



**Entorno virtual 3D de una planta de nivel, mediante la técnica Hardware-In-the-
Loop, para el control avanzado de procesos**

Alpúsig Cuichán, Silvia Emperatriz

Vicerrectorado de Investigación, Innovación y Transferencia de Tecnología

Centro de Posgrados

Maestría en Electrónica y Automatización mención Redes Industriales

Trabajo de titulación, previo a la obtención del título de
Magíster en Electrónica y Automatización mención Redes Industriales

Mgs. Pruna Panchi, Edwin Patricio

20 de Agosto del 2021



UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN
MENCIÓN REDES INDUSTRIALES

Certificación

Certifico que el trabajo de titulación, **“Entorno virtual 3D de una planta de nivel, mediante la técnica Hardware-In-the-Loop, para el control avanzado de procesos”** fue realizado por la señora Ing. **Alpúsig Cuichán, Silvia Emperatriz** el mismo que ha sido revisado y analizado en su totalidad, por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Latacunga, 20 de agosto de 2021

Firma:

.....
Mgs. Pruna Panchi, Edwin Patricio

Director

C.C.: 0502651003



Document Information

Analyzed document Proyecto de titulación Silvia Alpúsig.pdf (D110292478)
Submitted 7/9/2021 4:13:00 AM
Submitted by Pruna Panchi Edwin Patricio
Submitter email epruna@espe.edu.ec
Similarity 3%
Analysis address epruna.espe@analysis.arkund.com

Sources included in the report

Universidad de las Fuerzas Armadas ESPE / Arkund Trabajo de titulación Pepe Ibañez.pdf

Document Arkund Trabajo de titulación Pepe Ibañez.pdf (D96289388)
 Submitted by: ipescobar@espe.edu.ec
 Receiver: ipescobar.espe@analysis.arkund.com 1

Universidad de las Fuerzas Armadas ESPE / Proyecto de Investigación_Zambrano_Caicedo.pdf

Document Proyecto de Investigación_Zambrano_Caicedo.pdf (D36144601)
 Submitted by: epruna@espe.edu.ec
 Receiver: epruna.espe@analysis.arkund.com 3

URL: <http://www.altadisponibilidadlogitek.com/wp-content/uploads/White-paper-Virtualization-in-Industrial-Plants-Industrial-Automation-ESSP-1.pdf>
 Fetched: 7/9/2021 4:14:00 AM 1

URL: http://blender.tecnawebs.es/conceptos_basicos.html
 Fetched: 7/9/2021 4:14:00 AM 2

Firma:

.....
Mgs. Pruna Panchi, Edwin Patricio

Director

C.C.: 0502651003



UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN
MENCIÓN REDES INDUSTRIALES

Responsabilidad de Autoría

Yo **Alpúsig Cuichán, Silvia Emperatriz**, con cédula de ciudadanía Nro. 0502779697 declaro que el contenido, ideas y criterios del trabajo de titulación: **“Entorno virtual 3D de una planta de nivel, mediante la técnica Hardware-In-the-Loop, para el control avanzado de procesos”** es de mí autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 20 de agosto de 2021

.....
Alpúsig Cuichán, Silvia Emperatriz

C.C.: 0502779697



**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN
MENCIÓN REDES INDUSTRIALES**

Autorización de Publicación

Yo **Alpúsig Cuichán, Silvia Emperatriz** con cédula de ciudadanía Nro. 0502779697 autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **“Entorno virtual 3D de una planta de nivel, mediante la técnica Hardware-In-the-Loop, para el control avanzado de procesos”** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Latacunga, 20 de agosto de 2021

.....
Alpúsig Cuichán, Silvia Emperatriz

C.C.: 0502779697

Dedicatoria

Dedico este trabajo a Dios y a la Virgen María por la vida y la oportunidad de vivirla cada día junto a mis seres queridos.

A mis padres Juan y Emperatriz por darme un ejemplo diario de trabajo constante y amor incondicional.

A mi esposo Gustavo por su apoyo para la culminación de mis objetivos, sobre todo por el amor, respeto y comprensión que me brinda día a día.

A mis hijos Daniel y Felipe por su cariño y amor que permiten que me desarrolle personal y profesionalmente.

A mis hermanos Byron y Juan Carlos por el apoyo y el ejemplo de arduo trabajo para sacar adelante a sus familias.

A Meli, Vero, mis sobrinos y familiares que me apoyaron con su granito de arena para que pueda tener el tiempo de dedicación y culminar este trabajo, mis abuelitas Tomasa y María que desde el cielo sé que están conmigo apoyándome, el esfuerzo de este trabajo es para todos ustedes.

Silvia Emperatriz Alpúsig Cuichán

Agradecimiento

Quiero expresar mi gratitud a Dios por regalarme salud y vida, por cuidarnos día a día y permitirnos seguir compartiendo momentos con nuestros seres queridos.

A Gustavo, Daniel y Felipe que son el motor que impulsan para cumplir mis sueños, gracias por su amor y comprensión que complementa mi vida.

A mis padres y hermanos por su ejemplo de trabajo y lucha constante para no rendirse ante las adversidades, buscando siempre día mejores para nuestras familias.

A mis amigos y compañeros de trabajo con quienes comparto experiencias de aprendizaje y de vida.

A los docentes de la Universidad de la Fuerzas Armadas ESPE sede Latacunga, de manera especial a Ivón y Edwin quienes me impulsaron para culminar con éxito este trabajo.

Silvia Emperatriz Alpúsig Cuichán

Tabla de contenidos

Carátula	1
Certificación	2
Responsabilidad de Autoría	4
Autorización de Publicación	5
Dedicatoria	6
Agradecimiento	7
Tabla de contenidos	8
Índice de Tablas	11
Índice de Figuras	12
Resumen	16
Abstract	17
Contenidos Generales	18
Antecedentes	18
Planteamiento del problema	19
Objetivos	20
<i>Objetivo General</i>	20
<i>Objetivos Específicos</i>	20
Justificación, importancia y alcance del proyecto	20
Hipótesis de investigación	22
<i>Categorización de las variables de investigación</i> <i>(variables independiente y dependiente)</i>	22
Fundamentación Teórica	24
Marco Conceptual	24
<i>Realidad Virtual</i>	24
Realidad Virtual Inmersiva	24
Realidad Virtual de Escritorio	25
<i>Software Libre</i>	26
Unity 3D	26
<i>Comunicación serie entre Unity y Arduino</i>	27
Librerías	27
<i>Diagrama de Instrumentación y Tuberías P&ID</i>	28

<i>Procesos industriales</i>	28
Proceso autorregulado	28
Proceso integrante	29
<i>Control de Procesos</i>	31
Control básico SISO	31
Control en cascada	32
<i>Controlador PID</i>	33
Métodos de sintonía para procesos autorregulados	34
Métodos de sintonía para procesos Integrantes	34
Método de sintonización del tanteo	35
<i>Diseño de procesos industriales con herramientas de dibujo asistido por computador 3D (Blender)</i>	36
Malla	37
Ray-tracing	37
Render	37
<i>Hardware-In-the-Loop</i>	37
<i>PLC (Controlador lógico Programable)</i>	38
Metodología de la Investigación	40
Esquema de la Planta Industrial	41
Tratamiento del Agua de Alimentación de la Caldera	42
Generación de Vapor Mediante una Caldera	43
Respuesta Dinámica y Control del Sistema	44
<i>Características Dinámicas del Sistema</i>	46
Control en Cascada del Nivel de Agua de la Caldera	49
<i>Diseño de los Controladores Primario y Secundario</i>	49
Diseño del entorno virtual 3D	53
<i>Modelado del Entorno Virtual</i>	53
<i>Desarrollo de la aplicación en Unity3D</i>	61
Diseño del entorno virtual.	61
Programación de las características de la aplicación.	68
<i>Adquisición y Generación de Señales</i>	76
<i>Implementación de los Algoritmos de Control en el PLC</i>	78
Resultados	85

	10
Rendimiento de la Aplicación	85
Desempeño de la estrategia de control	86
<i>Comparación entre los métodos de sintonía utilizados para el control en cascada</i>	87
<i>Comparación Entre el Control Clásico de Lazo Único y el Control en Cascada</i>	88
Comprobación de la hipótesis	95
Análisis	100
Conclusiones y Recomendaciones	101
Conclusiones	101
Recomendaciones	102
Bibliografía	103

Índice de Tablas

Tabla 1 <i>Operacionalización de las Variables</i>	23
Tabla 2 <i>Métodos de Sintonía para un Controlador PID para Procesos Autorregulados</i> 34	34
Tabla 3 <i>Métodos de sintonía para un controlador PID para procesos integrantes</i>	35
Tabla 4 <i>Lista de instrumentos del proceso</i>	45
Tabla 5 <i>Constantes de sintonía para el controlador secundario</i>	49
Tabla 6 <i>Constantes de sintonía para el controlador primario</i>	52
Tabla 7 <i>Parámetros de rendimiento para cada método de sintonía para el control en cascada</i>	87
Tabla 8 <i>Parámetros de Rendimiento para cada Estrategia de Control - Método de Sintonía de Ford</i>	89
Tabla 9 <i>Parámetros de Rendimiento para cada Estrategia de Control - Método de Sintonía de Astrom - Hagglund</i>	90
Tabla 10 <i>Parámetros de rendimiento para cada estrategia de control - Método de sintonía de Hay</i>	92
Tabla 11 <i>Parámetros de Rendimiento para cada Estrategia de Control - Método de Sintonía de Zou - Brigham</i>	93
Tabla 12 <i>Constantes de Sintonía de los Controladores Primario y Secundario obtenidas por el Método Empírico</i>	93
Tabla 13 <i>Parámetros de Rendimiento para cada Estrategia de Control - Método de Sintonía Empírico</i>	94

Índice de Figuras

Figura 1 <i>Realidad virtual inmersiva empleando gafas óculos</i>	25
Figura 2 <i>Realidad virtual de escritorio (teclado, mouse joysticks)</i>	25
Figura 3 <i>Librería .NET 2.0</i>	28
Figura 4 <i>Respuesta de un Proceso Autorregulado.</i>	29
Figura 5 <i>Respuesta de un Proceso Integrante</i>	30
Figura 6 <i>Control Automático de Procesos</i>	31
Figura 7 <i>Sistema de Control SISO</i>	32
Figura 8 <i>Estructura de un sistema de control en cascada</i>	33
Figura 9 <i>Planta industrial modelada en Blender</i>	36
Figura 10 <i>PLC Siemens S7-1200</i>	38
Figura 11 <i>Arquitectura del sistema virtual 3D de una planta de nivel mediante la técnica Hardware- In-the-Loop</i>	40
Figura 12 <i>Diagrama P&ID del sistema de generación de vapor</i>	42
Figura 13 <i>Estructura de control en cascada para la regulación del nivel de agua de la caldera</i>	46
Figura 14 <i>Comportamiento de un proceso autorregulado de primer orden más tiempo muerto</i>	47
Figura 15 <i>Comportamiento de un proceso integrante más tiempo muerto</i>	48
Figura 16 <i>Simulación del Controlador PID Secundario (caudal) para los 4 métodos de sintonía considerados</i>	50
Figura 17 <i>Respuesta Dinámica del Proceso Primario (nivel) incluido el Controlador Secundario (Caudal)</i>	51
Figura 18 <i>Simulación del Controlador PID Primario (nivel) para los 4 Métodos de Sintonía considerados</i>	52

Figura 19 <i>Modelo Tridimensional de una Caldera de Vapor diseñada en Blender</i> (Object Mode y Edit Mode)	54
Figura 20 <i>Procedimiento de corrección en la orientación de las caras de la malla de un objeto</i>	55
Figura 21 <i>Número de vértices de un objeto</i>	56
Figura 22 <i>Vértices duplicados removidos de la malla del objeto</i>	56
Figura 23 <i>Configuración del punto de rotación de un objeto</i>	57
Figura 24 <i>Creación del mapa UV de un objeto</i>	58
Figura 25 <i>Diseño de la Planta Virtual para la Generación de Vapor a través de una Caldera</i>	59
Figura 26 <i>Render del Transmisor de Caudal</i>	59
Figura 27 <i>Render del Transmisor de Nivel</i>	60
Figura 28 <i>Render de la válvula de control neumática y las válvulas manuales de bypass</i>	60
Figura 29 <i>Interfaz de Unity3D</i>	61
Figura 30 <i>Modelo importado añadido a la escena en Unity3D</i>	62
Figura 31 <i>Creación y asignación de materiales en Unity3D</i>	63
Figura 32 <i>Materiales Asignados a cada Componente del Entorno Virtual</i>	64
Figura 33 <i>Directional Light y Skybox añadidos a la escena</i>	65
Figura 34 <i>Configuraciones para la generación de mapas de iluminación</i>	66
Figura 35 <i>Entorno virtual en Unity3D – Caldera de vapor</i>	67
Figura 36 <i>Entorno virtual en Unity3D – Válvula de control y válvulas manuales de bypass</i>	67
Figura 37 <i>Entorno virtual en Unity3D – Etapa de tratamiento de agua</i>	68

Figura 38 <i>GameObject “Player” implementado para el movimiento del usuario en la escena.....</i>	69
Figura 39 <i>Canvas “World Space” con elementos UI.....</i>	70
Figura 40 <i>HMI diseñado con herramientas UI en Unity3D.....</i>	71
Figura 41 <i>Datos en el Transmisor de Caudal.....</i>	72
Figura 42 <i>Datos en el transmisor de nivel.....</i>	72
Figura 43 <i>Indicador de Nivel.....</i>	73
Figura 44 <i>Animaciones de la válvula de control neumática.....</i>	73
Figura 45 <i>Válvula Manual Manipulable.....</i>	74
Figura 46 <i>Componente de sonido dentro del entorno virtual.....</i>	75
Figura 47 <i>Respuesta del Sistema Simulado ante una Entrada Escalón.....</i>	76
Figura 48 <i>Acondicionamiento de Señales.....</i>	77
Figura 49 <i>Intercambio de datos entre Unity3D y Arduino.....</i>	78
Figura 50 <i>Vista de Configuración del PLC y los Módulos de Entradas y Salidas Analógicas (Tia Portal).....</i>	79
Figura 51 <i>Configuración de los Canales o Entradas Analógicas del Módulo SM 1231 (Tia Portal).....</i>	80
Figura 52 <i>Configuración de los Canales o Salidas Analógicas del Módulo SM 1232 (Tia Portal).....</i>	80
Figura 53 <i>Normalización y escalamiento de señales de entrada (Lenguaje KOP – Tia Portal).....</i>	81
Figura 54 <i>Normalización y escalamiento de señales de salida (Lenguaje KOP – Tia Portal).....</i>	81
Figura 55 <i>Escalamiento de la salida del controlador primario (Lenguaje KOP – Tia Portal).....</i>	82

Figura 56 Programación de los controladores PID (Lenguaje KOP - Tia Portal)	83
Figura 57 Ventana de configuración de parámetros del bloque PID_Compact	84
Figura 58 Parámetros de rendimiento de la aplicación	85
Figura 59 Respuesta del sistema virtual por control en cascada	86
Figura 60 Comparación entre métodos de sintonía para el control en cascada (PLC)	87
Figura 61 Gráfica comparativa para el control clásico de lazo único vs. control en cascada (Método de Ford)	89
Figura 62 Gráfica Comparativa para el Control Clásico de Lazo único vs. Control en Cascada (Método de Astrom - Hagglund).....	90
Figura 63 Gráfica Comparativa para el Control Clásico de Lazo único vs. Control en Cascada (Método de Hay)	91
Figura 64 Gráfica comparativa para el control clásico de lazo único vs. control en cascada (Método de Zou - Brigham).....	92
Figura 65 Gráfica Comparativa para el Control Clásico de Lazo Único vs. Control en Cascada (Método Empírico)	94
Figura 66 Resultado de la Pregunta 1	96
Figura 67 Resultado de la Pregunta 2.....	96
Figura 68 Resultado de la Pregunta 3.....	97
Figura 69 Resultado de la Pregunta 4.....	97
Figura 70 Resultado de la Pregunta 5.....	98
Figura 71 Resultado de la Pregunta 6.....	98
Figura 72 Resultado de la Pregunta 7.....	99
Figura 73 Resultado de la Pregunta 8.....	99

Resumen

Este trabajo presenta el desarrollo de un sistema basado en la simulación Hardware-In-the-Loop (HIL) para llevar a cabo pruebas de controladores avanzados de procesos de manera particular el control en cascada, el diseño del entorno virtual consta de una planta industrial la misma que está completamente caracterizada para emular el comportamiento real, de un sistema de control de nivel del agua de alimentación de una caldera de vapor, el entorno virtual ha sido implementado utilizando Blender y el motor gráfico Unity3D, logrando de este modo un alto nivel de realismo en la instrumentación utilizada así como en sus características físicas y operativas, además, se ha implementado un Arduino como tarjeta de adquisición de datos dispositivo capaz de leer y generar señales eléctricas de 0 a 5 VDC para las variables de entrada y salida entre la planta virtual y el controlador alojado en un PLC S7-1200. El proyecto se identifica como un aporte para el uso del entorno de realidad virtual didáctico para que los profesionales y estudiantes de control de procesos, realicen su entrenamiento en el manejo de plantas industriales y la estrategia de control en cascada de manera interactiva con un bajo costo de implementación en el uso de laboratorios.

Palabras Clave:

- **ENTORNO VIRTUAL 3D.**
- **HARDWARE-IN-THE-LOOP (HIL).**
- **CONTROLADOR EN CASCADA.**
- **ESTACIÓN DE PROCESOS.**
- **SISTEMA DE ENTRENAMIENTO.**

Abstract

This work presents the development of a system based on Hardware-In-the-Loop (HIL) simulation to carry out tests of advanced process controllers, particularly cascade control, the design of the virtual environment consists of an industrial plant the same one that is fully characterized to emulate the real behavior of a feedwater level control system of a steam boiler, the virtual environment has been implemented using Blender and the Unity3D graphic engine, thus achieving a high level of realism in the instrumentation used as well as in its physical and operational characteristics, in addition, an Arduino controller has been implemented as a data acquisition card, a device capable of reading and generating electrical signals from 0 to 5 VDC for the input and output variables between the virtual plant and controller located in a S7-1200 PLC. This project is identified as a contribution to the use of the didactic virtual reality environment for process control professionals and students to carry out their training in the management of industrial plants and the cascade control strategy in an interactive manner due to its low implementation cost in laboratories.

Key Words:

- **3D VIRTUAL ENVIRONMENT.**
- **HARDWARE-IN-THE-LOOP (HIL).**
- **CASCADE CONTROLLER**
- **PROCESSOR STATION.**
- **TRAINING SYSTEM.**

CAPÍTULO I

1. Contenidos Generales

1.1 Antecedentes

El uso de la realidad virtual como herramienta de aprendizaje y entrenamiento en diferentes campos entre ellos la ingeniería actualmente está en auge, el diseño de aplicaciones de realidad virtual han permitido el desarrollo de prototipos virtuales que permiten tener una idea clara del funcionamiento real de cualquier entorno (Raposo, Soares, Wagner, Corseuil, & Gatass, 2009), bajo este concepto se han aprovechado las bondades que ofrecen los simuladores virtuales, para integrar dispositivos de diferentes marcas y licencias patentadas que encarecen los modelos comerciales de procesos industriales, desarrollando simuladores de bajo costo. (Bin, Wu, Figueroa, & Monti, 2007) (De Farias, Rodriguez, Murillo , Lopes, & Avila , 2019). La simulación de cualquier proceso industrial permite conocer el comportamiento real de un proceso y su respuesta aplicando diferentes tipos de controladores avanzados (Velásquez, Collazos , & Mines, 2017).

Por otro lado la técnica de simulación de Hardware-In-the-Loop (HIL) proporciona plataformas eficaces para el ensayo de hardware y software, esta técnica se utiliza para el desarrollo y pruebas de control en el funcionamiento de máquinas y sistemas complejos en tiempo real, con la simulación HIL la parte física de una máquina o sistema es reemplazada por una simulación (Fang, Zhou, Ma, & Yang, 2018) (De Farias, Rodriguez, Murillo , Lopes, & Avila , 2019) (Bacic, 2005)

La técnica HIL también pretende realizar pruebas de rendimiento del sistema de control a través de las condiciones del proceso de producción, aplicando señales digitales o analógicas, para asegurar que el modelo y el sistema de simulación puede

realizar el monitoreo en tiempo real de los parámetros del equipo, proporcionando un entorno de simulación de control más realista. (Zhao & Tian, 2018).

1.2 Planteamiento del problema

Actualmente el aprendizaje y entrenamiento en sistemas de control avanzado de procesos; haciendo referencia al control en cascada se ha vuelto una necesidad importante para el desarrollo de destrezas técnicas en los estudiantes a nivel de tecnología y pregrado en los centros de educación superior.

Esto debido a que a nivel industrial los procesos ya no simplemente están basados en sistemas de control tradicionales, si no que por su complejidad y la aplicación de nuevas tecnologías se han empleado nuevas estrategias de control, que requieren de personal capacitado para su implementación y operación.

Uno de los inconvenientes para la falta de capacitación del futuro profesional en control automático es que se requiere de laboratorios implementados con estaciones y equipos que tengan la capacidad de realizar controles avanzados de procesos (Control en cascada), estas estaciones tienen un costo elevado en el mercado actual lo que causa que no se cuente con el número adecuado de estaciones por estudiante, es por esto que existe la necesidad de implementar nuevas herramientas en el proceso enseñanza – aprendizaje de control automático de procesos.

Por lo que se plantea desarrollar un sistema didáctico empleando nuevas tecnologías de simulación de bajo costo; entre las más destacadas tenemos el diseño de entornos en realidad virtual que tienen la capacidad de generar un acercamiento real entre el proceso y el estudiante, además que se aplicará la técnica HIL con la que se puede probar fácilmente de una manera práctica, barata y no peligrosa el modelo de la planta en tiempo real.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar un entorno virtual 3D de una planta de nivel, mediante la técnica Hardware-In-the-Loop, para el control avanzado de procesos.

1.3.2 Objetivos Específicos

- Obtener los modelos matemáticos que representen la respuesta dinámica de un sistema de control de nivel (cascada).
- Diseñar el entorno virtual 3D de la planta de nivel, mediante normas y estándares internacionales.
- Implementar los algoritmos de control PI-PID, para el control en cascada mediante la técnica Hardware-In-the-Loop (HIL).
- Validar el sistema virtual 3D del proceso de nivel mediante pruebas experimentales.

1.4 Justificación, importancia y alcance del proyecto

En la industria de producción moderna el nivel representa un parámetro de control importante, con el manejo de esta variable podemos monitorear y controlar la producción a través del nivel de un contenedor con diferentes productos (gas, Líquidos, crudo) (Hong & Deng, 2016). Por otro lado, los procesos industriales actuales han implementado sistemas de control modernos que van ganando gran notoriedad por sus resultados en la producción, entre éstos está el control en cascada que permite introducir una doble acción correctiva mediante dos lazos de control por realimentación (Adam, 2020)

Bajo este contexto se vuelve de gran importancia el entrenamiento del futuro profesional en el manejo de estaciones de nivel con la implementación de la estrategia de control en cascada que le permitirán volverse más competitivo en el campo laboral.

Para lograr este objetivo es necesario que los centros de educación superior cuenten con el número adecuado de estaciones de aprendizaje por alumno en los laboratorios de control de procesos, sin embargo, el costo que representa esta implementación dificulta cada vez más el proceso enseñanza – aprendizaje, por lo que se propone el desarrollo de nuevas herramientas de simulación a nivel industrial para que los alumnos puedan practicar en un entorno lo más cercano al proceso real.

El diseño de un entorno virtual 3D de un proceso industrial para la enseñanza y aprendizaje de control de procesos va a permitir mejorar la calidad de los métodos y técnicas de enseñanza en ingenieros y alumnos de las carreras técnicas, la técnica Hardware-In-the-Loop (HIL) permite la aplicación de un controlador físico a una simulación en tiempo real de la planta, es por esto que se considera un controlador real mientras que la planta es virtual, lo que permitirá reducir costos en equipos e instrumentos siendo accesibles para la utilización de los estudiantes de tecnología y pregrado.

Actualmente los estudiantes de tecnología no cuentan con laboratorios de control de procesos implementados, sin embargo, cuentan con equipos y controladores destinados al proceso enseñanza – aprendizaje práctico que podrían ser utilizados en el entrenamiento sobre el funcionamiento de entornos industriales siendo parte de un sistema simulado.

El alcance del proyecto se limita al desarrollo del entorno virtual e implementación del modelo matemático de la estación de nivel, aplicando la técnica de simulación Hardware in the Loop (HIL), la estrategia de control en cascada será desarrollada mediante un controlador lógico programable PLC, utilizando una computadora portátil con software de diseño.

1.5 Hipótesis de investigación

El desarrollo de entornos en realidad virtual con técnicas de simulación HIL proporcionará un sistema de bajo costo para el proceso enseñanza – aprendizaje de control en cascada de la variable Nivel.

1.5.1 Categorización de las variables de investigación (variables independiente y dependiente)

En consecuencia, de la hipótesis planteada se identifican dos variables:

- Variable Independiente: Sistemas de realidad virtual y técnicas de Hardware-in-the-Loop.
- Variable Dependiente: Aprendizaje práctico de control en cascada.

La operacionalización de las variables se muestra en la siguiente Tabla 1:

Tabla 1*Operacionalización de las Variables*

Variables	Tipo	Definición Conceptual	Definición Operacional	Dimensiones	Indicadores
Aprendizaje práctico de control en cascada	Dependiente	Este tipo de aprendizaje es conocido como learning by doing; y busca no solo poner en práctica lo que uno estudia, sino que se aprenda directamente al ponerse en la situación y el contexto laboral planteado	El aprendizaje práctico consiste en simulaciones de casos prácticos según el conocimiento teórico que van adquiriendo los alumnos universitarios en las aulas.	<ul style="list-style-type: none"> - Actitudes y percepciones - Adquirir e integrar el conocimiento Facilidad de uso - Uso significativo del conocimiento 	<ul style="list-style-type: none"> - Encuestas - Resultados de porcentaje de disminución en dificultades del aprendizaje.
Sistemas de realidad virtual y técnicas de Hardware-In-the- Loop	Independiente	La Realidad Virtual (RV) es una simulación de un Ambiente Virtual (AV) creado por computadora en el que el usuario tiene la sensación de estar dentro de ese entorno al mismo tiempo que puede interactuar usando tecnología de hardware llena de posibilidades, con aplicaciones en múltiples áreas tales como educación	Los sistemas de realidad virtual aplicada a control en cascada de procesos permiten obtener herramientas de aprendizaje prácticas para el manejo de entornos industriales.	<ul style="list-style-type: none"> Funcionamiento del entorno virtual Implementación del sistema Puesta en servicio del sistema 	<ul style="list-style-type: none"> Pruebas funcionales para validación del entorno virtual Semejanzas del entorno virtual del proceso de nivel desarrollado en realidad virtual con una estación de procesos física.

CAPÍTULO II

2. Fundamentación Teórica

2.1 Marco Conceptual

2.1.1 *Realidad Virtual*

Según A. Rowell : “La Realidad Virtual es una simulación interactiva por computador desde el punto de vista del participante, en la cual se sustituye o se aumenta la información sensorial que recibe”. El uso de la realidad virtual se ha popularizado en los últimos años. Una de las disciplinas más utilizadas dentro de la realidad virtual es la simulación, la cual consiste en representar objetos 3D del mundo real con animaciones e interacciones para una experiencia de aprendizaje atractiva.

La realidad virtual está siendo implementada en muchas plataformas como educación, medicina, marina, entrenamiento, ingeniería y operaciones militares (P. Hafner, 2013). Según (Atkins, 2009) la realidad virtual se trata de entornos 3D interactivos, accesibles, personalizables y programables, con numerosas potencialidades para prácticas e investigaciones educativas. Existen dos tipos de realidad virtual las cuales son:

2.1.1.1 **Realidad Virtual Inmersiva**

Un sistema de realidad virtual inmersiva busca que el usuario pierda contacto con el mundo real al percibir los estímulos que se proveen de un entorno virtual. (Cantón Enríquez, Arellano Pimentel, & Hernández López, 2017). La realidad virtual inmersiva se puede distinguir de dos formas: La primera en la cual se utilizan gafas o cascos que al usuario le permiten estar aislados del mundo exterior. Y la segunda, las llamadas cuevas de realidad virtual, las cuales son una sala en que las paredes rodean

al usuario produciendo imágenes tridimensionales en diferentes tipos de proyección y produciendo la inmersión total.

Figura 1

Realidad virtual inmersiva empleando gafas óculos



Nota. Tomado de (Cantón Enríquez, Arellano Pimentel, & Hernández López, 2017)

2.1.1.2 Realidad Virtual de Escritorio

La realidad virtual de escritorio es aquella que permite interactuar con el entorno virtual a través de un teclado, ratón, joystick, etc. Generalmente se utiliza pantallas y computadores donde se muestra un mundo virtual, las cuales poseen la capacidad de reproducir contenido multimedia. La realidad virtual de escritorio es menos costosa de implementarla a comparación con la realidad virtual inmersiva. (Fällman, 1999).

Figura 2

Realidad virtual de escritorio (teclado, mouse joysticks)



Nota. Tomado de (Togores, 2019)

La virtualización de los laboratorios, procesos o sistemas industriales llega con el fin de propiciar un funcionamiento más inteligente, seguro y rápido, simplificando la gestión de los sistemas en general y obteniendo una mayor eficiencia del mismo.

(Logitek, 2021)

Virtualizar un laboratorio trae muchas ventajas, dentro de ellas está la formación y capacitación de profesionales a partir de la realidad virtual, permitiendo formar técnicos especializados, los cuales interactúen con máquinas y su funcionamiento con la ayuda de un entorno virtual. Además, la realidad virtual nos ayuda a realizar simulaciones de riesgo y enseñar al personal de trabajadores a seguir los protocolos de seguridad y manejar los equipos evitando así las situaciones de riesgo y accidentes.

(Altertecnica, 2018)

2.1.2 Software Libre

De acuerdo con (Oscar Arriola Navarrete y Avila González, 2008) el software libre proporciona algunas “libertades” para los usuarios entre las que se destaca la posibilidad de: ejecutar, modificar, copiar y distribuir el software, en este tipo de software se proporciona el código fuente del programa, pudiendo modificarlo y adecuarlo para mejorar así sus prestaciones.

2.1.2.1 Unity 3D

El software Unity 3D es un motor gráfico para la creación de video juegos, esta plataforma de desarrollo flexible y con grandes prestaciones permite crear entornos virtuales 2D y 3D multiplataforma con animaciones, sonidos e interacciones logrando un nivel de realismo muy similar al entorno real.

En un principio el software Unity 3D fue creado para desarrollar video juegos sin embargo con la evolución de la tecnología este software se ha ido actualizando

incluyéndose herramientas para el desarrollo de aplicaciones de realidad virtual y realidad aumentada. (Vegas, 2019)

El software Unity 3D dispone de diversas herramientas y cualidades para el desarrollo de aplicaciones dentro de las cuales están las siguientes:

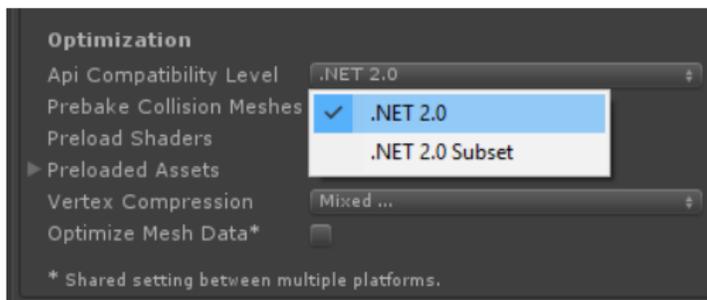
- El editor visual de Unity posee herramientas intuitivas que desarrollan juegos inmersivos para mejorar la capacidad en el manejo del programa.
- Unity puede usarse junto con otros programas de modelación 3D como por ejemplo Blender, 3ds Max, Maya, etc., haciendo más fácil la incorporación de estos objetos al entorno virtual, si se realizan cambios a los objetos creados con estos productos se actualizan automáticamente en todas las instancias de ese objeto durante todo el proyecto sin necesidad de volver a importar manualmente. (Unity, 2020)

2.1.3 Comunicación serie entre Unity y Arduino

Unity posee un paquete de comunicación serial asíncrona, lo que permite la configuración del puerto serie para intercambiar mensajes entre Unity y Arduino, se requiere el lenguaje de programación C# y scripts de Arduino para lograr dicho objetivo. Es importante indicar que se deben utilizar etiquetas tanto para Unity y Arduino, de manera que se pueda evitar confusiones al momento de asociar los datos en la comunicación. (Zucconi, 2015)

2.1.3.1 Librerías

Aun cuando Unity posee scripts de programación para la comunicación en C#, el software no suele incluir todas las librerías necesarias para realizar la comunicación, por lo que es necesario incluir la librería completa de .NET 2.0 en sus ejecutables.

Figura 3*Librería .NET 2.0*

Nota. Tomado de (Coughran, 2021)

2.1.4 Diagrama de Instrumentación y Tuberías P&ID

Un diagrama de instrumentación y tuberías o más conocido como diagrama P&ID por sus siglas en inglés, es un esquema que muestra la interconexión de equipos e instrumentos para el control de un proceso. Estos diagramas son comúnmente utilizados en el área de ingeniería. (Lucidchar, 2021)

2.1.5 Procesos industriales

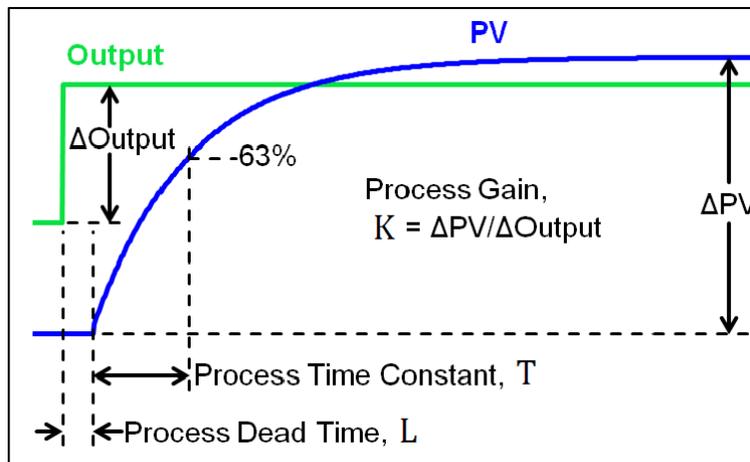
Existen dos tipos de procesos industriales que se clasifican según su respuesta: Autorregulados e integrantes. (King, 2016).

2.1.5.1 Proceso autorregulado

Un proceso autorregulado o autorregulatorio tiene la característica de fijar la variable del proceso a un nuevo valor sin necesidad de que el controlador haga una corrección, cuando se presenta un cambio en el valor de referencia (Villajulca, 2020). Generalmente estos procesos se pueden representar con un modelo matemático de primer orden el cual se lo puede obtener a partir de su gráfica característica que se presenta a continuación:

Figura 4

Respuesta de un Proceso Autorregulado.



Nota. Tomado de (Coughran, 2021)

De acuerdo a las características de la Figura 4 se determina la función de transferencia de un modelo FOPDT (First Order Plus Dead Time):

$$G(s) = \frac{K \cdot e^{-Ls}}{1 + Ts} \quad (1)$$

En dónde:

T = Constante de tiempo del Proceso.

L = Tiempo muerto del proceso.

La ganancia K se obtiene a partir de:

$$K = \frac{\Delta PV}{\Delta Output} \quad (2)$$

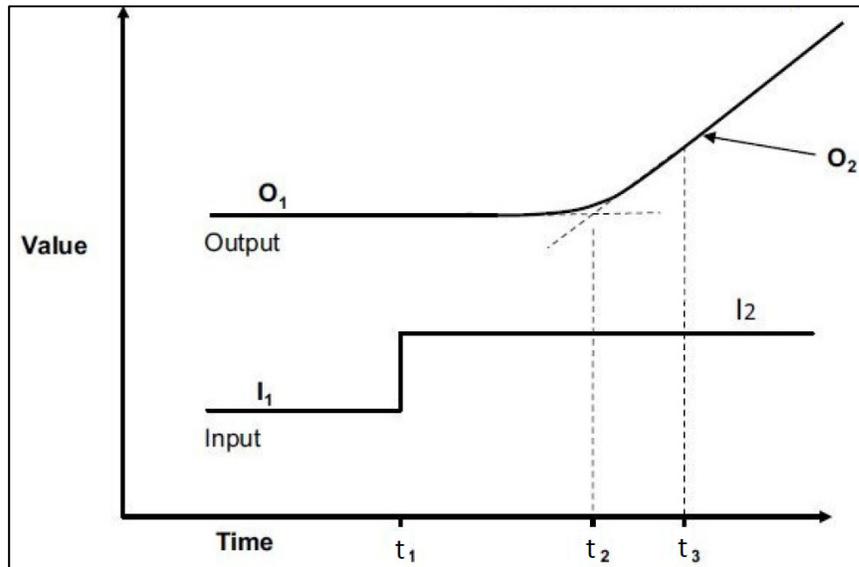
2.1.5.2 Proceso integrante

Según (Camacho, 1998), describe los procesos integrantes como sistemas que están caracterizados por un crecimiento sostenido de la variable de proceso, esto quiere decir que su salida tiene forma de rampa, cuando se hace un cambio en el valor de

referencia. Generalmente esto ocurre cuando se presenta un desbalance de la energía de flujo de entrada y salida del proceso.

Figura 5

Respuesta de un Proceso Integrante



Nota. Tomado de (Otero & Optimo Group Inc., 2020)

De las características del proceso integrante indicadas en la Figura 5 se puede representar el modelo matemático con:

$$G(s) = \frac{K * e^{-Ls}}{s} \quad (3)$$

En dónde la ganancia del proceso integrante K se obtiene a partir de:

$$K = \frac{m}{t_3 - t_2} \quad (4)$$

Para obtener m se utiliza la siguiente ecuación:

$$m = \frac{O_2 - O_1}{t_3 - t_2} \quad (5)$$

El tiempo muerto está representado por:

$$L = t_2 - t_1 \quad (6)$$

2.1.6 Control de Procesos

El control de procesos es una rama de la automatización que involucra la medición y el análisis de las variables que están presentes en el funcionamiento de un proceso, así como la toma de decisiones y la ejecución de acciones de control para gobernar el mismo. (López, 2001)

Actualmente el control de procesos se realiza utilizando tecnología electrónica la cual está basada en la adquisición de datos que son extraídos directamente del proceso para ser analizados con el fin de tomar decisiones de control con un mínimo de intervención humana.

Figura 6

Control Automático de Procesos



Nota. Tomado (AT Automation, 2019)

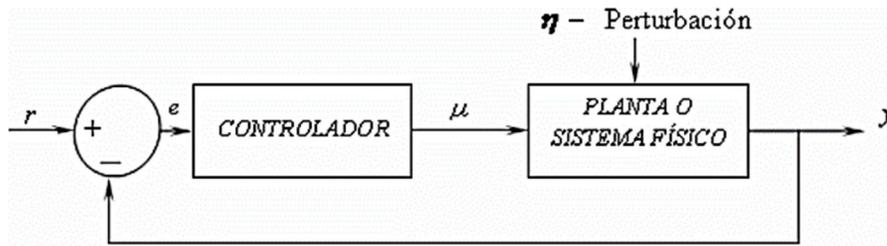
2.1.6.1 Control básico SISO

En un sistema de control SISO (Single Input, Single Output), para mantener la salida deseada del sistema deben hacerse los ajustes necesarios a la entrada y de esta forma se lograría el propósito de control, sin embargo, existe la necesidad de observar constantemente cómo es la respuesta ante las variaciones en la entrada, por ello surge

la necesidad de cerrar el lazo de control para monitorear de manera permanente la salida del sistema y compararla con una señal de referencia o salida deseada. La Figura 7 muestra un sistema de control SISO en lazo cerrado. (Giraldo B. D., 1997)

Figura 7

Sistema de Control SISO



Nota. Tomado de (Diego A. Mejía Giraldo, 2007)

En dónde:

- r es la referencia o set point del sistema
- μ es la señal del controlador
- $e = r - y$; es el error del sistema usado por el controlador
- y , es la salida del sistema

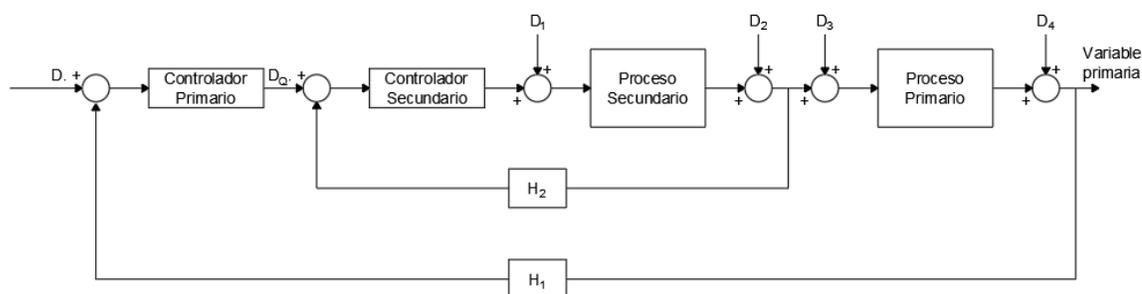
2.1.6.2 Control en cascada

Teniendo en cuenta a (Smith & Corripio, 1991) el control en cascada es una estrategia de control utilizada ampliamente en diversas aplicaciones en las industrias de proceso, desde que su esquema de control se introdujo por Franks y Worley en 1956 (Worley & Franks, 1996), se emplea principalmente debido a su eficacia para eliminar las perturbaciones existentes en un sistema de control. De acuerdo (Nalbantoğlu & Kaya, 2015) si se utiliza un sistema clásico de control de retroalimentación SISO hay que esperar para observar la acción correctiva hasta que la perturbación haga que la

variable controlada se aleje de la variable deseada. Por otro lado, un esquema de control en cascada tomará una acción correctiva inmediata utilizando un controlador de lazo interno como se puede identificar en el esquema de la Figura 8.

Figura 8

Estructura de un sistema de control en cascada



Nota. Tomado (Murrill, 2000)

Para lograr que el circuito de control sea más estable y de respuesta rápida se debe considerar en el diseño de la estrategia de control en cascada, que el controlador secundario o conocido también como circuito interno debe ser más rápido, que el primario o circuito externo. (Smith & Corripio, 1991)

2.1.7 Controlador PID

La capacidad de los controladores PID para compensar la mayoría de los procesos industriales ha logrado que alrededor del 95% de los lazos de control empleados en la industria sean PID, este controlador sigue representando una respuesta robusta, práctica y fiable en el control de procesos. La identificación de las siglas PID está relacionada con los modos de control que lo forman: Proporcional (P), Integral (I) y Derivativo (D) mismas que se refieren a las acciones de control sobre el error. (Sánchez, Vilanova, & Arrieta, 2014), (O'Dwyer, Handbook of PI and PID Controller Tuning Rules, 2006). El controlador PID ISA estándar es una estructura

clásica e ideal, también es conocido como controlador no interactuante, ya que el tiempo integral no influye en el tiempo derivativo o viceversa (O'Dwyer, PID Controller Structures, 2006). En la ecuación (7) se muestra la formula del controlador PID ISA estándar:

$$C(s) = K \left(1 + \frac{1}{s T_i} + s T_d \right) \quad (7)$$

2.1.7.1 Métodos de sintonía para procesos autorregulados.

Si se desea integrar un controlador PID para procesos autorregulados se aplicará los siguientes métodos de sintonía los cuales se presentan en la Tabla 2.

Tabla 2

Métodos de Sintonía para un Controlador PID para Procesos Autorregulados

Método	Kc	Ti	Td
Ziegler-Nichols	$1.2 \frac{T}{KL}$	2L	0.5 L
Cohen-Coon	$\frac{T}{KL} \left(\frac{4}{3} + \frac{L}{4T} \right)$	$\frac{L (32T + 6L)}{13T + 8 L}$	$\frac{4TL}{11T + 2L}$
Lambda	$\frac{1}{K} \left(\frac{\frac{L}{2} + T}{\frac{L}{2} + T_{cl}} \right)$ <i>Tcl = 3T lambda robusto</i> <i>Tcl = T lambda agresivo</i>	$T + \frac{L}{2}$	$\frac{TL}{L + 2T}$
Parr	$1.25 \frac{T}{KL}$	2.5 L	0.4 L

Nota. Tomado de (O'Dwyer, PID Controller Structures, 2006)

2.1.7.2 Métodos de sintonía para procesos Integrantes

Si se desea aplicar un controlador PID para procesos integrantes se aplicará los siguientes métodos de sintonía los cuales se presentan en la Tabla 3.

Tabla 3

Métodos de sintonía para un controlador PID para procesos integrantes.

Método	Kc	Ti	Td
Ford	$\frac{1.2}{KL}$	2L	0.37L
Astrom - Hagglund	$\frac{0.94}{KL}$	2L	0.5L
Hay	$\frac{0.4}{KL}$	3.2L	0.8L
Zou - Brigham ($\lambda = 3\tau_m$)	$\frac{2}{K(\lambda + 0.5L)}$	$2\lambda + L$	$\frac{\lambda + 0.25L}{2\lambda + L}L$

Nota. Tomado de (O'Dwyer, PID Controller Structures, 2006)

2.1.7.3 Método de sintonización del tanteo

El método de sintonización del tanteo requiere que el controlador y el proceso estén funcionando normalmente. El proceso general se basa en poner en marcha el proceso con bandas anchas en todas las acciones, y estrecharlas poco a poco individualmente, hasta obtener la estabilidad deseada. (Jhonathan Caicedo, 2018)

Para estabilizar un proceso por medio del método de tanteo se debe seguir los siguientes pasos:

- Se inicia con ganancia integral tendiendo al infinito y derivativa igual a cero.
- Se comienza por aumentar la ganancia proporcional Kc hasta alcanzar la inestabilidad. (Jhonathan Caicedo, 2018)

- Se aumenta el tiempo integral T_i mediante pequeños incrementos con pequeños desplazamientos del SP y hasta obtener un comportamiento cíclico. (Jhonathan Caicedo, 2018)
- Se reduce en este punto la ganancia K_c y se ajusta el tiempo derivativo T_d (estrechando la banda proporcional) hasta alcanzar una respuesta óptima (Jhonathan Caicedo, 2018)

2.1.8 Diseño de procesos industriales con herramientas de dibujo asistido por computador 3D (Blender)

Blender es un programa de modelado en 3D que provee herramientas que permiten un diseño con un alto nivel de detalle y realismo. Es importante destacar que se debe optimizar los modelos creados en este software, es decir, se requiere de la eliminación de vértices duplicados en las mallas, lo cual contribuye en gran medida a la fluidez de la aplicación (Pinzás, 2009). Una ventaja importante que brinda este software es la creación de archivos con formato *.fbx para que el objeto 3D pueda ser utilizado en Unity. En la Figura 9 se muestra una planta industrial modelada en Blender.

Figura 9

Planta industrial modelada en Blender



2.1.8.1 Malla

Es una estructura formada por vértices, aristas y caras. Constituyen el armazón del objeto modelado, sobre ella se aplican materiales, texturas y luces para simular una apariencia física más realista. Pueden ser abiertas o cerradas, tener todos los puntos unidos o no, pero son muy importantes para que la representación final del objeto 3D sea correcta. (Tecnawebs, 2020)

2.1.8.2 Ray-tracing

Se utiliza principalmente para producir superficies reflectantes y espejadas. También se utiliza para crear transparencia y refracción. Con el Ray-tracing todas las luces que se utilicen en Blender pueden proyectar sombras, puede tener un efecto sorprenderte, pero puede tener altos tiempos de renderizado. (Ray-Tracing, 2020)

2.1.8.3 Render

La etapa de render toma como entrada los elementos definidos en las fases anteriores (Modelado, Materiales y Texturas, Iluminación y Animación) y produce como salida una imagen bidimensional que representa a la escena. (Blender, 2020)

2.1.9 *Hardware-In-the-Loop*

La técnica Hardware-In-The-Loop (HIL), es una simulación en tiempo real de todos los subsistemas que conforman un sistema complejo de ingeniería, los cuales se modelan y acoplan a través de modelos numéricos para una representación completa. Dicha simulación puede llevarse a cabo en un sistema embebido o en una computadora. (Park, Park, Wang , & Kwak, 2008) (Muraspahic, Gu, Farji, & Iskandarani, 2012). Es así que, un algoritmo de control puede validarse de una manera realista utilizando los sensores y actuadores simulados para verificar la respuesta del sistema

una vez que el controlador actúa en base a ellos, esto a través de una interfaz de comunicación entre el sistema simulado y el controlador real implicados en la arquitectura HIL (Rad, Maties, Hancu , & Lapusan, 2012).

Hoy en día, la simulación HIL se utiliza ampliamente por su bajo costo, nivel de detalle, alta efectividad y seguridad que ofrece en varios campos, entre los cuales se puede mencionar la electrónica, robótica, industria automotriz, industria energética y recientemente en la enseñanza de control de procesos y automatismos (Rad, Maties, Hancu , & Lapusan, 2012) (Tejado, Serrano, Pérez, Torres, & Vinagre, 2016).

2.1.10 PLC (Controlador lógico Programable)

Un PLC o controlador lógico programable es una computadora de un solo procesador, que emula el comportamiento de un diagrama en escalera eléctrico capaz de controlar varios tipos de equipos industriales y sistemas completos automatizados. Los PLC son actualmente utilizados en muchas partes de un sistema automáticos en la industria. Estos equipos son muy eficientes y rentables en aplicaciones que involucran control secuencial y sincronización de proceso, dentro de la manufactura, industria química y procesos industriales (Swapnil Arun Nameka, 2020). Existen muchas marcas dentro del mercado como Siemens, Allen Bradley, Mitsubishi, etc.

Figura 10

PLC Siemens S7-1200



Nota. Tomado de (Siemens, 2021)

Un controlador Lógico programable está compuesto de tres módulos: Módulo CPU, módulo de fuente de alimentación y Módulo de entradas y salidas.

- Módulo CPU: Está compuesto de un procesador central y de una memoria. Este módulo realiza todos los cálculos y el procesamiento de datos que recibe de las entradas y para producir las salidas correspondientes.
- Módulo de alimentación: Es aquel que suministra energía a los circuitos del PLC los cuales funcionan con 5Vcc. Responsables de encender el sistema.
- Módulo de E/S: Es el módulo responsable de conectar los sensores y actuadores al sistema PLC. (Automation, Mobile, 2017)

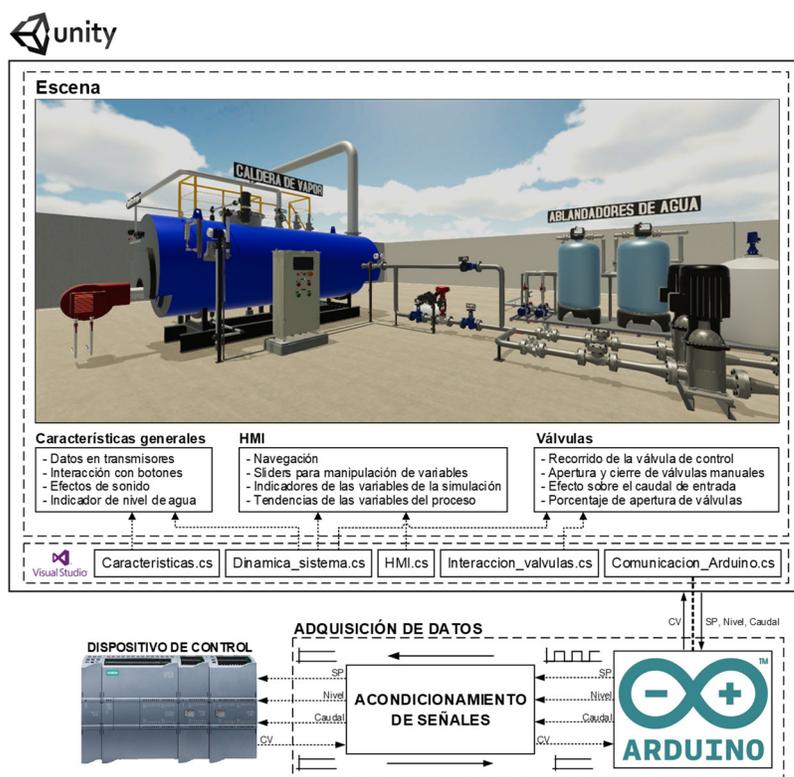
CAPÍTULO III

3. Metodología de la Investigación

En este capítulo, se detalla el desarrollo del sistema propuesto para la implementación del control en cascada en una planta industrial virtual a través de la técnica *Hardware-In-the-Loop (HIL)*. Esta técnica hace uso de una simulación en tiempo real del proceso en estudio y de un método que permita la interacción de las variables de dicha simulación con un dispositivo externo, mismo que contendrá el algoritmo de control que se requiere poner a prueba. La Figura 11 describe la arquitectura del sistema a través de un diagrama que permite comprender la relación entre cada una de las etapas que lo conforman.

Figura 11

Arquitectura del sistema virtual 3D de una planta de nivel mediante la técnica Hardware-In-the-Loop.



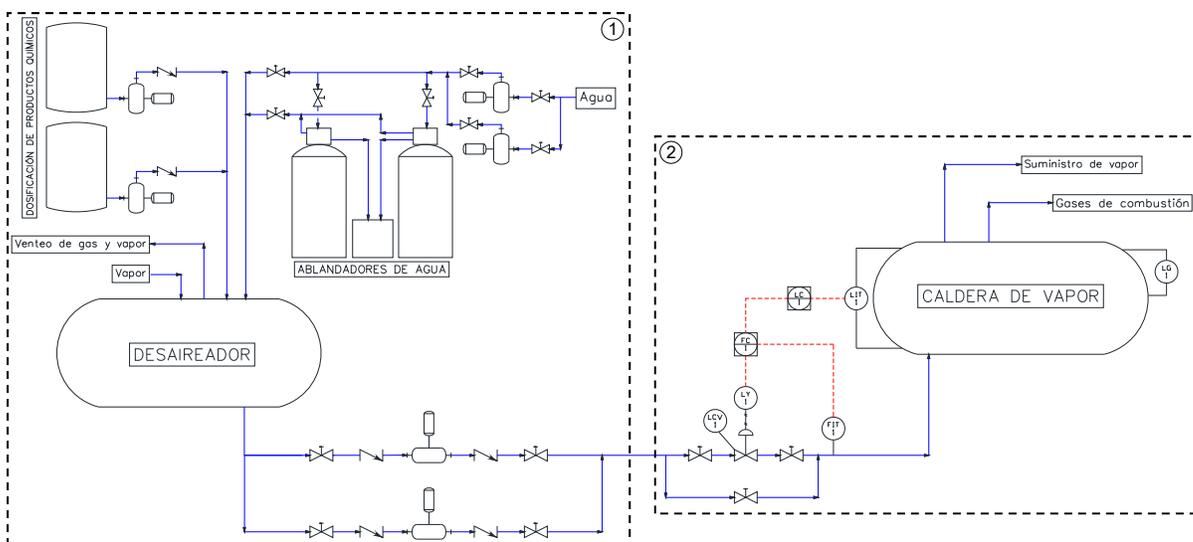
La implementación del proyecto inicia con la selección de un proceso industrial para su virtualización. En este caso, se eligió un sistema de control de nivel del agua de alimentación de una caldera de vapor. El proceso seleccionado ha sido digitalizado en Blender, un programa de diseño 3D gratuito y de código abierto y, posteriormente, caracterizado en Unity3D, una plataforma de desarrollo en tiempo real que permite la creación de experiencias interactivas. Luego, una vez que se han definido las características de la planta virtual, se implementa la programación que permitirá simular su comportamiento en tiempo real a través de modelos matemáticos y animaciones complementarias para obtener un proceso virtual realista. En este punto, se define el método de intercambio de datos entre la planta simulada y el dispositivo externo, que para este caso es un controlador lógico programable (PLC), que contendrá el algoritmo de control para ponerlo a prueba en el sistema desarrollado. Finalmente, a través del método de comunicación implementado anteriormente, se realizan las conexiones necesarias entre el controlador externo y la planta simulada para realizar los ensayos requeridos por el usuario.

3.1 Esquema de la Planta Industrial

El proceso seleccionado para su virtualización es un sistema de control de nivel del agua de alimentación de una caldera de vapor. Dicho sistema ha sido elegido por toda la instrumentación que requiere y por los métodos de control que pueden aplicarse para la regulación de sus variables, características importantes tomando en cuenta el objetivo del proyecto que se basa en la implementación y comprobación de controles avanzados de procesos y el entrenamiento de estudiantes y profesionales en esta área. En la Figura 12, se muestra el diagrama P&ID del sistema seleccionado.

Figura 12

Diagrama P&ID del sistema de generación de vapor



El sistema propuesto en la Figura 12 consta de 2 etapas: la primera es el tratamiento del agua de alimentación (1) y la segunda es la generación de vapor a través de una caldera (2). A continuación, se presenta una descripción y un análisis del sistema, imprescindibles para la implementación de este proyecto.

3.2 Tratamiento del Agua de Alimentación de la Caldera

El agua de alimentación de la caldera es tratada antes ser utilizada para la generación de vapor, de esta manera se evitan problemas causados por corrosión e incrustaciones (Smith, 2016). En esta etapa, el primer paso es el ablandamiento del agua que consiste en eliminar iones de calcio y magnesio, los cuales dan lugar a una *dureza* en el agua; los ablandadores de agua contienen una resina que secuestra los iones de calcio y magnesio y los intercambia por iones de sodio, generando sales que son solubles al incrementar la temperatura. Luego, el agua *suavizada* pasa a un proceso de desgasificación o desaieración que sirve para remover los gases disueltos no condensables, principalmente oxígeno y dióxido de carbono, a través de un

incremento en la temperatura; el desaireador eleva la temperatura del agua hasta su punto de ebullición y elimina la mayor parte de dichos gases a través de un venteo ubicado en la parte superior del dispositivo. En este punto, se efectúa un tratamiento químico para eliminar oxígeno residual del agua, para esto se utilizan productos que contienen secuestrantes de oxígeno que son suministrados en el tanque de almacenamiento de agua del desaireador. Estos procedimientos aseguran la calidad del agua de alimentación de la caldera previniendo su eventual deterioro (Gilman, 2010; Smith, 2016).

3.3 Generación de Vapor Mediante una Caldera

En esta etapa, el agua previamente tratada es llevada hacia una caldera pirotubular¹ por medio de bombas que funcionan en intervalos de tiempo (Smith, 2016). En esencia, el principio de funcionamiento de una caldera se basa en la exposición de agua a altas temperaturas. Esto se consigue por medio de un gas de combustión caliente que fluye a través de tubos internos y, a través de una transferencia de calor por convección, evapora el agua que se encuentra en el depósito de agua de la caldera. El vapor generado se destina a múltiples usos como la generación de energía, calefacción, esterilización de instrumentos, etc. (Chattopadhyay, 2000; Smith, 2016).

En una caldera de vapor, se requiere que el nivel de agua se mantenga dentro de un rango seguro especificado, de esta manera se obtiene un funcionamiento fiable y un vapor de alta calidad (Gilman, 2010; Xu, Shaoyuan, & Wenjian, 2004). El control del nivel de agua en una caldera presenta dificultades por varias razones, una de ellas es que la presión de vapor suele presentar variaciones significantes, mismas que pueden crear caídas de presión en la válvula de control utilizada para la regulación y que pueden, por consiguiente, provocar que el caudal de entrada se vea reducido y a su vez

¹ Caldera pirotubular: Caldera en la cual los gases calientes de combustión circulan a través de tubos internos. El agua rodea los tubos para absorber el calor y finalmente evaporarse.

el nivel del agua en el interior. También pueden darse cambios abruptos en el flujo de vapor debido a la demanda variable del mismo en una etapa posterior. Otro factor que puede influir en el sistema es la activación o desactivación de las bombas de alimentación de la caldera, esto depende de la demanda de vapor y puede dar lugar a cambios en el caudal de entrada y por ende en el nivel de agua (Gilman, 2010; Love, 2007; Xu et al., 2004).

Por las razones expuestas en el párrafo anterior, entre otras, es que una de las estrategias más utilizadas para regular el nivel del agua de una caldera es el control en cascada que, a través de un controlador secundario, permite compensar los efectos causados por las variaciones inherentes de este sistema al actuar sobre estas de una forma más rápida que el control clásico de lazo único y de una manera anticipativa para que la variable primaria, en este caso el nivel de agua, no se vea afectada (Love, 2007; Marlin, 1995; Xu et al., 2004). Además, el sistema presenta las condiciones necesarias para implementar la estrategia de control antes mencionada, entre ellas que la dinámica del caudal de entrada (proceso secundario) es más rápida que la de nivel (proceso primario) (Marlin, 1995). Finalmente, para la implementación de este método de control, se requiere añadir a la instrumentación del sistema una válvula de control neumática (elemento de control final) y un transmisor de caudal (FIT) en la línea de entrada de agua de la caldera y un transmisor de nivel por presión diferencial (LIT), todos estos requerimientos se encuentran considerados en el diagrama P&ID propuesto en la figura anterior que incluye también los lazos de control antes mencionados.

3.4 Respuesta Dinámica y Control del Sistema

Esta sección presenta la definición de los modelos que permitirán simular el comportamiento del sistema y el diseño del método de control que se utilizará como caso de estudio, incluyendo la programación en el controlador lógico programable (PLC) escogido.

En primer lugar, luego de haber analizado las características operativas del sistema, es necesario establecer las particularidades del proceso para su simulación y control. Para esto, se debe tomar en cuenta la estructura del control en cascada que se requiere implementar (Ver Figura 13). El esquema propuesto cumple con las condiciones de diseño mencionadas en la fundamentación teórica de este trabajo, dado que el caudal de entrada (variable secundaria) incide directamente sobre el nivel de agua de la caldera (variable primaria). Además, como se menciona en la sección anterior, la variable secundaria presenta una respuesta dinámica más rápida que la variable primaria y es susceptible de variaciones que pueden afectar al proceso, mismas que deberán ser compensadas por el controlador secundario. En el diagrama de bloques de la Figura 13, p_1 y p_2 representan perturbaciones en los procesos primario y secundario, respectivamente. Cabe resaltar que se utilizará una válvula de control neumática como elemento de control final para la regulación y transmisores de caudal y nivel para la medición de las variables.

Los rangos de operación de las variables del proceso han sido definidos de la siguiente manera:

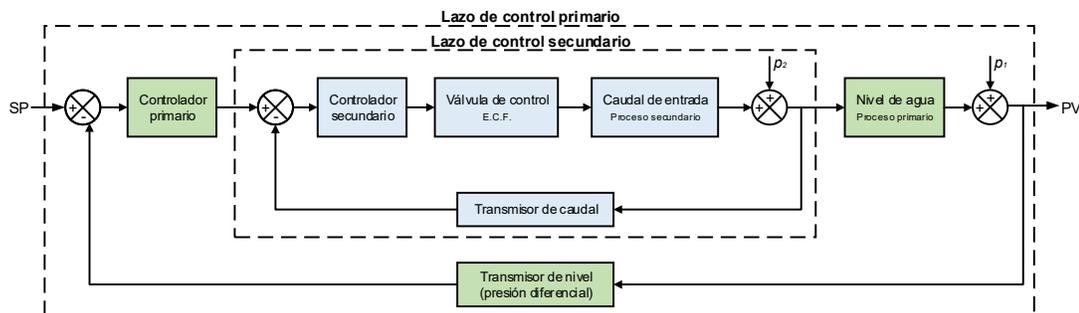
Tabla 4

Lista de instrumentos del proceso.

TAG	Descripción	Rango
LIT - 1	Transmisor indicador de nivel por presión diferencial, variable primaria del proceso	0 – 2 m
FIT-1	Transmisor indicador de caudal, variable secundaria del proceso	0 – 20 GPM

Figura 13

Estructura de control en cascada para la regulación del nivel de agua de la caldera



3.4.1 Características Dinámicas del Sistema

La caldera de vapor tiene como entrada el caudal de agua de alimentación que se evapora una vez que es llevada a su punto de ebullición. Entonces, la variación del volumen de agua dentro de la caldera puede definirse como la diferencia entre el caudal de alimentación (q_f) y el caudal del vapor generado (q_s) por la caldera, para lo cual se define la ecuación (8) (Halihal, 2019).

$$\frac{dV}{dt} = q_f - q_s \quad (8)$$

Dado que el volumen puede expresarse como el producto del área (A) de la caldera y la altura (h) de la misma, se plantea la ecuación 9.

$$A \frac{dh}{dt} = q_f - q_s \quad (9)$$

A partir de la ecuación (9), se generan los comportamientos requeridos para caracterizar la respuesta dinámica del caudal de entrada (proceso secundario) y del nivel de agua de la caldera (proceso primario). Cabe mencionar que a partir de dicha ecuación es posible simular una respuesta para el sistema de manera general, para lo cual no se han considerado los parámetros físico-químicos involucrados en el proceso de generación vapor seleccionado. Es decir, solamente se ha verificado que el

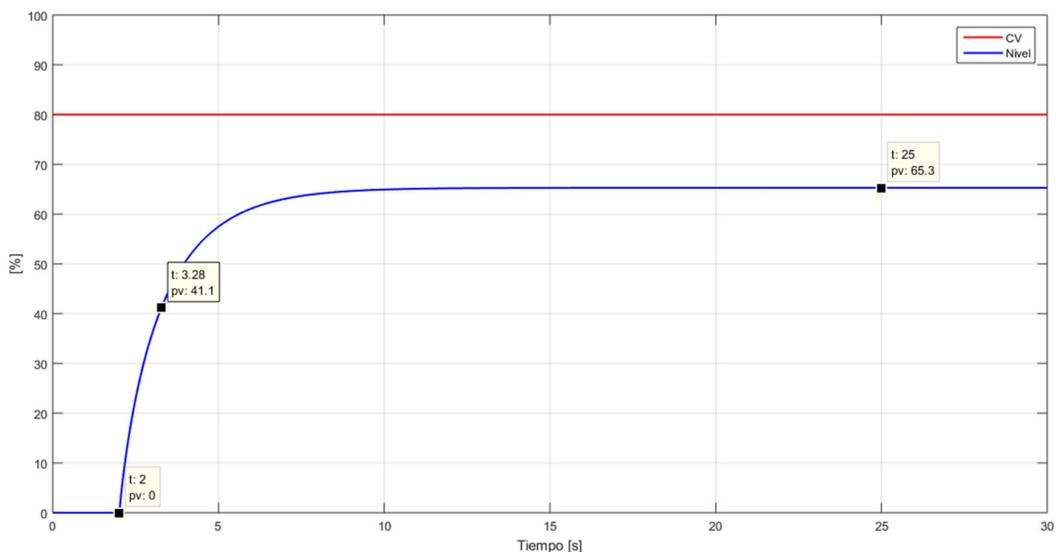
comportamiento simulado para cada una de las variables corresponda de manera coherente con su dinámica real.

Luego, se han definido dos modelos matemáticos (funciones de transferencia) para caracterizar la respuesta dinámica del caudal de entrada y del nivel de agua de la caldera.

El caudal se describe a través de un modelo FOPDT (First Order Plus Dead Time), mismo que puede representarse a partir del cálculo de sus parámetros en base a una gráfica generada utilizando los fundamentos descritos anteriormente, esto se muestra en la Figura 14. A través de estos datos de entrada (CV) y salida (Caudal), se establece una función de transferencia ecuación (10) reemplazando los valores correspondientes en la ecuación (1).

Figura 14

Comportamiento de un proceso autorregulado de primer orden más tiempo muerto



Los valores de los parámetros de la función de transferencia se calculan en base a las características de la curva de la Figura 14, donde:

$$K = 65.3/80 = 0.81625$$

$$T = 3.28 - 2 = 1.28$$

$$L = 2$$

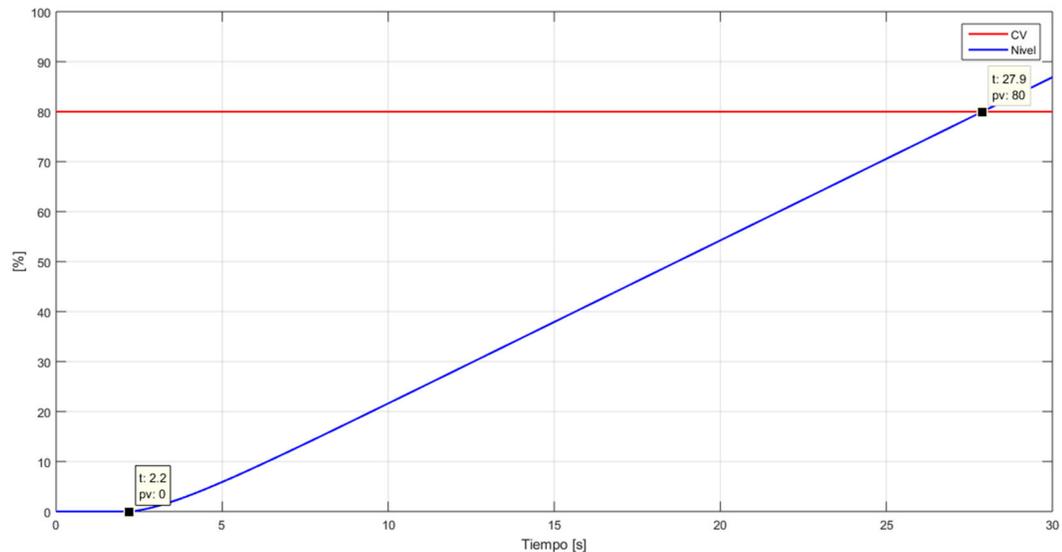
Reemplazando estos valores en la ecuación se obtiene:

$$G(s) = \frac{0.81625}{1 + 1.28s} e^{-2s} \quad (10)$$

El nivel de agua se describe a través de un modelo IPDT (Integral Plus Dead Time). Al igual que en el caso anterior, se define el modelo requerido a través de un conjunto de datos generados previamente (Ver Figura 15) y reemplazando los datos necesarios en la ecuación (3). La función de transferencia para este proceso se muestra en la ecuación (11).

Figura 15

Comportamiento de un proceso integrante más tiempo muerto



Los valores de los parámetros de la función de transferencia se calculan en base a las características de la curva de la Figura 16, donde:

$$m = \frac{80}{27.9 - 2.2} = 3.113$$

$$K = \frac{3.113}{80} = 0.0389$$

$$L = 2.2$$

Reemplazando estos valores en la ecuación se obtiene:

$$G(s) = \frac{0.0389}{s} e^{-2.2s} \quad (11)$$

Estas funciones de transferencia serán utilizadas para implementar la simulación del comportamiento de la planta en el entorno virtual y permitirán diseñar la estrategia del control en cascada.

3.5 Control en Cascada del Nivel de Agua de la Caldera

3.5.1 Diseño de los Controladores Primario y Secundario.

La estrategia de control en cascada requiere dos controladores, uno para cada variable involucrada en su diseño, tal como se indica en la Figura 13. Entonces, el primer paso es diseñar el controlador secundario que, en este caso, utilizará una estructura PID. Para esto, se hace uso de los parámetros de la función de transferencia de caudal para calcular las constantes de sintonía del controlador. Se han considerado 4 métodos de sintonía para controladores PID y, a partir del rendimiento que ofrezca cada uno para este proceso, se realizará un análisis comparativo para escoger el control más óptimo (sin sobreimpulso). En la Tabla 5, se muestran los valores para las constantes de sintonía del controlador según los métodos escogidos (las ecuaciones utilizadas para encontrar los valores de las constantes se encuentran en la sección "Métodos de sintonía de procesos autorregulados").

Tabla 5

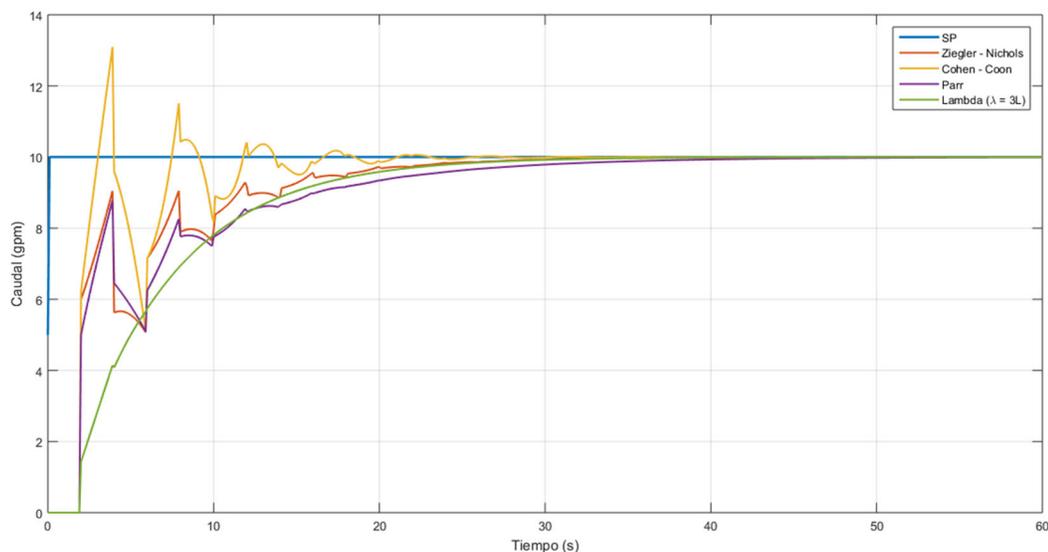
Constantes de sintonía para el controlador secundario

Método/Constantes	Kp	Ti [s]	Td [s]
Ziegler - Nichols	0.9409	4	1
Cohen - Coon	1.3648	3.296	0.7139
Parr	0.9801	5	0.8
Lambda ($\lambda=3L$)	0.399	2.28	0.5614

En la Figura 16, se presenta una simulación del controlador aplicado a este proceso, misma que permite comparar las características que ofrece cada método de sintonía considerado para este caso (las ecuaciones utilizadas para encontrar los valores de las constantes se encuentran en la sección “Métodos de sintonía de procesos autorregulados”).

Figura 16

Simulación del Controlador PID Secundario (caudal) para los 4 métodos de sintonía considerados



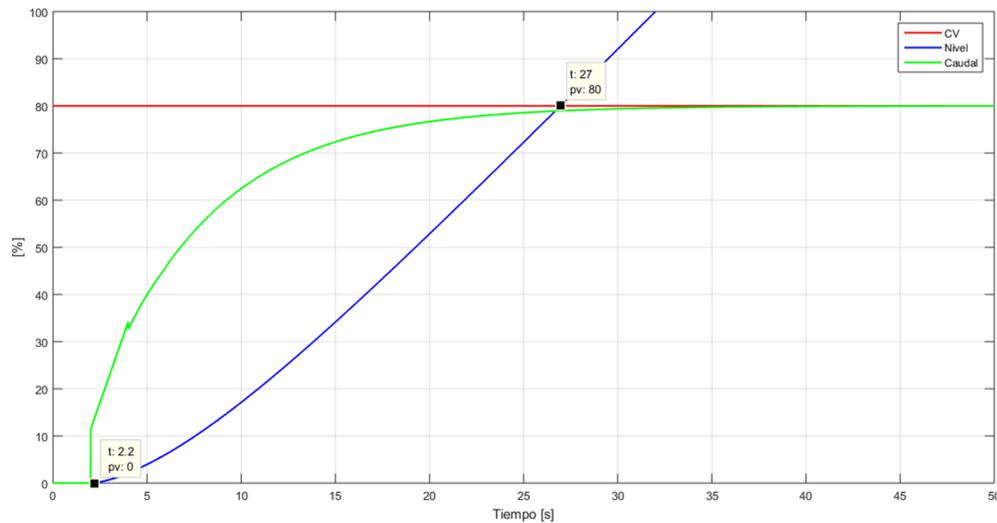
Se observa que el tiempo de estabilización es similar para los 4 casos, pero existe un mejor desarrollo de la variable en la curva resultante del método Lambda ya que no presenta cambios abruptos en su etapa transitoria. Tomando en cuenta este factor, se eligen las constantes calculadas a partir del método Lambda (con $\lambda=3L$) para ser utilizadas en el controlador secundario.

Ahora, se requiere hallar una nueva función de transferencia para el diseño del controlador primario, tomando en cuenta que el lazo de control interno cuenta con un controlador que ahora hace parte de la dinámica del proceso primario. Para esto, los datos obtenidos en la simulación anterior, con el método de sintonía elegido, se incluyen

como parte de una nueva simulación para obtener los datos de salida que permitirán calcular los parámetros del nuevo modelo requerido. La Figura 17 muestra el comportamiento del proceso primario tomando en cuenta las consideraciones de diseño mencionadas anteriormente y en la ecuación (12) se muestra la función de transferencia que define la dinámica de este proceso.

Figura 17

Respuesta Dinámica del Proceso Primario (nivel) incluido el Controlador Secundario (Caudal)



Los valores de los parámetros de la función de transferencia se calculan en base a las características de la curva de la Figura 15, donde:

$$m = \frac{80}{27 - 2.2} = 3.226$$

$$K = \frac{3.226}{80} = 0.0404$$

$$L = 2.2$$

Reemplazando estos valores en la ecuación se obtiene:

$$G(s) = \frac{0.0404}{s} e^{-2.2s} \quad (12)$$

Utilizando los parámetros de la ecuación (12), se calculan las constantes de sintonía para el controlador primario que también hará uso de una estructura PID. Se han considerado 4 métodos de sintonía para realizar un análisis comparativo posterior según el rendimiento obtenido con cada uno de estos. En la Figura 18 se muestran las curvas resultantes para cada método de sintonía escogido y en la Tabla 6 se muestran las constantes calculadas (las ecuaciones utilizadas para encontrar los valores de las constantes se encuentran en la sección “Métodos de sintonía de procesos integrantes”).

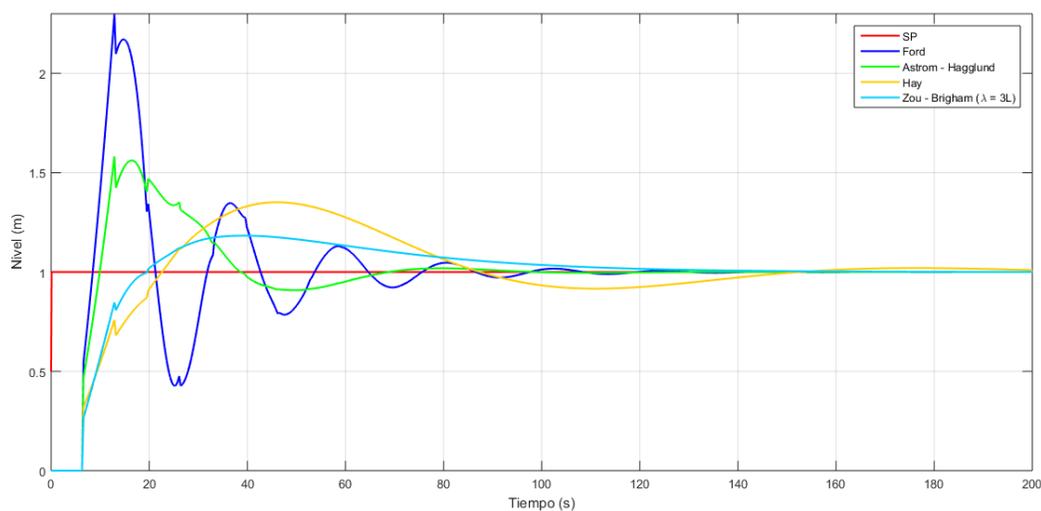
Tabla 6

Constantes de sintonía para el controlador primario

Método/Constantes	Kp	Ti [s]	Td [s]
Ford	16.6517	4.4	0.814
Astrom - Hagglund	10.5761	4.4	1.1
Hay	4.5005	7.04	1.76
Zou - Brigham ($\lambda=3L$)	6.4292	15.4	1.0214

Figura 18

Simulación del Controlador PID Primario (nivel) para los 4 Métodos de Sintonía considerados.



Es importante tomar en cuenta que, si las constantes de sintonía del controlador secundario cambian, también se modificarán las características de la variable caudal y en consecuencia la dinámica del proceso primario. Por lo tanto, si se decide cambiar algún parámetro dentro del lazo de control interno, será necesario repetir todo el procedimiento de diseño del controlador primario. Otra consideración importante es la ecuación del controlador PID que se utiliza en las simulaciones, la cual debe ser similar con la utilizada por el controlador real (PLC), en este caso se utiliza la ecuación PID ISA (ideal), de esta manera los datos simulados constituyen una buena herramienta de diseño que permite comparar y elegir entre distintas técnicas de una manera confiable.

3.6 Diseño del entorno virtual 3D

3.6.1 Modelado del Entorno Virtual

El entorno virtual ha sido diseñado en Blender tomando como base el diagrama P&ID del sistema e imágenes referenciales de los instrumentos y equipos requeridos. Es así que, partiendo de modelos básicos (primitivas), se modelan objetos tridimensionales complejos con un gran nivel de detalle.

De manera general, el espacio de diseño de Blender se denomina “Viewport” y cuenta con varios modos de diseño, mismos que se utilizan según se necesite. Los principales modos de diseños utilizados son: “Object Mode” y “Edit Mode”. Las modificaciones de diseño, principalmente, se realizan en Edit Mode ya que permite la edición de la malla² de los modelos y ofrece varias herramientas para un diseño de precisión. Además, se puede hacer uso de distintos modificadores³ que permiten añadir características muy variadas a los modelos. En la Figura 19, se muestra el modelo de una caldera de vapor diseñada en Blender, misma que se utilizará para detallar los

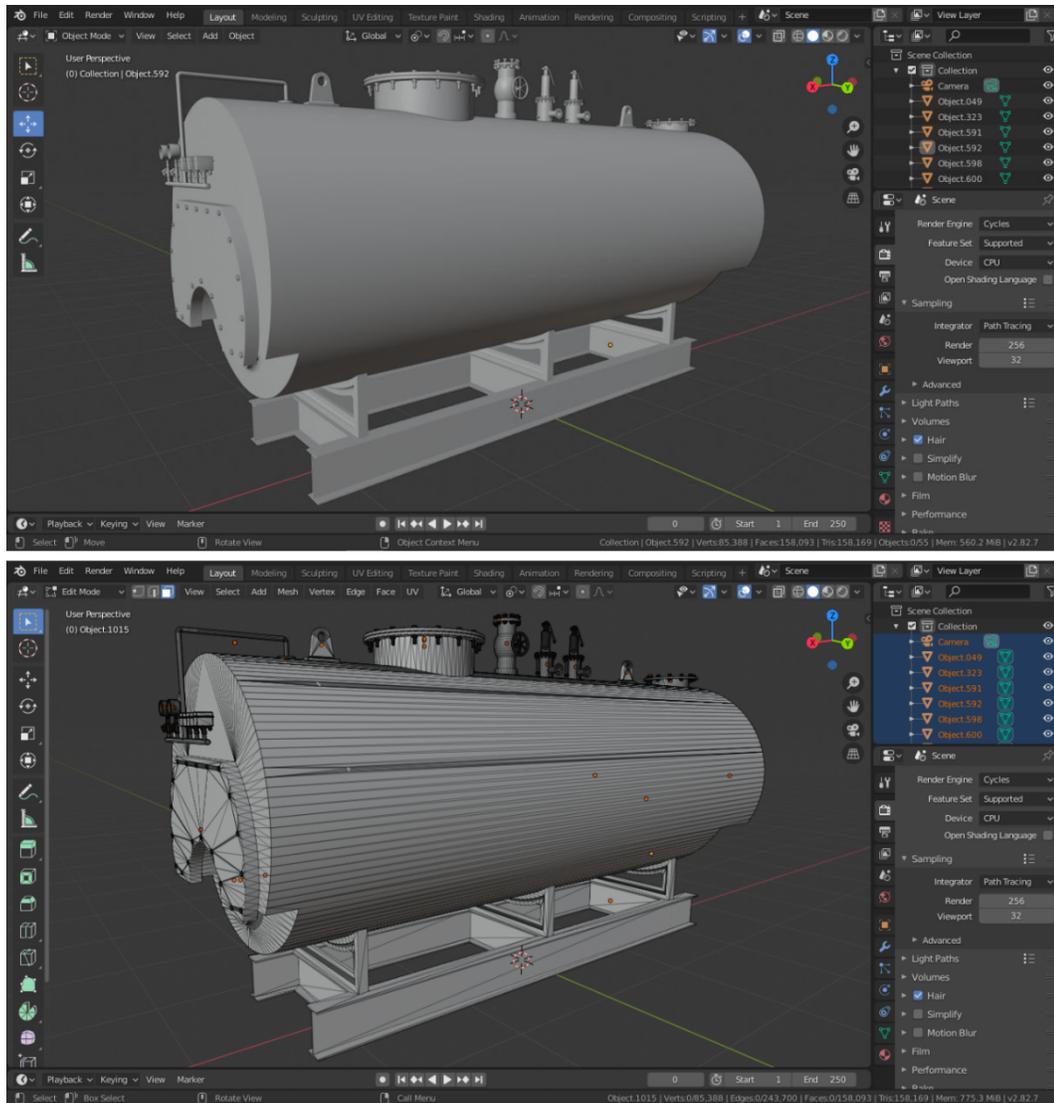
² Malla: Es una colección de vértices, aristas y caras que dan lugar a un objeto tridimensional.

³ Modificadores: En programas de diseño 3D, son operaciones automáticas que modifican la geometría de un objeto.

principales procedimientos y consideraciones necesarios para que un modelo 3D sea utilizable dentro de un motor gráfico, en este caso Unity3D.

Figura 19

Modelo Tridimensional de una Caldera de Vapor diseñada en Blender (Object Mode y Edit Mode)

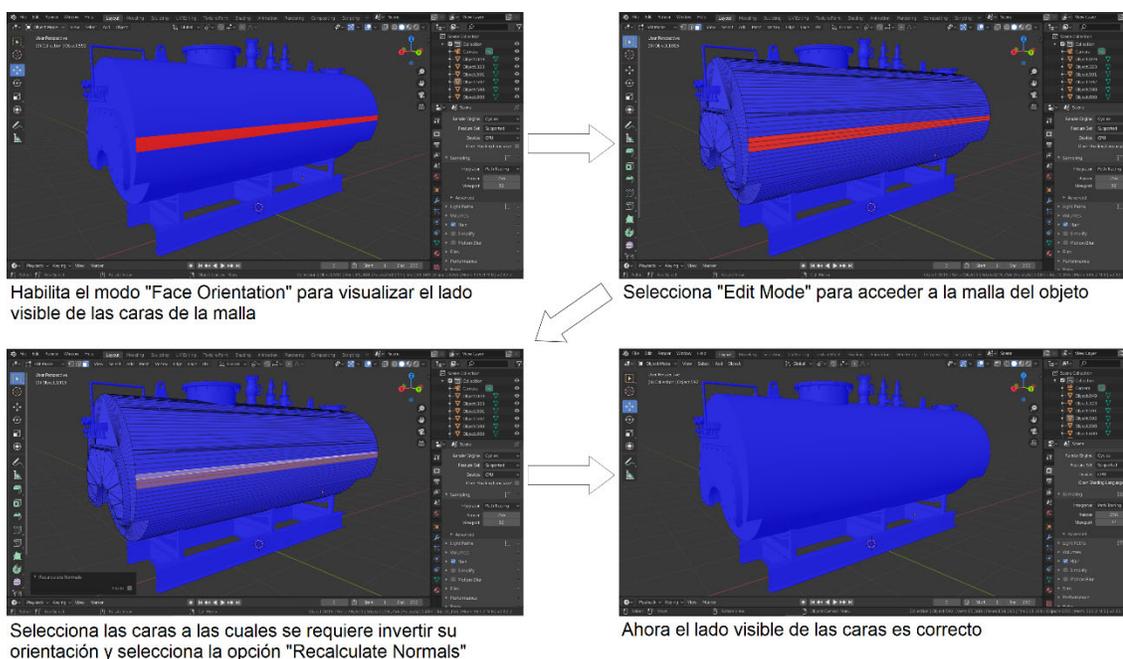


Dado que los modelos 3D serán utilizados para crear una aplicación en Unity3D, uno de los puntos esenciales a tomar en cuenta es que las mallas de los objetos estén configuradas correctamente. Para esto, se requiere verificar la orientación de las caras

de la malla de modo que esta sea visible desde el punto de vista del usuario. Esto no se puede observar a simple vista en el Viewport, para esto se utiliza una opción que marca en azul el lado visible de las caras y en rojo el lado opuesto que, en Unity3D, no se visualizará. Esto se puede corregir en Edit Mode seleccionando las caras con error e invirtiendo su orientación de acuerdo a lo indicado en la Figura 20.

Figura 20

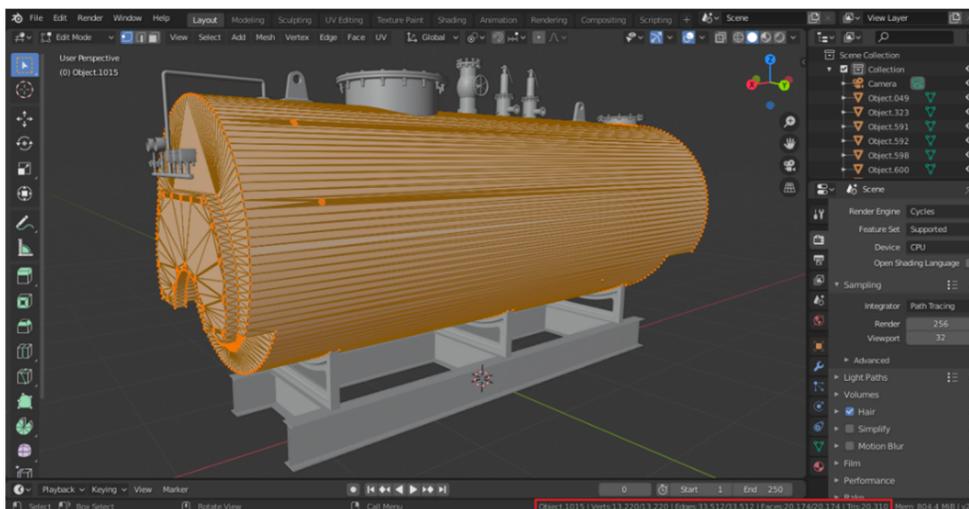
Procedimiento de corrección en la orientación de las caras de la malla de un objeto



Otra consideración importante es la optimización de los modelos a través de la eliminación de vértices duplicados o innecesarios en la malla, mismos que no aportan al detalle del objeto 3D. Este procedimiento es imprescindible ya que, si un objeto está siendo visualizado constantemente en Unity3D y la cantidad de vértices que lo conforman es demasiado alta, puede darse un bajo rendimiento o una pérdida de fluidez en la aplicación por los recursos computacionales que se necesitan para realizar esta tarea. La cantidad de vértices de un objeto puede visualizarse en la parte inferior del Viewport, como se muestra en la Figura 21.

Figura 21

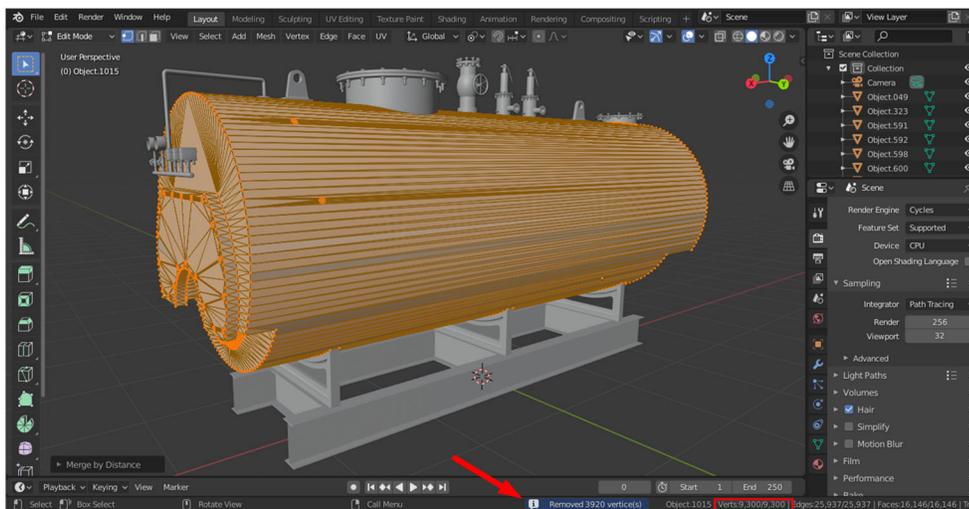
Número de vértices de un objeto



Para eliminar los vértices innecesarios del objeto, Blender ofrece una opción que calcula la distancia entre éstos y, si esta es menor o igual a un valor predefinido, elimina uno de ellos (Merge by Distance). En la Figura 22 se muestra el objeto optimizado.

Figura 22

Vértices duplicados removidos de la malla del objeto

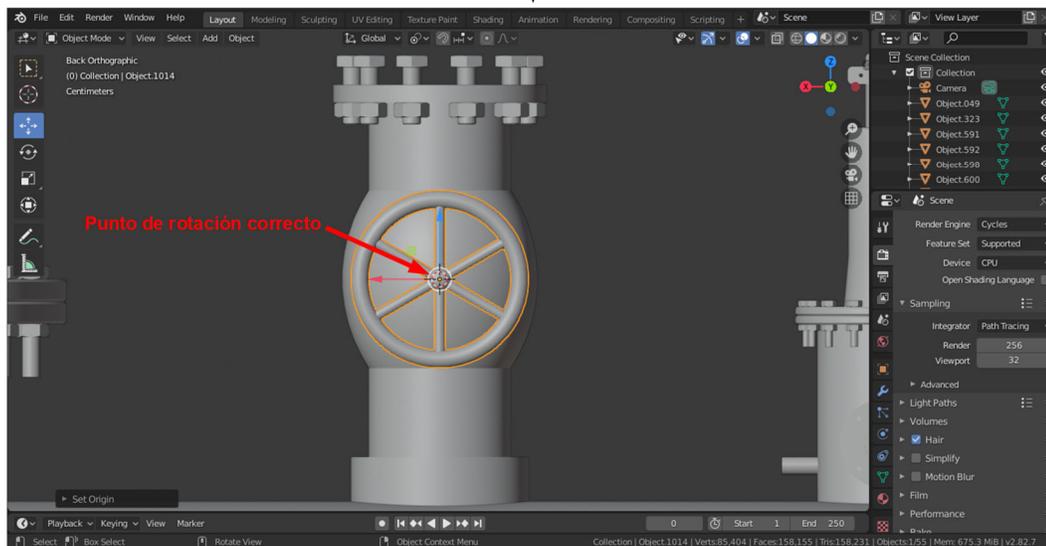
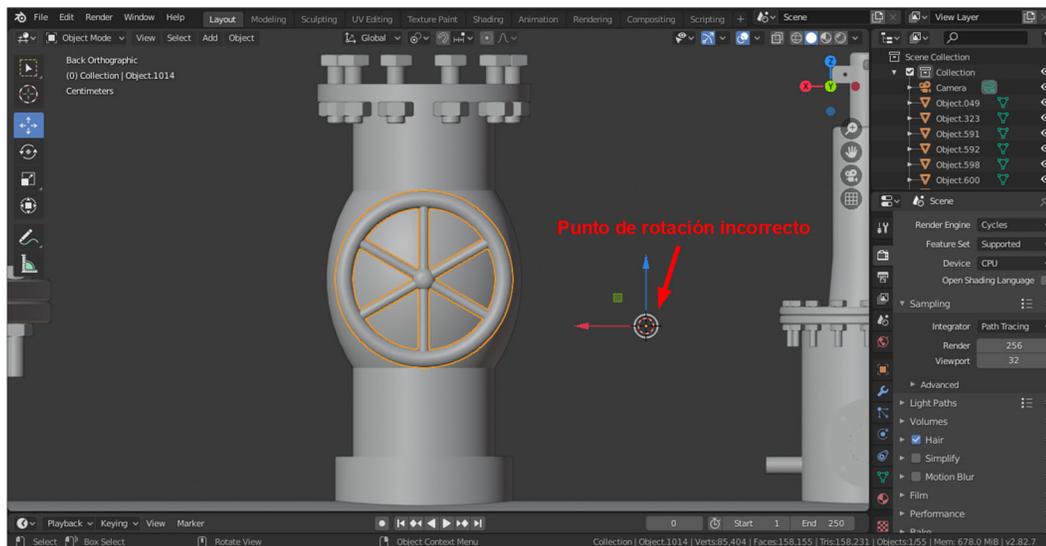


Cada objeto tiene un punto de rotación o pivote en torno al cual gira. Dicho punto de rotación se puede configurar según los requerimientos de la aplicación. En la Figura

23, se muestra la configuración de un punto de rotación para el actuador de una válvula, lo cual resulta beneficioso para su posterior animación en el motor gráfico.

Figura 23

Configuración del punto de rotación de un objeto



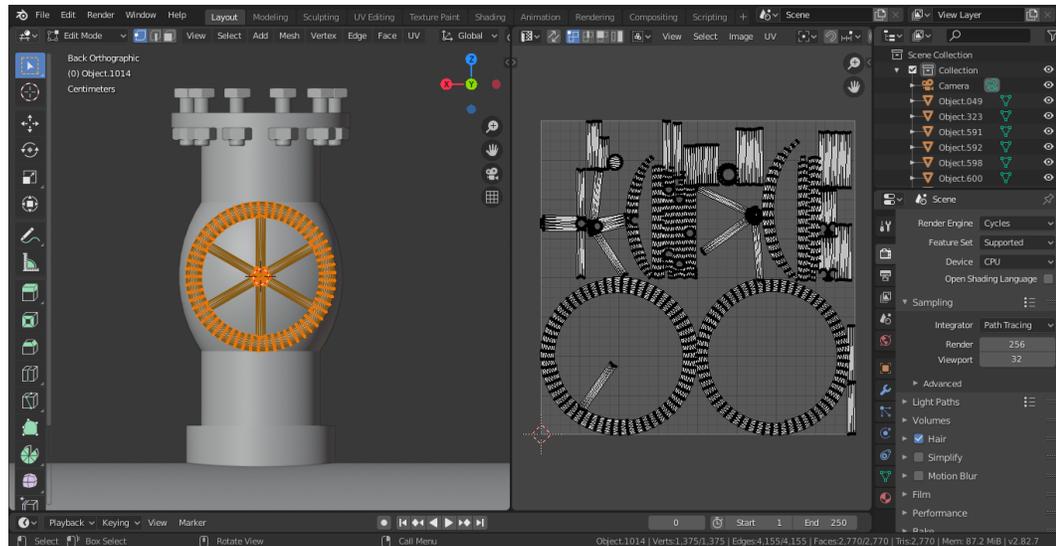
Una característica importante dentro del proceso de diseño es la creación de mapas UV⁴. Esto es necesario si se requiere utilizar texturas sobre un modelo y para la

⁴ Mapa UV: Representación en dos dimensiones de un objeto 3D para su texturización.

creación de mapas de iluminación (Lightmaps) en Unity3D. Blender proporciona herramientas para realizar este proceso de manera manual o automática. En la Figura 24 se muestra el mapa UV del modelo del actuador de una válvula manual.

Figura 24

Creación del mapa UV de un objeto



Las configuraciones detalladas anteriormente, entre otras, son esenciales para un modelo que se exportará a Unity3D ya que aseguran una buena calidad gráfica y permiten que su posterior animación y caracterización sea rápida e intuitiva.

Blender cuenta con un motor de renderizado llamado “Cycles” que utiliza ray-tracing⁵ para obtener renders⁶ realistas. La Figura 25 presenta un render del diseño de la planta virtual completa, para lo cual se han asignado materiales o “shaders” a cada uno de los modelos y se han añadido fuentes de iluminación a la escena. Cabe recalcar que dichos materiales deberán ser creados y añadidos nuevamente en Unity3D ya que no son compatibles con otros programas.

⁵ Ray-tracing: Tecnología que permite recrear en forma realista la incidencia de la luz sobre los objetos.

⁶ Render: Imagen digital que se crea a partir de un modelo diseñado con herramientas informáticas.

Figura 25

Diseño de la Planta Virtual para la Generación de Vapor a través de una Caldera



En la Figura 26 y en la Figura 27, se muestran renders de los transmisores de nivel (por presión diferencial) y caudal utilizados dentro del proceso.

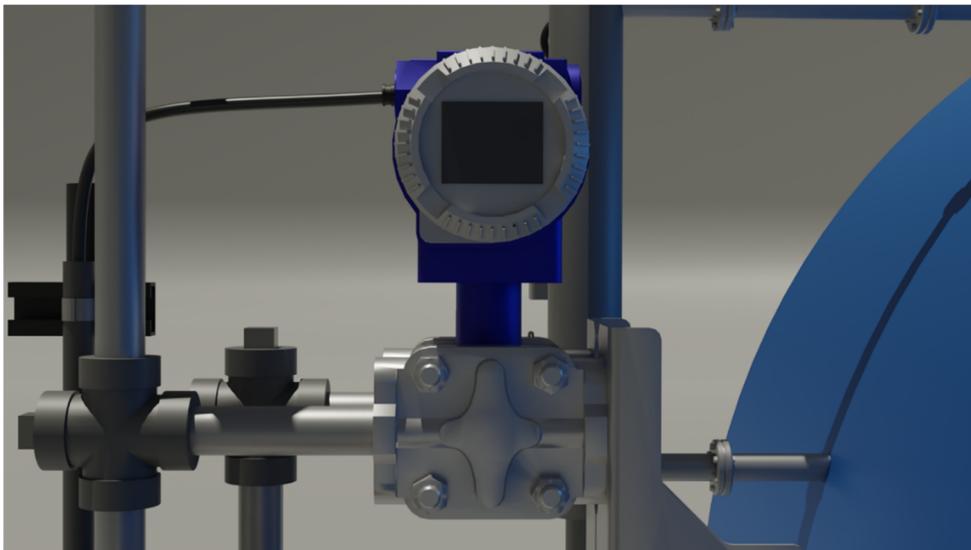
Figura 26

Render del Transmisor de Caudal



Figura 27

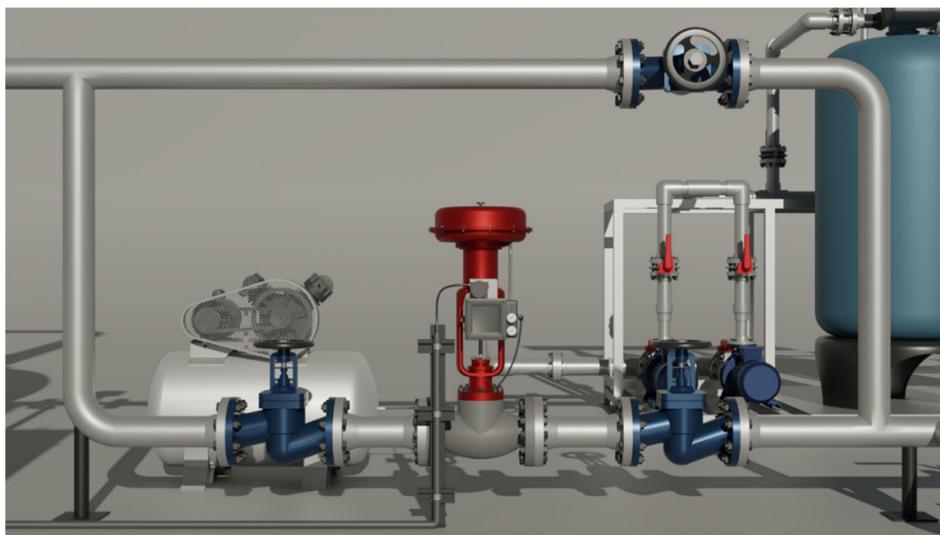
Render del Transmisor de Nivel



En la Figura 28, se muestra un render de la válvula de control neumática con un arreglo de válvulas manuales que se utilizan para bypass.

Figura 28

Render de la válvula de control neumática y las válvulas manuales de bypass



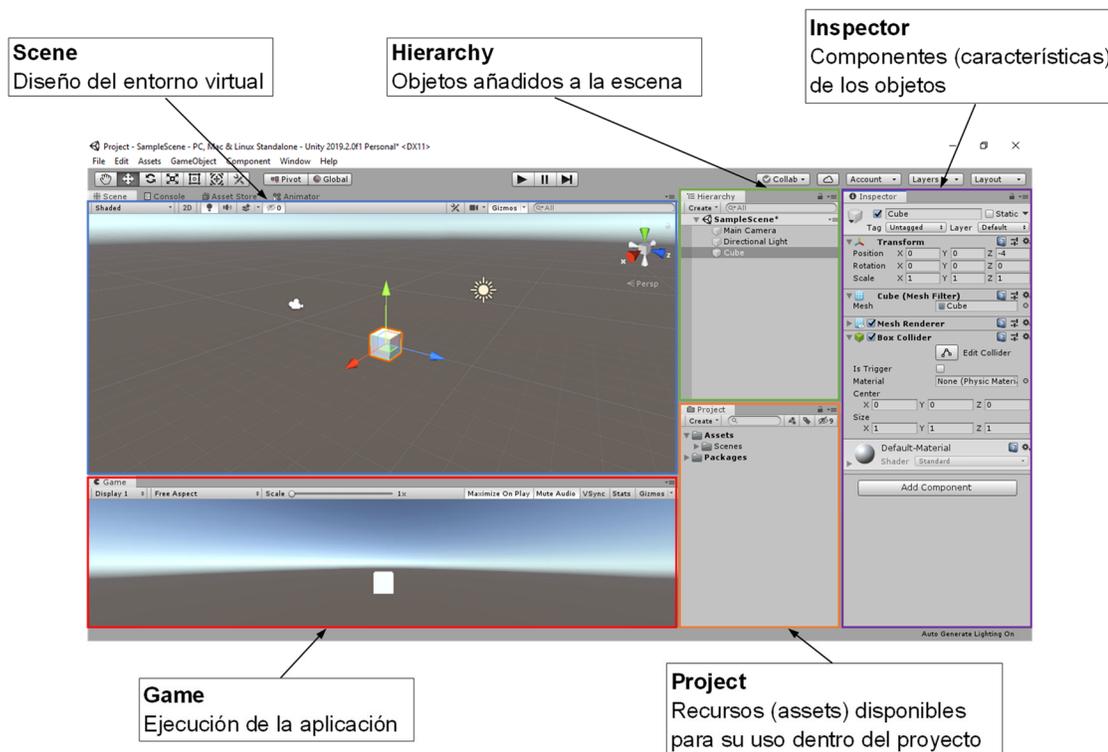
3.6.2 Desarrollo de la aplicación en Unity3D

3.6.2.1 Diseño del entorno virtual.

La aplicación que simulará el comportamiento de la planta virtual ha sido implementada en Unity3D. De manera general, en Unity3D se utiliza un layout⁷ básico compuesto por las ventanas “Scene”, “Game”, “Hierarchy”, “Project”, “Inspector”. En la Figura 29, se muestra la interfaz de Unity3D con el layout mencionado anteriormente, mismo que puede personalizarse para la comodidad del usuario.

Figura 29

Interfaz de Unity3D

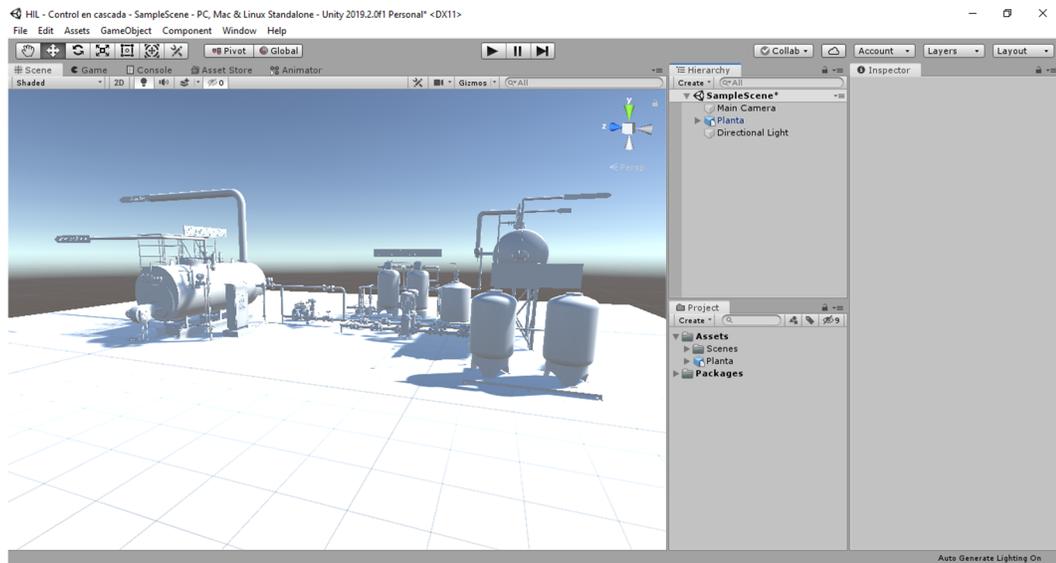


El modelo previamente diseñado en Blender se exporta en formato FBX, para luego ser importado en Unity3D, esto se realiza desde la ventana “Project”. En la Figura 30 se observa el modelo importado y añadido a la escena.

⁷ Layout: Forma en la que están distribuidos los elementos.

Figura 30

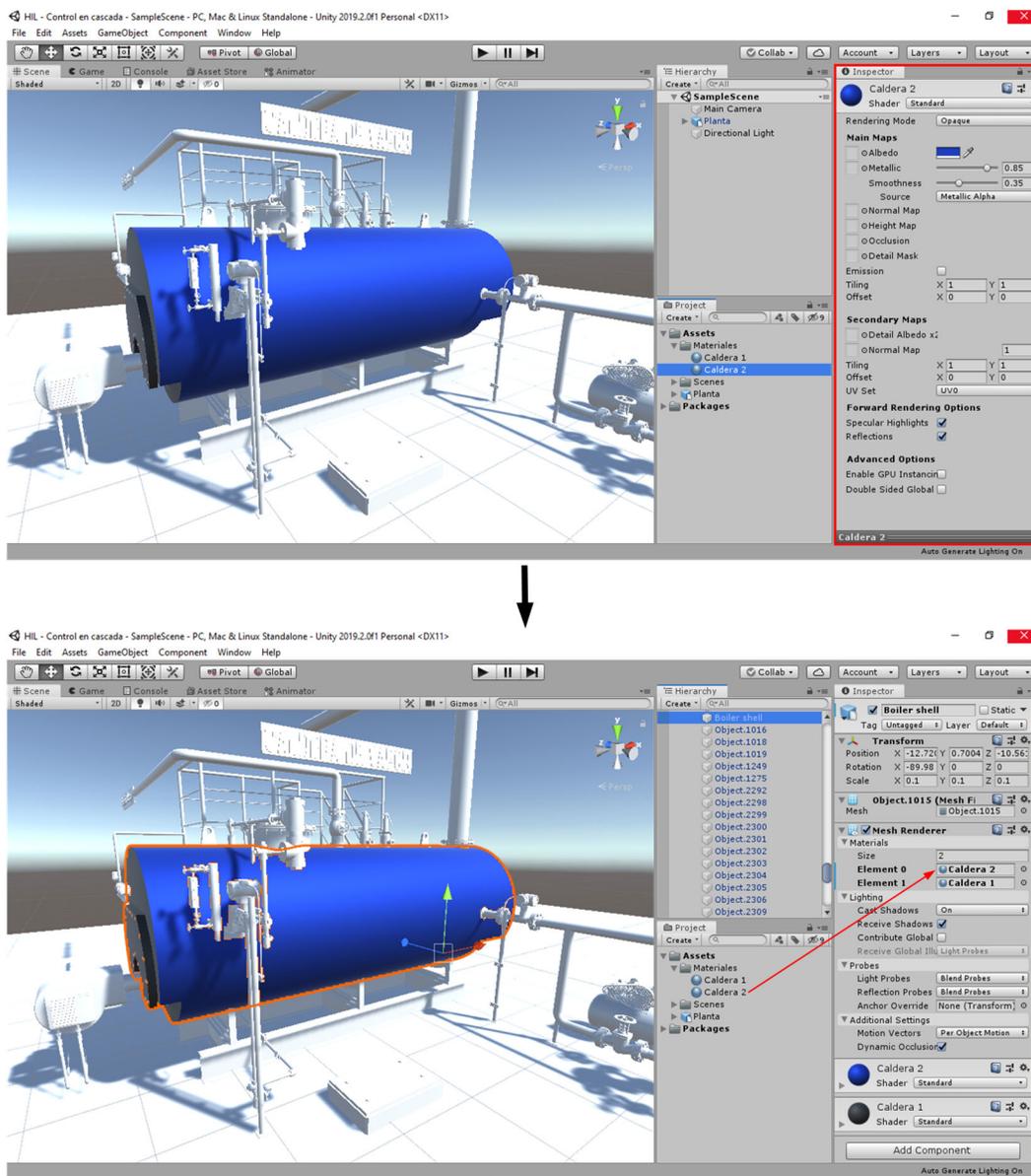
Modelo importado añadido a la escena en Unity3D



Como puede observarse en la Figura 30, los objetos carecen de materiales ya que, como se mencionó anteriormente, la manera de gestionar los materiales es distinta en cada programa. Por lo tanto, el primer paso es la creación y asignación de materiales para cada uno de los modelos. Un material puede tener varios atributos según los requerimientos del diseñador y, para una correcta organización de los recursos, es recomendable crear una carpeta que los contenga. En la Figura 31, se muestra la ventana para la configuración de un material y su posterior asignación a un objeto.

Figura 31

Creación y asignación de materiales en Unity3D



En la Figura 32 se muestra el entorno virtual una vez que se han asignado los materiales necesarios a cada uno de sus componentes.

Figura 32

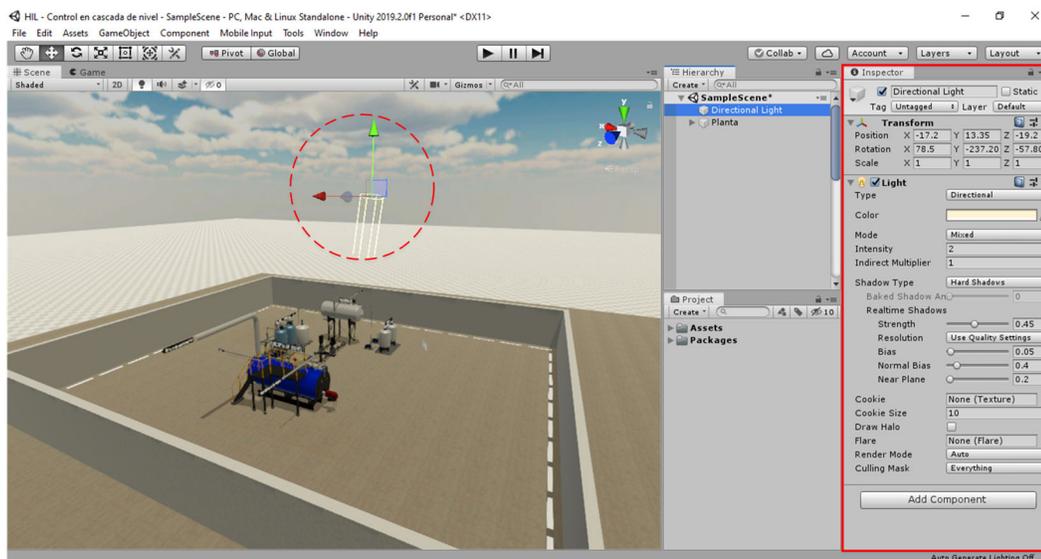
Materiales Asignados a cada Componente del Entorno Virtual



La iluminación del entorno es un aspecto esencial para obtener un buen rendimiento y una buena calidad gráfica en la aplicación. En Unity3D, se manejan mapas de iluminación que pueden crearse a partir de fuentes de iluminación configuradas previamente. En este caso, se utiliza una “Directional Light” y un “Skybox” para lograr una iluminación global sobre todo el entorno, misma que se añade desde el menú principal, en la opción `GameObject > Light > Directional Light`. En la Figura 33, se muestra el skybox y la fuente de iluminación añadida a la escena y sus configuraciones, disponibles en la ventana Inspector.

Figura 33

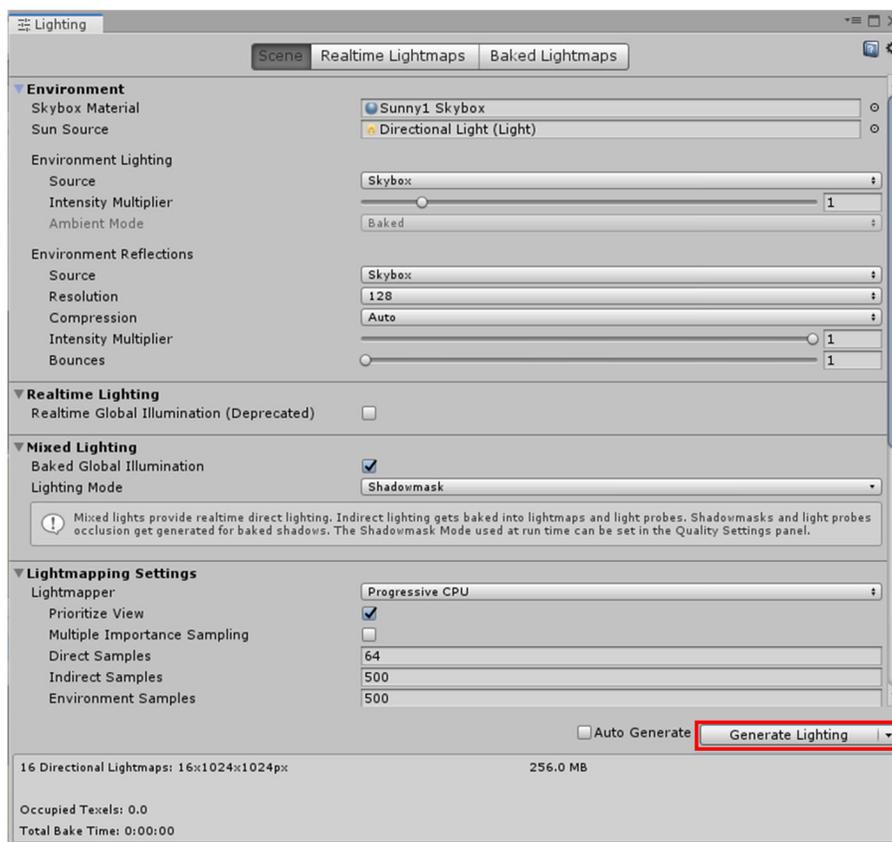
Directional Light y Skybox añadidos a la escena



El mapa de iluminación se crea a partir de los objetos marcados como “Static”, es decir, los objetos que no tienen movimiento en la escena (disponible en la ventana Inspector). Además, estos objetos deben contener sus respectivos mapas UV para poder aplicar el mapa de iluminación sobre estos, procedimiento que se realizó anteriormente en Blender. El mapa de iluminación se configura y genera a través de un asistente al cual se accede mediante el menú Windows > Rendering > Lighting Settings, mismo que se muestra en la Figura 34. Es importante tomar en cuenta que los mapas de iluminación deben crearse una única vez con el objetivo de mejorar el rendimiento, ya que también existe la opción de configurar una iluminación en tiempo real que, en la mayoría de los casos, podría ralentizar la aplicación. Una vez elegidas las opciones requeridas, se debe seleccionar la opción “Generate Lighting” y desmarcar “Auto Generate” para iniciar el proceso.

Figura 34

Configuraciones para la generación de mapas de iluminación



En las Figuras 35, 36 y 37, se muestra el entorno virtual una vez implementadas las principales consideraciones de diseño, es decir, aquellos aspectos visuales que ofrecen realismo y una buena experiencia de usuario.

Figura 35

Entorno virtual en Unity3D – Caldera de vapor

**Figura 36**

Entorno virtual en Unity3D – Válvula de control y válvulas manuales de bypass

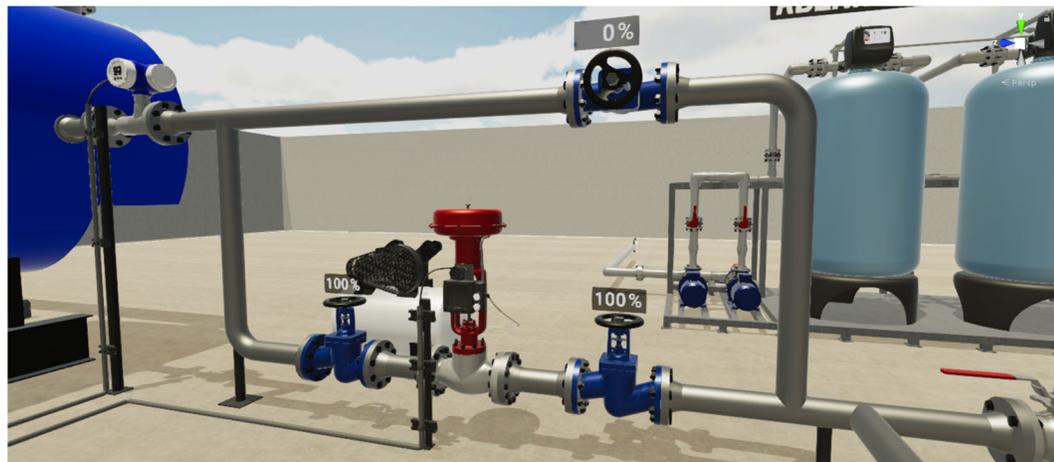


Figura 37

Entorno virtual en Unity3D – Etapa de tratamiento de agua



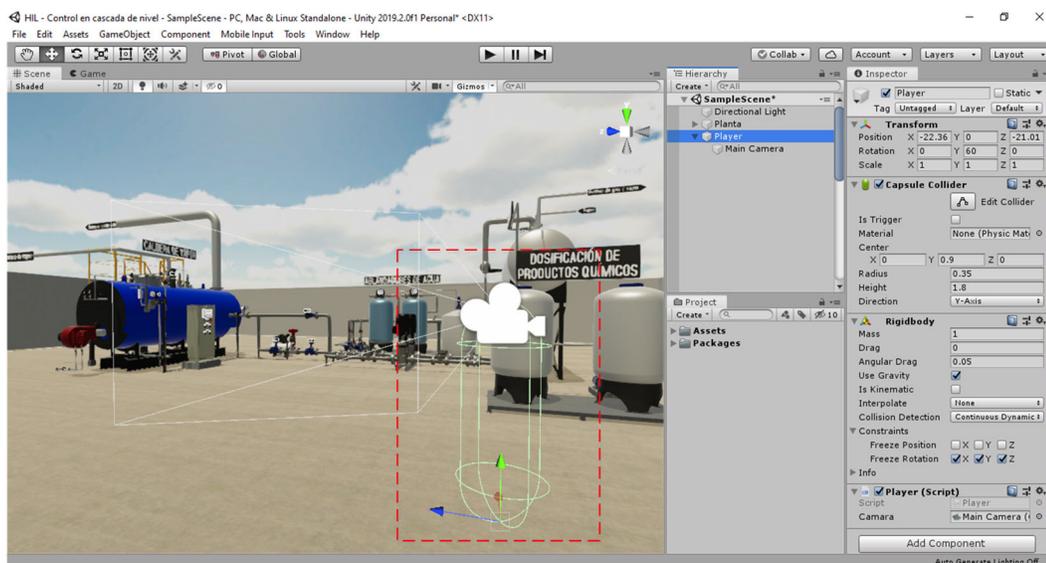
3.6.2.2 Programación de las características de la aplicación.

En primer lugar, se define la forma en la cual el usuario interactúa y se desplaza dentro del entorno virtual. En este caso, el movimiento se realiza a través del teclado y la rotación según el movimiento del mouse. Es así que, a través de un script⁸ desarrollado en C#, se implementa este comportamiento sobre un objeto (GameObject) que contiene una cámara que permite renderizar el entorno en tiempo real, con lo cual se obtiene una vista en primera persona para el usuario (ver Figura 38).

⁸ Script: Código de programación.

Figura 38

GameObject "Player" implementado para el movimiento del usuario en la escena

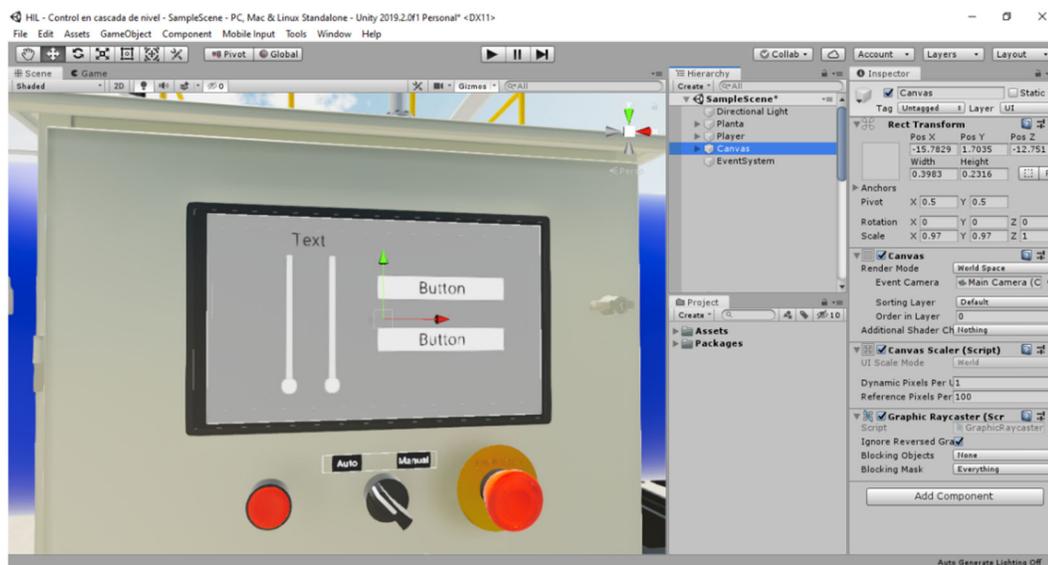


En la Figura 39, se observa que al GameObject "Player" se le ha añadido el script que permitirá el desplazamiento y, además, se han añadido los componentes "Capsule Collider" y "Rigidbody". Estos componentes permiten establecer características físicas sobre el objeto, con lo cual podrá colisionar con otros objetos que también contengan un componente "Collider", característica muy útil para una mejor y más realista experiencia de usuario.

Ahora, se requiere un HMI (interfaz humano-máquina) que permita el monitoreo de las variables del proceso y el cambio en la consigna. Para esto, se utilizan objetos tipo UI (User Interface) entre los cuales se encuentran botones, sliders, texto, etc. Estos objetos deben colocarse sobre un "Canvas" que sirve como base para el diseño y que debe estar configurado como "World Space" en el parámetro "Render Mode" para poder ubicarlo en cualquier lugar dentro de la escena. En la Figura 39, se muestra un "Canvas" y sus opciones de configuración, accesibles desde la ventana "Inspector".

Figura 39

Canvas “World Space” con elementos UI

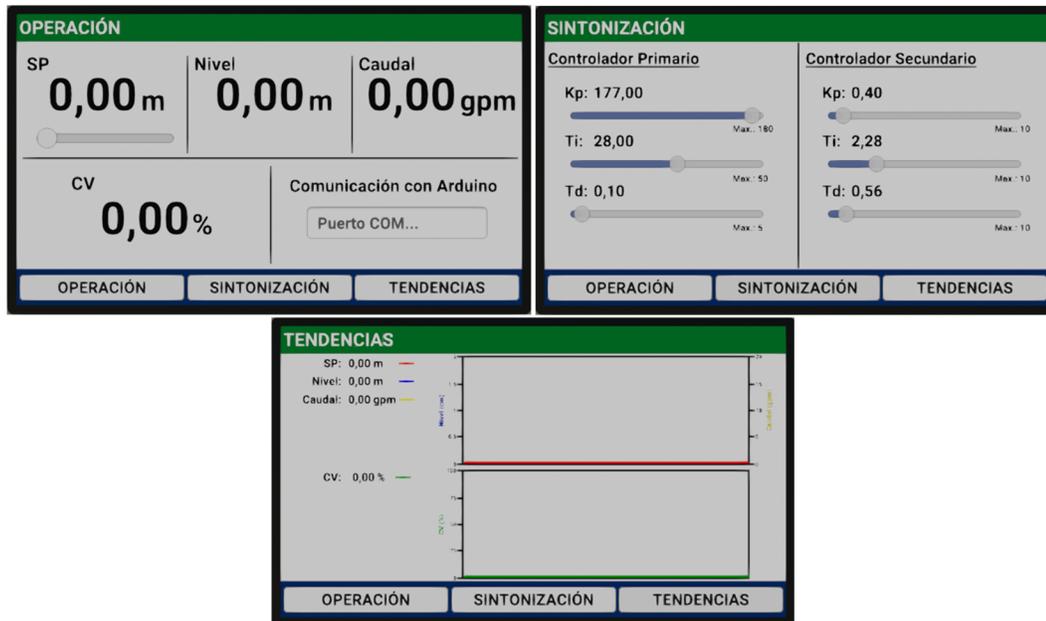


Cuando se añade un objeto “Canvas” a la escena, se crea automáticamente un objeto “EventSystem”, tal como se observa en la Figura 39. Este objeto contiene componentes que permiten la interacción del cursor del mouse con los objetos UI añadidos a la escena.

El HMI diseñado consta de 3 ventanas: operación, sintonización y tendencias. En la Figura 40, se muestran las ventanas de dicho HMI, mismas que son accesibles desde botones colocados en la parte inferior de la pantalla. La navegación en el HMI, la presentación de los datos y la manipulación de las sliders se implementa a través de un script añadido al canvas principal; las tendencias se grafican en tiempo real por medio de un script que utiliza un componente llamado “Line Renderer”.

Figura 40

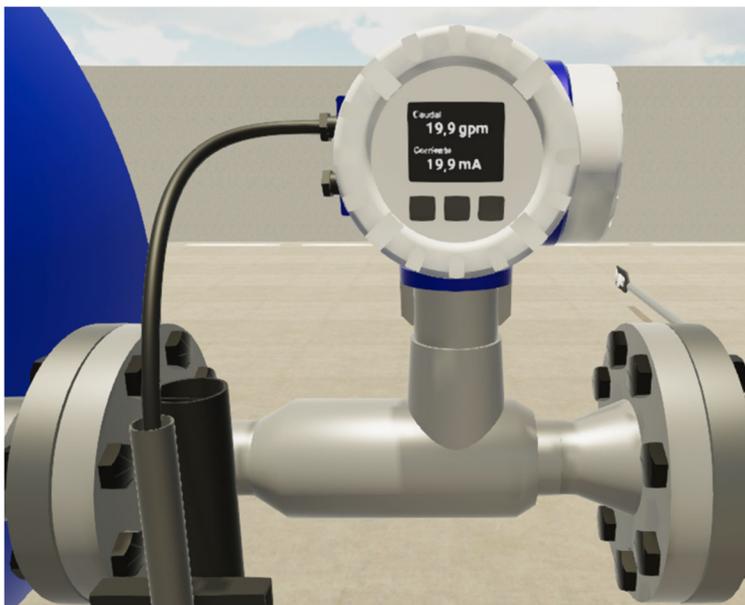
HMI diseñado con herramientas UI en Unity3D



Los datos de los transmisores de caudal y nivel se presentan utilizando herramientas UI al igual que en el caso del HMI. En la Figura 41 y en la Figura 42 se muestran los transmisores con sus respectivos datos.

Figura 41

Datos en el Transmisor de Caudal

**Figura 42**

Datos en el transmisor de nivel



Además, se incluye un indicador de nivel en la caldera, tal como se muestra en la Figura 43.

Figura 43

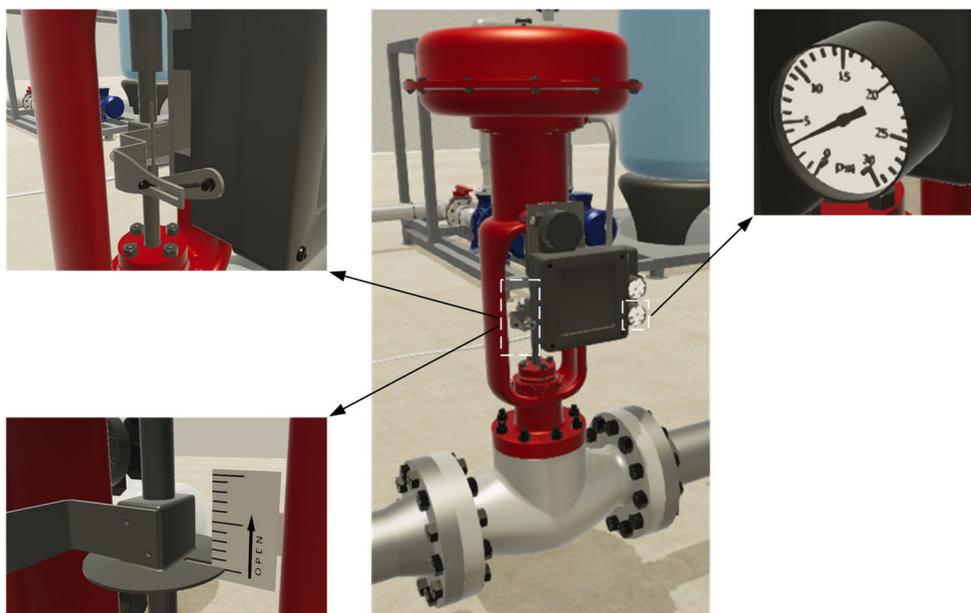
Indicador de Nivel



La válvula de control contiene un script que controla su animación según un dato de entrada en un rango de 0 a 100%. Es así que se ha implementado el movimiento del actuador y la deflexión del manómetro que indica la presión suministrada a la válvula como se muestra en la Figura 44.

Figura 44

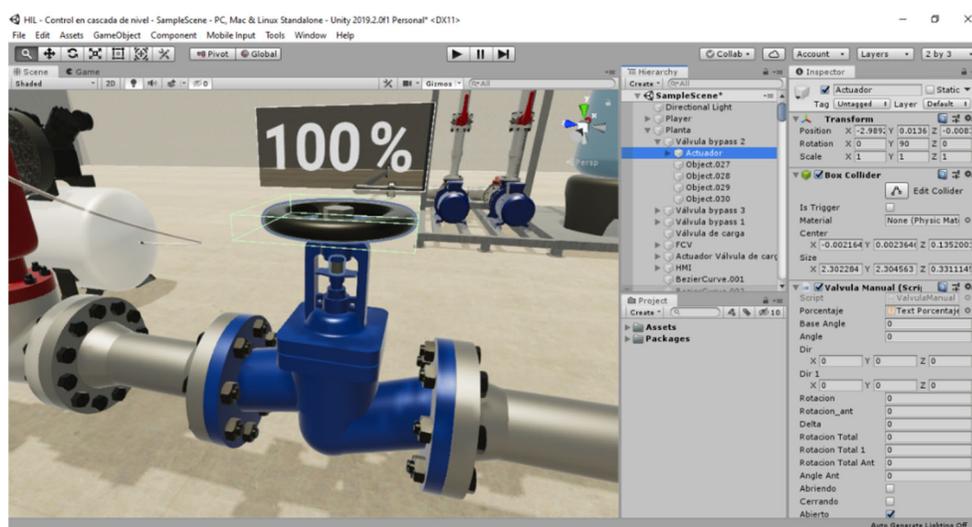
Animaciones de la válvula de control neumática



En el caso de las válvulas manuales, se implementó la capacidad de manipular el actuador de las mismas con el cursor a través de un movimiento circular del mismo, lo cual resulta cómodo y realista para el usuario. Además, el porcentaje de apertura de las válvulas manuales incidirá sobre el caudal de entrada, característica muy útil dentro de la simulación del proceso. Para esto, el actuador de la válvula debe contener un collider para la detección del cursor y un script con el algoritmo correspondiente para esta tarea, se incluye también un indicador para el porcentaje de apertura (Ver Figura 45).

Figura 45

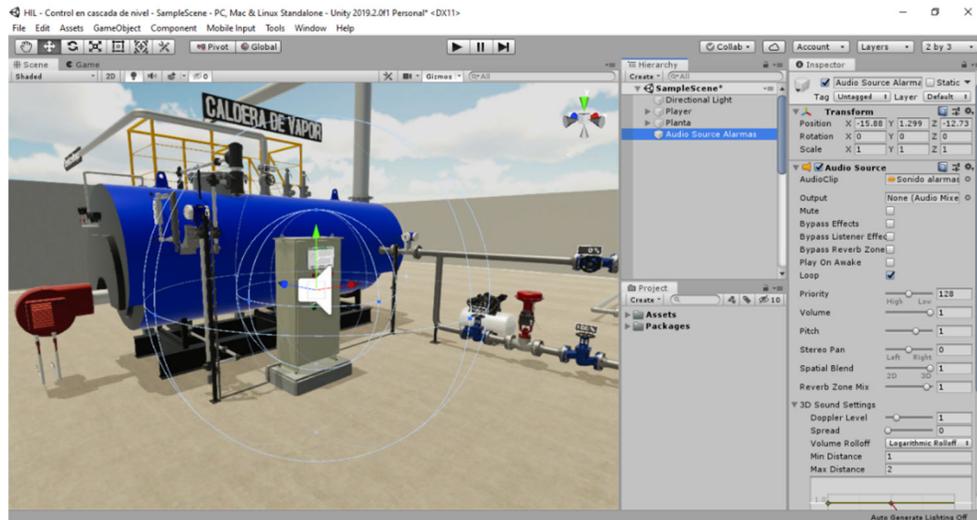
Válvula Manual Manipulable



Los sonidos de alarmas, bombas, interruptores y botones son importantes para complementar la aplicación y la experiencia del usuario. El componente que permite añadir sonidos en el entorno se denomina “Audio Source” y contiene varias opciones que pueden configurarse según las necesidades del desarrollador, en este caso los sonidos se propagarán de manera esférica y se escucharán según la posición del usuario en la escena, además, se activarán a partir de un script. En la Figura 46, se muestra uno de estos componentes utilizados para los sonidos de alarma.

Figura 46

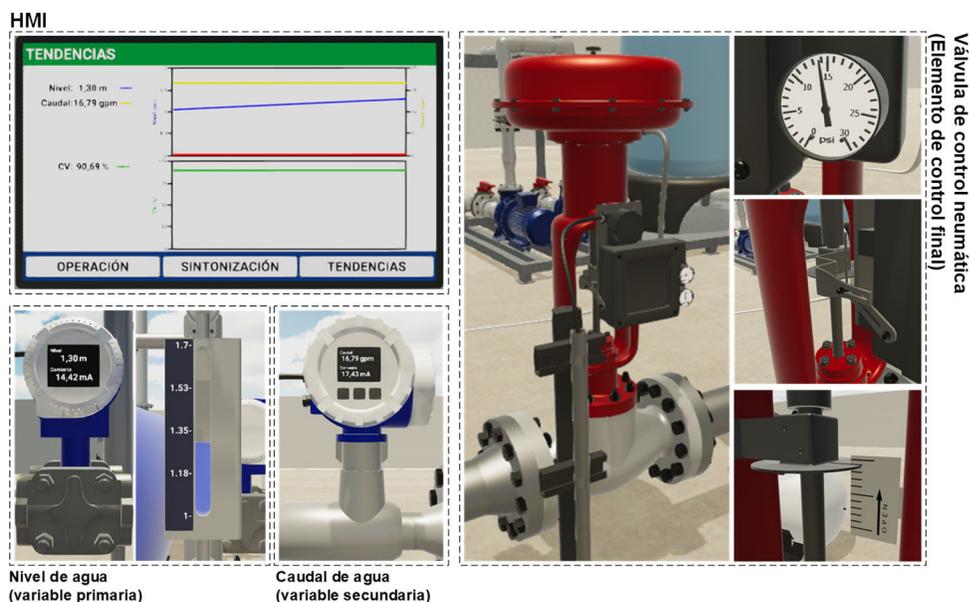
Componente de sonido dentro del entorno virtual



Una vez que se han definido las características visuales del entorno, se implementa el comportamiento del proceso en base a las ecuaciones definidas en la sección anterior. A través de un script, se simulan los comportamientos para el caudal de entrada y para el nivel de agua en la caldera basados en las ecuaciones (10) y (11), mismos que se programan dentro de un bucle repetitivo con un tiempo de muestreo de 0.02 segundos. En este script, se definen las variables involucradas en dichos comportamientos y se asignan los datos respectivos en el HMI y en los transmisores. En la Figura 47, se muestra la respuesta del sistema simulado cuando se ha aplicado una entrada escalón.

Figura 47

Respuesta del Sistema Simulado ante una Entrada Escalón



Como puede observarse en la Figura 47, al aplicar una variación en el elemento de control final, las variables involucradas en el proceso reaccionan según lo esperado y se presentan en tiempo real en los transmisores y en el HMI. De esta manera, se obtiene un proceso industrial virtual apto para ser utilizado en ensayos con controladores de todo tipo.

3.6.3 Adquisición y Generación de Señales

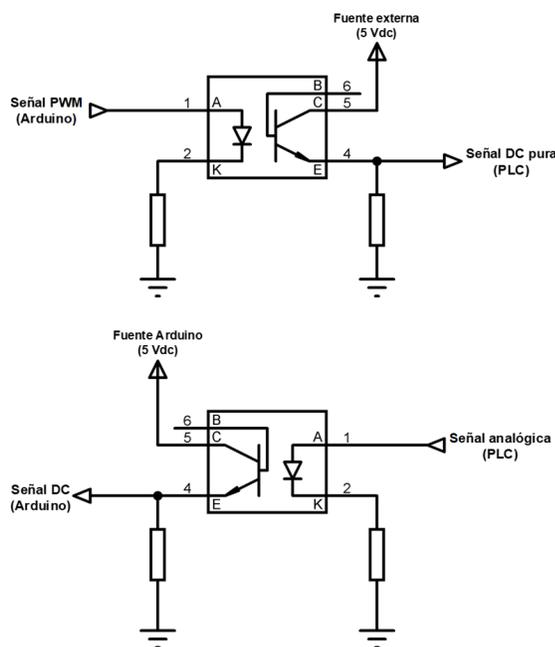
La simulación Hardware-In-the-Loop necesita de un método de comunicación entre el proceso simulado y el controlador que se requiere poner a prueba, este aspecto se detalla en secciones anteriores. En este caso, se ha optado por la generación de señales eléctricas para cada variable del proceso desde el computador en donde se ejecuta el entorno virtual. Esto requiere de una tarjeta de adquisición de datos para, a través de un puerto COM, leer los valores de las variables desde el entorno en tiempo real y traducirlas a un rango de voltaje utilizable por un controlador externo. Además,

esta tarjeta también servirá para leer el dato del parámetro de control calculado en el controlador (PLC) y transformarlo en una variable numérica para su uso en el sistema virtual implementado.

Para este proyecto, se ha escogido como tarjeta de adquisición de datos a un Arduino Mega 2560 por sus prestaciones y disponibilidad en el mercado, lo cual resulta ideal dado que el sistema se utilizará con fines didácticos. Arduino es capaz de leer y generar valores de voltaje continuo de 0 a 5V, por lo cual, dicho rango se manejará para todas las variables de entrada y salida, consideración tomada en la etapa de diseño de controladores. En este caso, se utiliza el modelo Arduino Mega 2560 que genera señales tipo PWM, mismas que deberán acondicionarse para convertirlas en señales DC, utilizables por el PLC como se mostró en la Figura 11. En la Figura 48, se muestra el circuito utilizado para el acondicionamiento de cada una de las señales por medio de optoacopladores.

Figura 48

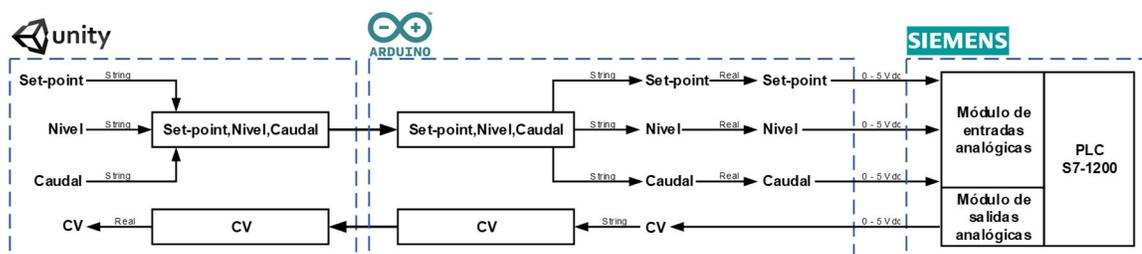
Acondicionamiento de Señales



Los datos de la simulación se intercambian entre el entorno virtual en Unity3D y Arduino a través de una comunicación serial por medio del puerto COM del computador. Para esto, se implementan scripts, en Unity3D y Arduino, que convierten los datos tipo *Real* a tipo *String*⁹ y los concatenan en una sola trama para enviarlos; luego, la trama recibida se desentrama y los datos se convierten nuevamente en tipo *Real* para su uso respectivo. De esta manera, se evita la pérdida de resolución de los datos y se optimiza la comunicación. En la Figura 49, se presenta un diagrama que describe el manejo de datos antes mencionado.

Figura 49

Intercambio de datos entre Unity3D y Arduino



3.6.4 Implementación de los Algoritmos de Control en el PLC

Los algoritmos de control han sido implementados en un PLC Siemens S7-1200, mismo que incluye los controladores primario y secundario con sus respectivas constantes de sintonía. Además, se utiliza un módulo de entradas analógicas (SM 1231 AI4) y un módulo de salidas analógicas (SM 1232 AQ2) para la adquisición y generación de las señales necesarias para el control.

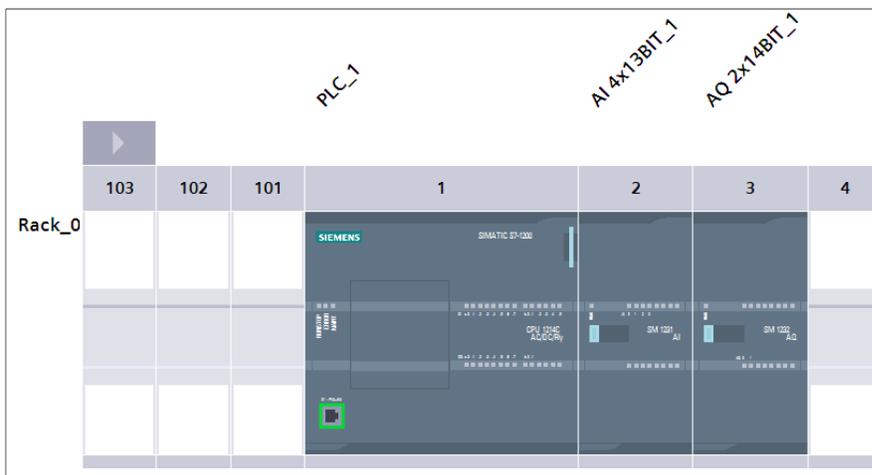
El primer paso para la implementación de la programación del PLC es la configuración del dispositivo y de los módulos de entradas y salidas analógicas en un

⁹ String: Tipo de dato que representa cadenas de caracteres.

nuevo proyecto creado en Tia Portal. En la configuración “Dispositivos y redes” se muestra el PLC y los módulos añadidos al proyecto (Ver Figura 50).

Figura 50

Vista de Configuración del PLC y los Módulos de Entradas y Salidas Analógicas (Tia Portal)



En este punto, se deben configurar las entradas y salidas de los módulos para que manejen rangos de tensión o corriente según se necesite. Para este caso, se utilizarán niveles de voltaje de 0 a 5V en las entradas y salidas, por lo cual el módulo SM 1231 se configura para que sus entradas acepten rangos de tensión de -5V a 5V (ver Figura 51) y del mismo modo se configura el módulo SM 1232 para un rango de -10V a 10V ya que no permite elegir un rango menor (ver Figura 52). Además, en este asistente de configuración, se indican las direcciones de cada canal para ser utilizadas en la programación.

Figura 51

Configuración de los Canales o Entradas Analógicas del Módulo SM 1231 (Tia Portal)

Canal0

Dirección de canal: IW96

Tipo de medición: Tensión

Rango de tensión: +/- 5 V

Filtrado: Fuerte (32 ciclos)

Activar diagnóstico de rebase por exceso

Activar diagnóstico de rebase por defecto

Figura 52

Configuración de los Canales o Salidas Analógicas del Módulo SM 1232 (Tia Portal)

Canal0

Dirección de canal: QW112

Tipo de salida analógica: Tensión

Rango de tensión: +/- 10 V

Valor sustitutivo para canal en caso de transición de RUN a S...: 0.000 V

Activar diagnóstico de cortocircuito

Activar diagnóstico de rebase por exceso

Activar diagnóstico de rebase por defecto

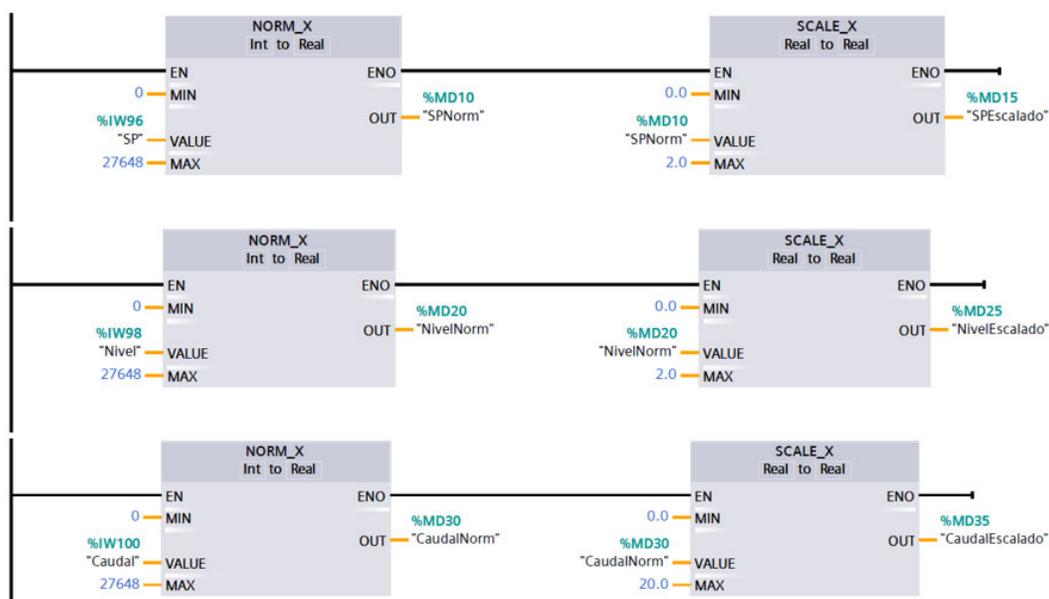
El PLC recibirá como datos de entrada el set-point, el nivel de agua y el caudal en las entradas o canales IW96, IW98 e IW100 del módulo SM 1231, respectivamente. Los datos recibidos deben normalizarse y escalarse para ser utilizados en cada controlador, es así que un rango de 0 a 5V se normaliza de 0 a 27648, según las especificaciones del convertor analógico-digital del dispositivo. En la Figura 53 se muestra la programación en Tia Portal que permite la normalización y escalamiento de las variables de entrada al módulo para lo cual se hacen uso de memorias o variables tipo Real¹⁰, todo esto en el bloque Main¹¹.

¹⁰ Real: Tipo de dato que representa un número real con determinada precisión.

¹¹ Main: Bloque de programación principal que se ejecuta en forma cíclica en Tia Portal.

Figura 53

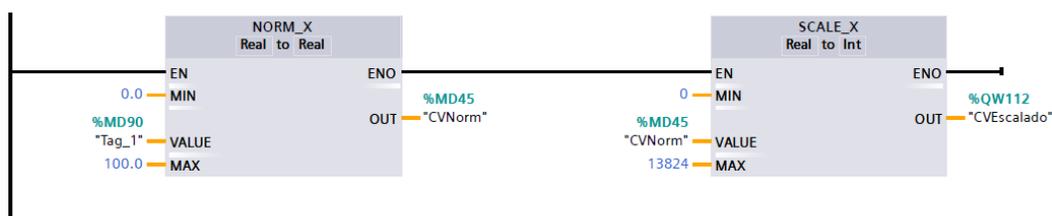
Normalización y escalamiento de señales de entrada (Lenguaje KOP – Tia Portal)



La salida QW112 del módulo SM 1232 se utilizará para generar la señal de control que se aplicará a la válvula neumática proporcional instalada en la entrada de agua de la caldera. Al igual que en el caso anterior, el parámetro de control se normaliza y escala para obtener un rango de voltaje utilizable para el elemento de control final. Dado que la salida se encuentra configurada en un rango de -10V a 10V, para un dato de 0 a 100% el escalamiento se hará de 0 a 13824, con esto se obtiene el rango de voltaje de 0 a 5V requerido. En la Figura 54 se muestra la programación desarrollada para este escalamiento

Figura 54

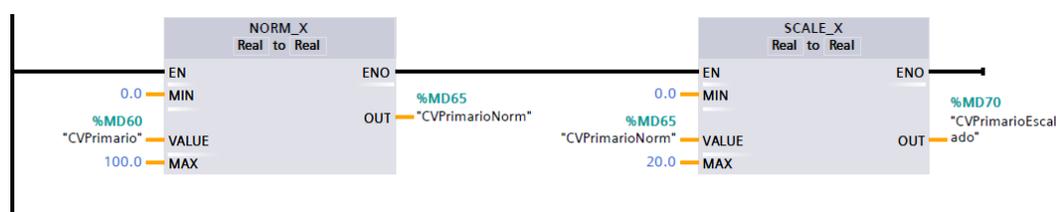
Normalización y escalamiento de señales de salida (Lenguaje KOP – Tia Portal)



Es necesario un escalamiento adicional para la salida del controlador primario ya que, como se muestra en la Figura 13, esta se utiliza como entrada para el controlador secundario. Por esta razón, es necesario convertir el rango de salida del controlador primario al rango de operación de la variable secundaria. La programación necesaria para dicho escalamiento se muestra en la Figura 55.

Figura 55

Escalamiento de la salida del controlador primario (Lenguaje KOP – Tia Portal)

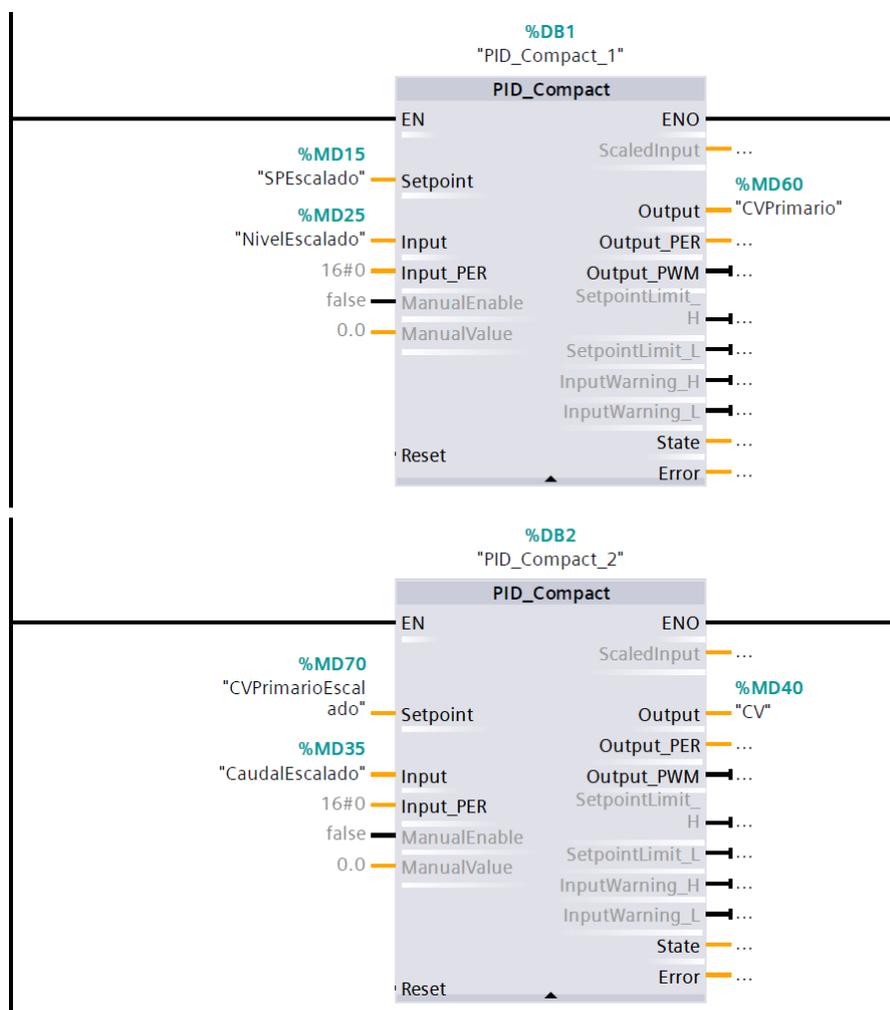


Los controladores se implementan en una función llamada *Cyclic_Interrupt*¹² y se configuran manualmente. Los bloques utilizados para el control se denominan "PID_Compact" y contienen todos los parámetros necesarios para implementar el controlador PID requerido. La programación y configuración de las variables escaladas requeridas por los bloques se muestra en la Figura 56. El bloque *PID_Compact_1* es el controlador primario y el bloque *PID_Compact_2* es el controlador secundario.

¹² *Cyclic_Interrupt*: Bloque de programación que interrumpe la ejecución cíclica del programa para ejecutarse en intervalos definidos tiempo de manera independiente.

Figura 56

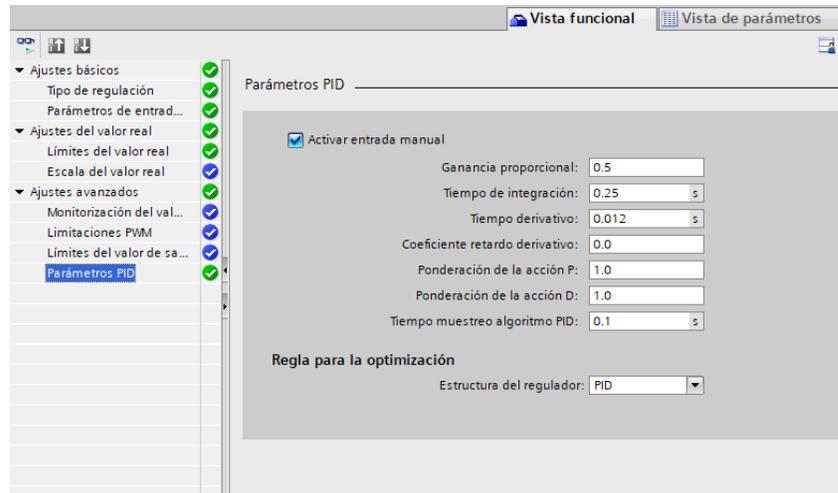
Programación de los controladores PID (Lenguaje KOP - Tia Portal)



Al seleccionar un bloque PID_Compact, se despliega un asistente de configuración que permite ingresar los valores de las constantes de sintonía y varias configuraciones adicionales relacionadas al controlador. En la Figura 57 se muestra el asistente antes mencionado.

Figura 57

Ventana de configuración de parámetros del bloque PID_Compact



A través de este asistente se pueden configurar varias opciones para complementar el rendimiento del controlador, tal como el inicio automático del controlador, tiempo de muestreo, ganancia del filtro derivativo, etc.

Considerando que el algoritmo de control del PLC S7-1200 es PID ISA (ideal), se utilizan las constantes obtenidas para cada uno de los controladores (primario y secundario), de acuerdo a los métodos de sintonía calculados en el apartado correspondiente.

CAPÍTULO IV

4. Resultados

Esta sección presenta los resultados obtenidos luego del diseño e implementación del sistema basado en Hardware-In-the-Loop y controlado mediante la estrategia de control en cascada.

4.1 Rendimiento de la Aplicación

La aplicación desarrollada en Unity3D ha sido optimizada a través de diferentes métodos que proporciona el mismo motor gráfico. Estas optimizaciones, además de las consideraciones tomadas en la etapa de diseño en Blender, permiten una buena fluidez en la aplicación, es decir, la interacción con los objetos y el desplazamiento a través del entorno se ejecutan sin ningún retardo, característica importante para la experiencia del usuario. El rendimiento de la aplicación puede monitorearse desde la ventana “Statistics”, tal como se muestra en la Figura 58.

Figura 58

Parámetros de rendimiento de la aplicación



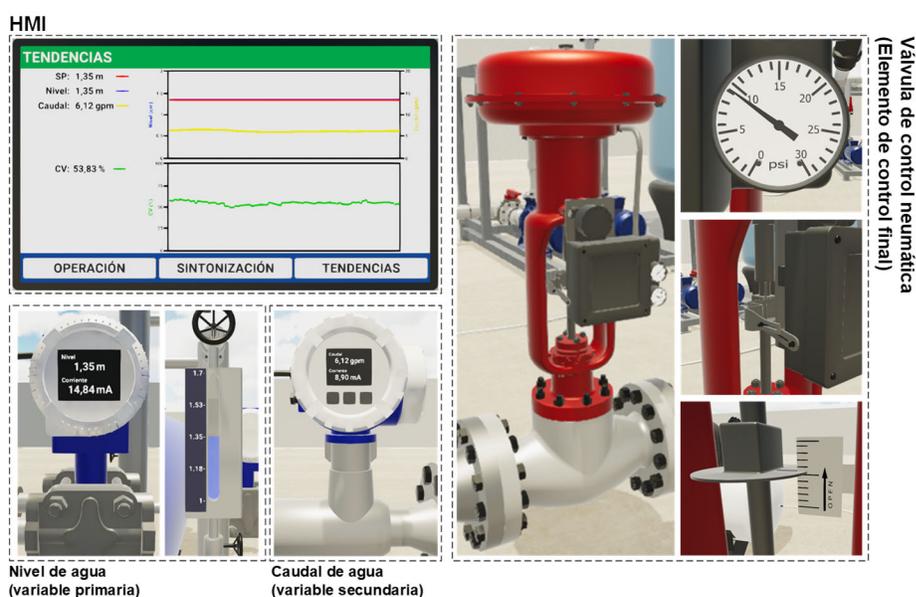
El envío y recepción de datos entre la entorno virtual y Arduino también se realiza en tiempo real ya que se han depurado y perfeccionado los códigos

implementados para este fin, añadiendo instrucciones que permiten disminuir los retardos y utilizando tiempos de espera congruentes entre ambos scripts.

Consecuentemente, la generación y adquisición de señales es eficiente y permite que el control ejecutado por el PLC resulte óptimo al no presentar retardos significativos. La Figura 59, presenta la respuesta del sistema regulado a través del control en cascada.

Figura 59

Respuesta del sistema virtual por control en cascada



4.2 Desempeño de la estrategia de control

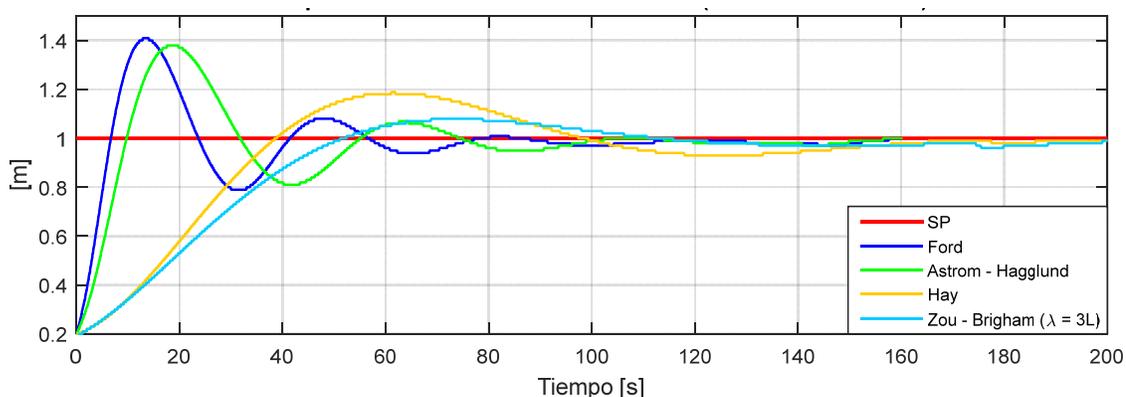
Para evaluar el rendimiento del control en cascada implementado, se han obtenido los datos de las variables de interés del sistema en modo automático bajo varios parámetros y se han generado gráficas que permiten realizar comparaciones y determinar las ventajas de la estrategia de control. La obtención de datos se realizó a través de la exportación de los mismos en archivos CSV desde el entorno virtual en Unity3D mientras el control se encontraba ejecutándose.

4.2.1 Comparación entre los métodos de sintonía utilizados para el control en cascada

En la Figura 60, se presenta una gráfica comparativa entre los resultados obtenidos para cada método de sintonía utilizados para el control en cascada.

Figura 60

Comparación entre métodos de sintonía para el control en cascada (PLC)



Se observa que los resultados experimentales obtenidos para cada método son similares a los obtenidos por simulación anteriormente, lo cual valida los procedimientos utilizados en cada caso. En la Tabla 7, se presentan parámetros que permiten evaluar cada método de sintonía utilizado para poder compararlos entre sí.

Tabla 7

Parámetros de rendimiento para cada método de sintonía para el control en cascada

Método de sintonía	Sobreimpulso	Tiempo de establecimiento	IAE
Ford	41%	152.4 s	46.95
Astrom-Hagglund	39.4%	96.13 s	14.84
Hay	20.2%	157.9 s	26.88
Zou-Brigham ($\lambda = 3L$)	6.93%	102.1 s	24.56

Se observa que el porcentaje de sobreimpulso es menor para el método de Zou-Brigham, pero su respuesta es más lenta. El valor de la integral del error absoluto (IAE) para el método de Ford es el mayor ya que presenta más oscilaciones que el resto y el tiempo de establecimiento es considerablemente alto, lo cual es evidenciable en la gráfica. Estos parámetros, los cuales brindan información relevante sobre cada método de sintonía, son de gran importancia para la selección definitiva del controlador óptimo.

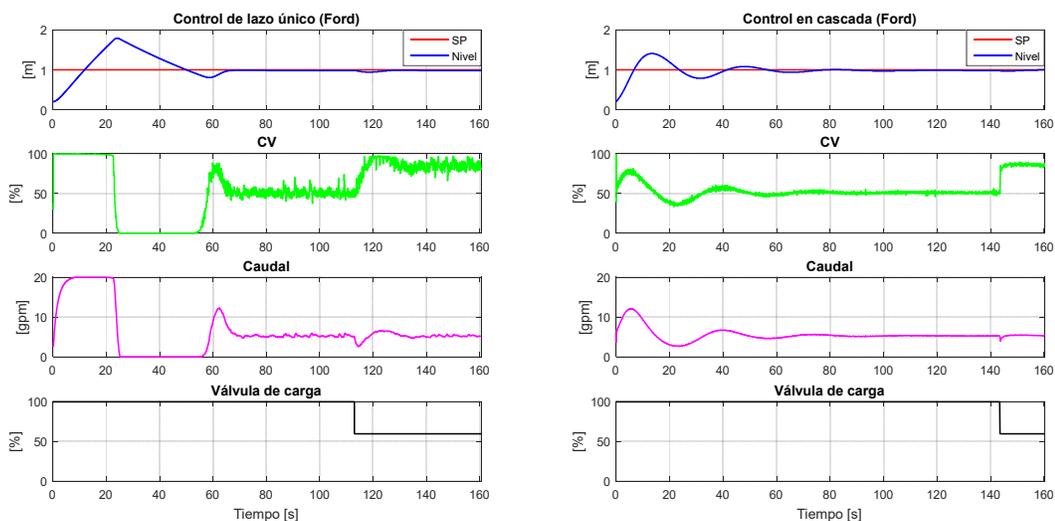
4.2.2 Comparación Entre el Control Clásico de Lazo Único y el Control en Cascada

La principal ventaja de la estrategia de control en cascada es la capacidad que tiene de compensar los efectos causados por perturbaciones antes de que estos influyan sobre la variable primaria del proceso, tal como se menciona en la sección correspondiente. Para evidenciar las ventajas que tiene el control en cascada sobre el control clásico de lazo único, se ha implementado un controlador PID simple para la regulación del nivel de la caldera y se han generado gráficas comparativas para cada método de sintonía considerado

En la Figura 61, se presentan gráficas comparativas para el método de sintonía de Ford. Se observa que para el control clásico simple el sistema presenta una respuesta transitoria más abrupta y la compensación que ofrece ante una perturbación es más lenta en comparación con el control en cascada por lo cual la variable primaria se ve afectada. En cambio, el control en cascada ofrece una respuesta rápida cuando se aplica una perturbación, evitando así que la variable primaria presente cambios no deseados. Además, se observa que el parámetro de control (CV) es más estable y menos agresivo en el caso del control en cascada, lo cual resulta beneficioso dado que esta señal se aplica al elemento de control final.

Figura 61

Gráfica comparativa para el control clásico de lazo único vs. control en cascada (Método de Ford)



En la Tabla 8, se presentan los parámetros de rendimiento para cada estrategia de control utilizando el método de sintonía de Ford. La principal diferencia que se observa es que el control de lazo único resulta más rápido que el control en cascada, pero el sobreimpulso es significativo. Se agrega carga a las dos estrategias de control, observando que, en el control en cascada, el lazo secundario compensa la perturbación y no se ve afectada la estabilidad del lazo primario.

Tabla 8

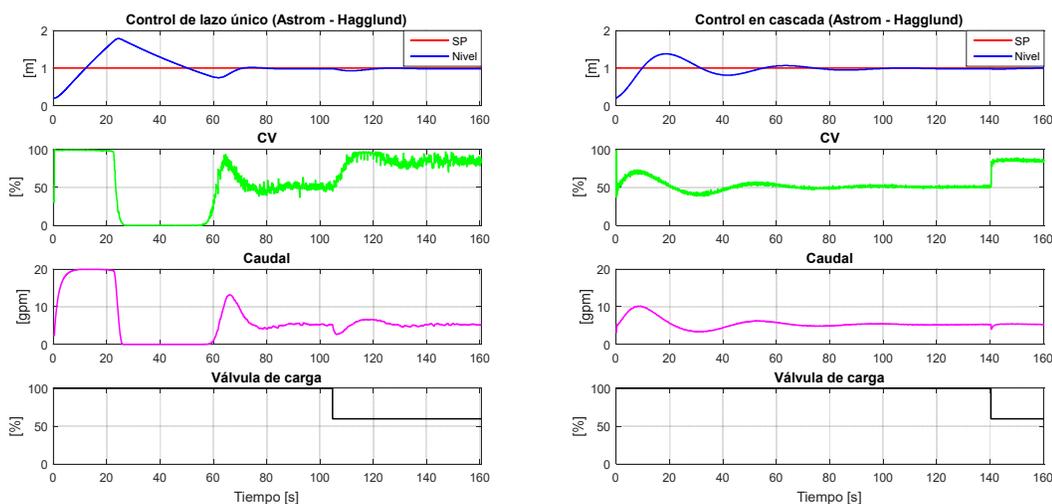
Parámetros de Rendimiento para cada Estrategia de Control - Método de Sintonía de Ford

Estrategia de control	Sobreimpulso	Tiempo de establecimiento	IAE
Control de lazo único	79%	67.17 s	24.07
Control en cascada	41%	152.4 s	46.95

En la Figura 62, se presentan gráficas comparativas para el método de sintonía de Astrom - Hagglund. El desempeño de esta estrategia de control es similar al caso anterior.

Figura 62

Gráfica Comparativa para el Control Clásico de Lazo único vs. Control en Cascada (Método de Astrom - Hagglund)



En la Tabla 9, se presentan los parámetros de rendimiento para cada estrategia de control utilizando el método de sintonía de Astrom - Hagglund. De manera similar al caso anterior, el control de lazo único describe una respuesta más rápida que el control en cascada, pero el sobreimpulso que presenta es mucho mayor.

Tabla 9

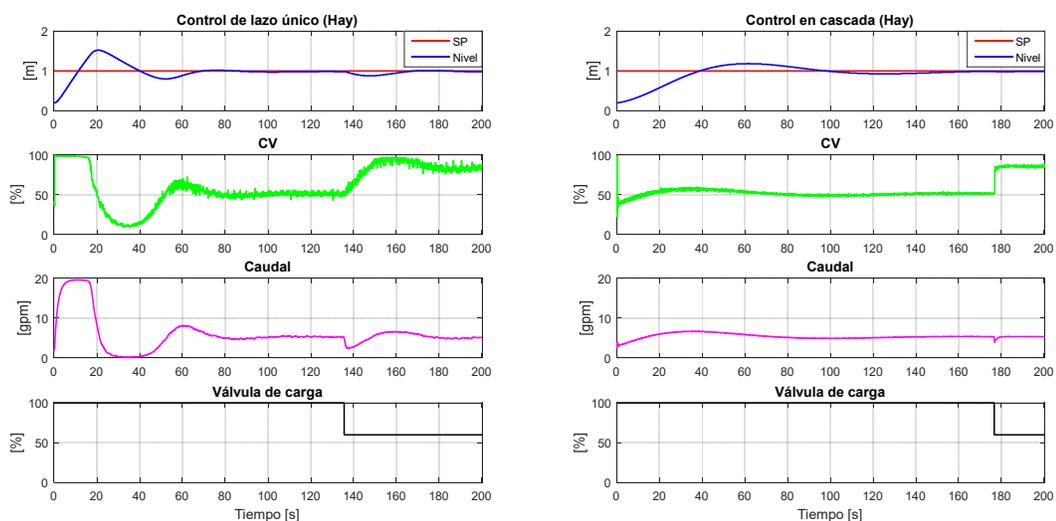
Parámetros de Rendimiento para cada Estrategia de Control - Método de Sintonía de Astrom - Hagglund

Estrategia de control	Sobreimpulso	Tiempo de establecimiento	IAE
Control de lazo único	79%	76.02 s	25.4
Control en cascada	39.4%	96.13 s	14.84

En la Figura 63, se presentan gráficas comparativas para el método de sintonía de Hay. Se observa que el control en cascada presenta una respuesta transitoria más lenta y el parámetro de control es más estable en comparación con el control de lazo único. Además, la compensación de la perturbación aplicada es mucho más óptima en el control en cascada ya que la variable primaria no presenta variaciones cuando se cierra la válvula de la línea de entrada de agua, en cambio, en el control de lazo único, se observa una variación significativa en el nivel.

Figura 63

Gráfica Comparativa para el Control Clásico de Lazo único vs. Control en Cascada (Método de Hay)



En la Tabla 10, se presentan los parámetros de rendimiento para cada estrategia de control utilizando el método de sintonía de Hay. En este caso, el control en cascada presenta un sobreimpulso menor en comparación con los anteriores métodos de sintonía analizados.

Tabla 10

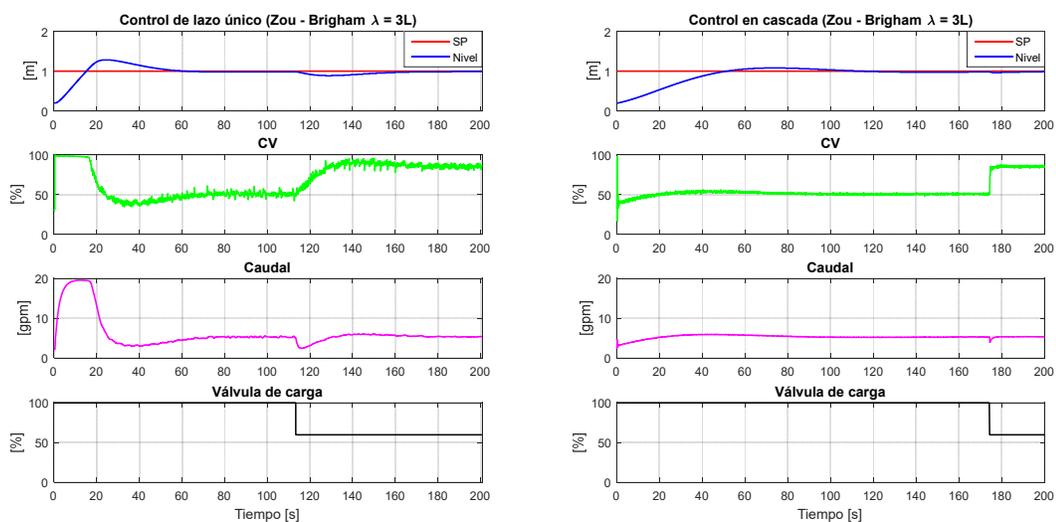
Parámetros de rendimiento para cada estrategia de control - Método de sintonía de Hay

Estrategia de control	Sobreimpulso	Tiempo de establecimiento	IAE
Control de lazo único	54.55%	66.25 s	20.57
Control en cascada	20.2%	157.9 s	26.88

En la Figura 64, se presentan gráficas comparativas para el método de sintonía de Zou – Brigham. Este método de sintonía presenta una respuesta más lenta que los anteriores y esto se puede modificar variando el parámetro λ . La ventaja que ofrece este método sobre los demás es que dicho parámetro permite un mayor control sobre el resultado final. A pesar de esto, la compensación que ofrece el control en cascada es óptima dado que es el controlador secundario el que reacciona a la perturbación aplicada.

Figura 64

Gráfica comparativa para el control clásico de lazo único vs. control en cascada (Método de Zou - Brigham)



En la Tabla 11, se presentan los parámetros de rendimiento para cada estrategia de control utilizando el método de sintonía de Zou – Brigham.

Tabla 11

Parámetros de Rendimiento para cada Estrategia de Control - Método de Sintonía de Zou - Brigham

Estrategia de control	Sobreimpulso	Tiempo de establecimiento	IAE
Control de lazo único	30.61%	62.27 s	18.62
Control en cascada	6.93%	102.1 s	24.56

Se observa que, en todos los casos, el control en cascada presenta un comportamiento más lento que el control de lazo único y esto es evidenciable en sus parámetros de rendimiento, pero, en cambio, el sobreimpulso es menor y la compensación de las perturbaciones es significativamente mejor. Cabe recalcar que los controladores pueden implementarse utilizando otras estructuras (P, PI, PID) según las necesidades del proceso y del diseñador.

Finalmente, se ha considerado un método de sintonía empírico (método de tanteo) para un análisis más amplio del rendimiento de la estrategia de control en cascada y de las alternativas de diseño disponibles. Este método se ha aplicado a los dos controladores, siguiendo las normas de sintonía descritas en secciones anteriores. En la Tabla 12 se presentan las constantes de sintonía obtenidas a través del método empírico y en la Figura 65 se muestran gráficas comparativas para dicho método.

Tabla 12

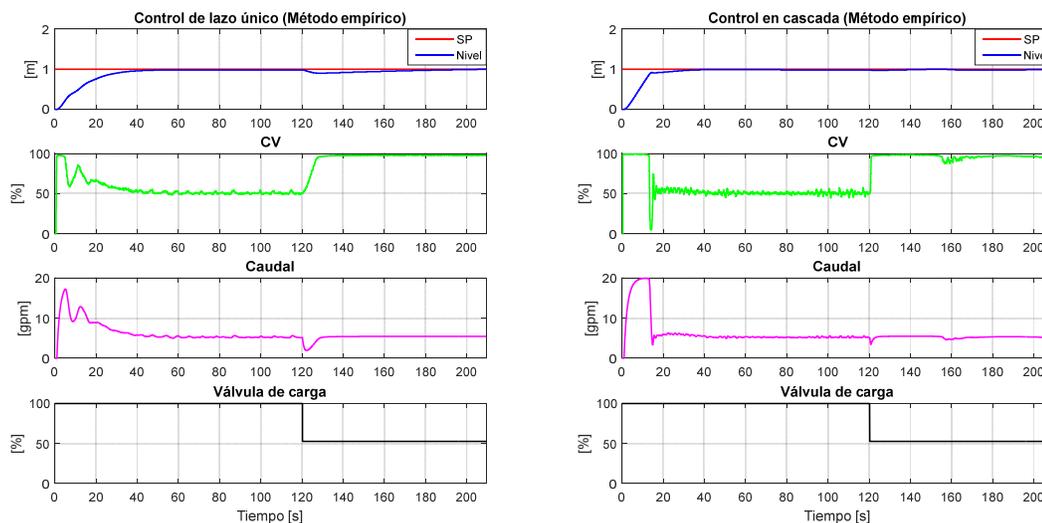
Constantes de Sintonía de los Controladores Primario y Secundario obtenidas

por el Método Empírico

Controlador/Constantes	Kp	Ti	Td
Controlador secundario	190	8.85	0.0012
Controlador primario	0.435	0.891	0.08

Figura 65

Gráfica Comparativa para el Control Clásico de Lazo Único vs. Control en Cascada (Método Empírico)



En la Tabla 13, se presentan los parámetros de rendimiento para cada estrategia de control utilizando el método de sintonía empírico.

Tabla 13

Parámetros de Rendimiento para cada Estrategia de Control - Método de Sintonía Empírico

Estrategia de control	Sobreimpulso	Tiempo de establecimiento	IAE
Control de lazo único	0%	42.36 s	19.75
Control en cascada	0%	29.52 s	12.29

El método empírico presenta resultados más satisfactorios en comparación con los métodos analíticos anteriormente descritos, haciendo uso de una sintonía más agresiva. Cabe señalar que la rapidez de la respuesta del sistema depende en gran medida de la sintonía del controlador secundario y la elección del método de sintonía depende de las características y requerimientos del proceso industrial.

4.3 Comprobación de la hipótesis

El Entorno virtual 3D de una planta de nivel, mediante la técnica Hardware-In-the-Loop, para el control avanzado de procesos implementado, es una solución de bajo costo ya que su valor estimado es de: 1500 dólares, en comparación con sistemas didácticos comerciales que ofrecen las mismas características cuyos precios oscilan entre los 20000 y 25000 dólares. (FESTO, 2021).

El Entorno virtual 3D de una planta de nivel, mediante la técnica Hardware-In-the-Loop para el control avanzado de procesos, fue analizado en cada una de sus etapas, la etapa de animación 3D de los equipos e instrumentos, así como el proceso presentan un funcionamiento óptimo, similar a un sistema de control de nivel. En la etapa de HIL se emulan señales eléctricas de transmisores y elementos de control final, señales que son adquiridas de forma adecuada por el autómatas programable, finalmente en la etapa de controladores el sistema permite el diseño e implementación de la estrategia de control en cascada, indicando claramente que el sistema en cascada es mucho más eficiente que un control en lazo simple. Así también se consideró la facilidad en el uso del sistema lo que permite ser utilizado por estudiantes en el área de control automático.

Adicionalmente se realizaron pruebas de funcionamiento del Entorno virtual 3D de una planta de nivel, mediante la técnica Hardware-In-the-Loop, para el control avanzado de procesos para determinar la usabilidad. Los usuarios del sistema son 10 estudiantes de la asignatura Control de Procesos del noveno nivel de la carrera de Ingeniería Electrónica e Instrumentación (Periodo NOV 2020 – ABR 2021), los mismos que poseen conocimientos teóricos de control de procesos y conceptos relacionados al control en cascada.

Las respuestas de las preguntadas planteadas a los estudiantes se encuentran a continuación:

1. ¿Ha identificado y comprendido todas las etapas del sistema de entrenamiento?

Figura 66

Resultado de la Pregunta 1



2. ¿El entorno virtual presenta un alto nivel de realismo?

Figura 67

Resultado de la Pregunta 2



3. ¿La interacción con los componentes de la planta es intuitiva?

Figura 68

Resultado de la Pregunta 3



4. ¿El entorno virtual le ha permitido identificar todas las partes del proceso industrial?

Figura 69

Resultado de la Pregunta 4



5. ¿Las variables del proceso se presentan de una forma clara?

Figura 70

Resultado de la Pregunta 5



6. ¿El sistema de entrenamiento le ha proporcionado información relevante sobre el control de procesos?

Figura 71

Resultado de la Pregunta 6



7. ¿Es capaz de identificar instrumentos industriales reales luego de observar la planta virtual?

Figura 72

Resultado de la Pregunta 7



8. ¿Luego de analizar la planta virtual y la estrategia de control en cascada implementada, comprende los fundamentos de este método de regulación?

Figura 73

Resultado de la Pregunta 8



4.4 Análisis

De las encuestas realizadas se determina que el entorno virtual presenta un alto nivel de realismo, así como permite identificar las etapas del proceso, los usuarios identificaron de forma clara las variables del proceso, así como indican que el sistema les ha proporcionado información relevante sobre el control en cascada y comprenden de forma práctica el diseño e implementación de esta estrategia de control.

Por lo expuesto anteriormente se valida la hipótesis planteada.

CAPÍTULO V

5. Conclusiones y Recomendaciones

5.1 Conclusiones

El entorno virtual 3D de la planta de nivel, está conformado por dos modelos matemáticos, el lazo principal tiene la respuesta dinámica de un proceso integrante (Nivel del tanque) y el lazo secundario tiene la respuesta dinámica de un proceso autorregulado (Caudal de la tubería), los mismos que permiten la simulación del sistema de control en cascada.

El entorno virtual 3D de la planta de nivel fue desarrollado en base al sistema de alimentación de agua en una caldera de vapor (utilizado en la industria de petróleo y gas), en el diseño se empleó las normas ISA S5.1, e ISA S5.3, así como los instrumentos y equipos industriales fueron diseñados a detalle con un alto nivel de realismo, proporcionando a los usuarios un sistema virtual con gran similitud a un proceso industrial real.

La técnica Hardware-In-The-Loop, hace uso de un entorno virtual interactivo que permite manipular varios elementos de la planta industrial para evaluar el rendimiento de la estrategia de control en cascada implementado en un Controlador Lógico Programable (PLC).

Del diseño e implementación de varios algoritmos (PI-PID) y la estrategia de control en cascada, se determina que el controlador más eficiente para el lazo primario es el algoritmo PID con las constantes de sintonía ($K_p=0.43$; $T_i=0.891s$; $T_d=0.08s$), y para el lazo secundario es el algoritmo PID con las constantes de sintonía ($K_p=190$; $T_i=8.85s$; $T_d=0.0012s$). Estos controladores proporcionan un sistema de control sin sobreimpulso con un tiempo de establecimiento de 29.52 segundos y la Integral Absoluta del Error (IAE) de

12.29, y presenta una mejor respuesta en comparación con el control en lazo único.

El nivel de detalle del entorno virtual, permite que los usuarios se familiaricen con las características físicas y operativas de los equipos e instrumentos industriales que intervienen en un sistema de control de nivel (control en cascada), asociados a la alimentación de agua en una caldera de vapor, como se muestran en los resultados obtenidos de las encuestas realizadas a los estudiantes de la asignatura Control de Procesos.

El entorno virtual es una alternativa de bajo costo para el entrenamiento del sistema de control en cascada, ya que permite visualizar la respuesta dinámica de los instrumentos y equipos industriales, además proporciona las herramientas necesarias para el aprendizaje en control automático.

5.2 Recomendaciones

Debido a que el diseño del entorno 3D representa una herramienta de entrenamiento en control de procesos, es recomendable seleccionar un sistema que contenga instrumentos y equipos utilizados en procesos industriales reales para que el estudiante se pueda familiarizar fácilmente con las características físicas y operativas de los mismos.

Para la implementación del control en cascada es importante considerar que la variable primaria del proceso, en este caso el nivel de alimentación de agua de la caldera presenta una respuesta lenta, y el flujo en la tubería que corresponde a la variable secundaria presenta una respuesta rápida, lo que permita un desempeño óptimo de la estrategia de control en cascada dentro del proceso.

Bibliografía

- Adam, E. (2020). *Instrumentación y Control de Procesos Notas de clase*. Santa Fe: UNL.
- Altertecnica. (01 de Enero de 2018). *Altertecnica*. Recuperado el 21 de Diciembre de 2020, de <https://altertecnica.com/realidad-aumentada-realidad-virtual-industria-40/>
- AT Automation. (01 de Enero de 2019). *automation PROCESS FACTORY&BUILDING SOLUTIONS*. Recuperado el 01 de Enero de 2021, de <https://atautomation.com.mx/>
- Atkins. (2009). Virtual experience: Observations on Second Life. En M. Purvis y B. Savarimuthu (Eds.). *Computer Science: ComputerMediated social networking* (págs. 7-17). Berlin: Springer.
- Automation, Mobile. (01 de Enero de 2017). *Mobile Automation*. Recuperado el Abril de 2021, de <https://www.mobileautomation.com.au/plc-industrial-application/>
- Bacic, M. (2005). On hardware-in-the-loop simulation. *44th IEEE Conference on Decision and Control*, 5.
- Bin, L., Wu, X., Figueroa, H., & Monti, A. (2007). A Low-Cost Real-Time Hardware-in-the-Loop Testing Approach of Power Electronics Controls. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*, 13.
- Blázquez, D. (22 de Enero de 2006). *Optimización de gráficos 3D para video juegos*. Recuperado el 22 de Agosto de 2020, de <http://www.exelweiss.com/blog/18/optimizacion-de-graficos-3d-para-videojuegos/>
- Blender. (01 de febrero de 2020). *Blender 2.93 Reference Manual*. Recuperado el 25 de febrero de 2021, de https://docs.blender.org/manual/en/latest/interface/window_system/workspaces.html
- Camacho, O. E. (1998). Sliding mode control: a robust approach to integrating systems with dead time. *Second IEEE International Caracas Conference on Devices, Circuits and Systems. ICCDCS 98. On the 70th Anniversary of the MOSFET and 50th of the BJT.*, (págs. 401-406). Caracas.
- Cantón Enríquez, D., Arellano Pimentel, J. J., & Hernández López, M. Á. (2017). Uso didáctico de la realidad virtual inmersiva con interacción natural de usuario enfocada a la inspección de aerogeneradores. *Apertura*, 8-11.
- Coughran, M. T. (2021). Lambda Tuning—the Universal Method for PID Controllers in Process Control. *Emerson*, 3.
- De Farias, A., Rodriguez, R., Murillo , A., Lopes, R., & Avila , S. (2019). Low-Cost Hardware-in-the-Loop Platform for Embedded Control Strategies Simulation. *IEEE Access*, 14.

- Diego A. Mejía Giraldo, R. A. (2007). Técnicas de control adaptativas aplicadas a sistemas multivariables (M.I.M.O.). *Scientia Et Technica*, vol. XIII, núm. 35, 37-38.
- Ellis, G. (2012). Desarrollo y verificación de modelos. En G. Ellis, *Guía de diseño de sistema de control* (págs. 279-280).
- Fakir Felipe, B. d. (2017). HARDWARE IN THE LOOP SIMULATION FOR INDUSTRIAL PROCESS CONTROL. *GASI*, 1-8.
- Fällman. (1999). VR in Education: An Introduction to Multisensory Constructivist Learning Environments. *Universitets pedagogisk konferens Umeåuniversitet*.
- Fang, K., Zhou, Y., Ma, P., & Yang, M. (2018). Credibility Evaluation of Hardware-in-the-Loop Simulation Systems. *The 30th Chinese Control and Decision Conference (CCDC)*, 6.
- FESTO. (01 de Enero de 2021). *FESTO*. Recuperado el 01 de Mayo de 2021, de https://www.festo.com/es/es/e/educacion/sistemas-de-aprendizaje/automatizacion-de-procesos/instrumentacion-industrial-y-control-de-procesos-id_33990/
- Giraldo B. D., T. G. (1997). *Teoría de Control*. Universidad Tecnológica de Pereira.
- Hong, Y., & Deng, N. (2016). Design of water tank level cascade control system based on siemens S7-200. *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, 3.
- Jhonathan Caicedo, G. Z. (2018). Sintonización por método de Tanteo. En G. Z. Jhonathan Caicedo, *SISTEMA SCADA UTILIZANDO PROTOCOLOS INDUSTRIALES ETHERNET, MODBUS Y WIRELESS PARA EL MONITOREO Y CONTROL DE PROCESOS* (pág. 63). Latacunga: ESPEL.
- King. (2016). *Process Control*. United Kingdom: Wiley.
- Logitek. (25 de Marzo de 2021). *La virtualización en las plantas industriales*. Recuperado el 25 de Marzo de 2021, de High Availability by Logitek: <http://www.altadisponibilidadlogitek.com/wp-content/uploads/White-paper-Virtualization-in-Industrial-Plants-Industrial-Automation-ESSP-1.pdf>
- López, A. A. (2001). *El control de procesos industriales y su influencia en el mantenimiento*. Lima: Universidad de Lima.
- Lucidchar. (20 de Enero de 2021). *Lucidchar*. Recuperado el 20 de Enero de 2021, de <https://www.lucidchart.com/pages/es/que-son-los-diagramas-de-tuberias-e-instrumentacion>
- Muraspahic, S., Gu, P., Farji, L., & Iskandarani, Y. (2012). Hardwa-re-in-the-loop implementation for an active heave compensated drawworks. *Open Engineering*,, 201-211.
- Murrill, P. W. (2000). *Fundamentals of Process Control Theory*. North Carolina.

- Nalbantoğlu, M., & Kaya, İ. (2015). Cascade Controller Design using Controller. *Control and Computing (ICSTCC)*, 32-36.
- O'Dwyer, A. (2006). *Handbook of PI and PID Controller Tuning Rules*. London: Imperial College Press.
- O'Dwyer, A. (2006). PID Controller Structures. En A. O'Dwyer, *Handbook of PI and PID Controller Tuning Rules* (pág. 7). London: Imperial College Press.
- Oscar Arriola Navarrete y Avila González, A. (2008). El software libre y la enseñanza de la catalogación: una relación amistosa. *Codice. Vol. 4, núm. 2*, 21-32.
- Otero, F. C., & Optimo Group Inc. (01 de Diciembre de 2020). *Optimización de Control de Procesos*. Recuperado el 01 de Diciembre de 2020, de Optimum Process Performance – Minimum Process Variability: www.optimogrp.com
- P. Hafner, V. H. (2013). Teaching methodology for Virtual Reality Practical Course in Engineering Education. *in Proc. Of the 2013 Int. Conf. on Virtual and Augmented Reality in Education, vol. 25*, (págs. 251-260). Obtenido de Skills2learn.com.: <http://www.skills2learn.com/what-is-virtual-reality.html>.
- Park, S. C., Park, C., Wang, G. N., & Kwak, J. (2008). PLCStudio: Simulation based PLC code verification. *Winter Simulation Conference IEEE.*, 222-228.
- Pinzás, A. R. (2009). Software libre aplicado al dibujo industrial: el caso Blender. *Revista de la Facultad de Ingeniería Industrial*, 1-6.
- Rad, C., Maties, V., Hancu, O., & Lapusan, C. (2012). Hardware-In-The-Loop (HIL) Simulation Used for Testing Actuation System of a 2-DOF Parallel Robot. *Applied Mechanics and Materials*, 334-343.
- Raposo, A., Soares, L., Wagner, G., Corseuil, E., & Gatass, M. (2009). Environ: Integrating VR and CAD in Engineering Projects. *Projects in VR*, 5.
- Ray-Tracing*. (01 de Octubre de 2020). Recuperado el 01 de Octubre de 2020, de [.cdschools.org](https://www.cdschools.org): https://www.cdschools.org/cms/lib/PA09000075/Centricity/Domain/81/5th-Chapter_9.pdf
- Sánchez, H. S., Vilanova, R., & Arrieta, O. (2014). Implementación de controladores PID: Equivalencia y Optimalidad. *Jornadas de Automática*, 1 - 6.
- Siemens. (5 de Marzo de 2021). *Siemens*. Recuperado el 5 de Marzo de 2021, de <https://www.siemens.com/global/en.html>
- Smith, C. A., & Corripio, A. B. (1991). *Control Automático de Procesos Teoría y Práctica*. México D.F.: LIMUSA.
- Swapnil Arun Nameka, R. Y. (15 de Abril de 2020). Programmable Logic Controller (PLC) and its Applications. *INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN TECHNOLOGY*, págs. 1-5.

- Tecnawebs. (01 de septiembre de 2020). *Técnicas de diseño web y programación Blender*. Recuperado el 01 de septiembre de 2020, de http://blender.tecnawebs.es/conceptos_basicos.html
- Tejado, I., Serrano, J., Pérez, E., Torres, D., & Vinagre, B. M. (2016). Low-cost hardware-in-the-loop tested of a mobile robot to support learning in automatic control and robotics. *IFAC-PapersOnLine*, 242-247.
- Togores, R. N. (1 de Enero de 2019). *Formas y Funciones*. Recuperado el 3 de Marzo de 2021, de <http://www.togores.net/3d-rv/tecnologiayrv/realidad-virtual-01>
- Unity. (13 de Septiembre de 2020). *Unity*. Recuperado el 13 de Septiembre de 2020, de <https://unity.com/es/how-to/beginner/using-blender-and-maya-unity>
- Vegas, E. (6 de Enero de 2019). *Emiliusvgs.com*. Recuperado el 6 de Septiembre de 2020, de Emiliusvgs.com: <http://emiliusvgs.com/tutorial-basico-realidad-aumentada-unity/>
- Velásquez, D., Collazos, V., & Mines, J. (2017). A Low-Cost Hardware-In-The-Loop Real Time Simulation of Control Systems. *National University of Engineering, Lima, Peru*, 4.
- Villajulca, J. C. (21 de Noviembre de 2020). *Instrumentación y Control*. Recuperado el 21 de Noviembre de 2020, de Instrumentación y Control: <https://instrumentacionycontrol.net/procesos-auto-regulatorios-self-regulating-solo-proporcional-integral/>
- Worley, C. W., & Franks, R. G. (1996). Quantitative Analysis of Cascade Control. *ENGINEERING, DESIGN, AND PROCESS DEVELOPMENT*, 1074 - 1079.
- Zhao, L., & Tian, H. (2018). Hardware-in-the-loop Simulation System for Process Control. *2018 IEEE International Conference on Mechatronics and Automation (ICMA)*, 5.
- Zucconi, A. (7 de Octubre de 2015). *How to integrate Arduino with Unity*. Recuperado el 30 de Octubre de 2020, de <https://www.alanzucconi.com/2015/10/07/how-to-integrate-arduino-with-unity/>