.append(promedio) # Agregar el promedio
a la lista
89                     if z + 1 < len(maximos) and z + 1 < min_len:
90                         promedio = (minimos[z] + maximos[z+1]) / 2 #
Promedio de cambios_inferiores[i] con
cambios_superiores[i+1]
91                         prom.append(promedio) # Agregar el promedio
a la lista
92             elif orientacion == 1: # Si orientación es 1, alternamos
de cambios_inferiores a cambios_superiores

```

```

93         for z in range(min_len):
94             if z < len(minimos):
95                 promedio = (minimos[z] + maximos[z]) / 2 #
Promedio de cambios_inferiores[i] con
cambios_superiores[i]
96                 prom.append(promedio) # Agregar el promedio
a la lista
97                 if z + 1 < len(minimos) and z + 1 < min_len:
98                     promedio = (maximos[z] + minimos[z+1]) / 2 #
Promedio de cambios_superiores[i] con
cambios_inferiores[i+1]
99                     prom.append(promedio) # Agregar el promedio
a la lista
100     t_prom = [] #Variable para obtener tiempos asociados a
los PV característicos
101     for y in range(0, len(tiem)-1) :
102         t_promedio = 0
103         t_promedio = (tiem[y] + tiem[y+1])/2
104         t_prom.append(t_promedio)
105     #Obtener máximos y mínimos pero ahora de la lista de
promedios
106     dir = None
107     maxi =[]
108     for o in range(1, len(prom)) : #Se recorre los valores
promedios
109         if prom[o] > prom[o - 1] and dir != "crecer": #Si el
valor actual es mayor que el anterior y la dirección No
era crecer, significa que es un mínimo
110             dir = "crecer" #Se cambia la dirección a crecer
111         elif prom[o] < prom[o - 1] and dir != "decrecer":

```

```

112         dir = "decrecer" #Se cambia la dirección a
decrecer
113         if prom[o-1] > i and orientacion == 1 :
114             maxi.append(prom[o - 1]) #Se guarda el
máximo en la lista
115         elif prom[o-1] < i and orientacion == 0 :
116             maxi.append(prom[o - 1]) #Se guarda el
máximo en la lista
117         if len(maxi) < 2 : #Se revisa que existan al menos dos
peaks
118             d = "No posee dos peaks consecutivos" #"No posee dos
peaks consecutivos "
119         else :
120             pri_max = maxi[0] - i
121             seg_max = maxi[1] - i
122             d = seg_max / pri_max
123
124
125     #Overshoot
126     if len(maximos) == 0 : #Se revisa si hay máximos
127         ov = "No tiene"
128     else :
129         if orientacion == 1 :
130             ov = ((max(maximos) - i)*100)/i #Se guarda el
valor máximo de la lista y se utiliza la definición de
Overshoot
131         else :
132             ov = abs(((max(maximos) - i)*100)/i) #Se guarda
el valor mínimo de la lista y se utiliza la definición de
Overshoot

```

```

133
134
135     #Desviación Estándar
136     o = lista[pv_name].std() #Función de Python que permite
obtener la Desviación Estándar
137
138
139     #MAE
140     sum = 0 #Variable de suma
141
142     for k in range(PV.index[0], (len(lista) + PV.index[0])) :
#Se recorre toda la lista
143         sum = sum + abs(PV[k] - i) #Se va sumando en valor
absoluto de la diferencia entre PV y referencia
144     MAE_k1 = sum / len(lista) #Se obtiene MAE considerando
el tiempo muestral transcurrido
145     MAE_k =
(MAE_k1*100) / (max(lista[pv_name]) - min(lista[pv_name]))
146
147     #Gráfica en formato tiempo en [s]
148     plt.figure(figsize=(10, 6))
149     plt.plot(tiempo, lista[sp_name], label="Setpoint")
150     plt.plot(tiempo, lista[pv_name], label="Process
Variable")
151     plt.plot(t_prom[:len(prom)], prom, label="Promedios",
color='r', linestyle='--', marker='o')
152     plt.xlabel('Fecha')
153     plt.ylabel('Valor (m3/h)')
154     plt.title(f'Setpoint y Process Variable para SP={i}')
155     plt.legend()

```

```

156     plt.gca().autofmt_xdate()
157     plt.show()
158
159
160     return t_r,ov,d,o,t_s, MAE_k
161
162
163 #Función alternativa de gráfica en formato fecha
164 def fgr(df_lista_temp,i) :
165     plt.figure(figsize=(10, 6))
166     plt.plot(df_lista_temp[date_name],
167             df_lista_temp[sp_name], label="Setpoint")
168     plt.plot(df_lista_temp[date_name],
169             df_lista_temp[pv_name], label="Process Variable")
170     plt.xlabel('Tiempo transcurrido en Segundos')
171     plt.ylabel('Valor de proceso')
172     plt.title(f'Setpoint y Process Variable para SP={i}')
173     plt.legend()
174
175
176     plt.show()
177
178 #Creación Variables
179 indices = df[sp_name] #GUARDAR TODOS LOS VALORES DE LA
180     COLUMNA SP
181
182 SP = set() #SE GUARDARÁN UNA ÚNICA VEZ LOS DISTINTOS SP
183     ENCONTRADOS EN EL LISTADO
184
185 lista = [] # SE GUARDARÁ LA FECHA/PV/SP/OUT DE LOS SP FIJOS
186     ENCONTRADOS

```

```

181 #Proceso para determinar referencias constantes
182 for i in range(len(indices)): #POR CADA i=
    [0,1,2.....,len(indices)] EN RANGO [0,len(indices)]
183     current_value = indices[i] #VALOR DE SP EN LA POSICIÓN i
    EN EL LISTADO INDICES
184     # SI EL TÉRMINO i+5 EXISTE (YA QUE ES MÁS PEQUEÑO QUE EL
    LARGO DE INDICES - 1) Y SI SON IGUALES SUS 5 CONSECUTIVOS
185     if i+5 <= len(indices)-1 and indices[i] == indices[i +
    1] == indices[i + 2] == indices[i + 3] == indices[i + 4]
    == indices[i + 5] :
186         #COMO SON IGUALES, ES UN SP FIJO Y SE GUARDA TODA SU
    FILA EN "LISTA" Y EL VALOR DEL SP EN "SP". NOTAR QUE POR
    COMO ESTÁ DEFINIDO "SP", SI YA EXISTE EL VALOR DEL SP EN
    LA LISTA, NO SE VUELVE A GUARDAR
187         lista.append(df.loc[i])
188         SP.add(indices[i])
189     # SI i ES MAYOR QUE 5 EXISTE Y SI SON IGUALES SUS 5
    ANTERIORES. ESTO PARA ANALIZAR EL INICIO Y FINAL DE LA
    TABLA SIN ERROR
190     elif i > 5 and indices[i] == indices[i - 1] == indices[i
    - 2] == indices[i - 3] == indices[i - 4] == indices[i -
    5] :
191         lista.append(df.loc[i])
192         SP.add(indices[i])
193 df_lista = pd.DataFrame(lista) #SE GUARDA COMO DATA FRAME LA
    LISTA CON FECHA/PV/SP/OUT CON TODOS LOS VALORES DE SP
    FIJOS
194
195
196 #Indicador de porcentaje en Modo Manual/Automático

```

```

197 modo_auto = (len(lista) * 100) / len(df)
198 modo_manual = 100 - modo_auto
199
200 #Proceso para determinar si NO existen periodos de tiempo
    con referencia constante
201 if SP == set() :
202     #Significa que todo el tiempo estuvo en modo manual, por
    lo que no se pueden obtener indicadores para conocer la
    precisión del controlador
203     print("Tiempo de Subida [s] : " , "-")
204     print("Tiempo de Asentamiento [s] : " , "-")
205     print("Razón de Amortiguamiento : " , "-")
206     print("Sobrepaso [%] : " , "-")
207     print("Desviación Estándar : " , "-")
208     print("Mean absolute error [%] : " , "-")
209     print("Modo Automático [%] : " , modo_auto )
210     print("Modo Manual [%] : " , modo_manual)
211     #Creación Reporte
212     resultados = pd.DataFrame({
213         'Indicadores': ["Tiempo de Subida [s]" , "Tiempo de
    Asentamiento [s]" , "Razón de Amortiguamiento" , "Sobrepaso
    [%]" , "Desviación Estándar " , "Mean absolute error [%]" ,
    "Modo Automático [%]" , "Modo Manual [%]" ],
214         'Resultados': ["-" , "-" , "-" , "-" , "-" , "-" , modo_auto ,
    modo_manual]
215     })
216     ubicacion_archivo =
    r"C:\\Users\\xxxx\\xxxxx\\xxxxx\\xxxx.xlsx" #Ubicación de
    almacenamiento del archivo, con nombre.
217

```

```

218     resultados.to_excel(ubicacion_archivo, index=False) #
    Guardar el DataFrame en el archivo Excel en la ubicación
    especificada

219
220
221
222
223
224 for i in SP : #POR CADA i = [SP[0],SP[1],....] EN EL LISTADO
    SP

225
226     #DEFINIR VARIABLES QUE SE REINICIEN CADA QUE SE LLEGA A
    UN NUEVO SP

227     cont = 0 #CONTADOR PARA CONOCER LA POSICIÓN EN EL
    LISTADO SP

228     lista_temp = [] #LISTADO TEMPORAL PARA TENER SUB-LISTA
    DE CADA SP INDIVIDUALMENTE

229
230     for j in df_lista[sp_name] : #POR CADA j =
    [df_lista["SP..."][0],....] EN EL EL LISTADO DE LA
    COLUMNA "SP..." DE LOS SP FIJOS

231
232     if i == j: #SI EL SP EN EL QUE ESTOY (i) ES IGUAL AL
    VALOR EN EL QUE ESTOY DEL LISTADO (j)

233         lista_temp.append(df_lista.iloc[cont]) #SE
    AGREGA AL LISTADO TEMPORAL LA FECHA/PV/SP/OUT DEL SP QUE
    SE ESTÁ ANALIZANDO DADA LA UBICACIÓN OBTENIDA EN CONT

234         cont += 1 #EL CONTADOR AUMENTA YA QUE SE PASA A UNA
    NUEVA FILA

235

```

```

236     df_lista_temp = pd.DataFrame(lista_temp) #YA SE
REVISARON TODAS LAS FILAS, POR LO QUE EL LISTADO SE
CONVIERTE EN DATA FRAME

237
238
239     #PUEDE QUE SEA EL MISMO SP EN DISTINTOS TIEMPOS, POR LO
QUE SE REvisa
240     bln = 0 #VARIABLE PARA SABER SI ES EL CASO, DONDE SU
VALOR VA A CAMBIAR SI ES ASÍ
241     #SE CREA UNA LISTA QUE ALMACENE EL ÍNDICE DONDE OCURREN
LOS PUNTOS DE INFLEXIÓN. EL PUNTO SE RECONOCE YA QUE LA
DIFERENCIA TEMPORAL ENTRE FECHAS ES MAYOR QUE LOS
SEGUNDOS ESPECIFICADOS
242     infl = df_lista_temp[df_lista_temp[date_name].diff() >
pd.Timedelta(seconds=10)].index
243     #SE GUARDA LA POSICIÓN DEL PRIMER TÉRMINO, PUES LAS
POSICIONES NO PARTEN DESDE 0 YA QUE ESTAN DEFINIDAS POR
LA LISTA EXPORTADA
244     posicion = df_lista_temp.index[0]
245     #POR CADA x = [infl[0],...] EN EL LISTADO DE PUNTOS DE
INFLEXIÓN (PDI)
246     for x in infl :
247         #fdc = VARIABLE PARA SABER QUE NO HAY MÁS PDI Y
ANALIZAR EL ÚLTIMO TRAMO DEL LISTADO TEMPORAL
248         fdc = 1
249         #cont2 = PARA CONOCER LA POSICIÓN EN EL LISTADO DE
LISTA TEMPORAL
250         cont2 = 0
251         #lista_temp2 = LISTADO TEMPORAL PARA TENER UNA
SUBLISTA DEL PRIMER TRAMO TEMPORAL

```

```

252     lista_temp2 = []
253     #POR CADA y = [df_list_..[0],...] EN EL LISTADO CON
LAS POSICIONES DEL LISTADO TEMPORAL
254     for y in df_lista_temp.index :
255         #SI MI INDICE DE INFLEXIÓN(x) ES DISTINTO A MI
INDICE ACTUAL(y) Y MI INDICE ACTUAL ES MAYOR O IGUAL A LA
POSICION DE PARTIDA
256         if x != y and y >= posicion :
257             #SE AGREGA AL SUBLISTADO LA FECHA/PV/SP/OUT
DEL TRAMO TEMPORAL
258             lista_temp2.append(df_lista_temp.iloc[cont2])
259             #EL CONTADOR AUMENTA YA QUE SE PASA A UNA
NUEVA FILA
260             cont2 += 1
261             #SI SE LLEGA AL PUNTO DONDE y = x
262             else :
263                 #SE SALE DEL BUCLE for
264                 break
265             #SE CAMBIA EL VALOR DE LA POSICIÓN DE PARTIDA PARA
ANALIZAR LOS SIGUIENTES INDICES
266             posicion = df_lista_temp.index[cont2]
267             #EL SUBLISTADO SE CONVIERTE EN DATA FRAME
268             df_lista_temp2 = pd.DataFrame(lista_temp2)
269
270             #INSPECCIÓN
271             if not df_lista_temp2.empty :
272                 #SE CAMBIA EL VALOR DE BLN PARA SABER QUE SI
EXISTEN PDI
273                 bln = 1
274

```

```

275     #INDICADORES
276     resultado = indicadores(df_lista_temp2, i)
277     if resultado is None :
278         print("No es un cambio de referencia
279 drástico, por lo que no es posible analizar a SP = ", i)
280     else :
281         t_r , ov, d, o, t_s, MAE_k = resultado
282         print("Tiempo de Subida [s] : " , t_r)
283         print("Tiempo de Asentamiento [s] : ", t_s)
284         print("Razón de Amortiguamiento : ", d)
285         print("Sobrepaso [%] : ",ov)
286         print("Desviación Estándar : ", o)
287         print("Mean absolute error [%] : " , MAE_k)
288         print("Modo Automático [%] : " , modo_auto )
289         print("Modo Manual [%] : ", modo_manual)
290         # Creación Reporte
291         resultados = pd.DataFrame({
292             'Indicadores': ["Tiempo de Subida [s]"
293 , "Tiempo de Asentamiento [s]", "Razón de
294 Amortiguamiento", "Sobrepaso [%]", "Desviación
295 Estándar", "Mean absolute error [%]", "Modo Automático
296 [%]", "Modo Manual [%]" ],
297             'Resultados': [t_r ,t_s, d, ov, o,
298 MAE_k, modo_auto , modo_manual]
299         })
300         ubicacion_archivo =
301         r"C:\\Users\\xxxx\\xxxxx\\xxxxx\\xxxx.xlsx" #Ubicación de
302 almacenamiento del archivo, con nombre.

```

```

296         resultados.to_excel(ubicacion_archivo,
index=False) # Guardar el DataFrame en el archivo Excel
en la ubicación especificada

297
298     #FALTA ANALIZAR EL ÚLTIMO TRAMO TEMPORAL DE ESTE SP,
ES LO QUE SE HACE A CONTINUACIÓN. SE REVISA SI NO HAY MÁS
PDI

299     if fdc+1 > len(infl) :
300         #SE REPITE PROCESO ANTERIOR
301         lista_temp2 = []
302         for y in df_lista_temp.index :
303             #SE QUITA UNA CONDICIÓN YA QUE COMO ES EL Ú
LTIMO TRAMO, ACABA CUANDO TERMINEN LOS DATOS
304             if y >= posicion :
305
306         lista_temp2.append(df_lista_temp.iloc[cont2])
307         cont2 += 1
308         df_lista_temp2 = pd.DataFrame(lista_temp2)
309
310     #INDICADORES
311     resultado = indicadores(df_lista_temp2, i)
312     if resultado is None :
313         print("No es un cambio de referencia
drástico, por lo que no es posible analizar a SP = ", i)
314     else :
315         t_r , ov, d, o, t_s, MAE_k = resultado
316         print("Tiempo de Subida [s] : " , t_r)
317         print("Tiempo de Asentamiento [s] : ", t_s)
318         print("Razón de Amortiguamiento : ", d)
319         print("Sobrepaso [%] : ",ov)

```

```

319         print("Desviación Estándar : ", o)
320         print("Mean absolute error [%] : " , MAE_k)
321         print("Modo Automático [%] : " , modo_auto )
322         print("Modo Manual [%] : ", modo_manual)
323         # Creación Reporte
324         resultados = pd.DataFrame({
325             'Indicadores': ["Tiempo de Subida [s]"
326 , "Tiempo de Asentamiento [s]", "Razón de
327 Amortiguamiento", "Sobrepaso [%]", "Desviación
328 Estándar", "Mean absolute error [%]", "Modo Automático
329 [%]", "Modo Manual [%]" ],
330             'Resultados': [t_r , t_s, d, ov, o,
331 MAE_k, modo_auto , modo_manual]
332         })
333         ubicacion_archivo =
334         r"C:\\Users\\xxxx\\xxxxx\\xxxxx\\xxxx.xlsx" #Ubicación de
335         almacenamiento del archivo, con nombre.
336
337         resultados.to_excel(ubicacion_archivo,
338 index=False) # Guardar el DataFrame en el archivo Excel
339         en la ubicación especificada
340
341         #SI NO EXISTIAN PDI, ENTONCES NO SE ENTRA AL CICLO FOR,
342         POR LO QUE SE DEBE PEDIR LA SALIDA
343         if bln == 0 :
344             #INDICADORES
345             resultado = indicadores(df_lista_temp, i)
346             if resultado is None :

```

```

338         print("No es un cambio de referencia drástico,
por lo que no es posible analizar a SP = ", i)
339     else :
340         t_r , ov, d, o, t_s, MAE_k = resultado
341         print("Tiempo de Subida [s] : " , t_r)
342         print("Tiempo de Asentamiento [s] : ", t_s)
343         print("Razón de Amortiguamiento : ", d)
344         print("Sobrepaso [%] : ",ov)
345         print("Desviación Estándar : ", o)
346         print("Mean absolute error [%] : " , MAE_k)
347         print("Modo Automático [%] : " , modo_auto )
348         print("Modo Manual [%] : ", modo_manual)
349         # Creación Reporte
350         resultados = pd.DataFrame({
351             'Indicadores': ["Tiempo de Subida [s]" , "Tiempo
de Asentamiento [s]" , "Razón de
Amortiguamiento" , "Sobrepaso [%]" , "Desviación
Estándar" , "Mean absolute error [%]" , "Modo Automático
[%]" , "Modo Manual [%]" ],
352             'Resultados': [t_r ,t_s, d, ov, o,
MAE_k,modo_auto , modo_manual]
353         })
354         ubicacion_archivo =
r"C:\\Users\\xxxx\\xxxxxx\\xxxxxx\\xxxx.xlsx" #Ubicación de
almacenamiento del archivo, con nombre.
355
356         resultados.to_excel(ubicacion_archivo,
index=False) # Guardar el DataFrame en el archivo Excel
en la ubicación especificada

```