

SEP

TECNOLÓGICO NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE TIJUANA

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA



INFORME TÉCNICO DE RESIDENCIA PROFESIONAL

“SENSOR VIRTUAL PARA LA ESTIMACIÓN DE
CONCENTRACIÓN DE ETANOL EN FERMENTACIÓN
CONTINUA”

PRESENTA:

FABIÁN GONZÁLEZ MORALES

18211551

BAJO LA ASESORÍA:

INTERNA: DR. RAMÓN RAMÍREZ VILLALOBOS

EXTERNA: DR. PAUL ANTONIO VALLE TRUJILLO

TIJUANA, BAJA CALIFORNIA, MÉXICO

ENERO DEL 2025

Agradecimientos

Tomaré esta oportunidad para expresar mi agradecimiento, en la medida que corresponda, a todos aquellos que, de una manera u otra, han sido partícipes de este hito personal. El proceso me ha costado mucho más de lo que anticipé en términos mentales, emocionales y físicos, y su finalización me habría resultado imposible de no haber estado parado sobre los hombros de los gigantes que apostaron por mí.

A mi familia, particularmente a mi núcleo, pero sin menospreciar la atención y el cuidado de mi familia extendida. Agradezco el apoyo incondicional y la tolerancia infinita que mis padres, Rogelio y Nancy, me brindaron aun cuando no entendían mi toma de decisiones o mi metodología de vida. Si en la vida logro consolidarme como un adulto funcional y honorable, será gracias a sus sacrificios y consejos, por ello siempre estaré en deuda con ellos. A mis hermanos, Jessica y Jonathan, quienes me aman y quieren lo mejor para mí.

A mis amistades, incluyendo aquellos individuos con los que no conviví, pero me marcaron más allá del tiempo compartido. Agradezco mucho haber compartido momentos inolvidables y experiencias de fraternidad con Bruno, mi brother con quien encontré un apoyo mutuo hacia el final del camino. A Jocelyne, que me dio la confianza de seguir intentando sacar adelante la carrera cuando más lejos se sentía. A Alejandra, con quien tuve una cómplice en crimen y un desahogo emocional. A Ruth, que desde lejos me dedicó tiempo y apoyo cuando más sentía necesitar de él. A Emmanuel, Manuel, Paola, Sidney, Omar, Karla y Ricardo, con quienes podía encontrar un fuerte compañerismo y risas por doquier.

A mis catedráticos y las personas responsables de mi formación profesional. Para Corina Plata y Carlos Vera, el epítome de docencia y ejemplos a seguir. A Ramon que me concedió la oportunidad de desarrollar su proyecto como residencia profesional. Al profesor Preciado, cuyo aporte profesional y deseo de enseñar me inspiró en desenvolverme en el área industrial. Finalmente, a los profesores que me hicieron la vida imposible: Sotelo, Valle, Gamboa y Z. Aprendí mucho, gracias.

Resumen

Se realiza la implementación de un sensor virtual para estimar la concentración de etanol en procesos de fermentación continua, empleando como modelo la levadura *Kluyveromyces marxianus*. Este microorganismo, utilizado ampliamente en la industria vinícola, se vuelve de gran utilidad debido a su capacidad para fermentar azúcares simples y actuar como agente de biocontrol contra microorganismos deteriorantes. Sin embargo, los métodos tradicionales de medición directa de etanol presentan una latencia significativa, lo que dificulta realizar ajustes oportunos y mantener el control del proceso.

Con el objetivo de superar estas limitaciones, se emplea un modelo mecanicista que describe la interacción dinámica entre la biomasa, la glucosa y el etanol durante la fermentación continua de *Kluyveromyces marxianus*. Este sensor virtual permite generar estimaciones en tiempo real de las concentraciones de estas variables, utilizando únicamente valores fácilmente medibles del proceso, mediante el uso de dos programas en microcontroladores Arduino: uno para simular el comportamiento del sistema y otro para estimar las concentraciones mediante el sensor virtual.

Los resultados de las simulaciones y estimaciones se evalúan utilizando métricas cuantitativas y análisis gráficos que demuestran su grado de precisión y la convergencia del sensor virtual hacia los valores del sistema modelado. La implementación del sensor virtual hace posible el uso de una herramienta eficiente para la monitorización en tiempo real de la concentración de etanol, permitiendo mejorar la calidad del producto, reducir la variación del proceso y optimizar la eficiencia operativa en la producción de vino.

Índice de contenido

Agradecimientos.....	ii
Resumen	iii
Índice de contenido	iv
Índice de figuras	vi
CAPÍTULO 1. INTRODUCCIÓN.....	1
1.1 Descripción de la institución o empresa.....	2
1.2 Objetivos.....	3
1.2.1 Objetivo general	3
1.2.2 Objetivos específicos	3
1.3 Justificación	3
CAPÍTULO 2. MARCO TEÓRICO.....	4
2.1 Fermentación continua en la industria	4
2.2 Microcontroladores	6
2.2.1 Arduino Uno	6
2.2.2 Arduino Mega	7
2.3 Sensores virtuales	7
2.3.1 Método de Euler.....	7
2.3.2 Método de Runge-Kutta de cuarto orden	8
2.4 Modelado mecanicista de sistemas biológicos	9
2.5 Representación de datos y métodos de evaluación	11
2.5.1 Diagramas de error	12
2.5.2 Histogramas	12
2.5.3 Diagramas de caja	13
2.5.4 Error medio absoluto	13
2.5.5 Raíz del error cuadrático medio	14
2.5.6 Coeficiente de correlación.....	14
2.5.7 Coeficiente de determinación	14
2.5.8 Coeficiente de incertidumbre.....	14
2.6 Impacto industrial.....	15
CAPÍTULO 3. DESARROLLO	16
3.1 Implementación del modelo mecanicista	17
3.2 Implementación del sensor virtual	20

3.3 Evaluación de la convergencia del sensor virtual	25
3.3.1 Métodos cualitativos.....	26
3.3.2 Métodos cuantitativos.....	26
3.4 Resultados.....	27
3.4.1 Simulación del modelo mecanicista	27
3.4.2 Estimación del sensor virtual.....	28
3.4.3 Evaluación de la convergencia.....	28
CAPÍTULO 4. CONCLUSIONES.....	34
4.1 Conclusiones del proyecto.....	34
4.2 Trabajo futuro	35
4.3 Competencias desarrolladas	35
REFERENCIAS.....	37
ANEXOS	41

Índice de figuras

Figura 2.1 <i>Reacción química de la fermentación alcohólica</i>	4
Figura 3.1 <i>Diagrama de conexiones físicas entre Arduino TX y Arduino RX</i>	25
Figura 3.2 <i>Series de tiempo de las variables del sistema y del sensor virtual</i>	29
Figura 3.3 <i>Relación entre variables del sistema y estimaciones del sensor virtual</i> ...	30
Figura 3.4 <i>Histogramas de las variables del sistema y del sensor virtual</i>	31
Figura 3.5 <i>Diagramas de caja de las variables del sistema y del sensor virtual</i>	32

CAPÍTULO 1. INTRODUCCIÓN

El control de variables críticas durante el proceso de fermentación continua es esencial para garantizar la calidad y consistencia del producto en la industria vinícola. Entre estas variables, la concentración de etanol tiene un efecto predominante debido a su impacto en las características del vino y, por ende, en la aprobación del catador. Sin embargo, los métodos convencionales para medir directamente el etanol presentan tiempos de respuesta prolongados que limitan la capacidad de realizar ajustes en tiempo real.

En respuesta a este tipo de limitaciones, surgen los sensores virtuales como una solución, ofreciendo la posibilidad de estimar variables difíciles de medir utilizando datos fácilmente accesibles. En el caso de la fermentación con *Kluyveromyces marxianus* (*K. marxianus*), un modelo mecanicista permite describir la interacción dinámica entre la biomasa, la glucosa y el etanol, proporcionando una base sólida para el desarrollo de un sensor virtual.

A continuación, se describe el proceso de implementación y evaluación de un sensor virtual para estimar la concentración de etanol en tiempo real. Se abordan los fundamentos del modelo mecanicista utilizado, los detalles de su implementación en microcontroladores Arduino y el análisis de los datos obtenidos para validar la precisión del sensor virtual. Además, se presentan las técnicas empleadas para evaluar la convergencia de las estimaciones y se discuten los resultados obtenidos en función de su aplicabilidad en la industria vinícola.

Planteamiento del problema

Durante la producción vinícola, el monitoreo preciso de variables clave en el proceso de fermentación continua es crucial para garantizar la calidad del producto final. Se emplea extensamente la levadura *Kluyveromyces marxianus* debido a su capacidad de fermentar azúcares simples y actuar como agente de biocontrol contra microorganismos deteriorantes. Sin embargo, la medición de la concentración del

etanol sintetizado se encuentra con una barrera significativa debido a la latencia de los métodos disponibles, cuyos resultados pueden tardar hasta cuatro días. Esta demora imposibilita realizar ajustes oportunos al sistema, aumentando la variación del proceso y comprometiendo la calidad del producto.

La falta de monitoreo continuo no solo dificulta el control de las condiciones del sistema, sino que también incrementa los costos operativos asociados al desperdicio de recursos y a la pérdida de eficiencia en la producción. Este problema resulta especialmente crítico en la industria vinícola, donde la capacidad de predecir y ajustar la dinámica del sistema fermentativo es fundamental para mantener la consistencia y competitividad del producto.

En función de esta problemática, una solución viable consiste en implementar un sensor virtual basado en el modelo mecanicista para *K. marxianus* mediante el cual se permita generar estimaciones de la concentración de etanol en tiempo real a partir de las variables más fácilmente medibles del proceso, reduciendo así la latencia y facilitando un control más efectivo del proceso fermentativo.

1.1 Descripción de la institución o empresa

En el Instituto Tecnológico de Tijuana El Instituto Tecnológico de Tijuana (ITT) cuenta con diversos laboratorios y proyectos multidisciplinarios en la búsqueda constante de innovación y mejora en el ámbito tecnológico. Particularmente en la carrera de Ingeniería Biomédica, el ITT promueve el realizar investigación científica y desarrollo tecnológico al igual que integrar soluciones informáticas para lograr avances en el área biomédica, como es el caso para el proyecto presente.

1.2 Objetivos

1.2.1 Objetivo general

Desarrollar un sensor virtual para la solución del problema de estimación de la producción de etanol en fermentación continua.

1.2.2 Objetivos específicos

1. Desarrollar un algoritmo que permita la implementación del modelo mecanicista de la producción de etanol en un microcontrolador.
2. Desarrollar un algoritmo que permita la implementación del sensor virtual en un microcontrolador.
3. Evaluar la convergencia del sensor virtual mediante la comparación del error entre las soluciones proporcionadas por el sensor virtual y las obtenidas a partir del modelo matemático y/o datos experimentales.

1.3 Justificación

En el proceso de fermentación continua, es fundamental contar con herramientas que permitan monitorear y controlar de manera efectiva variables críticas que pueden ser difíciles de medir directamente. Mediante estas se pueden detectar posibles alteraciones o anomalías en el proceso de fermentación y permitir realizar ajustes inmediatos en función de la dinámica de las condiciones del sistema, lo cual facilita la toma de decisiones para corregir las mismas y mejorar la eficiencia.

Considerando esta necesidad, el desarrollo de un sensor virtual permite monitorear de manera continua la concentración de etanol sin la latencia que presenta un sensor físico. Asimismo, produce una mejora en la precisión y calidad del proceso fermentativo al igual que la reducción del uso de recursos y de los gastos operativos asociados. Este aspecto se vuelve de suma importancia en el ámbito de la producción de vino a escala industrial, donde la eficacia del proceso presenta un efecto directamente proporcional a la rentabilidad.

CAPÍTULO 2. MARCO TEÓRICO

El presente capítulo establece las bases teóricas necesarias para comprender la implementación y evaluación de un sensor virtual diseñado para la estimación de la concentración de etanol en fermentación continua. Se inicia con una revisión breve de los procesos de fermentación, haciendo hincapié en la importancia de *K. marxianus* como agente fermentativo en la industria vinícola. Posteriormente, se introduce el concepto de sensores virtuales y su papel en la estimación de variables críticas mediante métodos numéricos, como Euler y Runge-Kutta de cuarto orden. Asimismo, se aborda el modelado mecanicista de sistemas biológicos para fundamentar su capacidad para describir dinámicas complejas en procesos fermentativos. Finalmente, se analizan las herramientas de representación y evaluación de datos, junto con su relevancia industrial para contextualizar la contribución del proyecto en el ámbito del control de la calidad y la reducción de costos operativos.

2.1 Fermentación continua en la industria

La fermentación es un proceso bioquímico que se lleva a cabo en condiciones anaeróbicas mediante el cual ciertos microorganismos, como ciertas levaduras, transforman azúcares (usualmente monosacáridos como glucosa o fructosa) en etanol y dióxido de carbono [1], tal como se ilustra en la Figura 2.1. Es por eso que este proceso es esencial en la producción de alcoholes y bebidas fermentadas, donde se busca un control estricto sobre las condiciones para garantizar un producto de calidad. La reacción química de la fermentación alcohólica mediante levaduras se escribe de la siguiente manera.

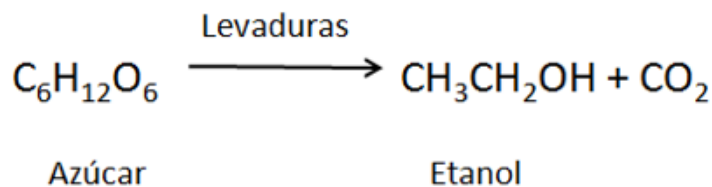


Figura 2.1 Reacción química de la fermentación alcohólica

La elección del agente fermentativo depende de diversos factores como la eficiencia en la conversión de sustratos, tolerancia al etanol, rapidez en el crecimiento y capacidad para mantener un ambiente controlado contra microorganismos contaminantes [2].

En el caso de la fermentación continua, el proceso opera mediante un flujo constante de medio de cultivo a través del reactor, manteniendo un volumen constante en aplicaciones industriales [3]. A diferencia de otros métodos, la fermentación continua no emplea dispositivos que impidan que la biomasa permanezca en el recipiente de cultivo. Este sistema puede ser controlado de distintas maneras dependiendo del parámetro que se regula para mantenerlo constante: limitación de nutrientes (quimiostato), pH constante (pH-auxostato), densidad óptica constante (turbidostato) o sustrato constante (nutristato) [3].

El proceso comienza como una fermentación por lotes y se realiza la transición hacia un sistema continuo al alcanzar la fase de crecimiento exponencial o al ser limitado por el sustrato. Su adopción industrial ha sido limitada, a pesar de su capacidad de mantener un flujo constante de producción, debido al riesgo de contaminación, deriva genética en el cultivo y complejidades en el control del proceso [4].

Entre las levaduras más ampliamente utilizadas en la industria se encuentra *K. marxianus*, una levadura termotolerante conocida por su capacidad para fermentar glucosa y fructosa, capaz de producir etanol de manera eficiente incluso bajo condiciones adversas. Además de su uso como agente de fermentación, *K. marxianus* es de gran utilidad debido a su capacidad de biocontrol, lo cual permite evitar la proliferación de microorganismos deteriorantes sin necesidad de conservantes químicos [5]. Su importancia en la industria vinícola se basa en estas cualidades, en su capacidad de mejorar la calidad del producto final y en la reducción del impacto ambiental mediante un proceso más sostenible [6].

2.2 Microcontroladores

Los microcontroladores son circuitos integrados diseñados para ejecutar tareas específicas en sistemas embebidos. A diferencia de los microprocesadores, los microcontroladores están optimizados para operar con recursos limitados, integrando en un solo chip la unidad de procesamiento central, unidades de memoria, puertos de entrada y salida, y periféricos [7]. Tienen un uso extenso en electrodomésticos, sistemas médicos e industriales, entre otros, debido a su bajo costo, tamaño reducido y capacidad para operar de manera autónoma. Sin embargo, presentan limitaciones en términos de capacidad de procesamiento, memoria y velocidad, lo que los hace menos adecuados para tareas que requieren cálculos complejos o manejar grandes volúmenes de datos [7].

Entre las plataformas de microcontroladores más empleadas se encuentra Arduino, un sistema que combina hardware de código abierto con un entorno de desarrollo integrado (IDE) basado en un lenguaje de programación derivado de C y C++ [8]. Arduino ha adquirido una gran popularidad gracias a su facilidad de uso, accesibilidad y a una amplia comunidad que aporta ideas para su implementación en proyectos educativos, experimentales e industriales. Su diseño modular permite a usuarios integrar sensores, interfaces de comunicación y dispositivos periféricos con facilidad, consolidándolo como una herramienta ideal para el desarrollo de prototipos rápidos y aplicaciones de bajo costo [9].

2.2.1 Arduino Uno

El Arduino Uno es uno de los modelos más reconocidos de la plataforma Arduino, diseñado para proyectos educativos y de prototipado rápido. Este microcontrolador utiliza el chip ATmega328P, opera a una frecuencia de reloj de 16 MHz y dispone de 32 KB de memoria flash, suficiente para ejecutar programas sencillos [10]. Cuenta con 14 pines para entradas y salidas digitales al igual que 6 entradas analógicas, pero presenta limitaciones en términos de capacidad de memoria y puertos disponibles, lo

que lo hace menos ideal para aplicaciones que requieren manejar múltiples periféricos o procesos más complejos [10].

2.2.2 Arduino Mega

El Arduino Mega está diseñado para aplicaciones más avanzadas que requieren un mayor número de entradas, salidas y capacidad de memoria. Equipado con el microcontrolador ATmega2560, este modelo incluye 54 pines digitales y 16 entradas analógicas [11]. Su memoria flash de 256 KB y frecuencia de reloj de 16 MHz permiten manejar programas más grandes y realizar tareas más complejas en comparación con el Arduino Uno [11]. Debido a su capacidad para integrar múltiples sensores, actuadores y módulos de comunicación es ampliamente utilizado en proyectos que involucran sistemas multicomponentes como robots complejos o sistemas de monitoreo industrial.

2.3 Sensores virtuales

Un sensor virtual, también conocido como observador, es una herramienta computacional que, con base en modelos matemáticos y datos medibles, permite estimar variables difíciles o imprácticas de medir directamente. Este método se utiliza ampliamente en procesos industriales debido a su capacidad para reducir costos operativos y mejorar la eficiencia del monitoreo [12].

La implementación de sensores virtuales frecuentemente requiere resolver sistemas de ecuaciones diferenciales ordinarias que describen la dinámica de los sistemas a estimar. Para ello, se emplean métodos de resolución numérica, entre los que destacan el método de Euler y el método de Runge-Kutta de cuarto orden, debido a su eficacia y simplicidad computacional.

2.3.1 Método de Euler

El método de Euler es uno de los métodos más básicos para resolver ecuaciones diferenciales de forma iterativa. Este método utiliza una aproximación lineal para

estimar el valor futuro de una variable en función de su pendiente actual [13]. La fórmula general del método de Euler para resolver ecuaciones diferenciales ordinarias (EDOs) de la forma:

$$\frac{dy}{dt} = f(t, y) \quad y(t_0) = y_0 \quad (1)$$

es:

$$y_{n+1} = y_n + h f_n \quad (2)$$

Donde $f(t, y)$ es una función conocida, las condiciones iniciales son conocidas, y_n es el valor actual de la variable, h es el tamaño del paso de integración, y f_n es la derivada de la variable en el instante t_n . Resulta simple de implementar y computacionalmente eficiente sin embargo presenta limitaciones significativas en precisión y estabilidad cuando se utiliza en sistemas no lineales o dinámicas rápidas. Por esta razón, aunque el método de Euler es útil como punto de partida, no es la opción preferida para sistemas complejos [13].

2.3.2 Método de Runge-Kutta de cuarto orden

El método de Runge-Kutta de cuarto orden (RK4) mejora considerablemente las limitaciones del método de Euler, que emplea únicamente la pendiente inicial del intervalo para estimar el próximo valor, al calcular cuatro pendientes intermedias dentro de cada paso de integración. Esto permite una estimación más precisa del valor siguiente de las variables. El método se basa en las ecuaciones, mostradas a continuación, para calcular el valor de las pendientes y el valor siguiente de la variable [14]. De acuerdo a lo expresado en las siguientes ecuaciones, se le otorga un mayor peso a k_2 y k_3 , las pendientes intermedias, indicando que el método considera que representan mejor el comportamiento promedio del sistema dentro del intervalo.

$$\left. \begin{aligned} y_{n+1} &= y_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4), \\ t_{n+1} &= t_n + h \\ k_1 &= f(t_n, y_n), \\ k_2 &= f\left(t_n + \frac{h}{2}, y_n + h\frac{k_1}{2}\right), \\ k_3 &= f\left(t_n + \frac{h}{2}, y_n + h\frac{k_2}{2}\right), \\ k_4 &= f(t_n + h, y_n + hk_3). \end{aligned} \right\} \quad (3)$$

Mientras que el método de Euler genera un error global acumulado proporcional al tamaño del paso de integración, RK4 lo reduce a una cuarta potencia del mismo, de aquí el motivo del nombre “de cuarto orden”. Esto significa que incluso con pasos de integración relativamente grandes, el método conserva un nivel alto de precisión. Asimismo, la implementación de RK4 ayuda a evitar oscilaciones no deseadas en las soluciones numéricas, especialmente en sistemas con alta sensibilidad a las condiciones iniciales [15].

2.4 Modelado mecanicista de sistemas biológicos

El modelado mecanicista en sistemas biológicos tiene su origen en la necesidad de comprender y predecir con precisión el comportamiento de procesos dinámicos complejos, como los que ocurren en la fermentación. A diferencia de los modelos empíricos, que se basan en datos experimentales para establecer relaciones estadísticas entre variables, los modelos mecanicistas parten de las relaciones fundamentales del sistema como las leyes de conservación de masa y energía, las reacciones bioquímicas involucradas y las cinéticas de crecimiento biológico [16].

Esto permite no solo describir matemáticamente las interacciones entre las variables clave del sistema en condiciones actuales, sino que también permiten predecir, hasta

cierto grado, cómo responderá el sistema ante cambios en las condiciones operativas, siendo fundamentales para diseñar estrategias de control, optimización y escalado en la producción industrial.

El modelado de un sistema de fermentación se ha realizado previamente para *Kluyveromyces marxianus* donde, como resultado, se producen tres EDOs de primer orden, acopladas no lineales, que consideran las tasas de crecimiento de biomasa, consumo de glucosa y producción de etanol con diez parámetros ajustados a partir de datos experimentales de 17 cepas de *K. marxianus* mediante un algoritmo computacional en MATLAB [17]. Estas tasas se han establecido como indica el conjunto de ecuaciones diferenciales ordinarias expresadas a continuación, donde las concentraciones de biomasa, glucosa y etanol son representadas por las variables x , y y z , respectivamente.

$$\dot{x} = \frac{\rho_1 xy}{\rho_2 + y} - \rho_3 xz - \rho_4 x, \quad (4)$$

$$\dot{y} = -\rho_5 xy - \rho_6 yz - \rho_7 y, \quad (5)$$

$$\dot{z} = \rho_8 xz + \rho_9 yz - \rho_{10} z, \quad (6)$$

Los parámetros (ρ_1 a ρ_{10}) y las variables empleadas en la Tabla 2.1 se definen en función de propiedades específicas de este modelo y sirven como base en un trabajo de residencia profesional donde se desarrolló una metodología de implementación de un sensor virtual de orden completo, tipo Luenberger, para la estimación de la producción de etanol en fermentación continua mediante una tarjeta electrónica [18].

Tabla 2.1 Descripción, valores y unidades de las variables y parámetros para el modelo mecanicista de *K. marxianus*.

Parámetros	Descripción	Valores	Unidades
$x(t)$	Concentración de biomasa	-	g/L
$y(t)$	Concentración de glucosa	-	g/L
$z(t)$	Concentración de etanol	-	g/L
ρ_1	Tasa máxima de crecimiento de biomasa	$289,385,381,419 \times 10^{-3}$	h^{-1}
ρ_2	Constante de afinidad con el sustrato	2.281	g/L
ρ_3	Tasa de inhibición del crecimiento de biomasa debido a la acumulación de productos	$1,066 \times 10^{-3}$	L/(g x h)
ρ_4	Tasa de muerte de biomasa	$7,25 \times 10^{-3}$	h^{-1}
ρ_5	Tasa de consumo para el crecimiento de biomasa	$56,893 \times 10^{-3}$	L/(g x h)
ρ_6	Tasa de consumo para la producción de etanol	$71,842 \times 10^{-3}$	L/(g x h)
ρ_7	Tasa de descomposición espontánea de glucosa	$824,233 \times 10^{-9}$	h^{-1}
ρ_8	Producción de etanol asociada con la tasa de crecimiento de biomasa	$[19,088,49,816] \times 10^{-3}$	L/(g x h)
ρ_9	Glucosa convertida en etanol	$[46,352,70,349] \times 10^{-3}$	L/(g x h)
ρ_{10}	Tasa de degradación de etanol	$149,899 \times 10^{-3}$	h^{-1}

2.5 Representación de datos y métodos de evaluación

El análisis y evaluación de datos son esenciales para validar modelos predictivos en aplicaciones industriales, ya que permiten medir la precisión y confiabilidad de las

estimaciones comparando valores reales y estimados. Los diagramas de error, histogramas y diagramas de caja facilitan la identificación de patrones, sesgos y errores sistemáticos, mientras que métricas como el error medio absoluto, la raíz del error cuadrático medio y los coeficientes de correlación, determinación e incertidumbre proporcionan una medida numérica del desempeño del modelo. La integración de estas herramientas permite un análisis más profundo y efectivo, asegurando que los sensores virtuales cumplan con los estándares de calidad y desempeño exigidos en contextos industriales.

2.5.1 Diagramas de error

Los diagramas de error son herramientas gráficas utilizadas para comparar los valores reales de una variable con sus estimaciones, mostrando directamente las discrepancias entre ambas [19]. Este tipo de gráfico es esencial para evaluar la precisión de los sensores virtuales, ya que permite identificar tendencias específicas en los errores de predicción.

Los diagramas de error no solo muestran la magnitud de las desviaciones, sino que también revelan si estas son sistemáticas o aleatorias, lo cual es crítico para diagnosticar y ajustar modelos. Un sensor virtual eficaz debería mostrar un error cercano a cero y una distribución de puntos ajustada a una línea diagonal a 45° en las gráficas comparativas. Estas herramientas son ampliamente utilizadas en aplicaciones industriales para evaluar la viabilidad de modelos predictivos [20].

2.5.2 Histogramas

Los histogramas proporcionan una representación visual de la distribución de los datos, facilitando el análisis de las diferencias entre dos conjuntos de datos [21]. En el contexto de los sensores virtuales, estas gráficas permiten detectar sesgos en las estimaciones y evaluar si las distribuciones de los datos estimados son consistentes con las observadas.

Un histograma ideal mostraría una superposición casi perfecta entre las distribuciones de los valores reales y estimados, indicando que el modelo predice correctamente la variabilidad del sistema. Asimismo, los histogramas pueden revelar la presencia de errores sistemáticos como desviaciones hacia un rango específico de valores. Esta herramienta es particularmente valiosa en el análisis de procesos industriales, ya que permite verificar si el sistema cumple con los estándares de calidad empleados [22].

2.5.3 Diagramas de caja

Los diagramas de caja son representaciones gráficas que resumen las distribuciones de datos mediante sus cuartiles, resaltando aquellos valores atípicos y la variabilidad del conjunto. Los diagramas de caja permiten evaluar si las distribuciones de las estimaciones coinciden con las de los valores reales, proporcionando información visual sobre la precisión y estabilidad del modelo [23].

Estos diagramas son esenciales para identificar posibles inconsistencias o sesgos en las predicciones. En aplicaciones industriales, se espera que los diagramas de caja de las variables estimadas y reales tengan rangos y medianas similares, con pocos o ningún valor atípico significativo. Esto asegura que el sensor virtual no solo sea preciso, sino también confiable bajo diferentes condiciones operativas [22].

2.5.4 Error medio absoluto

El error medio absoluto (MAE) es una métrica que mide la magnitud promedio de los errores absolutos entre las estimaciones y los valores reales [24]. Es una medida intuitiva y fácil de interpretar, ya que utiliza las mismas unidades que las variables analizadas. Para su uso en sensores virtuales, un MAE bajo indica que las estimaciones son consistentemente cercanas a los valores reales. El MAE es particularmente útil en aplicaciones donde los errores grandes y pequeños deben ser tratados con igual importancia [22].

2.5.5 Raíz del error cuadrático medio

La raíz del error cuadrático medio (RMSE) evalúa la dispersión de los errores, dando un peso adicional a los errores grandes debido a la naturaleza cuadrática de su cálculo. Esta métrica es ideal para sistemas donde los valores atípicos pueden tener un impacto significativo en la precisión global [24]. En aplicaciones industriales, un RMSE bajo es indicativo de un modelo que maneja bien tanto las predicciones promedio como los valores extremos [22].

2.5.6 Coeficiente de correlación

El coeficiente de correlación (R) mide la fuerza y dirección de la relación lineal entre dos conjuntos de datos. Un valor de R cercano a uno indica que las estimaciones tienen una alta correlación con los valores reales, lo que refleja una buena calidad en las predicciones [25]. Esta métrica es ampliamente utilizada en la industria para validar modelos predictivos antes de su implementación. Valores bajos de R sugieren la necesidad de recalibrar el modelo o mejorar los datos de entrada para asegurar una predicción confiable [25].

2.5.7 Coeficiente de determinación

El coeficiente de determinación (R^2) es una métrica crucial para evaluar la calidad general del modelo predictivo que mide la proporción de la variabilidad de los datos reales que es explicada por el modelo. Un valor de R^2 cercano a uno indica que el modelo captura casi toda la variabilidad de los datos reales, mientras que valores bajos señalan deficiencias en el ajuste del modelo [26]. En la industria, un R^2 alto asegura que el modelo predictivo es adecuado para su uso en entornos operativos, minimizando riesgos asociados con estimaciones inexactas [26].

2.5.8 Coeficiente de incertidumbre

El coeficiente de incertidumbre (U de Theil) compara el desempeño del modelo predictivo con un método ingenuo, como usar valores promedio para la estimación. Un

valor cercano a cero indica que el modelo es significativamente mejor que un método ingenuo, mientras que valores más altos sugieren la necesidad de corrección [27].

Esta métrica es valiosa en contextos industriales, ya que evalúa no solo la precisión del modelo, sino también su utilidad práctica frente a alternativas más simples y menos costosas.

2.6 Impacto industrial

La determinación de la concentración de etanol en procesos de fermentación es un aspecto crítico para garantizar la calidad del producto final, particularmente en la industria vinícola. Actualmente, existen diversas opciones disponibles para medir este parámetro como los cromatógrafos de gases y líquidos, espectrofotómetros, y densímetros [28]. Sin embargo, estas técnicas presentan barreras limitantes, entre ellas: tiempos prolongados de análisis, costos elevados de equipo y mantenimiento, y la necesidad de personal altamente capacitado para operar los sistemas. Los métodos basados en cromatografía pueden tardar varios días en generar resultados, lo que impide la implementación de correcciones oportunas durante el proceso fermentativo, generando potenciales pérdidas en eficiencia y calidad del producto.

En el ámbito industrial, el monitoreo continuo y preciso de variables clave como la concentración de etanol es esencial para cumplir con los estándares de control de calidad. La capacidad de medir y analizar estas variables en tiempo real permite identificar variaciones no deseadas en el proceso, reduciendo la variabilidad del producto y garantizando la conformidad con las especificaciones establecidas [29]. Un control más efectivo del proceso fermentativo mediante sensores virtuales también se traduce en menores desperdicios y una mejor utilización de los insumos, como los azúcares y nutrientes, lo que aumenta la rentabilidad del proceso. Además, la capacidad de detectar y corregir anomalías de manera inmediata minimiza la probabilidad de producir lotes fuera de especificación, lo que es crucial para mantener la confianza del cliente y la competitividad en el mercado global.

CAPÍTULO 3. DESARROLLO

El sistema implementado consiste en dos programas, cada uno en un Arduino distinto, que simulan y estiman el comportamiento de la fermentación continua de *K. marxianus* a través del modelo mecanicista desarrollado previamente [17]. Estos programas, denominados TX (transmisor) y RX (receptor), ejecutan su código simultáneamente y de manera unidireccional, permitiendo así monitorear y analizar variables críticas como biomasa, glucosa y etanol.

El programa TX tiene la función principal de modelar el comportamiento del sistema real, generando y transmitiendo datos simulados de la variable de interés $x(t)$, que representa la biomasa. Por otro lado, RX opera como un sensor virtual que recibe los valores de biomasa enviados desde TX y, únicamente a partir de estos valores, realiza estimaciones de las variables de estado: la biomasa, glucosa y el etanol.

El sistema se fundamenta en un proceso iterativo derivado del método de Euler, el cual utiliza el método de Runge-Kutta de cuarto orden (RK4) para mejorar la precisión y estabilidad en la estimación de las soluciones del sistema [15]. Las estimaciones se realizan mediante la implementación del mismo modelo mecanicista, pero realizándose una suma ponderada a cada ecuación diferencial con el producto del error en la biomasa y su ganancia respectiva, la cual optimiza la convergencia del sistema.

Ambos programas emplean comunicación serial a 57600 baudios para el intercambio de información. Los valores transmitidos se registran y almacenan en archivos CSV mediante software de monitoreo serial. Estos archivos contienen series temporales de las variables simuladas y estimadas, permitiendo su posterior procesamiento y análisis en herramientas como Excel o MATLAB.

A continuación, se presenta el desarrollo de cada uno de los programas implementados, detallando su estructura, funciones y la metodología utilizada para la simulación y estimación del proceso de fermentación continua.

3.1 Implementación del modelo mecanicista

El código TX tiene como propósito principal simular el sistema modelando la dinámica de concentración de biomasa, glucosa y etanol en un proceso de fermentación continua, con base en el modelo mecanicista que define las tasas de cambio de estas variables, para posteriormente transmitir los valores calculados de estas variables a través del puerto serial para ser registrados en tiempo real.

Las primeras líneas del código establecen las constantes, variables iniciales y parámetros del sistema. Estas incluyen el intervalo de tiempo para la integración numérica (dt), las condiciones iniciales de las variables de estado (x , y , y z correspondientes a biomasa, glucosa y etanol, respectivamente) y un conjunto de constantes ($p1$ a $p10$). Estas constantes son esenciales para caracterizar el comportamiento del sistema y se obtuvieron a partir del modelo matemático desarrollado previamente.

```
1 // Valores de tiempo
2 const float dt = 1E-3; // Incremento de tiempo
3 double ta = 0; // Tiempo acumulado

4 // Condiciones iniciales del sistema
5 float x = 0.5; // Biomasa inicial (g/L)
6 float y = 21.0; // Glucosa inicial (g/L)
7 float z = 0.10; // Etanol inicial (g/L)

8 // Constantes del modelo mecanicista
9 const float p1 = 312.378E-3;
10 const float p2 = 2.2814;
11 const float p3 = 1.0658E-3;
12 const float p4 = 7.2751E-3;
13 const float p5 = 56.8928E-3;
14 const float p6 = 71.8424E-3;
15 const float p7 = 824.2332E-9;
16 const float p8 = 31.563E-3;
17 const float p9 = 60.706E-3;
18 const float p10 = 149.899E-3;
```

A continuación, se inicializa la comunicación serial a una velocidad de 57600 baudios y se incluye un retardo de un segundo para garantizar que el medio de recolección de datos esté listo antes de iniciar la transmisión. Para ello se conecta el Arduino

transmisor a CoolTerm, una aplicación gratuita empleada como terminal de puerto serial, después de subir el código y antes que inicie su transmisión.

```
1 // Configuración inicial del sistema
2 void setup() {
3   Serial.begin(57600); // Inicia comunicación serial a 57600 baudios
4   delay(1000);         // Pausa para establecer conexión
5 }
```

Una vez configurado el Arduino, se realiza la primera fase del bucle principal donde se ejecuta un ciclo infinito en el que se emplea un retardo de 100 milisegundos para no sobresaturar el puerto serial. El sistema comienza con las condiciones iniciales de las tres variables como los valores actuales del sistema, llama a la función *rungeKutta4* para calcular el siguiente estado del sistema, actualiza las variables de estado, las imprime en formato CSV por el puerto serial y se repite el proceso.

```
1 // Fase 1 del bucle principal del programa
2 void loop() {
3   delay(100); // Retardo de 100 ms entre iteraciones
4   // Envía tiempo y valores actuales por puerto serial en formato CSV
5   Serial.print(ta, 3);
6   Serial.print(",");
7   Serial.print(x);
8   Serial.print(",");
9   Serial.print(y);
10  Serial.print(",");
11  Serial.println(z);
12  rungeKutta4(x, y, z, ta, dt); // Llama a la función para calcular el
    siguiente estado
13 }
```

Mediante esta primera fase se logra obtener un archivo en formato CSV que representa la evolución de las variables de estado a través del tiempo, en valores discretos, formándose así una simulación del sistema de fermentación continua de *K. marxianus*, la cual se usará como referencia en la evaluación de la convergencia. Posteriormente se debe editar el apartado del bucle inicial para que el Arduino TX envíe exclusivamente los valores de la biomasa de manera que se emule el comportamiento de un sensor físico entregando valores de mediciones, como lo sería en una implementación real.


```

1 // Fase 2 del bucle principal del programa
2 void loop() {
3   delay(100);           // Retardo de 100 ms entre iteraciones
4   // Envía valor actual de biomasa por puerto serial
5   Serial.println(x);
6   rungeKutta4(x, y, z, ta, dt); // Calcular el siguiente estado
7 }

```

En el apartado final se encuentran las dos funciones empleadas para realizar la simulación del sistema. La primera función implementa el método iterativo de RK4, utilizado para resolver las ecuaciones diferenciales que describen el sistema. Primero se declaran y calculan los valores de las cuatro pendientes, para ello se emplea la segunda función del programa: *systemDynamics*. Se calculan las pendientes de las variables de estado cuatro veces para generar las doce pendientes totales y a partir de estas hacer la suma ponderada que se usa para actualizar los valores.

```

1 // Implementación del método de Runge-Kutta de cuarto orden
2 void rungeKutta4(float &x, float &y, float &z, double &ta, float dt) {
3   // Declaración de las variables para las pendientes
4   float k1x, k1y, k1z;
5   float k2x, k2y, k2z;
6   float k3x, k3y, k3z;
7   float k4x, k4y, k4z;
8   // Calcular las pendientes intermedias para cada variable del sistema
9   systemDynamics(x, y, z, k1x, k1y, k1z);
10  systemDynamics(x + 0.5 * dt * k1x, y + 0.5 * dt * k1y, z + 0.5 * dt *
    k1z, k2x, k2y, k2z);
11  systemDynamics(x + 0.5 * dt * k2x, y + 0.5 * dt * k2y, z + 0.5 * dt *
    k2z, k3x, k3y, k3z);
12  systemDynamics(x + dt * k3x, y + dt * k3y, z + dt * k3z, k4x, k4y, k4z);
13 // Actualización de estado del sistema y la suma ponderada de pendientes
14 x += (dt / 6) * (k1x + 2 * k2x + 2 * k3x + k4x);
15 y += (dt / 6) * (k1y + 2 * k2y + 2 * k3y + k4y);
16 z += (dt / 6) * (k1z + 2 * k2z + 2 * k3z + k4z);
17 ta += dt; // Incrementar el tiempo acumulado
18 }

```

Por último, en la función *systemDynamics* se establecen las ecuaciones diferenciales del sistema que modelan la interacción entre las variables y emplean los diez parámetros establecidos como constantes al inicio del programa. La función calcula las tasas de cambio (dx , dy , dz) de cada variable en función del estado actual del sistema.

```

1 // Modelo mecanicista que modela la dinámica del sistema
2 void systemDynamics(float x, float y, float z, float &dx, float &dy,
   float &dz) {
3 // Ecuaciones diferenciales del modelo
4 dx = ((p1 * x * y) / (p2 + y)) - (p3 * x * z) - (p4 * x);
5 dy = -(p5 * x * y) - (p6 * y * z) - (p7 * y);
6 dz = (p8 * x * z) + (p9 * y * z) - (p10 * z);
7 }

```

3.2 Implementación del sensor virtual

El código RX se desempeña como el sensor virtual que estima las concentraciones de biomasa, glucosa y etanol a partir de las mediciones transmitidas por el Arduino TX. Este sensor virtual se basa en la misma dinámica que el modelo de simulación, pero incorpora términos de corrección para reducir el error de estimación. Además, los valores estimados y recibidos se registran en tiempo real para su análisis posterior.

Similar al código TX, se definen las constantes, variables iniciales y parámetros necesarios para el modelo con una alteración ligera a la condición inicial de la variable de biomasa para comprobar que el sistema se puede corregir aun empezando con un error en la medida inicial.

```

1 // Valores de tiempo
2 const float dt = 1E-3; // Incremento de tiempo
3 double ta = 0; // Tiempo acumulado

4 // Estimaciones iniciales del sensor virtual
5 float xe = 1.0; // Biomasa estimada inicial (g/L)
6 float ye = 21.0; // Glucosa estimada inicial (g/L)
7 float ze = 0.10; // Etanol estimado inicial (g/L)

8 // Constantes del modelo mecanicista
9 const float p1 = 312.378E-3;
10 const float p2 = 2.2814;
11 const float p3 = 1.0658E-3;
12 const float p4 = 7.2751E-3;
13 const float p5 = 56.8928E-3;
14 const float p6 = 71.8424E-3;
15 const float p7 = 824.2332E-9;
16 const float p8 = 31.563E-3;
17 const float p9 = 60.706E-3;
18 const float p10 = 149.899E-3;

```

Asimismo, se agregan las variables para las ganancias del sensor virtual ($L1$, $L2$, $L3$), constantes fundamentales para ajustar la velocidad y precisión de la convergencia hacia los valores reales. Se declara una bandera, a manera de una variable booleana, que permita al programa identificar cuando se recibe un nuevo valor de biomasa. Por último, se declara una variable para guardar los valores de biomasa provenientes del Arduino TX.

```
1 // Ganancias del sensor virtual
2 const float L1 = 2.0;
3 const float L2 = 1.5;
4 const float L3 = 0.4;

5 bool newValueReceived = false; // Bandera que indica si se ha recibido
    un nuevo valor
6 float x = 0.0;                // Valor de biomasa recibido
```

Las funciones empleadas en este programa son casi idénticas a las funciones del transmisor con la excepción de un par de cambios. El primer cambio se encuentra en la función de Runge-Kutta de cuarto orden que se usa para calcular los valores estimados del sensor virtual. Se conserva la misma lógica e implementación de este método, pero se cambian los nombres de las variables para indicar que estas son de los valores estimados. A su vez la función toma como argumento adicional la variable del valor recibido de biomasa, esto con el propósito de pasarle el valor a la función del sensor virtual.

```
1 // Método Runge-Kutta de cuarto orden para actualizar el estado del
    sensor virtual
2 void rungeKutta4(float x, float &xe, float &ye, float &ze, double &ta,
    float dt) {
3 // Declaración de las variables para las pendientes
4 float k1xe, k1ye, k1ze;
5 float k2xe, k2ye, k2ze;
6 float k3xe, k3ye, k3ze;
7 float k4xe, k4ye, k4ze;
8 // Cálculo de pendientes para actualizar el estado del sensor virtual
9 sensorVirtual(x, xe, ye, ze, k1xe, k1ye, k1ze);
10 sensorVirtual(x, xe + 0.5 * dt * k1xe, ye + 0.5 * dt * k1ye, ze + 0.5 *
    dt * k1ze, k2xe, k2ye, k2ze);
11 sensorVirtual(x, xe + 0.5 * dt * k2xe, ye + 0.5 * dt * k2ye, ze + 0.5 *
    dt * k2ze, k3xe, k3ye, k3ze);
```

```

12 sensorVirtual(x, xe + dt * k3xe, ye + dt * k3ye, ze + dt * k3ze, k4xe,
    k4ye, k4ze);
13 // Actualización del estado del sensor virtual con la combinación
    ponderada de las pendientes
14 xe += (dt / 6) * (k1xe + 2 * k2xe + 2 * k3xe + k4xe);
15 ye += (dt / 6) * (k1ye + 2 * k2ye + 2 * k3ye + k4ye);
16 ze += (dt / 6) * (k1ze + 2 * k2ze + 2 * k3ze + k4ze);
17 ta += dt; // Incrementar el tiempo acumulado
18 }

```

El segundo cambio significativo en las funciones entre los programas se encuentra en la función *sensorVirtual*, una versión adaptada de *systemDynamics*. Es a partir de la implementación de estas ecuaciones diferenciales que el programa adquiere su carácter de sensor virtual al incorporar términos adicionales que corrigen las estimaciones.

Las nuevas ecuaciones se basan en las mismas ecuaciones del modelo mecanicista con la adición de un término al final de cada una que realiza una suma con el producto de su ganancia respectiva y el error entre la medición externa *x* y el valor estimado internamente *xe*. Estos términos de corrección garantizan que las estimaciones converjan hacia los valores reales del sistema y la magnitud de cada una de las ganancias determina la estabilidad, precisión y velocidad de convergencia en su ecuación diferencial.

```

1 // Modelo mecanicista con suma ponderada del producto de las ganancias y
    el error
2 void sensorVirtual(float x, float xe, float ye, float ze, float &dx,
    float &dye, float &dze) {
3 dx = ((p1 * xe * ye) / (p2 + ye)) - (p3 * xe * ze) - (p4 * xe) + L1 *
    (x - xe);
4 dye = -(p5 * xe * ye) - (p6 * ye * ze) - (p7 * ye) + L2 * (x - xe);
5 dze = (p8 * xe * ze) + (p9 * ye * ze) - (p10 * ze) + L3 * (x - xe);
6 }

```

Se configura la comunicación serial entre el Arduino RX y la computadora a la misma velocidad de 57600 baudios que en el programa anterior, pero se aumenta el retardo a cinco segundos para realizar las conexiones físicas entre Arduinos y se imprime un encabezado con los nombres de las columnas para organizar los datos en formato CSV.

```

1 // Configuración inicial del sistema
2 void setup() {
3   Serial.begin(57600); // Inicia comunicación serial a 57600 baudios
4   delay(5000);         // Pausa para establecer conexión
5   Serial.println("t,x,e,ye,ze"); // Encabezado de los datos que se
    enviarán por serial
6 }

```

El bucle principal del programa RX comienza verificando si hay datos disponibles en el puerto serial. Si se detectan datos, el programa los lee como una cadena de texto, la limpia de caracteres innecesarios y la convierte a un valor numérico que se almacena en la variable flotante *x*, el cual representa la medición de biomasa recibida desde el transmisor. Una vez procesado, se activa un indicador lógico que asegura que el programa proceda con los cálculos exclusivamente cuando se reciba un valor nuevo.

Cuando el indicador lógico es verdadero, el programa procede a imprimir en el puerto serial los valores actuales del tiempo *ta*, la medición recibida *x* y las estimaciones actuales del sensor virtual *xe*, *ye* y *ze*, organizados en formato CSV para su registro y posterior análisis. Posteriormente, llama a la función *rungeKutta4* para actualizar los valores estimados utilizando las ecuaciones del sensor virtual. Al final de cada ciclo, el indicador lógico se reinicia para esperar un nuevo conjunto de datos.

```

1 // Bucle principal del programa
2 void loop() {
3   if (Serial.available()) { // Verifica si hay datos disponibles en el
    puerto serial
4     String input = Serial.readStringUntil('\n'); // Lee la cadena hasta el
    salto de línea
5     input.trim(); // Elimina espacios en blanco al inicio y fin
6     if (input.length() > 0) {
7       x = input.toFloat(); // Convierte el valor recibido a flotante
        (biomasa)
8       newValueReceived = true; // Se activa la bandera indicando nuevo valor
9     }
10  }

11  if (newValueReceived) {
12    // Envía tiempo y estimaciones actuales por puerto serial en formato CSV
13    Serial.print(ta, 3);
14    Serial.print(",");
15    Serial.print(x);

```

```

16 Serial.print(",");
17 Serial.print(xe);
18 Serial.print(",");
19 Serial.print(ye);
20 Serial.print(",");
21 Serial.println(ze);
22 // Cálculo del siguiente estado del sensor virtual usando Runge-Kutta
23 rungeKutta4(x, xe, ye, ze, ta, dt);
24 newValueReceived = false; // Reinicia la bandera
25 }
26 }

```

Una vez compilados ambos programas en el IDE de Arduino, estos están listos para ser subidos a sus respectivos microcontroladores. Para ello, es necesario establecer una conexión física mediante un cable USB entre la computadora, que debe tener el IDE instalado, y el Arduino cuya salida se desea monitorear. Se pueden usar distintos tipos de Arduinos para ejecutar los programas, pero para el proyecto presente se emplea un Arduino Uno como el transmisor (TX) y un Arduino Mega como el receptor (RX).

Primero, se carga el código transmisor correspondiente a la fase 1 del bucle principal en el Arduino TX, con el objetivo de registrar la evolución en el tiempo discretizado de las tres concentraciones. Una vez cargado el código, se inicia la grabación de datos en CoolTerm, lo que permite registrar toda la información transmitida a través del puerto serial hacia la computadora. La simulación se ejecuta hasta alcanzar 100 horas del sistema, se detiene la grabación y los datos crudos se guardan en un archivo en formato CSV. Posteriormente, se carga el código transmisor con la fase 2 del bucle principal en el Arduino TX y se desconecta de la computadora, quedando el código almacenado y listo para ejecutarse automáticamente al proporcionar alimentación al microcontrolador.

A continuación, se conecta el Arduino RX a la computadora para cargar el código receptor. Gracias al retardo programado en la configuración inicial de este código, se posee un tiempo de cinco segundos para realizar la conexión e iniciar la grabación de datos mediante CoolTerm antes que se envíe la línea de encabezado con los nombres

de las columnas. Al ejecutarse la configuración inicial, el receptor queda a la espera de recibir datos para iniciar el proceso de estimación.

Acto seguido, se realiza la conexión física entre ambos Arduinos, siguiendo las indicaciones mostradas en la Figura 3.1. Se debe procurar conectar primero el pin TX del transmisor al pin RX del receptor, luego los pines GND de ambos dispositivos y, finalmente, los pines de 5V, garantizando así la captura de todos los datos desde el inicio de la operación del Arduino TX. Ambos programas se ejecutan simultáneamente hasta alcanzar las 100 horas del sistema, tras lo cual se detiene la grabación y los datos crudos se almacenan en un archivo en formato CSV.

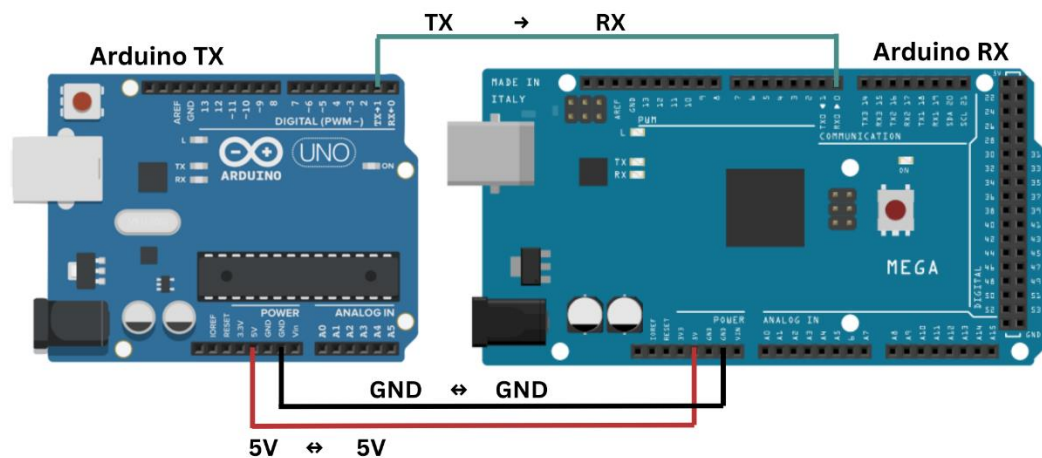


Figura 3.1 Diagrama de conexiones físicas entre Arduino TX y Arduino RX

3.3 Evaluación de la convergencia del sensor virtual

El análisis de la convergencia entre las variables estimadas por el sensor virtual y las variables simuladas del sistema se realiza al generar distintos gráficos y métricas de evaluación, los cuales permiten una descripción comparativa, con base en los datos crudos, del comportamiento del sistema implementado. Se emplean métodos cualitativos y cuantitativos que facilitan la interpretación visual del desempeño del sensor virtual mediante herramientas matemáticas, en este caso se hace uso del software de MATLAB.

3.3.1 Métodos cualitativos

Las primeras graficas generadas son las series temporales de las tres variables del sistema (biomasa, glucosa y etanol) junto con las estimaciones del sensor virtual. Estas gráficas permiten observar la evolución de las concentraciones a lo largo del tiempo, evaluando visualmente el nivel de concordancia entre ambas. Adicionalmente, se incluye una gráfica tridimensional que ilustra la trayectoria conjunta de las tres variables de estado, tanto simuladas como estimadas, en el espacio de estados para identificar el comportamiento global de ambos sistemas en un espacio multidimensional.

Posteriormente se emplean las gráficas de comparación directa entre las variables del sistema y las del sensor virtual. Para ello, se grafica cada variable estimada frente a su contraparte simulada, para visualizar de forma clara el nivel de similitud y la existencia de posibles desviaciones sistemáticas en la estimación. Asimismo, se crea una gráfica de error, mediante la cual se representa la diferencia entre las variables del sistema y las estimaciones del sensor virtual en función del tiempo tal que se pueda identificar momentos de mayor discrepancia y evaluar si los errores tienden a disminuir a medida que avanza la simulación.

De manera subsecuente, se realizan histogramas para evaluar la distribución de los datos simulados y estimados. Estas gráficas comparativas permiten visualizar las diferencias en la dispersión y frecuencias relativas entre ambas fuentes de datos. Además, se generan diagramas de caja para resaltar la amplitud de las estimaciones, su variabilidad y la presencia de posibles valores atípicos. Mediante estos recursos gráficos se hace posible la interpretación visual para identificar patrones generales en el comportamiento de los datos.

3.3.2 Métodos cuantitativos

En complemento a las gráficas, se calculan métricas cuantitativas que permiten una evaluación más precisa del desempeño del sensor virtual. Entre las métricas

empleadas se encuentran el error absoluto medio (MAE), la raíz del error cuadrático medio (RMSE), el coeficiente de determinación (R^2), coeficiente de correlación (R) y el coeficiente de incertidumbre (U de Theil). Estas métricas permiten medir la precisión y exactitud de las estimaciones, reflejando la magnitud de los errores y la relación estadística entre las variables simuladas y estimadas.

El cálculo de estas métricas se realiza a partir de las diferencias entre las series temporales del sistema y del sensor virtual, procesadas en MATLAB. Los resultados obtenidos complementan el análisis visual, ofreciendo una perspectiva más objetiva que respalda las conclusiones sobre la capacidad del sensor virtual para estimar las variables de interés con precisión.

3.4 Resultados

Se presentan ordenadamente los resultados que se obtienen al llevar a cabo los puntos de la sección anterior: la implementación del modelo mecanicista, el desarrollo del sensor virtual y la evaluación de la convergencia. Estas etapas reflejan el cumplimiento de los objetivos específicos planteados para el proyecto y alineándose a las prioridades establecidas en el proceso de producción vinícola.

3.4.1 Simulación del modelo mecanicista

El desarrollo del código TX logra implementar de manera efectiva el modelo mecanicista que describe las dinámicas de la fermentación continua de *K. marxianus* y permite generar un archivo en formato CSV que contiene los datos crudos de las concentraciones de biomasa, glucosa y etanol a lo largo de 100 horas del sistema en tiempo discreto que se transmiten a través del puerto serial para su registro y posterior análisis. El código empleado para este propósito se añade como repositorio y se encuentra anexado al final del presente documento.

3.4.2 Estimación del sensor virtual

El código RX permite implementar un sensor virtual que estima las concentraciones de biomasa, glucosa y etanol en función de los valores de biomasa transmitidos desde TX. Al igual que el programa transmisor, este código se encuentra documentado en los anexos. La ejecución de este programa genera un archivo adicional que representa el comportamiento estimado correspondiente a las mismas 100 horas del sistema en formato CSV. La integración de términos de corrección garantiza que las estimaciones converjan hacia los valores reales del sistema.

3.4.3 Evaluación de la convergencia

El procesamiento de los datos generados se realiza mediante el IDE de MATLAB, un sistema de cómputo numérico, donde se programa la lectura de los datos y la presentación de los gráficos resultantes. Se obtiene como resultado una serie de recursos gráficos y métricas de evaluación que comparan las variables estimadas y los datos del sistema simulado.

Según lo presentado en la Figura 3.2, se encuentran ligeras discrepancias en los valores iniciales de biomasa lo cual se acredita a las condiciones iniciales distintas. Las concentraciones de glucosa muestran una disminución agresiva debido al consumo de sustrato por la biomasa y ambas curvas indican seguir el mismo patrón. Se identifica la contraparte de este efecto en las concentraciones de etanol con el comportamiento esperado de un proceso fermentativo sin embargo en este caso la diferencia entre ambas curvas se hace ligeramente más discernible.

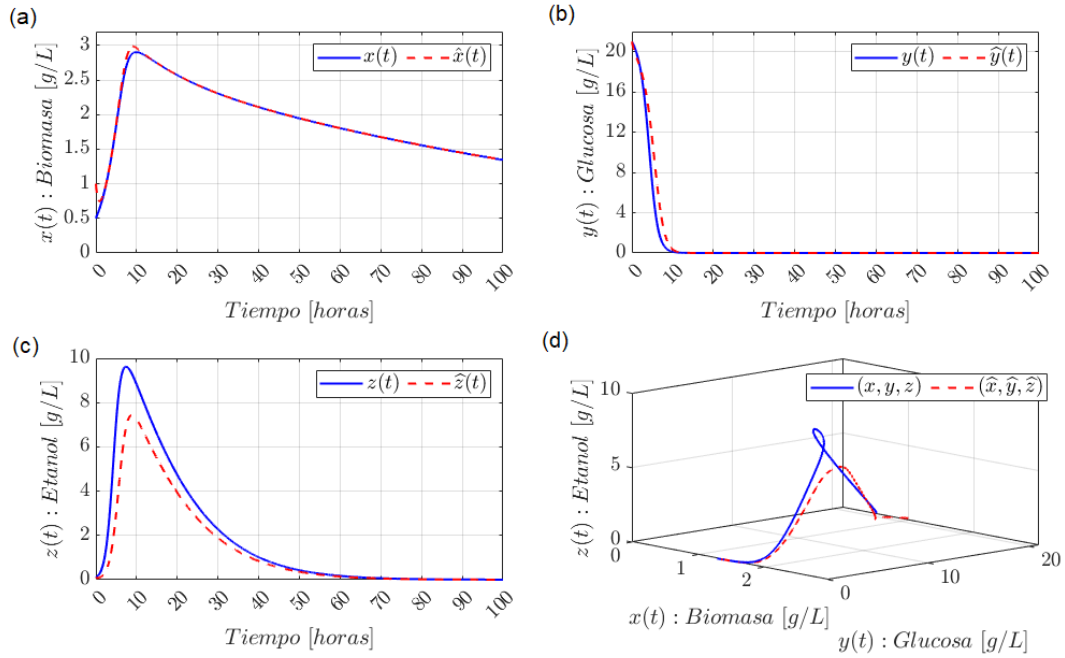


Figura 3.2 Series de tiempo de las variables del sistema y del sensor virtual

Nota. $x(t)$ y $\hat{x}(t)$ contra tiempo (a), $y(t)$ y $\hat{y}(t)$ contra tiempo (b), $z(t)$ y $\hat{z}(t)$ contra tiempo (c), trayectoria tridimensional entre $x(t)$, $y(t)$, $z(t)$, y $\hat{x}(t)$, $\hat{y}(t)$, $\hat{z}(t)$ (d).

La visualización tridimensional en la Figura 3.2 refleja la proximidad de las trayectorias simuladas y estimadas, lo cual resalta la capacidad del sensor virtual para capturar el comportamiento global del sistema. Los resultados indican una alta concordancia entre ambas curvas por lo general, lo que sugiere que el sensor virtual de estados es capaz de seguir con precisión el comportamiento dinámico del sistema.

Las gráficas de cada variable del sistema contra la misma variable estimada idealmente deben ser una línea diagonal. La biomasa y la glucosa presentan un patrón no muy alejado de esto, representado gráficamente en la Figura 3.3, mientras que la gráfica de etanol es la que parece tener el mayor desvío. En la gráfica de errores se puede observar que estos disminuyen con el tiempo, convergiendo a valores cercanos a cero, confirmando que el sensor virtual es efectivo para reducir el error de estimación a lo largo del proceso.

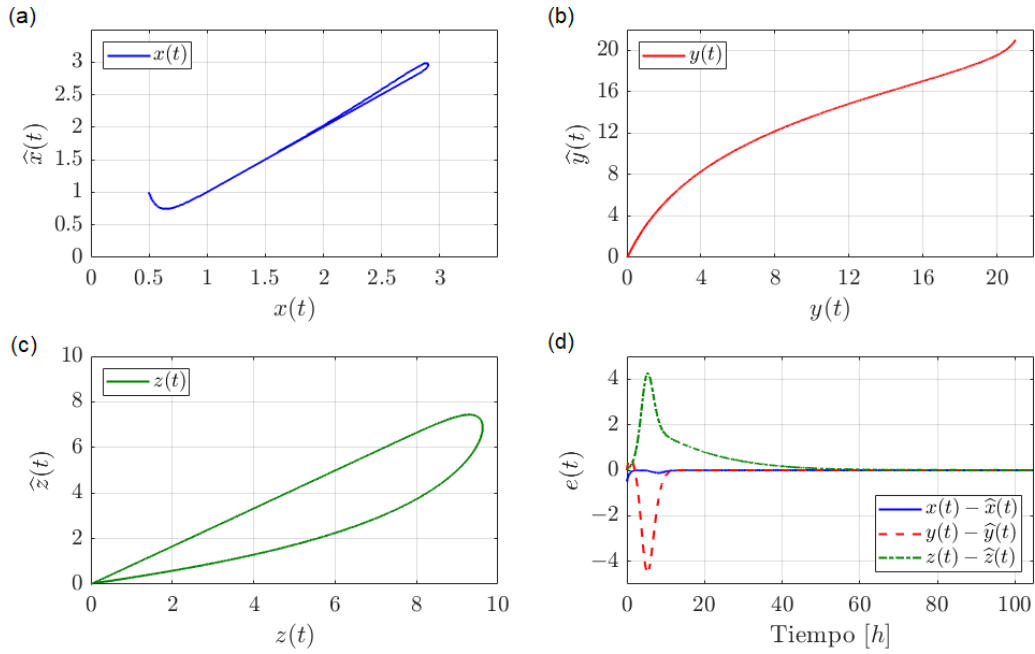


Figura 3.3 Relación entre variables del sistema y estimaciones del sensor virtual

Nota. $x(t)$ contra $\hat{x}(t)$ (a), $y(t)$ contra $\hat{y}(t)$ (b), $z(t)$ contra $\hat{z}(t)$ (c), error de las variables x , y , z respecto a $\hat{x}(t)$, $\hat{y}(t)$, $\hat{z}(t)$ contra tiempo (d).

Los histogramas visualizados en la Figura 3.4 muestran la distribución de las concentraciones simuladas y estimadas. La similitud en la forma y amplitud de las distribuciones sugiere que el sensor virtual representa adecuadamente las características estadísticas del sistema, pero indica que el rango de las estimaciones se queda por debajo del valor máximo del etanol del sistema simulado.

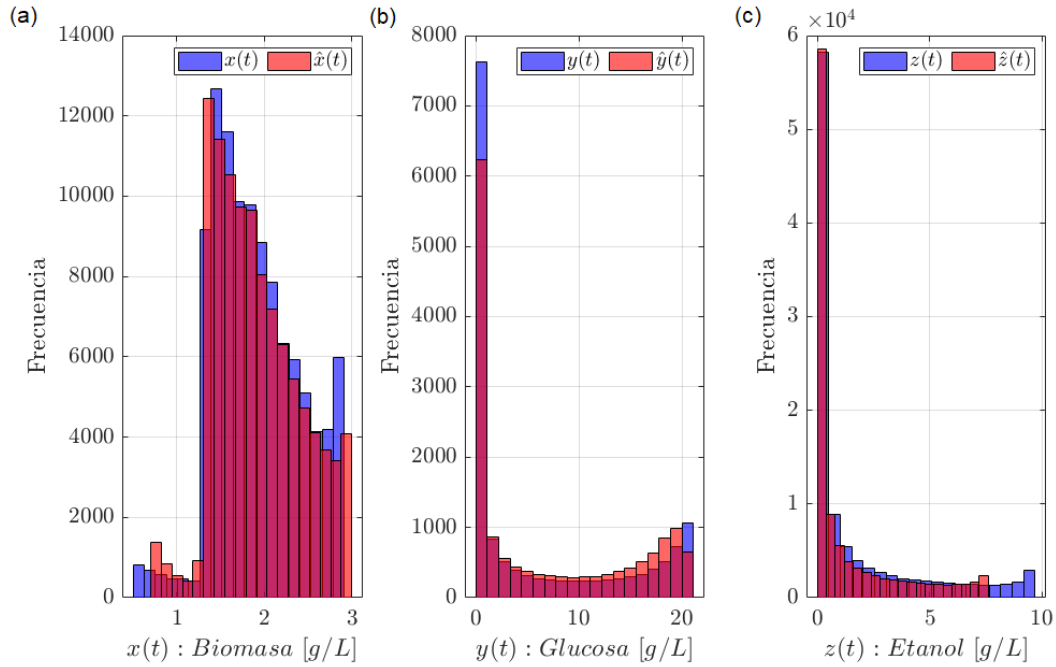


Figura 3.4 Histogramas de las variables del sistema y del sensor virtual

Nota. De izquierda a derecha: distribución de $x(t)$ y $\hat{x}(t)$, distribución de $y(t)$ y $\hat{y}(t)$, distribución de $z(t)$ y $\hat{z}(t)$.

Los diagramas de caja ofrecen una comparación del rango y la dispersión de los datos simulados y estimados. Como se observa en la Figura 3.5, la biomasa muestra los rangos más similares con una mediana casi idéntica, seguido por la glucosa donde se puede apreciar un ligero desfase de la mediana de lo cual se entiende que se hay más valores cercanos a cero para la variable del sistema. Para el etanol, las medianas parecen estar al mismo nivel a pesar que el rango del sistema abarca una mayor magnitud que el del sensor virtual, apoyando lo observado en las previas gráficas.

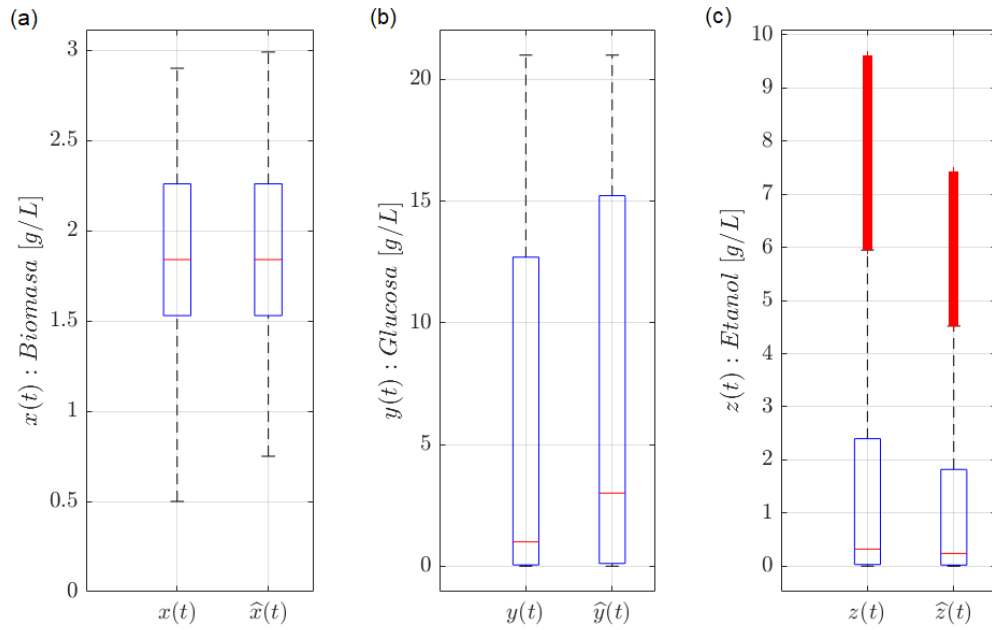


Figura 3.5 Diagramas de caja de las variables del sistema y del sensor virtual

Nota. De izquierda a derecha: comparación de $x(t)$ y $\hat{x}(t)$, comparación de $y(t)$ y $\hat{y}(t)$, comparación de $z(t)$ y $\hat{z}(t)$.

Para evaluar la precisión y la confiabilidad del sensor virtual íntegramente, se complementan los recursos gráficos con un análisis cuantitativo mediante las cinco métricas claves previamente calculadas. Debido a que el MAE y el RMSE miden en las mismas unidades que las variables, los resultados cercanos a cero indican un buen ajuste del modelo y una mayor precisión. Para ambas métricas, el valor más pequeño lo tiene la biomasa mientras que el etanol presenta un error ligeramente notable tal como lo evidencia la Tabla 3.1.

Tabla 3.1 Métricas de evaluación cuantitativa

Variable de estado	MAE (g/L)	RMSE (g/L)	R ²	U de Theil	R
Biomasa x(t)	0.008	0.033	0.995	0.008	0.998
Glucosa y(t)	0.176	0.733	0.958	0.096	0.984
Etanol z(t)	0.176	0.887	0.890	0.157	0.979

A continuación, se obtienen los resultados para R y R², con un valor ideal cercano a 1, para cuantificar qué tan bien se ajustan las estimaciones a los datos del sistema al igual que reflejar la fuerza y dirección de la relación lineal. Siguiendo el mismo patrón, los valores para biomasa presentan el mejor desempeño mientras que el valor menos favorable lo tiene el etanol. A pesar de no cumplir con el estándar que mantiene la variable de la biomasa o glucosa, aún son lo suficientemente altos sus valores para considerarse como buenos ajustes en aplicaciones prácticas

Por último, se realiza el cálculo de U de Theil para comparar el desempeño del modelo con una predicción ingenua donde un resultado cercano a cero indica un desempeño superior. Los valores bajos para las tres variables sugieren que el modelo supera ampliamente una predicción ingenua a pesar que las variables de glucosa y etanol de nuevo sean menos eficientes que la predicción de biomasa, presentan un rendimiento aceptable.

Estos resultados indican que el sensor virtual es altamente preciso para biomasa y glucosa, mostrando correlaciones cercanas al ideal pero el etanol presenta mayor error acumulado en valores extremos o variaciones puntuales. Esto da lugar a un área de oportunidad donde se realizan requerir ajustes en las ganancias del sensor virtual o en el modelo mecanicista para mejorar su rendimiento.

CAPÍTULO 4. CONCLUSIONES

4.1 Conclusiones del proyecto

Los resultados obtenidos confirmaron el cumplimiento de los objetivos específicos y su contribución al objetivo general de desarrollar un sensor virtual para estimar la concentración de etanol en fermentación continua. Primero, se desarrolló el programa transmisor basado en el modelo mecanicista, lo que permitió generar datos crudos que representaron de manera precisa la evolución de las concentraciones de biomasa, glucosa y etanol en un sistema de fermentación continua. Estos datos constituyeron la referencia necesaria para evaluar la efectividad del sensor virtual implementado. Segundo, se desarrolló el sensor virtual, diseñado para estimar las variables del sistema en tiempo real basándose únicamente en mediciones de biomasa, y demostró ser una herramienta eficaz. Finalmente, se evaluó la convergencia entre los datos simulados y las estimaciones del sensor virtual. Las gráficas revelaron una convergencia visual significativa entre las series temporales de las variables simuladas y estimadas. Adicionalmente, las métricas cuantitativas confirmaron esta tendencia, con coeficientes de determinación superiores a 0.89 y errores absolutos y relativos bajos en todas las variables. La biomasa y la glucosa mostraron los mejores resultados, con valores de R^2 cercanos a la unidad, mientras que el etanol presentó un mayor margen de error, probablemente relacionado con la naturaleza no lineal del modelo. Los análisis cuantitativos mostraron evidencia de que los errores fueron mínimos y las correlaciones fueron fuertes, lo cual validó la precisión y confiabilidad del sensor virtual. En conjunto, los resultados obtenidos validaron el uso del sensor virtual como una alternativa práctica, eficiente y viable para estimar las variables críticas en sistemas de fermentación continua.

4.2 Trabajo futuro

Con base en los resultados obtenidos, se identificaron áreas de mejora, como el ajuste de las ganancias del sensor virtual para optimizar la convergencia y reducir el error, mejorando su desempeño en escenarios operativos específicos. Adicionalmente, podría explorarse la implementación del sensor virtual en un entorno experimental para validar su funcionalidad bajo condiciones reales de operación.

4.3 Competencias desarrolladas

Realizar investigación científica y desarrollo tecnológico. El proyecto se fundamentó en la implementación de un modelo mecanicista para desarrollar un sensor virtual que permitió estimar en tiempo real la concentración de etanol en procesos de fermentación continua, como una alternativa práctica a métodos de medición convencionales con latencia significativa. Este proceso destacó la importancia de aplicar metodologías científicas para abordar problemas específicos en la industria vinícola.

Integrar soluciones informáticas en el campo de la Ingeniería Biomédica. La implementación del sensor virtual empleó herramientas tecnológicas como microcontroladores y software especializado para procesar datos, realizar cálculos avanzados y registrar los resultados. Estas actividades permitieron el monitoreo del proceso fermentativo mediante la simulación y estimación de las principales variables dinámicas del sistema.

Competencias instrumentales. Se desarrolló la capacidad para aplicar conocimientos matemáticos y computacionales en la resolución de problemas complejos mediante la programación y el análisis de datos. Esto incluyó el manejo de herramientas tecnológicas como MATLAB y software de monitoreo serial (CoolTerm) para el diseño, simulación y validación de sistemas virtuales. Adicionalmente, se fortalecieron habilidades para gestionar información técnica y generar reportes claros y organizados, fundamentales para la comunicación efectiva en el ámbito profesional.

Competencias interpersonales. Se desarrollaron las habilidades de comunicación efectiva, reflejadas en la redacción del informe técnico y en la presentación de resultados de manera comprensible y profesional. También se promovió el trabajo en colaboración con asesores y revisores del proyecto, favoreciendo el intercambio de ideas y el mejoramiento continuo.

REFERENCIAS

- [1] A. L. Lehninger, "Principles of Biochemistry," 4ª ed., W. H. Freeman, 2004.
- [2] J. Herrera, L. León, Y. Torres, N. Cano, A. Herrera, and M. Cuenca, "Evaluación y selección de levadura comercial para el proceso de fermentación alcohólica de hidromiel", *Publ. investig.*, vol. 13, no. 2, pp. 23–30, Jul. 2019, doi: 10.22490/25394088.3651.
- [3] E. K. Lindskog, "The Upstream Process: Principal Modes of Operation," en *Biopharmaceutical Processing*, G. Jagschies, E. Lindskog, K. Łacki, y P. Galliher, eds., 1ª ed., Elsevier, 2018, pp. 625–635. [En línea]. Disponible: <https://doi.org/10.1016/B978-0-08-100623-8.00031-1>
- [4] P. J. Verbelen, D. P. De Schutter, F. Delvaux et al., "Immobilized yeast cell systems for continuous fermentation applications," *Biotechnol. Lett.*, vol. 28, pp. 1515-1525, 2006. DOI: <https://doi.org/10.1007/s10529-006-9132-5>.
- [5] A. Karim, N. Gerliani, y M. Aïder, "Kluyveromyces marxianus: An emerging yeast cell factory for applications in food and biotechnology," *International Journal of Food Microbiology*, vol. 333, art. no. 108818, 2020. DOI: <https://doi.org/10.1016/j.ijfoodmicro.2020.108818>.
- [6] G. G. Fonseca, E. Heinzle, C. Wittmann, et al., "The yeast *Kluyveromyces marxianus* and its biotechnological potential," *Applied Microbiology and Biotechnology*, vol. 79, pp. 339–354, 2008. DOI: <https://doi.org/10.1007/s00253-008-1458-6>.
- [7] J. Schneider, "Microcontroller vs. Microprocessor: What's the Difference?", IBM, 13 de junio de 2024. [En línea]. Disponible en: <https://www.ibm.com/es-es/think/topics/microcontroller-vs-microprocessor>. [Accedido: 15 de enero de 2025].
- [8] Sanmit Chatterjee, "What are the Key Pros and Cons of the Arduino Programming Language?", Emeritus, 25 de enero de 2023. [En línea]. Disponible en:

<https://emeritus.org/blog/coding-arduino-programming-language/>. [Accedido: 15 de enero de 2025].

[9] Arduino, "Arduino - An Open-Source Electronics Platform," Arduino.cc. [En línea]. Disponible en: <https://www.arduino.cc/>. [Accedido: 15 de enero de 2025].

[10] Arduino, "Arduino Uno Rev3," Arduino.cc. [En línea]. Disponible en: <https://store.arduino.cc/arduino-uno-rev3>. [Accedido: 15 de enero de 2025].

[11] Arduino, "Arduino Mega 2560 Rev3," Arduino.cc. [En línea]. Disponible en: <https://store.arduino.cc/arduino-mega-2560-rev3>. [Accedido: 15 de enero de 2025].

[12] M. Kano y K. Fujiwara, "Virtual Sensing Technology in Process Industries: Trends and Challenges Revealed by Recent Industrial Applications," *Journal of Chemical Engineering of Japan*, vol. 46, no. 1, pp. 1–17, 2013. DOI: <https://doi.org/10.1252/jcej.12we167>.

[13] B. N. Biswas, S. Chatterjee, S. P. Mukherjee y S. Pal, "A discussion on Euler method: A review," *Electronic Journal of Mathematical Analysis and Applications*, vol. 1, no. 2, pp. 2090–2792, 2013.

[14] D. Iskandar y O. C. Tiong, "The application of the Runge-Kutta Fourth Order Method in SIR Model for simulation of COVID-19 Cases," *Proceedings of Science and Mathematics*, vol. 10, pp. 61–70, 2022.

[15] J. M. Franco, "Runge–Kutta methods adapted to the numerical integration of oscillatory problems," *Applied Numerical Mathematics*, vol. 50, no. 3–4, pp. 427–443, 2004.

[16] Y.-H. Du, M.-Y. Wang, L.-H. Yang, L.-L. Tong, D.-S. Guo y X.-J. Ji, "Optimization and Scale-Up of Fermentation Processes Driven by Models," *Bioengineering*, vol. 9, no. 9, p. 473, 2022. DOI: <https://doi.org/10.3390/bioengineering9090473>.

- [17] Y. Salazar, P. A. Valle, E. Rodríguez, N. O. Soto-Cruz, J. B. Páez-Lerma y F. J. Reyes-Sánchez, "Mechanistic Modelling of Biomass Growth, Glucose Consumption and Ethanol Production by *Kluyveromyces marxianus* in Batch Fermentation," *Entropy*, vol. 25, no. 3, p. 497, marzo 2023. DOI: <https://doi.org/10.3390/e25030497>.
- [18] A. J. Vargas Rodríguez, "Implementación de un observador para la estimación de producción de etanol." [Consultado: 03 de junio de 2024].
- [19] J. D. Hamilton, *Time Series Analysis*, Princeton University Press, 1994. ISBN: 0-691-04289-6.
- [20] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006. doi: 10.1016/j.ijforecast.2006.03.001.
- [21] L. Bessler, "Distributions, Histograms, Box Plots, and Alternative Tools," en *Visual Data Insights Using SAS ODS Graphics*, Apress, Berkeley, CA, 2023. doi: 10.1007/978-1-4842-8609-8_12.
- [22] D. Stroock, *Probability Theory*, Cambridge University Press, 2011, pp. 43. ISBN: 978-0-521-13250-3.
- [23] J. Shreffler and M. R. Huecker, "Exploratory data analysis: Frequencies, descriptive statistics, histograms, and boxplots," *StatPearls*, StatPearls Publishing, 2023.
- [24] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance," *Climate Research*, vol. 30, pp. 79–82, 2005. doi: 10.3354/cr030079
- [25] B. Ratner, "The correlation coefficient: Its values range between +1/–1, or do they?," *Journal of Targeting, Measurement and Analysis for Marketing*, vol. 17, pp. 139–142, 2009. doi: 10.1057/jt.2009.5.

- [26] L. J. Saunders, R. A. Russell, and D. P. Crabb, "The coefficient of determination: what determines a useful R^2 statistic?," *Investigative Ophthalmology & Visual Science*, vol. 53, no. 11, pp. 6830–6832, 2012.
- [27] P. Conceição and J. K. Galbraith, "Constructing Long and Dense Time-Series of Inequality Using the Theil Index," *Eastern Economic Journal*, vol. 26, no. 1, pp. 61–74, 2000.
- [28] H. B. Seo, H. J. Kim, O. K. Lee, J. H. Ha, H. Y. Lee, and K. H. Jung, "Measurement of ethanol concentration using solvent extraction and dichromate oxidation and its application to bioethanol production process," *Journal of Industrial Microbiology and Biotechnology*, vol. 36, no. 2, pp. 285–292, 2009.
- [29] D. C. Montgomery, "Statistical Quality Control", 7th ed., New York: Wiley, 2009.

ANEXOS

Los códigos de Arduino y MATLAB, junto con el archivo Excel que contiene los datos generados durante el proyecto, están disponibles en el siguiente repositorio:

<https://github.com/fobianglez/Sensor-virtual-para-la-estimacion-de-concentracion-de-etanol-en-fermentacion-continua.git>