



UNIVERSIDAD DE GUANAJUATO

CAMPUS IRAPUATO – SALAMANCA

DIVISIÓN DE INGENIERÍAS

**IMPLEMENTACIÓN DEL MÉTODO DE PROGRAMACIÓN
LINEAL A TRAVÉS DE REDES NEURONALES: CASO DE
ESTUDIO DE OPTIMIZACIÓN DE PROTOCOLOS DE
SÍNTESIS QUÍMICA EN EMPRESA ITRASIG.**

TESIS

**QUE PARA OBTENER EL GRADO DE:
MAESTRÍA EN ADMINISTRACIÓN DE TECNOLOGÍAS**

PRESENTA:

Ing. Michelle Alejandra Meléndez Cardiel

DIRECTORES DE TESIS:

Dr. Ernesto Isaac Tlapanco Ríos

Dr. Everardo Vargas Rodríguez

CONTENIDO

Contenido	i
Lista de tablas	iii
Lista de figuras	iv
Agradecimientos.....	v
1 RESUMEN	1
2 Introducción	2
2.1 Justificación y relevancia del tema en la línea investigación seleccionada	3
2.2 Antecedentes	5
2.3 Objetivos	10
2.3.1 Objetivo General.....	10
2.3.2 Objetivo Específico	11
3 Marco Teórico	12
3.1 Biología molecular.....	12
3.1.1 ADN y oligonucleótido	12
3.1.2 Fosforamiditas	13
3.1.3 Síntesis química	13
3.1.4 Etapas de la síntesis química	14
3.1.5 Determinación de la eficiencia de acoplamiento de la síntesis química	15
3.2 Programación lineal	16
3.2.1 Método de solución gráfica.....	17
3.2.2 Método de solución simplex	18
3.3 Redes Neuronales	19
4 Metodología	25
4.1 Caracterización del sistema	25

4.1.1	Características del equipo y columnas de prueba	26
4.1.2	Protocolos de síntesis o scripts	29
4.1.3	Procedimiento de síntesis.....	29
4.2	Diseño y pruebas experimentales.....	33
4.2.1	Identificación de variables	33
4.2.2	Volumen de reactivos (V)	34
4.2.3	Tiempo de proceso (T)	35
4.2.4	Presión de drenado (P).....	35
4.3	Diseño y pruebas experimentales.....	35
4.3.1	Escenarios experimentales y dataset	36
4.4	Modelado del proceso: función objetivo y restricciones	38
4.4.1	Función objetivo.....	38
4.4.2	Modelado de restricciones.....	41
4.5	Implementación de programación lineal con redes neuronales	43
5	Resultados del proceso de optimización.....	47
5.1	Resultados obtenidos con la red neuronal recursiva	49
5.2	Comparación de resultados obtenidos con la red recurrente y el método simplex.....	52
6	Conclusiones.....	53
7	Productos generados	54
8	Referencias	55
APÉNDICE 1	62

LISTA DE TABLAS

Tabla 1. Ejemplos de artículos publicados sobre aplicaciones de PL con IA en diversas ramas de estudio.....	9
Tabla 2. Tabla inicial simplex	19
Tabla 3. Funciones de activación más comunes.....	21
Tabla 4. Variables involucradas en el proceso de síntesis química	33
Tabla 5. Variables consideradas para la realización del diseño experimental con sus respectivos valores.....	34
Tabla 6. Resultados obtenidos de los escenarios experimentales	37
Tabla 7. Factores y niveles de las variables estudiadas	37
Tabla 8. Resultados obtenidos del diseño factorial 3k para el proceso de síntesis química.....	38
Tabla 9. Resultados obtenidos al aplicar diferentes valores de los parámetros de la red recurrente.....	50

LISTA DE FIGURAS

Figura 1. Doble hélice de ADN. Imagen modificada de (<i>Repaso de la estructura y replicación del ADN (artículo) Khan Academy, s/f</i>).....	13
Figura 2. Ciclo de síntesis del método de las fosoramiditas. Extraída de: (<i>Oligo chemical synthesis – Custom oligo production Eurogentec, s/f</i>).....	14
Figura 3. Ejemplo de método de solución gráfica.....	18
Figura 4.Arquitectura de una red neuronal.....	20
Figura 5. Metodología seguida para desarrollar el proyecto presentado en este trabajo	25
Figura 6. Fotografía de la máquina sintetizadora automática ABI	26
Figura 7. Cámara de síntesis y puertos de inyección.....	27
Figura 8. Características entre columnas de prueba de escalas de 100 nmoles universal y 5 nmoles.....	28
Figura 9. Diagrama de flujo que explica el proceso que se siguió para la elección de los valores	36
Figura 10. Grafica de los experimentos realizados con 3.5 psi	41
Figura 11. Grafica de los planos que forman el espacio de factibilidad.....	42
Figura 12. Arquitectura de red neuronal recursiva	43
Figura 13. Arquitectura de la red neuronal utilizada para la solución del trabajo de tesis.....	48
Figura 14. Comportamiento de las variables de estado a través de las épocas. ..	51
Figura 15. Comportamiento de la función objetivo a través de las épocas.	52

AGRADECIMIENTOS

Agradezco a mi mamá y a mi hermana por ser el pilar de mi vida en todos los aspectos, y por su constante apoyo en la realización de mi maestría y el desarrollo de este trabajo de tesis. Pero sobre todo por el amor que me dan todos los días el cual es mi inspiración para crecer siempre como persona.

A mis asesores, al Doctor Ernesto Tlapanco por su guía, apoyo y conexión a la empresa y al Doctor Everardo Vargas por su constante apoyo en el desarrollo de este trabajo de tesis y todo el conocimiento que me proporcionó en el camino. A ambos por creer en mí y en mi proyecto, y por su respaldo tan grande en todas las dificultades que se presentaron en la carrera.

Un especial y gran agradecimiento a todo el equipo de las empresas ITRASIG y T4 Oligo, por su increíble apoyo, paciencia y dejarme ser parte de su tan acogedora familia. Al Doctor Octavio García, al MCB Mauricio Díaz y a la Doctora Karina Gutiérrez por creer en mí, en mi trabajo y estar siempre en la mejor disposición de enseñarme y ayudarme con los obstáculos que se presentaron.

A mi comité de tesis, la Doctora Rocío Lizárraga y la Doctora Dinora Guzmán, por sus comentarios, sugerencias y observaciones que hicieron de este trabajo de tesis un trabajo impecable.

1 RESUMEN

Debido a la gran importancia de las aplicaciones del ADN artificial en áreas como medicina, genómica y biología molecular, se desarrollaron máquinas encargadas de sintetizar ADN automáticamente, no obstante, muy pocas empresas se dedican a la producción de oligonucleótidos en el mundo ya que es un giro relativamente nuevo; afortunadamente en Irapuato, Guanajuato, México se encuentra la única empresa que produce ADN artificial en todo Latinoamérica: T4 Oligo e ITRASIG.

Sin embargo, a causa de los secretos industriales, aún no existen estudios acerca del uso óptimo de dichas máquinas, provocando que los programas que se cargan al equipo sintetizador con las cantidades de reactivos y pasos a realizar, para llevar a cabo síntesis de ADN, son elaborados empíricamente por los trabajadores.

Es por ello por lo que se desarrolla la metodología de programación lineal implementada con redes neuronales, para lograr que se maximice la productividad y ganancias a través de la modificación de protocolos de síntesis con bases científicas que respalden el proceso, asegurando que la elección en tiempos y cantidad de reactivos que conlleva el proceso son los óptimos.

2 INTRODUCCIÓN

La toma de decisiones en una empresa es de suma importancia para evitar que se generen pérdidas, es por ello por lo que los procesos que se siguen deben de tener una base matemática que establezca que las actividades que se realizan son las mejores para la empresa. Constantemente en la industria se presentan problemáticas que frenan el buen desarrollo o provocan el estancamiento en el producto o servicio. Por ejemplo, desperdicio de tiempo, de mano de obra, o de recursos, son algunos de los contratiempos, y para resolverlos se implementan técnicas de investigación de operaciones, las cuales se encargan de optimizar cuestiones específicas, dependiendo del giro empresarial.

Para este trabajo de tesis, al identificar que es de suma importancia el aplicar de manera adecuada los métodos para probar y validar el funcionamiento, se planteó la idea e importancia de optimizar los procesos y los beneficios que se pueden lograr al contacto de la empresa Instituto Traslacional de Singularidad Genómica (ITRASIG), para evaluar y conocer las actividades que realizan para identificar cuáles procesos pueden optimizarse. Afortunadamente, se identificó que el proceso que realizan los empleados para ingresar los químicos a las máquinas de sintetización química de oligonucleótidos es por tanteo y basados en la experiencia de los trabajadores, sin fundamento científico matemático alguno.

Es por ello por lo que, en el presente trabajo, se implementó un sistema de optimización mediante la metodología de programación lineal resuelta por medio de una técnica de inteligencia artificial, logrando facilitar la obtención de la solución óptima a este problema. Donde se consideraron las variables que utilizan para este proceso y sin dejar pasar las restricciones que se presentan para su elaboración, es decir, limitaciones de la máquina, como especificaciones de presión a la que se puede trabajar, el diseño de las columnas de prueba y el volumen máximo de capacidad con el que se desarrolla el correcto funcionamiento.

Una vez implementado el sistema, fue utilizado en la empresa con las máquinas y se realizaron todas las pruebas necesarias para validar y conseguir los resultados esperados de optimización, para posteriormente escalarlo a las demás máquinas con las que ahí se trabaja. Obteniendo como resultado un proceso ya optimizado que es empleado actualmente por la empresa para sus máquinas.

2.1 Justificación y relevancia del tema en la línea investigación seleccionada

El proceso de toma de decisiones en la vida de las personas se desarrolla bajo incertidumbres. Expresiones como “probablemente”, “no muy claras” y “bastante peligrosas”, que se escuchan con frecuencia en la vida cotidiana, son el resultado de inseguridades. Por lo tanto, si no se tienen en cuenta las posibles dudas en el proceso de toma de decisiones, los resultados esperados pueden ser engañosos. (İskenderoğlu & Akdag, 2019)

Cada campo técnico y de ingeniería emplea las técnicas de optimización en la etapa de diseño del sistema en la que se busca la mejor solución sujeta a un criterio predefinido. El proceso de optimización tiene como objetivo encontrar la mejor solución entre muchas soluciones disponibles. El cuerpo de conocimiento asociado a los problemas de optimización se denomina programación matemática, abarcando las situaciones derivadas de las técnicas de programación lineal, no lineal, aleatoria, mono-objetivo, multiobjetivo, geométrica, separable, entre otras (Khan & Rafique, 2021).

La investigación de operaciones es una rama de las matemáticas aplicadas, donde la resolución a problemas de optimización es la base, dependiendo del campo al que se aplique en la industria, como en economía, administración, ingeniería, etc. Su objetivo principal es proporcionar las herramientas sustanciales para la toma de decisiones, con el fin de controlar u optimizar recursos (Torres, 2022).

La programación lineal (LP) es una de las técnicas de programación matemática más estudiadas y destacadas, la cual consiste en maximizar o minimizar una función de objetivo única considerando una serie de restricciones. La programación lineal se ha aplicado con éxito a todos los campos técnicos entre los que se pueden mencionar la ingeniería, la economía, la inteligencia computacional y la investigación de operaciones. La programación lineal tiene la fuerza para modelar una base teórica adecuada, clara y elegante, que puede describir situaciones altamente complejas (Khan & Rafique, 2021)

La inteligencia artificial (IA) es la habilidad de las máquinas para realizar actividades que comúnmente necesitarían inteligencia humana y para poder lograrlo requieren aprender los procesos a partir del análisis de datos, para posteriormente poder tomar decisiones. A diferencia de los humanos, las computadoras no descansan y por lo tanto desempeñan un mayor alcance en volumen de trabajo (Rouhiainen, 2018). Por lo que, la inteligencia artificial es una excelente herramienta para solucionar múltiples problemas de diversas áreas de estudio, como por ejemplo en el campo biomédico y genómico donde se requiere del manejo de grandes volúmenes de datos y la realización de tareas complejas y repetitivas que normalmente requieren mucho tiempo. En general, las técnicas de IA proporcionan grandes beneficios en diversas áreas del conocimiento, como por ejemplo: la automatización de procesos, el análisis celular, la modelación de estructuras moleculares, la planeación de síntesis química, entre otros (Matsingos et al., 2021).

Una técnica de la IA son las redes neuronales las cuales se pueden definir como un sistema que facilita establecer una relación de las entradas con las salidas. La idea de las redes neuronales está inspirada en el comportamiento de un cerebro humano, donde la principal función es el procesamiento en paralelo de la información, posibilitando el aprendizaje y generalización de situaciones. El objetivo del uso de una red neuronal es el resolver problemas complejos teniendo la capacidad de realizar predicciones (Rouhiainen, 2018). El uso de las redes neuronales representa una gran oportunidad para una amplia gama de aplicaciones, como por ejemplo en la biogenética, específicamente en la rama ómica, para la transcripción y replicación de ADN y ARN, entre otras. Esto debido a que permite realizar análisis más rápidos de grandes cantidades de datos, así como el poder implementar metodologías para mejorar protocolos y pronósticos de enfermedades y mutaciones (Sarmiento-Ramos, 2020).

La combinación de programación lineal con redes neuronales resulta ser un excelente camino para la optimización de procesos asegurando una solución eficaz debido a que puede reducir la gran cantidad de tiempo que conlleva el exceso de iteraciones presentes en el proceso de obtener una solución de PL, previniendo entorpecer el medio para encontrar la solución óptima del problema (Peña Florez et al., 2018).

Con la elaboración de este trabajo se generaron beneficios mayúsculos a la empresa T4 Oligo e ITRASIG, mediante el desarrollo e implementación de un modelado de optimización a través de la técnica de programación lineal, permitiendo optimizar las variables establecidas de acuerdo con las restricciones otorgadas.

2.2 Antecedentes

De acuerdo con la literatura, existen diversas técnicas de investigación de operaciones para optimizar, de las cuales destacan las siguientes: programación lineal, programación entera (PE), programación dinámica (PD), programación de redes (PR) y programación no lineal (PNL) (Taha, 2017). La aplicación de estas técnicas no tiene un área disciplinar exclusiva y puede encontrarse una amplia cantidad de modelados en casi todas las áreas del conocimiento.

Los métodos de solución tradicionales para dichas técnicas puede ser el gráfico o el simplex, donde cada uno se utiliza dependiendo del modelado matemático que se tenga a optimizar. En el caso del método gráfico, es muy útil para problemas de únicamente dos o tres variables, debido a que es sumamente sencillo de visualizar y entender, además de su fácil implementación sin necesidad de cálculos complejos o iteraciones. Sin embargo, presenta muchas desventajas al ser el método más sencillo, su uso se limita a problemas lineales con exclusivamente tres variables de decisión y tampoco garantiza la obtención de la solución óptima, pero sí una aproximación inicial a problemas aún más complejos.

Por otro lado, el método simplex sustituye al gráfico cuando se presentan más de dos variables y problemas relativamente grandes con diversas restricciones, además de que ofrece una solución precisa y óptima para problemas lineales. No obstante, debido a la manera de implementación de este método, la sensibilidad a la inicialización de los valores puede afectar significativamente al valor óptimo, provocando soluciones subóptimas pero factibles, sin mencionar que requiere un formato estándar específico inicial para poder ser aplicado y el no poder ser aplicado para problemas con función objetivo o restricciones no lineales.

La programación lineal es ampliamente usada en prácticamente todas las áreas y los métodos “tradicionales” son aplicados de manera exitosa para optimizar este tipo de problemas. Por ejemplo, (Tapia & Cevallos., 2022), usaron programación lineal, en particular el método de Simplex, para encontrar una combinación óptima de ventas de productos para una empresa que resultó afectada por la pandemia del

Covid-19, la cual causó una disminución en la demanda de compras incluso por clientes frecuentes. En consecuencia, con la aplicación de este modelado los autores lograron maximizar las utilidades de la empresa, mediante un modelado de función objetivo y restricciones lineales con 4 variables.

En otro ejemplo, (Ardenghi et al., 2022) presentaron el diseño de un humedal artificial para el tratamiento de aguas residuales que provienen de empresas ganaderas, el cual modelaron matemáticamente mediante programación lineal entera, donde modelaron su función objetivo con cuatro variables y dos restricciones lineales. Acertadamente, aplicaron el método simplex, ya que era el método que se acomodaba más a sus necesidades, y lograron alcanzar una optimización de capacidad de los humedales con lo que mejoraron sus resultados en comparación con los que se obtenían antes de aplicar el proceso de optimización.

En otro tipo de aplicaciones la investigación de operaciones también ha sido empleada en hoteles que son atendidos por robots, como es el caso del estudio de (Lee et al., 2020), quienes se encargaron de diseñar un modelo matemático basado en programación lineal para predecir la cantidad de robots necesarios para cubrir las actividades que harían los empleados de un hotel, tales como recibir las maletas, servicio al cuarto, traslado, entre otras. Obteniendo como resultado la minimización de unidades de robots, eficientizando la asignación de las tareas para cada uno, es decir, donde antes un robot se encargaba únicamente de una tarea después de aplicar la metodología podía encargarse de 3 tareas. Con esto se optimizó la asignación de tareas a cada robot mediante el método simplex por sus 4 variables y sus restricciones lineales, que en consecuencia logró minimizar el número de robots que necesita el hotel.

Cabe resaltar, que recientemente una gran cantidad de autores que apostaron por las técnicas de investigación de operaciones, en diferentes áreas de estudio, para la solución a sus problemas, también han estado optando por la implementación de herramientas basadas en inteligencia artificial para eficientizar el tiempo de procesamiento y obtención de resultados, para evaluar más soluciones factibles, y por su increíble flexibilidad de adaptarse a una amplia variedad de problemas de investigación de operaciones, pasando de problemas lineales a no lineales, sin

olvidar la facilidad de automatización del proceso sin necesidad de intervención humana en cada paso.

Por ejemplo, (Varghese & Srivastav, 2022) implementaron un modelado por medio de la técnica de programación lineal para el desarrollo de galletas nutritivas de bajo costo y alta energía que satisfagan los requisitos recomendados por el Instituto Nacional de Nutrición en niños desnutridos, teniendo como variables: la cantidad de ingredientes necesarios, la energía total que debe de contribuir, la cantidad de proteína y grasas y el precio. Además de otros atributos de calidad como son el sabor, el color, la textura y el gusto. En dicho trabajo los autores lograron implementar su trabajo a través de un algoritmo híbrido que ellos propusieron, que incluía lógica difusa, obteniendo resultados exitosos con gran facilidad debido a que un método tradicional como el simplex no hubiera encontrado una solución óptima a causa de la gran cantidad de variables y restricciones junto con la función objetivo no lineal, cumplieron con más del 30% de los requisitos de la cantidad diaria recomendada y redujeron el costo de ingredientes de 4.3 USD/kg a 1.52 USD/kg, asimismo logrando evitar la importación de productos, utilizando ingredientes comunes que se dan en la población.

En otro estudio, pero ahora en la industria aerodinámica, referente al diseño de una turbina, (Atalay et.al., 2020) presentaron resultados del modelado, computacionalmente, de una turbina eólica para optimizar los diseños de acuerdo con los parámetros de potencia que requieren. En dicho trabajo, los autores propusieron una nueva metodología de programación lineal difusa la cual integra la regresión lineal, ya que su función objetivo presentaba la no linealidad y una gran cantidad de variables a considerar. Gracias a esta ingeniosa idea los resultados de rendimiento alcanzaron el valor de 0.52 en eficiencia, siendo superior a los diseños anteriores.

Cabe mencionar que sí existen métodos “tradicionales” para la solución de modelos no lineales, como es el caso del método simplex modificado, pero aun presenta limitantes. En la literatura se pueden encontrar aproximaciones derivadas de métodos como el simplex, las cuales resuelven modelos no lineales particulares, por ejemplo, que la función objetivo puede ser cuadrática pero las restricciones deben ser forzosamente lineales. En otras aproximaciones se puede requerir que los modelos a resolver tengan un formato estándar, requiriendo que los problemas

se adapten específicamente a esta estandarización, lo cual conlleva a una gran complicación, ya que en la vida cotidiana, todos los problemas tienen diferentes características y comportamientos y no siempre se pueden adaptar a las particularidades de dichos métodos, es por ello por lo que muchos autores han estado estudiando la viabilidad de implementar herramientas o aproximaciones basadas en IA, que les permitan resolver casos específicos, particularmente modelos no lineales, sin mayor dificultad debido a la adaptabilidad de los algoritmos.

En este contexto, los métodos que combinan la programación lineal con IA se están volviendo cada vez más populares para realizar procesos de optimización, y existe una gran cantidad de trabajos publicados en la literatura. Por ejemplo, en la Tabla 1 se presenta una lista de trabajos de PL y se desglosa sus respectivas áreas de aplicación, junto con las técnicas empleadas.

En varios de los trabajos listados en la Tabla 1 se menciona que la diferencia de utilizar métodos tradicionales de solución a problemas de programación lineal contra los métodos que combinan técnicas de inteligencia artificial depende de la cantidad de variables, restricciones, combinaciones y la no linealidad. Esto porque los algoritmos tradicionales se pueden convertir en ineficientes debido a la limitación que induce su procesamiento secuencial el cual puede incrementar de manera significativa el tiempo del cálculo requerido para obtener una solución dependiendo de la estructura y las dimensiones del problema planteado (De La et al., 2017). Las técnicas de IA más usadas para implementarse con técnicas de IO son: (a) lógica difusa por ser de gran ayuda para problemas con variables no numéricas y con restricciones muy específicas para cada una de ellas, mientras que (b) las redes neuronales se implementan más cuando las funciones y/o restricciones son no lineales.

Gracias a la basta recopilación de trabajos en los cuales se visualiza de mejor manera los resultados exitosos al hacer uso de investigación de operaciones implementando alguna técnica de inteligencia artificial, se puede concluir que es una opción muy acertada para optimizar procesos, servicios o productos en casi cualquier rama de la ciencia, esto debido a la adaptabilidad a sistemas no lineales, tanto en la función objetivo como en las restricciones. Ahora bien, se toma en consideración para que sirva de punto de partida para la optimización del caso de

estudio formulado por la empresa ITRASIG, cuya área disciplinar es el laboratorio genómico químico, el cual consiste en maximizar la ganancia económica por la fabricación de oligonucleotidos. En este caso la empresa usa una máquina para fabricar estos oligo basados en la ejecución de un protocolo de proceso, el cual a su vez depende de varias variables. Así mismo para que el producto fabricado pueda ser útil debe de cumplir con ciertos parámetros técnicos y de calidad. Por lo que la empresa requiere determinar los valores de las variables de sistema que satisfagan las restricciones del proceso de fabricación y de calidad y que al mismo tiempo maximicen la ganancia que obtendría la empresa.

Para el modelado de este problema, de manera inicial se desconoce la correlación de las variables en la forma y la calidad y las características técnicas del producto fabricado. Además, no se pudieron encontrar publicaciones donde apliquen los métodos de PL para resolver problemas similares al que particular que plantea la empresa. Cabe mencionar que en la literatura si existen trabajos relativos a la rama de trabajo de la empresa, por lo que con base a los conocimientos del área de programación lineal y redes neuronales consideramos que es factible implementar una solución basada en IA para resolver el problema planteado. Esta consideración se hace partiendo del hecho que desconocemos el modelo que se debe optimizar y por las características del proceso es altamente probable que sea del tipo no lineal y como se describió en esta sección los métodos basados en técnicas de IA, como son las redes neuronales, brindan la flexibilidad de poder resolver este tipo de problemas.

Tabla 1. Ejemplos de artículos publicados sobre aplicaciones de PL con IA en diversas ramas de estudio

Aplicación	Metodología	Área	Referencia
Trabajos con características:			
a) Presentan múltiples variables que pueden o no estar relacionadas entre ellas, y restricciones específicas para cada una.			
b) Problemas donde sus datos no se pueden expresar únicamente con valores numéricos.			
Optimización de medicación en tratamientos de pacientes con VIH	Programación lineal difusa	Medicina	(Caetano et al., 2008)
Evaluación de técnicas de tratamiento para diferentes tipos de cáncer	Programación multicriterio y lógica difusa	Medicina	(Musa et al., 2019; Di. U. Ozsahin et al., 2019; I. Ozsahin, Ozsahin, Maisaini, et al., 2019; I. Ozsahin, Ozsahin, Nyakuwanikwa, et al., 2019; I.

			Ozsahin & Uzun Ozsahin, 2018)
Inversión en mercados	Programación lineal difusa	Económica	(İskenderoğlu & Akdag, 2019)
Diseño de hospitales en la distribución de espacios	Programación lineal difusa	Arquitectura	(Jamali et al., 2020)
Predicción de comportamiento de reactivos en síntesis química	Programación lineal y redes neuronales y lógica difusa	Biología molecular	(Eghbal Ahmadi & Mosayebi, 2021)
Minimizar costos y maximizar capacidad en vuelos	Programación lineal difusa	Industria de aerolíneas	(Khan & Rafique, 2021)
Reducción de tiempo de carga en vehículos eléctricos	Programación entera difusa	Ingeniería eléctrica	(Hussain et al., 2022)
Trabajos sobre modelos con características:			
a) Presentan funciones y/o restricciones no lineales			
b) Múltiples variables.			
Optimización de diagnóstico de cáncer	Programación lineal y redes neuronales	Medicina	(O. Mangasarian & Wolberg, 1990)
Predicción de cáncer	Programación lineal y arboles de decisión	Medicina	(Liu & Cios, 1992)
Diagnóstico y pronóstico de cáncer de mama	Programación lineal y redes neuronales	Medicina	(O. L. Mangasarian et al., 1995)
Predicción de enfermedades coronarias	Programación lineal y arboles de decisión	Medicina	(Pamela et al., 2013)
Planificación instrumental y de personal en hospital	Programación lineal y regresión de vectores de soporte	Administración hospitalaria	(Zlotnik et al., 2015)
Protección de sobre corriente en relees	Programación lineal y redes neuronales	Ingeniería electrónica	(Dehghanpour et al., 2018)
Optimización en diagnóstico de pacientes de mayor riesgo en hospitales	Programación entera mixta y redes neuronales	Medicina	(Chou et al., 2020)
Optimización en la identificación y estructuración de moléculas en oligonucleótidos para eficientizar propiedades farmacéuticas	Programación lineal y redes neuronales	Biología molecular	(Papargyri et al., 2020)
Vida útil de las redes sensoriales inalámbricas	Programación lineal y redes neuronales	Ingeniería electrónica	(Al-Mamari et al., 2022)
Predicción y optimización de reacciones químicas en el diseño de catalizadores	Programación lineal entera y redes neuronales	Química	(Schwaller et al., 2022)

2.3 Objetivos

2.3.1 Objetivo General

Encontrar la combinación óptima entre los tiempos del proceso, volumen de las sustancias utilizadas, presión de gas suministrado tal que maximice la ganancia obtenida por la producción los protocolos de síntesis que se realizan con la sintetizadora automática.

2.3.2 Objetivo Específico

- Identificar las variables y restricciones, que se involucran en el proceso de los protocolos de síntesis para diseñar el modelo matemático.
- Realizar pruebas experimentales, para la obtención de datos de entrenamiento para la red que establezca matemáticamente la relación entre las variables.
- Diseñar el modelo matemático, por medio de programación lineal para encontrar soluciones factibles de optimización.
- Implementar en software, el modelo de optimización de los protocolos de síntesis a través de redes neuronales.

3 MARCO TEÓRICO

3.1 Biología molecular

3.1.1 ADN y oligonucleótido

El ácido desoxirribonucleico, mejor conocido por sus siglas ADN, es un ácido nucleico que contiene la información genética utilizada en el desarrollo y funcionamiento de todos los organismos vivos y algunos virus; además de producir proteínas. Desde la perspectiva química, el ADN es un polímero de nucleótidos, es decir, un polinucleótido (Sekine, 2018). A su vez, cada nucleótido está conformado por tres elementos: un glúcido, una base nitrogenada y un grupo fosfato. De los cuales, cada uno puede ser:

- a) **Glúcido:** Desoxirribosa, formada por cinco átomos de carbono derivado de la ribosa.
- b) **Base nitrogenada:** Puede ser adenina (A), timina (T), citosina (C) o guanina (G).
- c) **Grupo fosfato:** Derivado del ácido fosfórico. (Paredes et al., 2017)

La forma de distinguir un polinucleótido de otro es la base nitrogenada, por esta razón, la secuencia de ADN se expone nombrando sus bases, siguiendo el criterio de combinación de A-T y G-C, ya que la adenina y la guanina son de mayor tamaño que la timina y la citosina; conformando uniformidad en la cadena (Illana, 2014). El ADN se presenta como una doble hélice, cuyas estructuras son antiparalelas, es decir, una hebra corre en dirección opuesta a la otra, teniendo como extremos 5' y 3' como se muestra en la Figura 1.

Sabiendo lo anterior, los oligonucleótidos sintéticos (ONS) son fragmentos cortos de ADN o RNA de cadena individual o doble, los cuales se construyen químicamente en un laboratorio, basado en el proceso de síntesis, en dirección de 3' a 5', y la secuencia de aminoácidos de la proteína de interés. (Patiño & Rugeles, 2006)

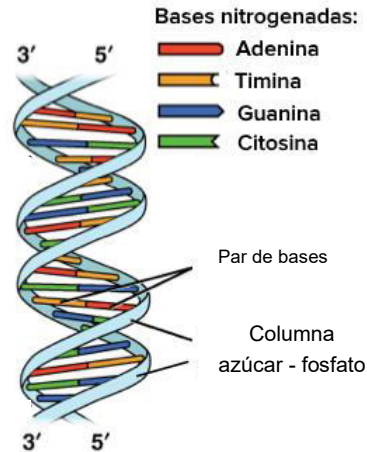


Figura 1. Doble hélice de ADN. Imagen modificada de (*Repaso de la estructura y replicación del ADN (artículo)* | Khan Academy, s/f)

3.1.2 Fosforamiditas

Las fosforamiditas son nucleósidos modificados a partir de grupos químicos de protección en los reactivos de la molécula, su función principal es impedir las reacciones no deseadas durante el proceso de síntesis. Existen varias moléculas usadas para proteger, pero las más usadas son un grupo dimetoxitritilo (DMT) en el extremo 5' y un grupo fosfito en el 3' del nucleósido (Brown, 1993).

La ventaja del uso de estos compuestos en la síntesis de oligonucleótidos es que son más estables, lo que permite tener un mejor almacenamiento y manipulación, además de una producción menos tardada y con mayor eficacia. Gracias a estas mejoras, la fabricación y distribución de oligonucleótidos a una escala comercial ha sido posible (Hogrefe & Biotechnologies, 2015).

3.1.3 Síntesis química

La síntesis química es el proceso por el cual se logra la construcción de compuestos moleculares complejos con intervención humana en un laboratorio, desde vitaminas y proteínas hasta ADN y ARN. La importancia de este proceso es el descubrimiento de nuevas reacciones y nueva química al momento de interactuar manualmente, logrando grandes avances en medicina, farmacología, bioquímica, genómica, entre otras ramas (Alazzouzi, 1996).

Los avances en la biología molecular traen consigo la síntesis de ADN mediante oligonucleótidos, los cuales son cadenas de ácido nucleico cortas, siendo gracias a este proceso que se ha logrado sintetizar genes. Existen diversos métodos de

síntesis de oligonucleótidos, pero el más utilizado es el de fosoramiditas, las cuales son nucleósidos modificados y tienen como función evitar reacciones no deseadas mientras se lleva a cabo la síntesis (Ongay-Larios & Cabrera Mora, 2021).

3.1.4 Etapas de la síntesis química

La síntesis química de oligonucleótidos por el método de fosoramiditas, mostrado en la Figura 2, es un ciclo que consta de 4 etapas: desprotección, acoplamiento, encapsulado y oxidación. Debido a que es un proceso muy repetitivo y algo tedioso y complicado de realizar, actualmente existen máquinas sintetizadoras automáticas que se encargan de hacer todo el trabajo (Ponce Abarca, 2021).

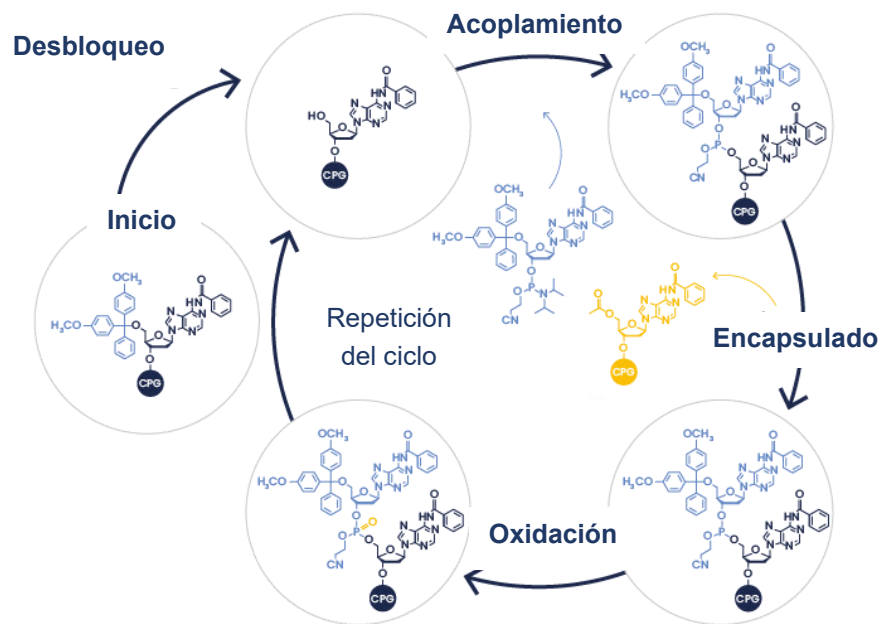


Figura 2. Ciclo de síntesis del método de las fosoramiditas. Extraída de: (*Oligo chemical synthesis – Custom oligo production | Eurogentec, s/f*)

Desprotección: El ciclo comienza cuando es removido el grupo protector, es decir, el dimetoxitritilo (DMT) del primer nucleósido unido al CPG mediante un tratamiento con ácido. Logrando que el hidroxilo del extremo 5', quede libre para unirse al siguiente nucleósido (Gallo, 2002).

Acoplamiento: Una vez que el nucleósido se activa, es momento de la adición del siguiente monómero, es decir, un nuevo nucleótido, a la cadena en compañía de un activador, el cual actúa como un ácido débil, logrando así, reaccionar con el hidroxilo

5' libre de la cadena anclada en el soporte sólido, uniéndose de forma covalente al nucleósido anterior, formando así el enlace (Vélez et al., 2003).

Encapsulado: Cualquier nucleósido sin acoplar es encapsulado para evitar otras reacciones. En promedio, menos del 2% de moléculas no reaccionan en el paso anterior, sin embargo, estas son bloqueadas permanentemente por medio de una reacción de acetilación del extremo 5'. Este proceso asegura que solamente las cadenas con secuencias correctas puedan crecer en los siguientes pasos (Beaucage & Reese, 2009).

Oxidación: Finalmente, el paso culminante del ciclo de síntesis consta de una reacción de oxidación para consolidar la cadena recién formada, esto se logra agregando una solución de yodo, colidina y agua en acetonitrilo, convirtiendo el fósforo trivalente a uno pentavalente, asegurando la estabilidad del enlace (Bermejo Martín, 2005).

Estas cuatro etapas se repiten para cada uno de los nucleótidos, hasta obtener la cadena de ADN que se requiere. Este procedimiento se efectúa en fase sólida y el soporte sólido (CPG) se encuentra contenido en las columnas de prueba.

3.1.5 Determinación de la eficiencia de acoplamiento de la síntesis química

La cuantificación permite estimar qué tan eficiente fue el acoplamiento en cada paso de incorporación y a su vez es un indicador preliminar confiable de la calidad de la síntesis. En solución ácida la molécula DMT produce un color naranja reluciente, y al diluirse es fácil detectarlo y cuantificarlo espectrofotométricamente. El rendimiento de la reacción de acoplamiento (R_x) en cada etapa es (1):

$$R_x = \frac{abs_x}{abs_{x-1}} \times 100\% \quad (1)$$

Donde

abs_x : es la lectura de absorbancia de la etapa x .

abs_{x-1} : es la lectura de absorbancia de la etapa anterior a x .

La diferencia entre estos valores indica la cantidad de oligonucleótidos libres que fallaron en acoplarse a la fosforamidita de la etapa siguiente (McBride & Caruthers, 1983).

Para obtener el rendimiento total de la síntesis (R_T), se puede emplear la expresión (2):

$$R_T = \frac{abs_n}{abs_2} \times 100\% \quad (2)$$

Donde

abs_n : es la última lectura.

abs_2 : es la segunda lectura.

Posteriormente, con el rendimiento total calculado, se obtiene la eficiencia promedio (E_p) con la ecuación (3):

$$E_p = (R_T)^{\frac{1}{N-1}} \quad (3)$$

Donde N es la longitud del oligonucleótido (Integrated DNA Technologies, 2011).

3.2 Programación lineal

El modelo de programación lineal es el método de investigación de operaciones más útil para la solución de problemas de optimización, el cual se compone de tres elementos fundamentales:

- a) **Variables.** Los datos encargados de la toma de decisiones que se pretenden determinar.
- b) **Objetivo.** Es la meta que se requiere optimizar (maximizar o minimizar).
- c) **Restricciones.** Son las limitaciones del sistema o proceso que se deben de considerar. (Almohamad & Duffuaa, 1993).

Por consiguiente, el primer paso para implementar un modelo de PL es identificar las variables de decisión, para posteriormente construir la función objetivo, partiendo de identificar si lo que se desea es maximizar o minimizar algún recurso,

como material, tiempo, ganancias, costos, etc. Para su desarrollo, es necesario encontrar relaciones entre las variables de decisión y así poder formarlas.

En el caso de las restricciones, están estructuradas como desigualdades o ecuaciones, y siempre deben ser lineales para que el sistema pueda ser resuelto por este método. De manera general, el método de la PL consta de encontrar los valores de las variables que satisfagan el conjunto de restricciones. A estos valores se les llaman soluciones factibles, pero la meta del problema es determinar la solución óptima, es decir, la mejor solución factible que maximice o minimice la función objetivo.

3.2.1 Método de solución gráfica

Para el caso de la solución gráfica de la PL, se siguen dos pasos:

1. Definir el espacio de factibilidad.
2. Determinar la solución óptima del espacio de factibilidad.

Para definir el espacio de factibilidad es necesario primero sustituir cada desigualdad por una ecuación, y posteriormente graficar cada una de las ecuaciones en un mismo plano. El efecto de la desigualdad de cada restricción es la guía para identificar cual espacio de las dos mitades que divide cada recta es la que se va a tomar en cuenta (Almohri et al., 2016).

Para determinar el lado correcto se selecciona (0,0) como punto de referencia, si este satisface la desigualdad, entonces el lado donde se encuentra será el semiplano factible. Este procedimiento se sigue para cada restricción, formando el espacio de soluciones factibles, donde todos los puntos que estén dentro o sobre el límite del área son factibles (Kästner & Langenbach, 2014).

Los vértices que forman el área de factibilidad son los posibles óptimos a maximizar o minimizar la función objetivo, únicamente se evalúa cuál de ellos provoca el valor más grande (maximización) o más pequeño (minimización) de la función objetivo, siendo esta la solución óptima. Por ejemplo, supongamos que se quiere maximizar la función objetivo $Z = 3x + 2y$, sujeta a las restricciones $x + 2y \leq 10$ y $2x + y \leq 12$, estas se “pasan” a igualdades para proceder a graficar como se muestra en la Figura 3, para identificar el área de factibilidad con ayuda de sus vértices A, B, C y D.

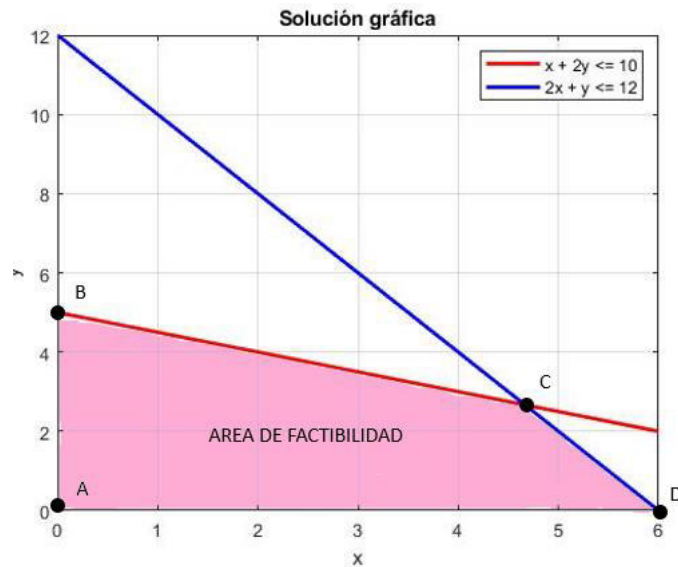


Figura 3. Ejemplo de método de solución gráfica.

3.2.2 Método de solución simplex

Para llevar a cabo el método de solución simplex se deben cumplir dos requerimientos en las restricciones de programación lineal.

1. Todas las restricciones son ecuaciones con lado derecho no negativo.
2. Todas las variables son no negativas (Del Rio Gomez, 2021).

Por lo tanto, el primer paso a desarrollar es convertir las desigualdades (\leq) en ecuaciones, esto se logra agregando una **variable de holgura** H al lado izquierdo de la restricción, por ejemplo, como se sigue en (4) (Hillier, 2015):

$$ax_1 + bx_2 \leq c \rightarrow ax_1 + bx_2 + H_1 = c \quad (4)$$

Donde a , b y c son constantes, x_1 y x_2 son las variables del sistema de PL, y H_1 es una variable no negativa de holgura.

Si es el caso de una restricción (\geq) significa un límite inferior en las actividades económicas de la programación lineal, así que la cantidad en la que el lado izquierdo excede el límite mínimo, lo cual se traduce en un superávit. Entonces su conversión a (=) se adquiere restando una **variable de sintética** S no negativa de lado izquierdo de la inecuación, un ejemplo se presenta en la ecuación (5) (Salas, 2017).

$$dx_1 + ex_2 \geq f \rightarrow dx_1 + ex_2 - S_1 = f \quad (5)$$

Una vez que se realizan las conversiones respectivas de las restricciones agregándose sus variables de holgura, se procede a formar la tabla inicial simplex (Tabla 2), la cual servirá de apoyo para las iteraciones de los cálculos.

Tabla 2. Tabla inicial simplex

Básica	z	x_1	x_2	H_1	S_1	Solución	
z	1	A	B	0	0	0	Fila z
H_1	0	a	b	1	0	c	Fila H_1
S_1	0	d	e	0	-1	f	Fila S_1

El número de variables de holgura y sintéticas agregadas está asociado con el número de restricciones. Posteriormente, la inicialización de iteraciones comienza con una solución básica factible (SBF), que es un conjunto de variables básicas tal que las demás son igual a cero, se selecciona una variable no básica (que no está en la SBF) para ingresar a la base y una variable básica para salir de la base (Manuel, 2016). Después se selecciona la variable no básica que puede entrar a la base para mejorar la solución y se identifica la variable básica que saldrá de la base para hacer espacio para la variable que entra. El siguiente paso consiste en actualizar la solución básica multiplicando la dirección simplex por el valor de entrada y sumándolo a la solución actual. Este proceso se repite hasta que no haya más mejoras posibles en la función objetivo. La solución resultante después de las iteraciones es la solución óptima del problema de programación lineal.

3.3 Redes Neuronales

Las redes neuronales artificiales, una rama de la inteligencia artificial, son un modelo computacional inspirado en el comportamiento que se observa en las neuronas del cerebro humano. Estas redes están conformadas por un conjunto de unidades de procesamiento llamadas neuronas artificiales, las cuales se encuentran entrelazadas unas con otras, donde los datos de entrada al momento de pasar por cada una de ellas son procesados secuencialmente y hasta obtener un conjunto de datos de salida (Maqueda Bojorquez, 2021). En la figura 4 se muestra el ejemplo de una red neuronal de 3 capas denominadas de entrada, intermedia/oculta y de salida. En este caso en la capa salida se tiene únicamente una neurona de salida.

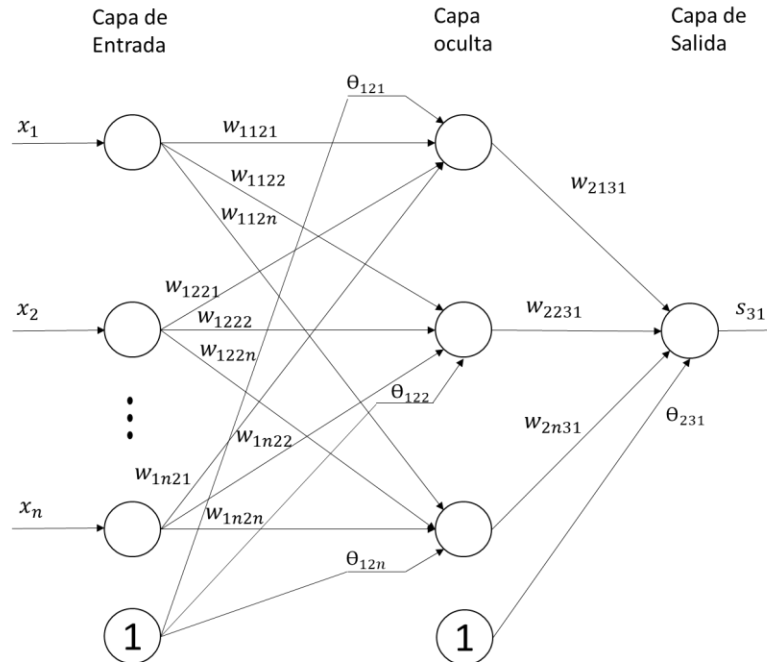


Figura 4.Arquitectura de una red neuronal

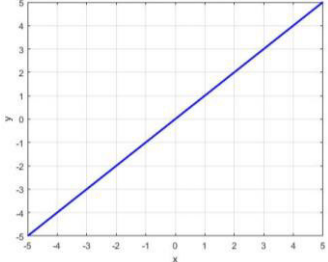
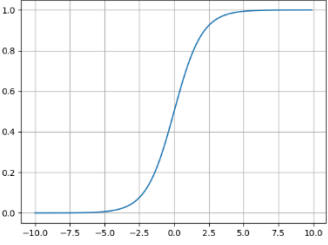
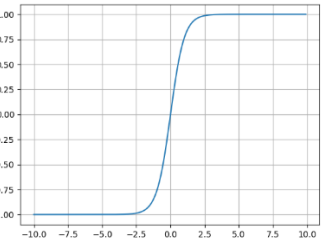
Los elementos más importantes que componen una red neuronal son los siguientes:

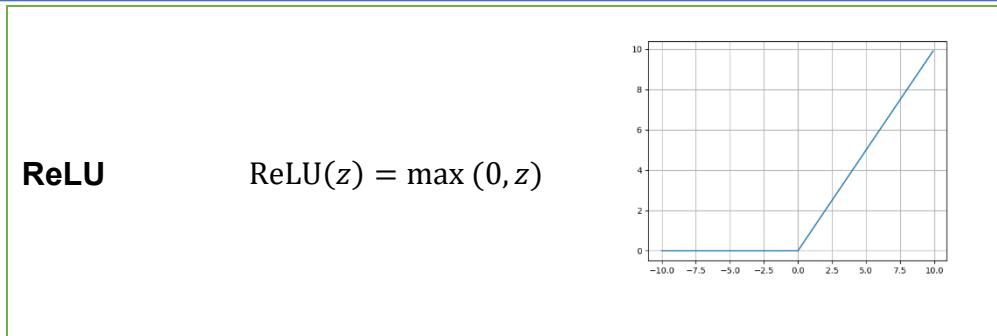
1. **Capa de entrada**, encargada de recibir los datos a procesar.
2. **Capas ocultas**, se encargan de todo el proceso de los datos. Las redes pueden tener una gran cantidad de capas ocultas, y cada capa analiza los datos que salieron de la capa anterior, la procesa en forma secuencial. El modelo más simple de red es el que contiene una sola capa oculta (Figura 4).
3. **Capa de salida**, donde se realiza la última etapa de procesamiento y se almacenan los datos ya procesados completamente, y por lo tanto son los resultados de la red neuronal.
4. **Pesos w** , son coeficientes que ponderan las conexiones entre las neuronas de una red. Estos pesos se adaptan dependiendo el proceso que se desee modelar y se obtienen aplicando un proceso de aprendizaje que minimiza el error entre la salida de la red y los valores de referencia. Normalmente los pesos iniciales son elegidos aleatoriamente (Golfe, 2021). Existen diferentes modelos de aprendizaje para las redes, uno de ellos es el llamado “back-propagation”.
5. **Bias (θ)** de una capa representa una neurona con un valor constante de 1 que se agrega a cada una de las capas de entrada e intermedias. Estas

tienen como efecto de agregar un “desplazamiento” en el argumento de las funciones de activación.

6. **Argumento de la función de activación z_{jI}** , en términos generales es la sumatoria de los productos de las salidas de las neuronas de una capa por los pesos que las conectan con una neurona de una capa posterior.
7. **Función de activación $\sigma_{jI}(z_{jI})$** , es una función que ayuda a que la red aprenda relaciones complejas entre los datos de entrada y salida. La elección de la función tiene gran impacto en la capacidad y rendimiento de la red neuronal, pueden utilizarse distintas funciones en diferentes partes del modelo. Por lo general todas las capas ocultas suelen utilizar la misma función de activación, pero todo depende del tipo de problema de aprendizaje. En la Tabla 3 se presentan algunas de las funciones de activación más comunes (Andrade Tepan, 2013).

Tabla 3. Funciones de activación más comunes.

Función	Definición	Gráfica
Lineal	$\sigma(z) = z$	
Sigmoide	$\sigma(z) = \frac{1}{1 + e^{-z}}$	
Tangente hiperbólica	$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	



La evaluación hacia delante de una red neuronal sirve para determinar la salida en función de una combinación de valores de las variables de entrada. Este proceso es secuencial y se va realizando capa por capa hasta llegar a la capa de salida.

Primero las salidas de las neuronas de la capa de entrada pueden denotarse como s_{1I} , donde los subíndices indican que se refieren a la I -ésima neurona de la capa de entrada ($J = 1$) y están definidos como:

$$s_{1I} = x_I$$

Mientras que las salidas de la I -ésima neurona de J -ésima capa ($J > 1$) estarían definidas por la expresión:

$$s_{JI} = \sigma_{JI}(z_{JI})$$

donde $\sigma_{JI}(z_{JI})$ es la función de activación de la neurona y su argumento z_{JI} puede calcularse por medio de la expresión:

$$z_{JI} = w_{(J-1)1JI}s_{(J-1)1} + w_{(J-1)2JI}s_{(J-1)2} + \dots + w_{(J-1)N_{(J-1)}JI}s_{(J-1)N_{(J-1)}} + \theta_{(J-1)JI}$$

Usando estas ecuaciones se puede ir calculando, de manera secuencial, la salida de las neuronas de cada capa hasta llegar a la neurona(s) de la capa de salida.

Por ejemplo, si se considera una red simple, con 2 neuronas en la capa de entrada, 2 en la capa intermedia y 1 en la capa de salida, se tendría que al agregar las neuronas de bias (con valor de 1), la red tendría 3 neuronas en las capas de entrada y en la intermedia. Para esta red los pesos que unen la neurona 1 de la capa de entrada con las neuronas de la capa intermedia se pueden denotar como: w_{1121} y w_{1122} . Por su parte, los pesos que unen la neurona 2 de la capa de entrada con las

neuronas de la capa intermedia son: w_{1221} y w_{1222} . Adicionalmente, se tendrían los pesos que conectan la neurona de bias de la capa entrada con las neuronas de la capa intermedia, que estarían definidos como: θ_{121} y θ_{122} . Por ultimo los pesos que unen a la neurona de bias de la capa 2 con la de la capa de salida sería θ_{131} . Con esta información y definiendo la función de activación que se desea utilizar, se puede estimar las salidas de las neuronas de la capa intermedia. Para la neurona primera neurona ($I = 1$) de la capa intermedia ($J = 2$) se tendría que el argumento de su función de activación es:

$$\begin{aligned} z_{21} &= w_{1121}s_{11} + w_{1221}s_{12} + \theta_{121} = [w_{1121} \quad w_{1221} \quad 1] \begin{bmatrix} s_{11} \\ s_{12} \\ \theta_{121} \end{bmatrix} \\ &= [w_{1121} \quad w_{1221} \quad 1] \begin{bmatrix} x_1 \\ x_2 \\ \theta_{121} \end{bmatrix} \end{aligned}$$

y la salida de la neurona estaría dado por:

$$s_{21} = \sigma_{21}(z_{21})$$

Siguiendo el mismo procedimiento se obtiene la salida de la neurona $I = 2$ de la capa intermedia ($J = 2$):

$$\begin{aligned} z_{22} &= w_{1122}s_{11} + w_{1222}s_{12} + \theta_{122} = [w_{1122} \quad w_{1222} \quad 1] \begin{bmatrix} s_{11} \\ s_{12} \\ \theta_{121} \end{bmatrix} \\ &= [w_{1122} \quad w_{1222} \quad 1] \begin{bmatrix} x_1 \\ x_2 \\ \theta_{121} \end{bmatrix} \end{aligned}$$

y la salida de la neurona estaría dado por:

$$s_{22} = \sigma_{22}(z_{22})$$

Una vez que se determinan las salidas de las neuronas de la capa $J = 2$, se pueden calcular las salidas de las neuronas de la siguiente capa ($J = 3$), que en nuestro ejemplo es la capa de salida con solo una sola neurona $I = 1$. Para este caso se tiene que:

$$z_{31} = w_{2131}s_{21} + w_{2231}s_{22} + \theta_{231} = [w_{2131} \quad w_{2231} \quad 1] \begin{bmatrix} s_{21} \\ s_{22} \\ \theta_{231} \end{bmatrix}$$

y la finalmente la salida de la neurona, y de la red, estaría definido por:

$$s_{31} = \sigma_{31}(z_{31})$$

Cabe mencionar que, en las redes neuronales convencionales, normalmente se elige la misma función de activación para todas las neuronas de las capas intermedias y la de la capa de salida puede cambiarse dependiendo del proceso que se este desarrollando. Por ejemplo, normalmente cuando se usa la red para efectuar una regresión se elige la función lineal como función de activación de la neurona de salida.

Existen diversas formas de clasificar las redes neuronales, puede ser por su función, por su arquitectura, por la forma en la que los datos recorren la red, tipo de entrenamiento, etc. Sin embargo, cada una tiene características específicas y la elección del tipo de red que se utilice depende del proceso a realizar.

En este trabajo se utiliza una red neuronal recurrente, la cual se diferencia por la importancia del tiempo en los datos, es decir, el estado del sistema actual depende del estado del paso anterior inmediato, la información se guarda en la memoria de las neuronas traducido al “ciclaje” o “recurrencia” de la información entre las neuronas de la capa de salida y las neuronas de la capa de entrada (Arana, 2021).

4 METODOLOGÍA

La metodología que se siguió para lograr los objetivos planteados para este proyecto de tesis se puede desglosar en 4 etapas, como se muestra en la Figura 5.

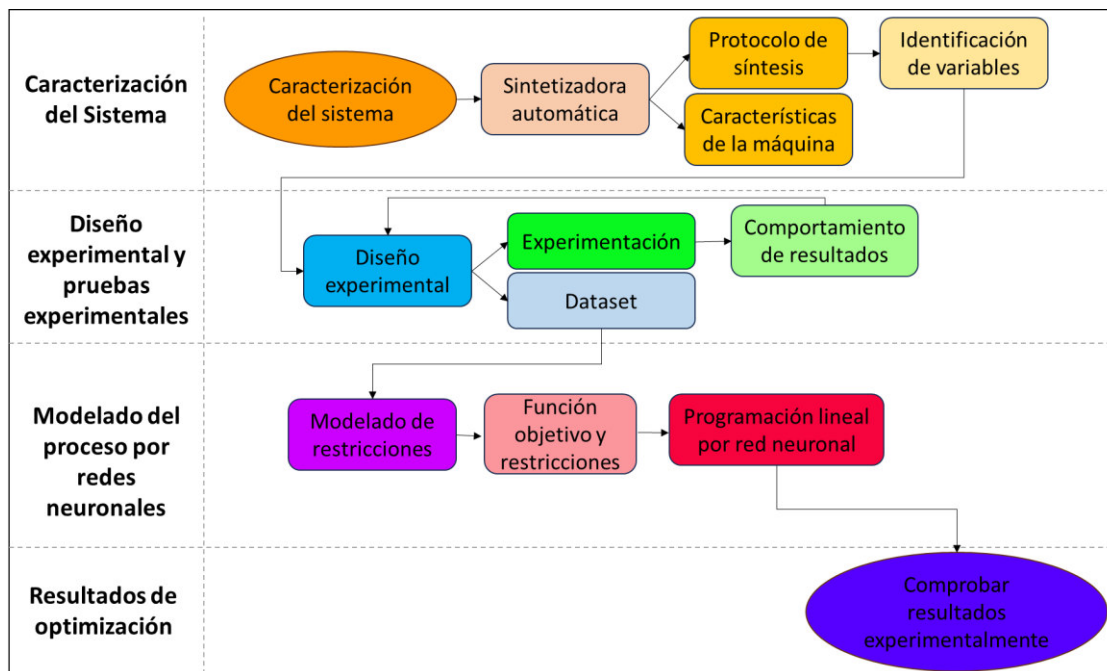


Figura 5. Metodología seguida para desarrollar el proyecto presentado en este trabajo

4.1 Caracterización del sistema

El primer paso para realizar el proyecto fue conocer a detalle el proceso de síntesis química de oligonucleótidos, tanto las características de la máquina, funcionamiento y lineamientos; características de las columnas de prueba, identificación de los reactivos que ocupa, sus volúmenes, entre otras cosas. Para conocer todo lo necesario para desarrollar el trabajo, se manipuló personalmente la máquina, desde el inicio hasta el fin del proceso. Así se logró identificar las restricciones en cuanto a capacidad de equipo y variables necesarias para llevar a cabo la síntesis.

Debido a que no únicamente se debe considerar la máquina, sino también los protocolos de síntesis donde se encuentran todos los parámetros y etapas de la síntesis, es por ello por lo que se extrajeron todas las variables presentes en dicho código para su posterior análisis y así poder identificar las variables y restricciones para modelar el diseño experimental. Cabe resaltar que la base fundamental para

toda la metodología son los protocolos de síntesis ya existentes para escala de 100 nmol y el procedimiento que conlleva este.

4.1.1 Características del equipo y columnas de prueba



Figura 6. Fotografía de la máquina sintetizadora automática ABI

La sintetizadora automática *ABI 3900* de Biolytic Lab Performance (Figura 6), es la máquina con la que se desarrolla la producción de oligonucleótidos en la empresa T4 Oligo. Esta funciona por medio de protocolos o scripts estructurados en formato de códigos que son cargados a su sistema a través de una interfaz y así fabricar la cantidad de oligonucleótidos deseados en formato de columnas.

El equipo cuenta con 3 elementos principales: cámara de síntesis, puertos de inyección y contenedores de reactivos. La cámara de síntesis mostrada en la Figura 7 tiene 4 bancos, cada uno con capacidad para 12 columnas de prueba, teniendo en total una capacidad de trabajo simultaneo de 48 columnas, además está diseñada para rotar sobre su propio eje, para lograr que se posicione cada columna debajo de la línea perteneciente a cada reactivo. Por ejemplo, si se quiere dispensar el reactivo llamado Deblock, el cual pertenece a la línea 1, entonces el rotor se encarga de mover cada columna hasta posicionarlas debajo de la línea 1 para que todas sean llenadas de este reactivo, y así sucesivamente con cada línea.

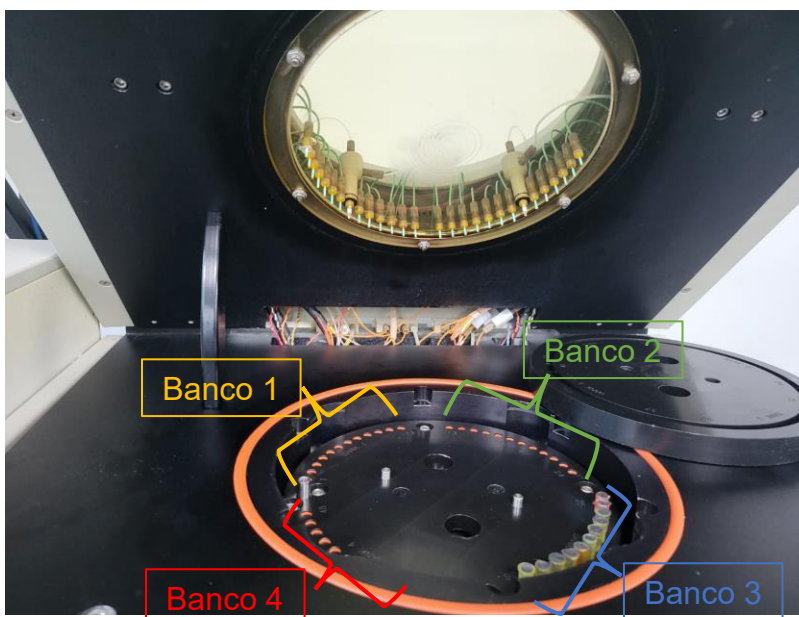


Figura 7. Cámara de síntesis y puertos de inyección

Los puertos de inyección son mangueras, de diámetro muy reducido, las cuales son las encargadas de suministrar el reactivo requerido a la cámara de inyección, cada línea cuenta con válvulas de precisión que controla el paso de la cantidad necesaria de reactivo a dispensar. Es una línea por cada reactivo, estas están fijas y no tienen movimiento.

Los contenedores, están diseñados en forma de botellas, son los encargados de almacenar todos los reactivos. Son de diferente capacidad, dependiendo del químico que contengan. Es posible transportar todos los reactivos gracias a las líneas de los puertos de inyección, debido a que en el extremo contienen una pequeña esponja que absorbe el químico y lo dispensan a la cámara de síntesis.

Las columnas de prueba son diferentes en tamaños, capacidad y material, dependiendo de la escala y tipo de síntesis a la que se trabaje. Estas, contienen un soporte sólido llamado frit hecho comúnmente de un material llamado CPG, el cual se encuentra entre dos filtros que permiten el flujo de los reactivos sin que el frit se salga y permanezca en la columna. En la Figura 8 se muestra la comparativa entre

una columna para escala de 100 nmoles de tipo universal y otra para escala de 5 nmoles.

100 nmol	5 nmol
<input type="checkbox"/> Material: CPG	<input type="checkbox"/> Material: CPG con Poliestireno
<input type="checkbox"/> Capacidad: 300 μ l	<input type="checkbox"/> Capacidad: 200 μ l
<input type="checkbox"/> Tamaño del frit: 1/8 in x 0.163 in	<input type="checkbox"/> Tamaño del frit: 2 mm x 2 mm
<input type="checkbox"/> Presión de drenado: 2.2 psi	<input type="checkbox"/> Presión de drenado: 3 a 5 psi




Figura 8. Características entre columnas de prueba de escalas de 100 nmoles universal y 5 nmoles

Cabe mencionar que dependiendo del material del que esté hecho el frit se puede retener o dejar pasar más rápido el reactivo, debido a la porosidad de cada material. Este factor afecta directamente a las variables de tiempos de drenado y de reacción, así como su presión de drenado. Además, para lograr un correcto proceso de síntesis, el entorno debe estar rigurosamente controlado, es decir, se debe garantizar idealmente la ausencia de humedad. Para tratar de mantener controlada esta variable se le suministra nitrógeno (también se puede utilizar argón), el cual es el encargado de generar un ambiente libre de humedad, tanto en las botellas contenedoras de reactivos, así como en la cámara de síntesis; además de drenar los reactivos una vez que se cumplió su tiempo de reacción y pasar a otra etapa de síntesis, logrando prevenir que se acumulen los químicos en las columnas y evitar derrames en toda la cámara.

Además, es importante mencionar que el equipo está conectado a una computadora con el software de la empresa Biolytic donde se tiene la interfaz de manejo del equipo y con el que se cargan, editan o crean los protocolos, así como llevar a cabo la calibración y asegurar el correcto funcionamiento de las válvulas, como lo es el dispensar exactamente la cantidad especificada en el lugar correcto de cada columna. El software asimismo permite cargar la estructura de diferentes oligos, con cualquier cantidad de bases con las que se desee trabajar. Al inicio de cada síntesis,

se le especifica los oligos con los que se trabajará y el equipo automáticamente repite el protocolo la cantidad de bases que tenga cada oligo.

4.1.2 Protocolos de síntesis o scripts

Los protocolos de síntesis o scripts son códigos modificables para controlar las funciones del equipo, constan de ciclos que equivalen a cada una de las etapas de la síntesis química, en cada ciclo se manejan variables como tiempos de drenado, tipos de drenado, tiempos de espera y volúmenes de sustancias; cada una adecuada a los requerimientos de cada etapa, como lo es la cantidad de veces que se repite una variable y qué posición ocupa.

Se tiene un protocolo diferente para cada escala y tipo de síntesis, es decir, los volúmenes tienen cantidades específicas para cada etapa, los tipos de drenado son utilizados en ciertas partes del código dependiendo de qué tipo de columna se use, las veces que se repite cada etapa. También, cuenta con el comando encargado de mover automáticamente a la posición de cada columna y abrir las válvulas de cada reactivo cuando se le indique en el código, y cerrarla automáticamente una vez que dispensó la cantidad indicada.

4.1.3 Procedimiento de síntesis

El procedimiento que a continuación se describe, es el que se llevó a cabo para realizar la síntesis química de oligonucleótidos establecida en cada una de las pruebas experimentales. Este se divide en 3 fases: a) check list, b) síntesis y c) postproceso; es importante resaltar que a lo largo de este trabajo se omiten diversos tipos de datos e información por cuestiones de confidencialidad de la empresa.

Antes de comenzar, cualquier persona que manipule el equipo debe de contar con una bata abotonada totalmente, pantalón de mezclilla, zapatos especiales, guantes de látex, cofia, cubrebocas y sin ningún tipo de accesorio colgante.

4.1.3.1 *Check list*

Esta fase consiste en 5 pasos, los cuales se describen a continuación:

1. Se verifica que los niveles de cada reactivo en los contenedores sea el suficiente para llevar a cabo la síntesis y que sean recientes, debido a que después de cierto tiempo los químicos suelen oxidarse provocando que pierdan funcionalidad para el procesamiento, es por ellos que se mantiene una bitácora de cada cuando se van cambiando los reactivos. Al momento de hacer los cambios de reactivos nuevos, en cada contenedor, además se

agregan pequeñas capsulitas encargadas de inhibir la humedad, llamados “chochitos”.

2. Se verifica que el nivel del tanque de nitrógeno, para que durante el proceso de síntesis no se quede sin gas y arruine el desarrollo de los oligonucleótidos, provocando también generar desperdicios, tanto de columnas de prueba como de reactivos y tiempo, entre otros. Esto porque sin gas para drenar las columnas estas pueden comenzar a llenarse y a derramar los reactivos, dañando así el equipo.
3. Se verifica que el contenedor donde son ingresados los desechos no esté lleno, ya que esto evita que, en lugar de que fluyan sin problemas los excesos químicos, se regresen a la cámara de síntesis e inunden las columnas y dañen no solo el proceso sino también el equipo.
4. Se comprueba que la presión de drenado en el equipo sea el adecuado, este depende de la escala a la que se va a trabajar o tipo de síntesis, para realizar este paso, el equipo primeramente debe estar cerrado al vacío, y con ayuda de una pantalla (display) se indica la presión, comenzamos a mover la perilla que controla la presión, ya sea para subir o bajarla. Una vez que el display marca la presión deseada, en el equipo se lleva a cabo un drenado completo para comprobar que efectivamente se mantiene la presión.
5. Se verifica que los puertos de inyección no están tapados y dispensan bien cada reactivo, para esto, se coloca un recipiente para que contenga el reactivo que se va a dispensar, y con ayuda de la interfaz del equipo se le indica que abra cada uno de los puertos de uno por uno, y así poder comprobar que funcionan correctamente. Si es el caso de que alguna de las líneas no dispensa reactivo, con ayuda de un dispositivo especial se destapa, el cual se introduce cuidadosamente dentro de la línea y posteriormente se extrae.

4.1.3.2 Síntesis

Esta fase consiste en 7 pasos, los cuales se describen a continuación:

1. Se colocan las columnas de prueba en el banco de prueba que se quiere trabajar, para llevar a cabo las pruebas experimentales de este trabajo de tesis, se utilizaron las primeras 6 posiciones del banco 1, y las 6 posiciones restantes son llenadas con tapones, ya que, de no ser así, ocasiona que el equipo no drene de manera adecuada. Para los 3 bancos restantes, se indica en la interfaz que los deshabilite, y así no es necesario de llenarlos de tapones. Los tapones son columnas de prueba, pero selladas completamente por los extremos para evitar que dejen pasar gas.
2. Se edita el protocolo de síntesis, para cada prueba. En este caso se trabajaba con el mismo protocolo, es decir, la misma estructura, únicamente se editaban los valores de los volúmenes y los tiempos para cada etapa, posteriormente se guardaba con un nombre con el que se pudiera identificar y consultar después si era necesario. Los valores que se modificaban eran previamente calculados, para así únicamente vaciarlos al protocolo manualmente en la sección de edición de protocolo de la interfaz.

Los siguientes pasos se realizan en la sección de síntesis de la interfaz de equipo:

3. Se selecciona el banco en el que se colocaron las columnas de prueba para ahí proseguir a cargar el protocolo y los oligos.
4. Se carga el protocolo, una vez que ya se editó, se cargaba el script y se verificaba que estuviera correcto, tanto en estructura como en información.
5. Se cargan las secuencias de oligonucleótidos que se van a sintetizar, para este paso, únicamente se accedía a una carpeta del equipo, donde se encuentran las secuencias previamente guardadas, y se seleccionaban las secuencias con las que se iban a trabajar, para fines de este proyecto, siempre se trabajó con la misma secuencia de oligonucleótidos. La ventana

para cargar los oligos es una tabla con 48 renglones, cada uno perteneciente a la posición de cada columna de prueba, es decir, si se trabaja con las primeras 6 posiciones del primer banco, entonces se llenaban los primeros 6 renglones con las secuencias deseadas, y los demás renglones se dejaban vacíos, así el equipo interpretaba como que únicamente se va a dispensar y drenar líquido en las primeras 6 posiciones de columnas de prueba.

6. Se deshabilitan los tres bancos de prueba restantes, ya que no se van a utilizar, así el equipo interpreta que están vacíos.

7. Al seleccionar el botón de iniciar síntesis, el equipo comienza el proceso, y en la pantalla de la interfaz indica en qué paso del script se encuentra, y el tiempo que tardará en completar la síntesis. Era muy importante estar observando el proceso, para evitar que hubiera derrames por mala elección de la presión de drenado o cantidad de volúmenes a dispensar.

4.1.3.3 Postproceso

Esta fase consiste en 3 pasos, los cuales se describen a continuación:

1. Una vez que el equipo indica que ha terminado, se extraen las columnas de la cámara de síntesis y se colocan en una tablilla de pruebas, para que puedan ser manipuladas de mejor forma en el postproceso.

2. Los empleados de la empresa se encargaban del postproceso, el cual ya está establecido por ellos, donde resumidamente, agregan diversos reactivos y llevan a una cámara donde exponen la solución a altas temperaturas por un determinado tiempo, todo esto para poder cuantificar el rendimiento. Cabe mencionar que siempre se hizo igual para todas las pruebas, para no afectar y agregar variables de más a los experimentos.

3. Los resultados de las pruebas son entregados días después de que se realizó, debido a la gran demanda de producción, para posteriormente vaciarlos a una base de datos, la cual sirvió para llevar a cabo los análisis necesarios.

4.2 Diseño y pruebas experimentales

Para la realización del diseño experimental se requiere identificar todas las variables presentes en el proceso y determinar cuales tienen mayor peso en el sistema. En nuestro caso este proceso de identificación lo realizamos con la ayuda del conocimiento de expertos que trabajan en la empresa.

4.2.1 Identificación de variables

Durante el proceso de síntesis química de oligonucleótidos, se deben de considerar dos factores primordiales: la máquina y el protocolo. Cada uno con diferentes variables involucradas, estas se listan en la Tabla 4.

Tabla 4. Variables involucradas en el proceso de síntesis química

Factor	Variables
Máquina	<ul style="list-style-type: none"> • Presión de drenado • Tipos de drenado • Cantidad de dispensadores • Cantidad de contenedores de reactivos • Tipos de columnas de prueba • Capacidad de columnas de prueba • Cantidad de columnas de prueba
Protocolo	<ul style="list-style-type: none"> • Escala de síntesis • Etapas de síntesis • Repeticiones de cada etapa de síntesis • Número de ciclos • Tipo de oligonucleótido • Bases de oligonucleótido • Tipo de reactivo • Volumen de reactivo • Tiempo de drenado • Tiempo de reacción

Debido a la amplia cantidad de variables involucradas en todo el proceso, se consideraron únicamente 3 variables, que denominamos como principales: **volumen, presión y tiempo**. En la Tabla 5 se muestran los valores de estas tres variables, los cuales fueron elegidos a prueba y error debido al poco conocimiento del comportamiento del sistema al momento de alterar las cantidades de sus factores, incluyendo el desconocimiento de qué tanto se podía variarlas y de que valores partir. Cabe aclarar que las variables que se omitieron de la Tabla 4, permanecieron constantes para no complicar el proceso de diseño experimental, y

sus valores se mantuvieron iguales a los de la configuración que se utiliza en el protocolo de síntesis química para escala de 100 nmoles. Es importante resaltar que para fines de este estudio no se toma en cuenta la estructura del protocolo, sino los valores de las variables que integra cada fase, teniendo como objetivo acertar las cantidades necesarias de cada variable en cada etapa para conseguir un rendimiento final de 3 a 5.

Tabla 5. Variables consideradas para la realización del diseño experimental con sus respectivos valores

VARIABLE	VALOR
Volumen (V)	2.92 ml
	47.52 ml
	31.68 ml
	15.84 ml
	126.72 ml
	28.80 ml
	36 ml
Presión (P)	21.6 ml
	3 psi
	3.5 psi
Tiempo (T)	5 psi
	3.775 horas
	3.475 horas
	3.375 horas
	3.275 horas
	3.075 horas

4.2.2 Volumen de reactivos (V)

El volumen de reactivos, el cual se consideró tendría mayor relevancia sobre el sistema y en la relación de aumento o disminución del rendimiento. Los valores hacen referencia a la cantidad total de volumen que se utiliza de una sola sustancia durante todo el proceso de síntesis, cabe resaltar que para cada sustancia se utilizan diferentes cantidades, pero en este caso, se toma de referencia el reactivo de Deblock. Cada cantidad que se presenta en la tabla 5 tiene un identificador de relación para las demás sustancias.

Para cada valor de V se tiene una cantidad v específica a dispensar en cada repetición del protocolo de síntesis, es decir, si se tiene un oligo de 15 bases entonces el ciclo de síntesis se repite 15 veces, y por lo tanto la cantidad v de deblock se dispensa 15 veces, y eso únicamente para una columna de prueba,

además debe considerarse el número de columnas a utilizar, es así como se obtuvo los valores de V , multiplicando v por el número de bases y por la cantidad de columnas que se quieren obtener.

Dicho esto, experimentalmente hablando, se contempló la misma cantidad de bases y de columnas para todas las pruebas experimentales y evitar así variables innecesarias para este trabajo de estudio. Adicionalmente, una de las consideraciones para elegir los valores del volumen fue la capacidad máxima de las columnas de prueba, para no sobrepasarlo evitando que se derramara el reactivo dentro de la máquina, asimismo el volumen mínimo que es capaz de dispensar la máquina.

4.2.3 Tiempo de proceso (T)

En el caso del tiempo (T), es el número de horas totales en completarse el proceso, a diferencia de lo establecido en la variable de V , en T sí se consideran los tiempos de todas las etapas. El total se obtuvo sumando los tiempos de todas las etapas y multiplicándose por la cantidad de bases y por el número de columnas colocadas en la cámara de síntesis. Igual que V , cada etapa tiene su tiempo específico, el cual es indicado en el protocolo.

4.2.4 Presión de drenado (P)

En cuestión de la presión, está definida en unidades de libras por pulgada cuadrada (psi) y es con la que la máquina drena el exceso de reactivos al finalizar cada etapa.

4.3 Diseño y pruebas experimentales

Una vez que se van obteniendo resultados de rendimiento de las pruebas elaboradas, pero permanecen bajos, se modifican los valores de las variables establecidas, hasta que se identifica que los resultados comienzan a tener relación de cambio conforme se modifican los valores de las variables. Se realizó este proceso, (Figura 9), debido a que se desconocía qué variables afectan más al rendimiento final de las pruebas y además que no existe ninguna manera de saberlo con certeza de forma a priori. Cabe mencionar que los protocolos que se usan en la

empresa fueron definidos a prueba y error, además no existe mucha información de cómo se desarrollan estos protocolos porque las empresas mantienen su elaboración como secretos industriales.

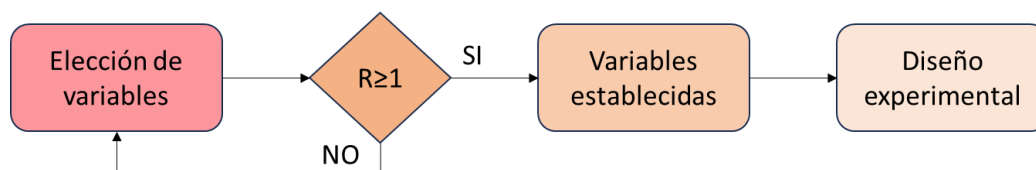


Figura 9. Diagrama de flujo que explica el proceso que se siguió para la elección de los valores

4.3.1 Escenarios experimentales y dataset

En total se realizaron 13 escenarios diferentes, con 6 réplicas cada uno, considerando como respuesta el valor promedio de rendimiento total (R_T) y porcentaje de aprobados (PA) en cada escenario, siendo 100% equivalente a 6 réplicas aprobadas. En la Tabla 6 se muestran los resultados arrojados de cada escenario realizado en la máquina sintetizadora. En los escenarios numerados del 1 al 10 de la Tabla 6, el tiempo permaneció constante, esto debido a que los expertos de la empresa consideraban que los tiempos de reacciones eran constantes sin importar la escala de síntesis, basándose en sus conocimientos adquiridos a lo largo de su carrera como especialistas bioquímicos. Sin embargo, al examinar los resultados de rendimiento total se observó que en la mayoría de los casos se obtenían niveles muy por debajo de 3. Por lo que se convenció a los colaboradores de la empresa de la posibilidad de que si se modificaban los tiempos de las reacciones se podían obtener mejores resultados, y en efecto, así fue.

A partir de los primeros resultados presentados en la Tabla 6 se desarrolló un diseño de experimentos, considerando la relación que tenía la alteración de los valores de las variables sobre el resultado de rendimiento, es decir, donde se obtuviera de rendimiento un valor igual o mayor a 3, que es el caso de la configuración número 9, 12, 13 y 14, de las cuales, tienen en común que se dispensó 21.6 ml de volumen de reactivos en cada etapa y una presión de drenado de 3.5 psi, con diferencia en sus valores de tiempo de reacción de cada etapa como se mencionó anteriormente.

Tabla 6. Resultados obtenidos de los escenarios experimentales

#	Volumen (ml)	Presión (psi)	Tiempo (h)	R_T	PA
1	126.72	3	3.475	0	0%
2	31.68	3.5	3.475	1.1175	0%
3	15.84	3	3.475	0.2491	0%
4	31.68	3	3.475	0.4525	0%
5	47.52	3	3.475	1.2275	0%
6	7.92	3	3.475	0.0975	0%
7	28.8	3.5	3.475	0.7933	0%
8	36	3.5	3.475	1.8566	0%
9	21.6	3.5	3.475	2.665	33.3%
10	21.6	5	3.475	1.5266	0%
11	21.6	3.5	3.375	3.76	83.3%
12	21.6	3.5	3.075	3.3283	100%
13	21.6	3.5	3.775	3.9133	100%

En la Tabla 7 se muestran los factores y niveles para la realización del diseño experimental basado en el tipo factorial completo 3^k , con 6 réplicas, esto debido a que se opta por mantener constante la presión con un valor de 3.5 psi, fundamentado en los resultados obtenidos de la Tabla 6, en consecuencia, el valor de k es 2, por el Volumen y Tiempo de reacción como factores, con sus 3 respectivos niveles cada uno.

Tabla 7. Factores y niveles de las variables estudiadas

	Variable	Niveles
Factor A	Volumen (ml) de reactivos utilizados en síntesis química, representado en porcentaje de los volúmenes usados en el protocolo de escala de 100 nmoles.	14.4
		21.6
		36
Factor B	Tiempo de proceso (h) utilizado en síntesis química, representado en porcentaje de los tiempos usados en el protocolo de escala de 100 nmoles.	3.075
		3.375
		3.775

Por consiguiente, se tienen 9 corridas experimentales, siendo todas las posibles combinaciones que se pueden ejecutar partiendo de los factores y niveles seleccionados. A continuación, en la Tabla 8 se muestran los resultados obtenidos de la experimentación, donde se puede observar que cuando se tiene un mayor volumen se requiere más tiempo de reacción, de lo contrario el rendimiento es más bajo al igual que el porcentaje de aprobados.

Todos los resultados obtenidos, presentados en las Tablas 6 y 8, fueron relativamente aceptables, por lo que todos los datos de todas las combinaciones son utilizados para formar el dataset con el que se modelarán las restricciones del proceso.

Tabla 8. Resultados obtenidos del diseño factorial 3^k para el proceso de síntesis química

#	Volumen (ml)	T (hr)	R	PA
1	14.4	3.075	3.0867	66.67%
2	21.6	3.075	3.3333	100%
3	36	3.075	2.7783	33.33%
4	14.4	3.375	3.92	83.33%
5	21.6	3.375	3.4617	100%
6	36	3.375	2.9767	33.33%
7	14.4	3.775	3.9	100%
8	21.6	3.775	3.37	83.33%
9	36	3.775	2.99	33.33%

4.4 Modelado del proceso: función objetivo y restricciones

4.4.1 Función objetivo

Para el modelado de la función objetivo, que en este caso será función de ganancia de la empresa cuando se cumple un rendimiento mayor a 3, se consideran 4 secciones de costos principales para llevar a cabo este proceso:

CNS: Costo de nitrógeno por síntesis

CES: Costo de energía eléctrica por síntesis

CRS: Costo de reactivos por síntesis

CGS: Costo de ganancias por síntesis

a) Costo de nitrógeno por síntesis (CNS)

La máquina sintetizadora que lleva a cabo el proceso requiere de tanques de nitrógeno para drenar el sobrante de sustancia que queda en las columnas al finalizar cada etapa de síntesis, ya que esto impide la acumulación de reactivos, evitando combinarse entre ellos y ser derramados dentro de la cámara sintetizadora.

Para este proceso, el nitrógeno es uno de los elementos más importantes para llevar a cabo la síntesis química, por lo que su costo es tomado en cuenta para calcular el costo del proceso.

La función de *CNS* se puede definir como el producto de una constante con unidades de volumen por hora por presión, por la presión (*P*) de drenado, el tiempo total (*T*) que dure la síntesis y el costo del metro cubico de nitrógeno, lo cual puede ser expresado con la ecuación (6):

$$\begin{aligned} CNS(\$) &= 0.1318 \left[\frac{\text{m}^3}{\text{h} \cdot \text{psi}} \right] \cdot P[\text{psi}] \cdot T[\text{h}] \cdot 223.5632 \left[\frac{\$}{\text{m}^3} \right] \\ &= 29.4701PT[\$] \end{aligned} \quad (6)$$

La constante de 0.1318 fue proporcionada por la empresa, el cual sus expertos han estimado de manera empírica de sus procesos de producción. Es importante hacer notar que la función del *CNS* es no lineal, debido a que implica el producto de dos variables lineales *P* y *T*, lo que implica que es una función de 2 grado.

b) Costo de energía eléctrica por síntesis (*CES*)

La gran cantidad de tiempo que duran las rondas de síntesis implica un consumo de energía eléctrica, lo cual conlleva a un costo que está sujeto a las tarifas de la Comisión Federal de Electricidad (CFE), que es la empresa que brinda el servicio de suministro de energía eléctrica. Es por ello por lo que se considera la potencia consumida por el equipo y la tarifa de uso industrial de energía eléctrica para hacer posible el cálculo. En el manual de la máquina se especifica que requiere una intensidad de corriente eléctrica de 5A a una tensión de 250V, de acuerdo con la fórmula de potencia (*W*) que se obtiene de multiplicar corriente eléctrica (Amperes) por la tensión eléctrica suministrada, se obtiene que la potencia del equipo es de 7.5kW, por lo que el costo de electricidad por ronda de síntesis puede expresarse con la ecuación (7)

$$CE = 1.25[\text{kW}] \cdot T[\text{h}] \cdot 3 \left[\frac{\$}{\text{kW} \cdot \text{h}} \right] = 3.75T[\$] \quad (7)$$

c) Costo de reactivos por síntesis (CRS)

Los datos de los costos de todas las sustancias utilizadas durante el proceso de síntesis fueron proporcionados por la empresa de acuerdo con sus proveedores. El costo de la botella de reactivo con 4000 ml tiene un costo de \$3392, entonces para calcular el costo por mililitro de reactivo, para posteriormente este valor multiplicarlo por la cantidad de volumen utilizado en una ronda de síntesis se expresa en la ecuación (8)

$$CR = \frac{3392}{4000} \left[\frac{\$}{ml} \right] \cdot V[ml] = 0.848V[\$] \quad (8)$$

d) Costo de ganancias por síntesis (CGS)

Un oligonucleótido es aceptado cuando el rendimiento final es mayor o igual a 3, entonces para representar que mientras más alto el valor del rendimiento final, más ganancias, se opta por multiplicarlo por una constante como se declara en la ecuación (9)

$$CGS = 1000R[\$] \quad (9)$$

Desarrollado los costos anteriores, la función objetivo z se conforma de restar CNS , CES y CRS de CGS representado en la formula (10)

$$z = -0.848V - 3.75T - 29.4701PT + 1000R \quad (10)$$

En términos generales esta función objetivo no se puede minimizar o maximizar empleando el método simplex, debido a que la función objetivo es no lineal. En este caso se pudiera emplear un método de programación cuadrática o algún método de inteligencia artificial.

4.4.2 Modelado de restricciones

Debido a que las pruebas experimentales presentaron resultados aceptables únicamente con una presión de drenado de 3.5 psi, se decide establecer la variable de presión como una constante, dicho esto, las pruebas realizadas con esta presión son las consideradas para modelar el espacio factible del sistema, para ello, se grafican los experimentos en un plano 3D, donde el eje x representa el volumen, el eje y el tiempo y el eje z el rendimiento. En la figura 10 se muestran dichos experimentos con sus respectivas replicas cada uno, cada punto equivale a una prueba.

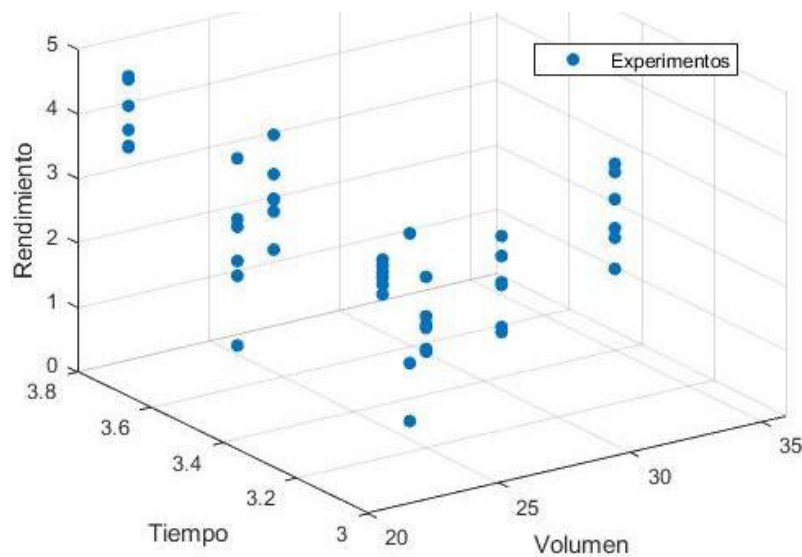


Figura 10. Grafica de los experimentos realizados con 3.5 psi

Posteriormente, se buscan los planos que encierren todos los puntos, para así formar el espacio de factibilidad, la ecuación de cada plano se considera como una restricción, y dependiendo de si el plano está por debajo o por encima de los puntos es el signo de desigualdad que se le asigna. En la Figura 11 se muestran los planos que aíslan los experimentos con rendimiento mayor o igual a 3 de los que no cumplen esta condición.

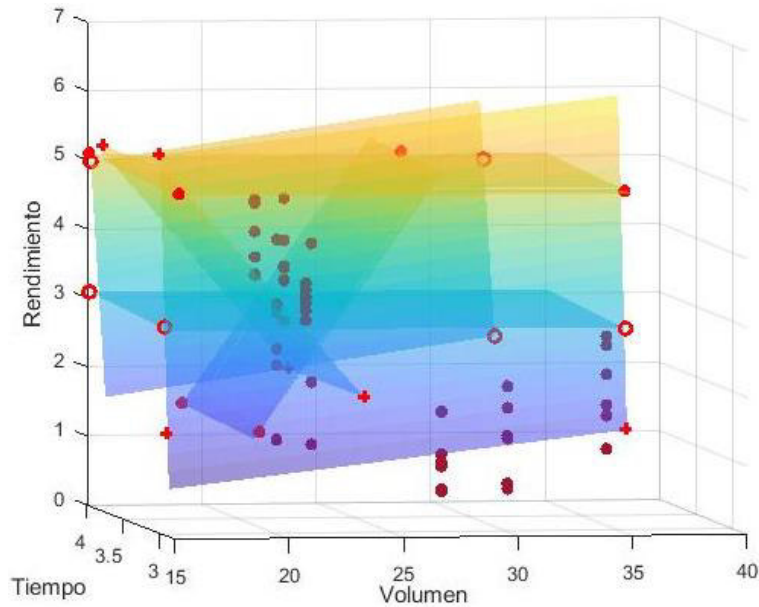


Figura 11. Grafica de los planos que forman el espacio de factibilidad.

Las ecuaciones de cada uno de los planos que acotan el espacio del volumen factible son las siguientes:

$$0V + 80T - 1.6R = 268 \quad (11)$$

$$V + 0T - 1.6R = 2 \quad (12)$$

$$0.5V + 0T - 1R = 22.5 \quad (13)$$

$$0V + 80T + 1.6R = 288 \quad (14)$$

$$0V + 8T - 3.2R = 14.4 \quad (15)$$

$$0V + 0T + 1R = 3 \quad (16)$$

Convirtiéndose en las siguientes desigualdades, para definir el espacio factible, respectivamente:

$$0V + 80T - 1.6R \geq 268 \quad (17)$$

$$1V + 0T - 1.6R \geq 21 \quad (18)$$

$$0.5V + 0T - 1R \leq 22.5 \quad (19)$$

$$0V + 80T + 1.6R \leq 288 \quad (20)$$

$$0V + 8T - 3.2R \geq 14.4 \quad (21)$$

$$0V + 0T + 1R \geq 3 \quad (22)$$

Donde además se debe de cumplir la no negatividad de las variables, $V, T, R \geq 0$.

4.5 Implementación de programación lineal con redes neuronales

La arquitectura de la red neuronal, Figura 12, usada para resolver este problema de programación lineal se basó en el trabajo de (Wang, 1994). La ventaja de esta arquitectura es que con cambios mínimos se pueden resolver problemas de:

- PL en los cuales tanto la función objetivo como las restricciones son lineales.
- programación cuadrática donde las restricciones son lineales y la función objetivo es de segundo grado.

En el caso de nuestro modelo en principio tiene una función objetivo cuadrática, pero se simplificó a lineal al dejar la presión constante. Sin embargo, en caso de que requiriera considerarla se podría resolver el problema empleando este tipo de red recurrente.

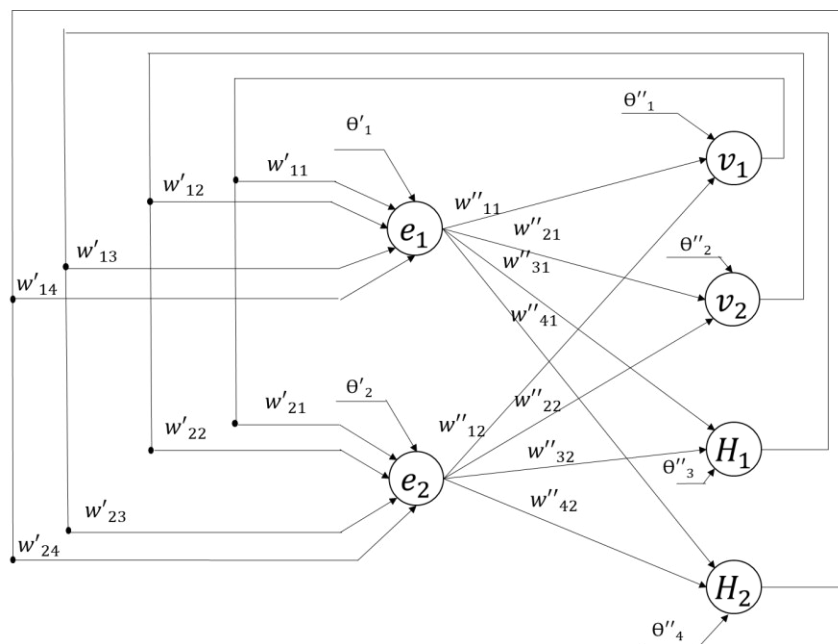


Figura 12. Arquitectura de red neuronal recursiva

En forma general el modelo consiste en dos capas: una de entrada y otra de salida. los pesos W' están definidos de manera a priori por los coeficientes de las restricciones. Por lo que en este caso no se entrena la red para calcular los pesos, sino que se entrena para buscar los valores de las salidas óptimas.

Para explicar los parámetros de la arquitectura de la red usaremos como base el ejemplo que empleamos para describir el método gráfico.

$$z = Av_1 + Bv_2 \quad (23)$$

Sujeto a las restricciones:

$$av_1 + bv_2 \geq D \quad (24)$$

$$cv_1 + dv_2 \geq E \quad (25)$$

$$y \quad v_1, v_2 \geq 0 \quad (26)$$

En este caso se comienza representando el sistema con sus variables de holgura y sintéticas, de manera semejante a lo que se hace en el método simplex:

$$z = Av_1 + Bv_2 + 0H_1 + 0H_2 \quad (27)$$

Sujeto a las restricciones:

$$av_1 + bv_2 + 1H_1 = D \quad (28)$$

$$cv_1 + dv_2 + 1H_2 = E \quad (29)$$

$$y \quad v_1, v_2, H_1, H_2 \geq 0 \quad (30)$$

Este modelo se puede representar en forma matricial.

$$z = \mathbf{c}\mathbf{v} \quad (31)$$

Sujeto a las restricciones:

$$\mathbf{A}\mathbf{v} = \mathbf{b} \quad (32)$$

$$y \quad v_1, v_2, H_1, H_2 \geq 0 \quad (33)$$

En este caso el vector c se forma a partir de los coeficientes de la función objetivo, de la siguiente manera:

$$\mathbf{c}^T = \begin{bmatrix} A \\ B \\ 0 \\ 0 \end{bmatrix} \quad (34)$$

Mientras que la matriz A se forma a partir de los coeficientes de las ecuaciones de restricciones.

$$\mathbf{A} = \left[\begin{array}{cc|cc} a & b & 1 & 0 \\ c & d & 0 & 1 \end{array} \right] \quad (35)$$

El vector \mathbf{v} se describe por las variables del sistema de la siguiente manera:

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ H_1 \\ H_2 \end{bmatrix} \quad (36)$$

Finalmente, el vector \mathbf{b} es establece como:

$$\mathbf{b} = \begin{bmatrix} D \\ E \end{bmatrix} \quad (37)$$

Los elementos de estos vectores y matrices se relacionan con los elementos de la red neuronal de la siguiente manera:

- a) Los pesos w' que conectan las salidas con las neuronas de entrada, están definidos por los elementos de la matriz \mathbf{A} , la cual incluye sus variables de holgura y variables sintéticas siguiendo las reglas del método simplex.
- b) Los pesos w'' que conectan la capa de entrada con la de salida, están definidos por la matriz transpuesta negativa de \mathbf{A} , esto es $-\mathbf{A}^T$.
- c) Los valores de bias θ' están definidos por los términos independientes de las restricciones, esto es por $-\mathbf{b}$
- d) Los bias θ'' están definidos por la multiplicación negativa del vector \mathbf{c} formado por los coeficientes de la función objetivo, por el parámetro obtenido por una función variante en el tiempo $T(t)$, esto es $\mathbf{c} T(t)$.

El parámetro variante en el tiempo $T(t)$, representa el efecto de la recursividad en la red, y proporciona las condiciones necesarias de la no negatividad y la monotonía decreciente de θ'' con respecto al tiempo. En este caso, cada iteración o época de entrenamiento implica un incremento en el tiempo t . La función polinómica exponencial definida en la ecuación (38) fue la seleccionada como la función monótona decreciente para el trabajo de esta tesis.

$$T(t) = \beta(1 + t)^{-\eta} \quad (38)$$

Donde $\beta > 0$ y $\eta > 0$ son parámetros constantes y se eligen dependiendo del modelo que se vaya a trabajar. En la siguiente sección se presentan los valores de los parámetros particulares que empleamos para optimizar nuestro problema de PL.

Los vectores de estado, que definen las salidas de cada neurona, de las capas de entrada y salida se denotan por $\mathbf{e}(t)$ y $\mathbf{v}(t)$, respectivamente. Para el caso del vector de estado de la capa de entrada se denota con la función denominada error instantáneo y que puede expresarse como:

$$\mathbf{e}(t) = \mathbf{A}\mathbf{v}(t) - \mathbf{b} \quad (39)$$

Debido a que la red es recursiva para calcular los valores de las salidas de las neuronas de la capa de entrada se necesita conocer los valores de las variables de decisión v , los cuales están definidos por el vector de estado de salida $\mathbf{v}(t)$ y para esta arquitectura se calculan con la función de activación:

$$v(t) = \frac{v_{max}}{1 + \exp^{-\xi u_i(t)}} \quad (40)$$

Siendo $v_{max} > 0$ y para garantizar una monotonía creciente, el parámetro ξ debe ser positivo, además de que es utilizado para controlar la pendiente de la función de activación sigmoidea, debido a que la sensibilidad de activación aumenta a medida que ξ aumenta, generalmente se selecciona un valor grande del parámetro ξ para acelerar la convergencia de la red neuronal recursiva. El rango de la función de activación sigmoidea se define como $[0, v_{max}]$, porque el mínimo y el máximo del estado de activación son 0 y v_{max} , respectivamente.

En esta función de activación se requiere conocer el término $u(t)$ el cual define el valor de las entradas netas instantáneas a las neuronas y es cambiante en el tiempo. Sin embargo, en el modelo no se conoce directamente $u(t)$ únicamente su derivada en el tiempo, la cual está definida por:

$$\frac{du(t)}{dt} = -\mathbf{A}^T \mathbf{e}(t) - T(t)\mathbf{c} \quad (41)$$

Por lo que en este caso $u(t)$ se puede evaluar integrando en el tiempo:

$$u(t) = \int_0^{t_{epocai}} [-\mathbf{A}^T \mathbf{e}(t) - T(t)\mathbf{c}] dt \quad (42)$$

donde t_{epocai} es el tiempo correspondiente a la i -ésima iteración o época de entrenamiento.

5 RESULTADOS DEL PROCESO DE OPTIMIZACIÓN

Una vez que se ha definido la función objetivo (10) a maximizar con sus restricciones (17)-(22), se puede proceder a formar la matriz **A** y los vectores **b** y **c**, los cuales son los necesarios para la solución del problema mediante el algoritmo implementado por redes neuronales.

Debido a que se decide que la variable *P* toma el valor constante de 3.5, la función objetivo se representa como:

$$z = -0.848V - 106.89535T + 1000R - 1 \times 10^9 S_1 - 1 \times 10^9 S_2 + 0H_1 + 0H_2 - 1 \times 10^9 S_3 - 1 \times 10^9 S_4 \quad (43)$$

Donde el vector **c** se forma a partir de los coeficientes de la función objetivo

$$\mathbf{c} = \begin{bmatrix} -0.848 \\ -106.89535 \\ 1000 \\ -1 \times 10^9 \\ -1 \times 10^9 \\ 0 \\ 0 \\ -1 \times 10^9 \\ -1 \times 10^9 \end{bmatrix} \quad (44)$$

La matriz **A** se forma a partir de los coeficientes de las restricciones.

$$\mathbf{A} = \begin{bmatrix} 0 & 80 & -1.6 & -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1.6 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0.5 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 80 & 1.6 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 8 & -3.2 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \quad (45)$$

Para el caso del vector **b**, son los coeficientes del lado derecho de las restricciones.

$$\mathbf{b} = \begin{bmatrix} 268 \\ 21 \\ 22.5 \\ 288 \\ 14.4 \\ 3 \end{bmatrix} \quad (46)$$

Con estos vectores y matrices se puede bosquejar la red neuronal (Figura 13) recurrente que representa nuestro sistema y que debemos resolver.

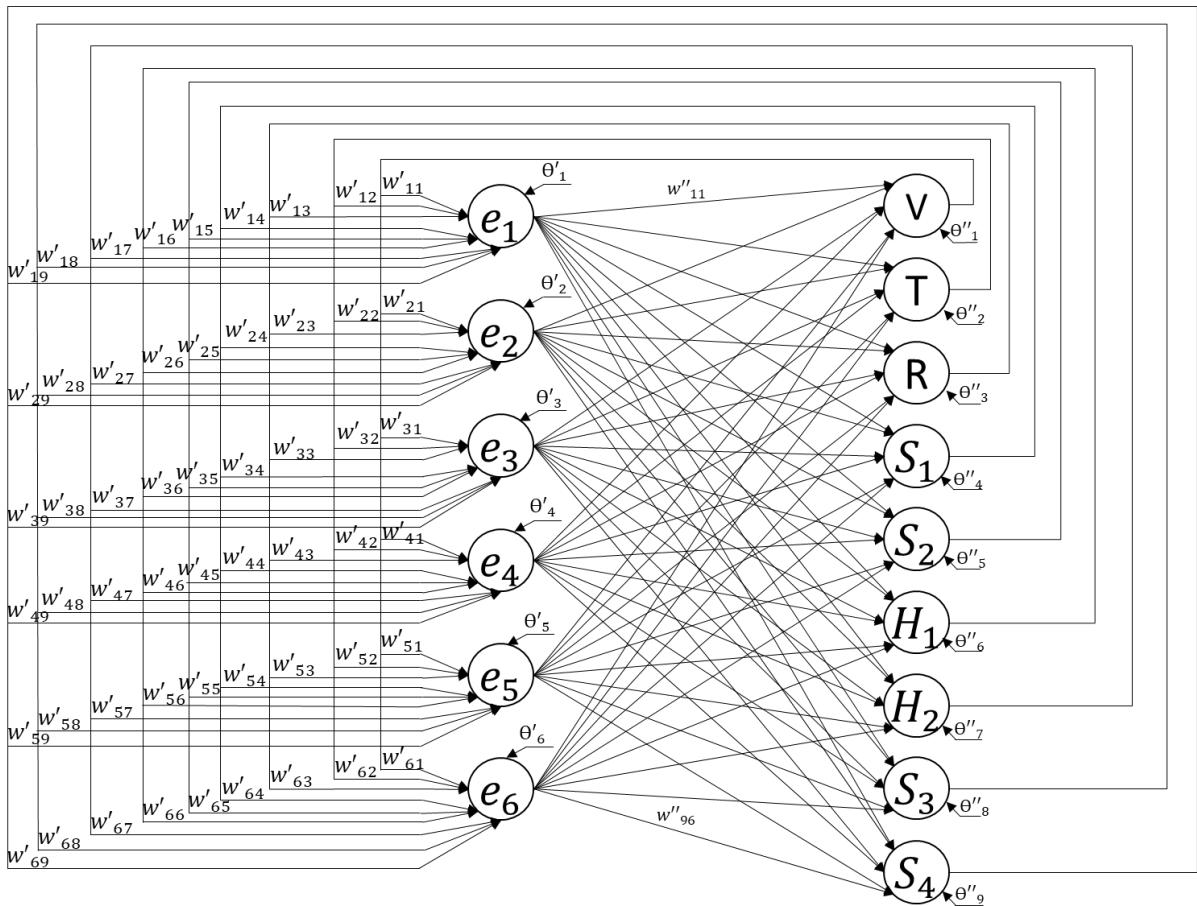


Figura 13. Arquitectura de la red neuronal utilizada para la solución del trabajo de tesis

5.1 Resultados obtenidos con la red neuronal recursiva

Para buscar las soluciones del sistema, se implementó un algoritmo en Matlab encargado para evaluar y entrenar la red neuronal recurrente mostrada en la Figura 13. Cabe recordar que a esta red el proceso de entrenamiento es diferente al de las redes convencionales, en este modelo los pesos son constantes y los definen los coeficientes de las restricciones. Aquí lo que se va “entrenando” es la red hasta que los valores de las variables de estado (salida) presentan un cierto nivel de monotonía y además satisfacen cada una de las restricciones.

El programa desarrollado en Matlab se presenta en el Apéndice 1. Para la ejecución de este programa se deben de definir algunos parámetros como son el número de épocas (*epoch*), es decir, la cantidad de veces que se repite el entrenamiento, para nuestro modelo seleccionamos un $epoch = 40 \times 10^3$. Por otra parte, las variables de estado $V, T, R, S_1, S_2, S_3, S_4, H_1$ y H_2 , se inicializan con valores aleatorios. Los valores que se le dan a los parámetros $v_{max}, \beta, \xi, \eta$ y Δt de las ecuaciones (38) y (40) son parámetros importantes que ayudan en la convergencia del proceso de aprendizaje. La combinación de estos parámetros conforma en sí un tipo de coeficiente de aprendizaje, para describirlo de manera análoga al de redes neuronales convencionales, puesto que “operan” directamente sobre la función de activación de las neuronas, que para este caso es una función sigmoide tal como se describe en la ecuación (40). Como en las redes convencionales, si el coeficiente de aprendizaje es muy bajo, la red puede requerir de un mayor número de épocas para converger, por el contrario, si es muy grande el proceso de aprendizaje puede llevar a que la solución diverja. Por lo que es necesario hacer algunos experimentos de entrenamiento para evaluar la convergencia de la red y además analizar si la solución es factible.

En el caso del parámetro v_{max} aparte de afectar en proceso de aprendizaje también limita el valor máximo que puede tener una salida, puesto que multiplica al sigmoide de la función de activación (ecuación (40)) y por lo tanto en el caso extremo el valor de las salidas va a ser v_{max} , por lo que, para seleccionar su valor, identificamos de las variables del sistema cual tenía el valor experimental más alto y fijamos v_{max} en un nivel cercano a dicho valor. En el caso de nuestros experimentos la variable que registro el mayor valor fue el volumen con 36 ml, por lo que elegimos valores cercanos a este nivel.

Para identificar los valores aceptables de los parámetros de “aprendizaje” de la red realizamos varios experimentos de entrenamiento. En primera instancia se busca que después de cierto número de épocas los valores de las variables de estado no cambien entre épocas, de manera tal que si permanecen estables en el tiempo pudieran considerarse soluciones factibles. En la Tabla 9 se presentan los resultados de la función objetivo para ciertos valores que la red considera variables de estado factibles. Así mismo se enlistan el número de épocas en las que la red mostró una tendencia estable en el proceso de aprendizaje. En este caso además de verificar la estabilidad en el tiempo de las variables se debe de comprobar que se satisfagan todas las restricciones.

Tabla 9. Resultados obtenidos al aplicar diferentes valores de los parámetros de la red recurrente.

Parámetros	V (ml)	T (hr)	R	S ₁ , S ₂ , S ₃ , S ₄	H ₁	H ₂	z _o (\$)	Épocas para converger	Satisface las restricciones
$v_{max} = 30, \beta = 1, \xi = 25, \eta = 1000, \Delta t = 2 \times 10^{-6}$	27.4399	3.4307	4.0050	0	4.757	7.065	3615.01	20000	Si
$v_{max} = 36, \beta = 1, \xi = 25, \eta = 1000, \Delta t = 2 \times 10^{-6}$	27.4561	3.4308	4.010	0	4.7221	7.0403	3620.03	32000	Si
$v_{max} = 42, \beta = 1, \xi = 25, \eta = 1000, \Delta t = 2 \times 10^{-6}$	27.4385	4.7792	4.0018	0	4.7389	7.0941	3467.65	Hasta la época 40000 la variable T aun no tiende a estabilizarse	No, la cuarta restricción no se satisface (ecuación (20)).
$v_{max} = 36, \beta = 1, \xi = 25, \eta = 1000, \Delta t = 4 \times 10^{-6}$	27.3624	0.0025	3.9938	0	4.8412	7.2029	3970.37	Hasta la época 40000 la variable T aun no tiende a estabilizarse	No, las restricciones 1, 2 y 5 (ecuaciones (17), (18) y (21)) no se satisfacen
$v_{max} = 36, \beta = 1, \xi = 15, \eta = 1000, \Delta t = 2 \times 10^{-6}$	27.3538	3.4290	3.9718	0	4.8524	7.3252	3582.09	Hasta la época 120000 las variables T, R y V aun no tienden a estabilizarse	No, las restricciones 1 y 2 (ecuaciones (17) y (18)) no se satisfacen
$v_{max} = 36, \beta = 1, \xi = 25, \eta = 500, \Delta t = 2 \times 10^{-6}$	27.5846	3.4354	4.0564	0	4.5563	6.7014	3665.814	50000	Si
$v_{max} = 36, \beta = 1, \xi = 25, \eta = 750, \Delta t = 2 \times 10^{-6}$	27.4221	3.4301	3.9958	0	4.7640	7.1533	3605.85	50000	Si
$v_{max} = 36, \beta = 1, \xi = 25, \eta = 350, \Delta t = 2 \times 10^{-6}$	27.4135	3.4299	3.9926	0	4.7750	7.1759	3602.78	80000	Si
$v_{max} = 36, \beta = 1.5, \xi = 25, \eta = 500, \Delta t = 2 \times 10^{-6}$	27.4281	3.4302	3.9979	0	4.7562	7.1377	3607.99	80000	Si
$v_{max} = 36, \beta = 1.5, \xi = 25, \eta = 1000, \Delta t = 2 \times 10^{-6}$	27.5397	3.4317	4.0394	0	4.6130	6.8285	3649.30	48000	Si
$v_{max} = 37, \beta = 2.5, \xi = 40, \eta = 1780, \Delta t = 1.1 \times 10^{-6}$	27.6409	3.4343	4.0811	0	4.4801	6.5059	3690.55	60000	Si

En la Tabla 9 se puede observar que los experimentos seleccionados, las soluciones óptimas de las variables sintéticas S_1 , S_2 , S_3 y S_4 son 0, lo cual es un requerimiento necesario para encontrar soluciones factibles.

Después de entrenar varias veces la red seleccionamos los siguientes valores $v_{max} = 37$, $\beta = 2.5$, $\xi = 40$, $\eta = 1780$ y un $\Delta t = 1.1 \times 10^{-6}$, debido a que con estos el sistema convergió y proporcionó el valor máximo de la función objetivo y satisface las restricciones. Con aproximadamente 60000 épocas la red neuronal proporciona resultados que indican que se tendrá una ganancia máxima en la función objetivo de $z = \$3,690.55$ pesos, usando variables de estado con valores de $V = 27.6409$ ml, $T = 3.4343$ horas y $R = 4.0811$. Para verificar que la respuesta es factible se sustituyen los valores óptimos variables de estado obtenidos en cada una de las restricciones y se obtiene lo siguiente:

En la desigualdad (17) resulta un valor de 268.2165, el cual cumple al ser ≥ 268 .

En la desigualdad (18) resulta un valor de 21.1112, el cual cumple al ser ≥ 21 .

En la desigualdad (19) resulta un valor de 17.9016, el cual cumple al ser ≤ 22.5 .

En la desigualdad (20) resulta un valor de 281.2761, el cual cumple al ser ≤ 288 .

En la desigualdad (21) resulta un valor de 14.4151, el cual cumple al ser ≥ 14.4 .

En la desigualdad (22) resulta un valor de 4.0811, el cual cumple al ser ≥ 3 .

En este caso se puede observar que se satisfacen todas las restricciones por lo que la solución obtenida es factible y en nuestro caso se considera como óptima.

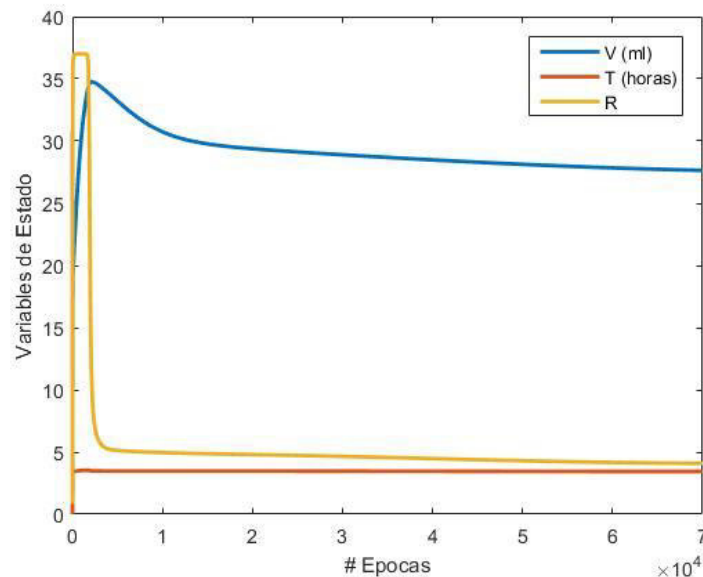


Figura 14. Comportamiento de las variables de estado a través de las épocas.

En la figura 14 se observa el comportamiento de las variables de estado V, T y R al pasar de las épocas, donde se comienza a encontrar una convergencia en la época número 20000 aproximadamente. De igual forma en la Figura 15, se puede observar como la función objetivo z comienza a converger en el mismo número de época.

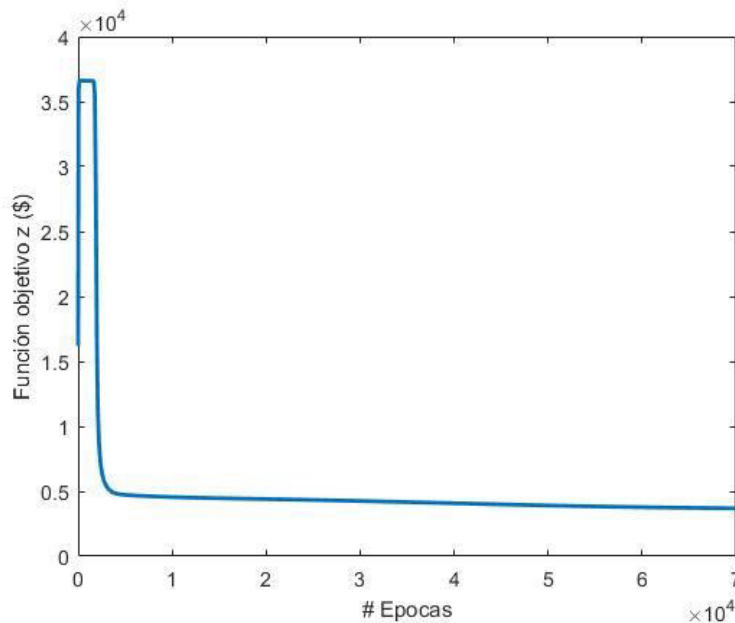


Figura 15. Comportamiento de la función objetivo a través de las épocas.

5.2 Comparación de resultados obtenidos con la red recurrente y el método simplex

A manera de comparación con un algoritmo de simplex tradicional, se hizo uso de una librería de RStudio para la solución de problemas de programación lineal, la cual se llama *lpSolve* donde se introducen los vectores c y b , en conjunto con la matriz A , indicando también que es un problema de maximización y los signos de las restricciones. Dando como resultados para $z = 3886.432$ equivalente a ganancias en pesos, con valores de las variables de estado de $V = 27.8571$ ml, $T = 3.5142$ horas y $R = 4.2857$.

Los valores optimizados de las variables de estado arrojados por la red neuronal fueron comprobados experimentalmente en la máquina, donde se tuvo un rendimiento final de 3.9, respaldando así el correcto funcionamiento de la metodología desarrollada en este trabajo.

6 CONCLUSIONES

El proceso de manufactura de oligonucleótidos utilizados en su gran mayoría para kits de diagnóstico de enfermedades respiratorias y ETS resultó optimizado, gracias al desarrollo de la metodología propuesta en este trabajo de tesis, logrando aumentar producción y ganancias en menor tiempo.

Con base a los valores obtenidos se puede concluir que la máxima ganancia se logra cuando el volumen es aproximadamente 27.4 ml, el tiempo de proceso de 3.43 horas y un rendimiento de aproximadamente 4, traducido a \$3615 de ganancia por proceso de síntesis, es decir que, si al día se realizan 3 rondas de síntesis, se tendrá el triple de ganancias en pesos, traducido a meses como una cantidad bastante considerable.

Respecto a la implementación de la red neuronal con PL, se concluye que es una manera más práctica de abordar, por el hecho de que los valores son más exactos en comparación del método simplex tradicional, sumando a su favor que la red es capaz de resolver problemas con funciones no lineales, adecuándose de manera sencilla en el algoritmo, factor que el método simplex no logra.

La biogenética es una ciencia que, si se aborda en futuros trabajos con inteligencia artificial, facilitará en gran medida el trabajo a los expertos del área, debido a la adaptabilidad de procesos iterativos y repetitivos.

Las redes neuronales son una excelente alternativa para problemas de optimización, no solo para áreas específicas de estudio, sino para cualquiera, gracias a su fácil ajuste de sistemas y los diversos recursos que proporciona.

7 PRODUCTOS GENERADOS

A lo largo del posgrado se desarrolló:

Poster y publicación de extenso en el 10° Encuentro en Ciencia y Tecnología (ENCITE 10), organizado por el Instituto Politécnico Nacional, con el trabajo titulado “Clasificación automática de textos estructurados utilizando inteligencia artificial”, el cual tuvo lugar el 17 de noviembre del año 2022.

Presentación y publicación de extenso en el Congreso Internacional de Investigación de Academia Journals Chetumal 2022, con el trabajo titulado “Metodologías Ágiles en Administración de Proyectos de Software”

8 REFERENCIAS

- Alazzouzi, E. (1996). *Síntesis en fase sólida de oligonucleótidos cíclicos*. <https://dialnet.unirioja.es/servlet/tesis?codigo=239696&info=resumen&idioma=SPA>
- Al-Mamari, G., Bouabdallah, F., & Cherif, A. (2022). A Survey of Sink Mobility Models to Avoid the Energy-Hole Problem in Wireless Sensor Networks. *IJACSA International Journal of Advanced Computer Science and Applications*, 13(5), 981–993. www.ijacsa.thesai.org
- Almohamad, H. A., & Duffuaa, S. O. (1993). A Linear Programming Approach for the Weighted Graph Matching Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(5), 522–525. <https://doi.org/10.1109/34.211474>
- Almohri, H. M. J., Watson, L. T., Yao, D., & Ou, X. (2016). Security optimization of dynamic networks with probabilistic graph modeling and linear programming. *IEEE Transactions on Dependable and Secure Computing*, 13(4), 474–487. <https://doi.org/10.1109/TDSC.2015.2411264>
- Andrade Tepan, E. C. (2013). *Estudio de los principales tipos de redes neuronales y las herramientas para su aplicación*. <http://dspace.ups.edu.ec/handle/123456789/4098>
- Arana, C. (2021). *Redes neuronales recurrentes: Análisis de los modelos especializados en datos secuenciales*. <https://www.econstor.eu/bitstream/10419/238422/1/797.pdf>
- Ardenghi, J. M., Buffo, F. E., & Vidal, M. (2022). *Problema de programación lineal entera para un tratamiento de aguas industriales*. <http://sedici.unlp.edu.ar/handle/10915/151844>
- Atalay, K. D., Dengiz, B., Yavuz, T., Koç, E., & Iç, Y. T. (2020). Airfoil–slat arrangement model design for wind turbines in fuzzy environment. *Neural Computing and Applications*, 32(17), 13931–13939. <https://doi.org/10.1007/S00521-020-04796-9/METRICS>

- Beaucage, S. L., & Reese, C. B. (2009). Recent advances in the chemical synthesis of RNA. *Current Protocols in Nucleic Acid Chemistry, SUPPL. 38*, 1–31. <https://doi.org/10.1002/0471142700.NC0216S38>
- Bermejo Martín, J. F. (2005). *Nuevos dendrímeros hidrosolubles de estructura carbosilano: estudios de biocompatibilidad y posibles aplicaciones biomédicas con oligonucleótidos en células mononucleares de sangre periférica*. <https://dialnet.unirioja.es/servlet/tesis?codigo=195086&info=resumen&idioma=SPA>
- Brown, D. M. (1993). A brief history of oligonucleotide synthesis. *Methods in molecular biology (Clifton, N.J.)*, 20, 1–17. <https://doi.org/10.1385/0-89603-281-7:1>
- Caetano, M. A. L., De Souza, J. A. M. F., & Yoneyama, T. (2008). Optimal medication in HIV seropositive patient treatment using fuzzy cost function. *Proceedings of the American Control Conference*, 2227–2232. <https://doi.org/10.1109/ACC.2008.4586823>
- Chou, C. A., Cao, Q., Weng, S. J., & Tsai, C. H. (2020). Mixed-integer optimization approach to learning association rules for unplanned ICU transfer. *Artificial Intelligence in Medicine*, 103, 101806. <https://doi.org/10.1016/J.ARTMED.2020.101806>
- De La, E., May, C., Ricalde, L. J., Jose, B., & Jimenez, C. (2017). *DISEÑO DE UN SISTEMA DE GESTIÓN ENERGÉTICA PARA REDES ELÉCTRICAS INTELIGENTES Application of heuristic techniques for analyzing energy systems. View project*. <https://www.researchgate.net/publication/332158784>
- Dehghanpour, E., Kazemi Karegar, H., Kheirollahi, R., & Soleymani, T. (2018). Optimal coordination of directional overcurrent relays in microgrids by using cuckoo-linear optimization algorithm and fault current limiter. *IEEE Transactions on Smart Grid*, 9(2), 1365–1375. <https://doi.org/10.1109/TSG.2016.2587725>
- Del Rio Gomez, D. (2021). *Un método simplex en programación lineal multiobjetivo*. <https://uvadoc.uva.es/handle/10324/50558>
- Eghbal Ahmadi, M. H., & Mosayebi, A. (2021). Fischer – Tropsch synthesis over Co-Ni/Al₂O₃ catalyst: Comparison between comprehensive kinetic modeling, Artificial Neural Network, and a novel hybrid GA-Fuzzy models. *Journal of the Taiwan Institute of Chemical Engineers*, 127, 32–45. <https://doi.org/10.1016/J.JTICE.2021.07.041>
-

- Gallo, M. (2002). *Desarrollo de oligonucleótidos modificados químicamente con potencial aplicación en terapia génica* [Universidad de Buenos Aires]. https://bibliotecadigital.exactas.uba.ar/collection/tesis/document/tesis_n3470_Gallo
- Golfe, A. (2021). *Diseño de un modelo compuesto de arquitecturas de red neuronal convolucional y recurrente para descripción de video en entornos outdoor/indoor*. <https://riunet.upv.es/handle/10251/165340>
- Hillier, F. S. (2015). *Investigación de operaciones*. <https://biblioteca.uazuay.edu.ec/buscar/item/77886>
- Hogrefe, R., & Biotechnologies, T. (2015). *A Short History of Oligonucleotide Synthesis*.
- Hussain, S., Kim, Y. S., Thakur, S., & Breslin, J. G. (2022). Optimization of Waiting Time for Electric Vehicles Using a Fuzzy Inference System. *IEEE Transactions on Intelligent Transportation Systems*, 23(9), 15396–15407. <https://doi.org/10.1109/TITS.2022.3140461>
- Illana, J. C. (2014). Biología molecular y estructura del ADN. *Anales de Química de la RSEQ*, 110(3), 234–234. <https://analesdequimica.es/index.php/AnalesQuimica/article/view/564>
- Integrated DNA Technologies. (2011). *Chemical Synthesis and Purification of Oligonucleotides*.
- İskenderoğlu, Ö., & Akdag, S. (2019). A Comparison of Portfolio Optimization Results with Fuzzy Konno-Yamazaki Linear Programming in Bull and Bear Markets: The Case of Turkey. *Economic Computation and Economic Cybernetics Studies and Research*, 53(4/2019), 195–208. <https://doi.org/10.24818/18423264/53.4.19.12>
- Jamali, N., Leung, R. K., & Verderber, S. (2020). A review of computerized hospital layout modelling techniques and their ethical implications. *Frontiers of Architectural Research*, 9(3), 498–513. <https://doi.org/10.1016/J.FOAR.2020.01.003>
- Kästner, D., & Langenbach, M. (2014). *Integer linear programming vs. graph-based methods in code generation*. <https://doi.org/10.22028/D291-25801>
- Khan, I. U., & Rafique, F. (2021). Minimum-cost capacitated fuzzy network, fuzzy linear programming formulation, and perspective data analytics to minimize the
-

- operations cost of American airlines. *Soft Computing*, 25(2), 1411–1429. <https://doi.org/10.1007/S00500-020-05228-5/METRICS>
- Lee, W. J., Kwag, S. II, & Ko, Y. D. (2020). Optimal capacity and operation design of a robot logistics system for the hotel industry. *Tourism Management*, 76, 103971. <https://doi.org/10.1016/J.TOURMAN.2019.103971>
- Liu, N., & Cios, K. J. (1992). Learning rules by integer linear programming. *IEEE International Symposium on Industrial Electronics*, 246–250. <https://doi.org/10.1109/ISIE.1992.279577>
- Mangasarian, O. L., Street, W. N., & Wolberg, W. H. (1995). Breast Cancer Diagnosis and Prognosis Via Linear Programming. <https://doi.org/10.1287/opre.43.4.570>, SS-94-01, 83–86. <https://doi.org/10.1287/OPRE.43.4.570>
- Mangasarian, O., & Wolberg, W. (1990). *Cancer diagnosis via linear programming*. <https://minds.wisconsin.edu/bitstream/handle/1793/59346/TR958.pdf>
- Manuel, I. J. (2016). *PROGRAMACIÓN LINEAL EL MÉTODO SIMPLEX*. https://www.researchgate.net/profile/Izar-Juan-Manuel/publication/329931840_SimplexProgLineal/links/655ce0beb86a1d521bfcea79/SimplexProgLineal.pdf
- Maqueda Bojorquez, D. E. (2021). *DE REDES NEURONALES RECURRENTE A MODELOS DE LENGUAJE: LA EVOLUCIÓN DEL PLN EN LA GENERACIÓN DE TEXTOS*. <https://doi.org/10.22201/dgtic.26832968e.2021.4.1>
- Matsingos, C., Urdaneta, A. M., Hernández, J. C., & Silva, R. A. P. (2021). Aplicaciones de la inteligencia artificial en la farmacología básica y clínica. *Medicina*, 43(4), 652–667. <https://doi.org/10.56050/01205498.1652>
- McBride, L. J., & Caruthers, M. H. (1983). An investigation of several deoxynucleoside phosphoramidites useful for synthesizing deoxyoligonucleotides. *Tetrahedron Letters*, 24(3), 245–248. [https://doi.org/10.1016/S0040-4039\(00\)81376-3](https://doi.org/10.1016/S0040-4039(00)81376-3)
- Musa, M. S., Ozsahin, Di. U., & Ozsahin, I. (2019). A Comparison for Liver Cancer Treatment Alternatives. *2019 Advances in Science and Engineering Technology International Conferences, ASET 2019*. <https://doi.org/10.1109/ICASET.2019.8714471>
-

Oligo chemical synthesis – Custom oligo production | Eurogentec. (s/f). Recuperado el 30 de octubre de 2023, de <https://www.eurogentec.com/en/synthesis-process>

Ozsahin, Di. U., Nyakuwanikwa, K., Wallace, T., & Ozsahin, I. (2019). Evaluation and Simulation of Colon Cancer Treatment Techniques with Fuzzy PROMETHEE. *2019 Advances in Science and Engineering Technology International Conferences, ASET 2019*. <https://doi.org/10.1109/ICASET.2019.8714509>

Ozsahin, I., Ozsahin, D., Maisaini, M., J, G. M.-W. C. R., & 2019, undefined. (2019). Fuzzy PROMETHEE analysis of leukemia treatment techniques. *World Cancer Research Journal*, 6, 1–5. <https://www.wcrj.net/wp-content/uploads/sites/5/2019/06/e1315-Fuzzy-PROMETHEE-analysis-of-leukemia-treatment-techniques.pdf>

Ozsahin, I., Ozsahin, Di. U., Nyakuwanikwa, K., & Simbanegavi, T. W. (2019). Fuzzy PROMETHEE for Ranking Pancreatic Cancer Treatment Techniques. *2019 Advances in Science and Engineering Technology International Conferences, ASET 2019*. <https://doi.org/10.1109/ICASET.2019.8714554>

Ozsahin, I., & Uzun Ozsahin, D. (2018). A Fuzzy PROMETHEE Approach for Breast Cancer Treatment Techniques. *Article in International Journal of Health Sciences*. www.ijmrhs.com

Pamela, P., Gayathri, P., & Jaisankar, N. (2013). *A Fuzzy Optimization Technique for the Prediction of Coronary Heart Disease Using Decision Tree*.

Papargyri, N., Pontoppidan, M., Andersen, M. R., Koch, T., & Hagedorn, P. H. (2020). *Chemical Diversity of Locked Nucleic Acid-Modified Antisense Oligonucleotides Allows Optimization of Pharmaceutical Properties*. <https://doi.org/10.1016/j.omtn.2019.12.011>

Paredes, E., Aduda, V., Ackley, K. L., & Cramer, H. (2017). Manufacturing of Oligonucleotides. En *Comprehensive Medicinal Chemistry III* (pp. 233–279). Elsevier. <https://doi.org/10.1016/B978-0-12-409547-2.12423-0>

Patiño, P., & Rugeles, M. (2006). Replicación del ADN. *Fondo Editorial Biogénesis*, 119–132. <https://revistas.udea.edu.co/index.php/biogenesis/article/download/325984/20783275/>

Peña Florez, L. A., Rodríguez-Rojas, Y. L., Peña Florez, L. A., & Rodríguez-Rojas, Y. L. (2018). Procedimiento de Evaluación y Selección de Proveedores Basado en el Proceso de Análisis Jerárquico y en un Modelo de Programación Lineal Entera Mixta. *Ingeniería*, 23(3), 230–251. <https://doi.org/10.14483/23448393.13316>

Repaso de la estructura y replicación del ADN (artículo) | Khan Academy. (s/f). Recuperado el 22 de noviembre de 2023, de <https://es.khanacademy.org/science/ap-biology/gene-expression-and-regulation/replication/a/hs-dna-structure-and-replication-review>

Rouhiainen, L. (2018). *INTELIGENCIA ARTIFICIAL 101 COSAS QUE DEBES SABER HOY SOBRE NUESTRO FUTURO INTELIGENCIA ARTIFICIAL.* www.planetadelibros.com

Salas, H. G. (2017). *PROGRAMACIÓN LINEAL APLICADA* (2a ed.). www.ecoediciones.com

Sarmiento-Ramos, J. L. (2020). Aplicaciones de las redes neuronales y el deep learning a la ingeniería biomédica. *Revista UIS Ingenierías*, 19, 1–18. <https://doi.org/10.18273/revuin.v19n4-2020001>

Schwaller, P., Vaucher, A. C., Laplaza, R., Bunne, C., Krause, A., Corminboeuf, C., & Laino, T. (2022). Machine intelligence for chemical reaction space. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 12(5), e1604. <https://doi.org/10.1002/WCMS.1604>

Sekine, M. (2018). Recent development of chemical synthesis of RNA. *Synthesis of Therapeutic Oligonucleotides*, 41–65. https://doi.org/10.1007/978-981-13-1912-9_3/FIGURES/20

Taha, H. A. (2017). *Investigación de operaciones* (Guillermo Trujano Mendoza, Ed.). https://books.google.com/books/about/Investigacion_de_operaciones.html?hl=es&id=3oHztjMSuL8C

Tapia, C. E. F., & Cevallos., K. L. F. (2022). Optimización de portafolios de venta de productos en la post crisis Covid-19, aplicando programación lineal y pronósticos. *Espíritu Emprendedor TES*, 6(2), 56–80. <https://doi.org/10.33970/eetes.v6.n2.2022.304>

- Torres, D. B. (2022). La Investigación de Operaciones. Herramienta fundamental en las Empresas. *Boletín administrativo*, 1–2. http://ojs.urbe.edu/index.php/boletin_administrativo/article/view/3801
- Varghese, C., & Srivastav, P. P. (2022). Formulation and sensory characterization of low-cost, high-energy, nutritious cookies for undernourished adolescents: An approach using linear programming and fuzzy logic. *Innovative Food Science & Emerging Technologies*, 75, 102904. <https://doi.org/10.1016/J.IFSET.2021.102904>
- Vélez, P. E., Rodríguez, E., & Moreno, P. A. (2003). *Curso Introductorio de Biología Molecular Construcción de Bibliotecas de ADN-Recombinante en el Bacteriófago lambda para el aislamiento de clones de genes* (Vol. 1).
- Wang, J. (1994). A deterministic annealing neural network for convex programming. *Neural Networks*, 7(4), 629–641. [https://doi.org/10.1016/0893-6080\(94\)90041-8](https://doi.org/10.1016/0893-6080(94)90041-8)
- Zlotnik, A., Gallardo-Antolín, A., Alfaro, M. C., Pérez, M. C. P., & Martínez, J. M. M. (2015). Emergency department visit forecasting and dynamic nursing staff allocation using machine learning techniques with readily available open-source software. *CIN - Computers Informatics Nursing*, 33(8), 368–377. <https://doi.org/10.1097/CIN.0000000000000173>

APÉNDICE 1

```
close all;

NEpochs_max=40e3;

c=[-0.848;-(3.75+29.4701*3.5); 1000];
c=-c;

A=[0.0 80 -1.6;
  1.0 0.0 -1.6;
  0.5 0.0 1;
  0.0 80 1.6;
  0.0 8 -3.2;
  0.0 0.0 1.0];

b=[268; 21; 22.5; 288; 14.4; 3];

Desi=[2;2;0;0;2;2];

S0=find(Desi==1);
S2=find(Desi==2)

Slack=eye(size(A,1));

Slack(S2,:)=Slack(S2,:)*(-1)

ASlack=[A Slack]

NRestricciones=size(A,1);

NVarSis=length(c);

vtslack=rand(size(Slack,2),1);

vt=[30;0.1;0.1];
vSlack=[vt;vtslack];

cSlack=zeros(size(vSlack));

NumeVar=size(A,2);

cSlack(1:NumeVar)=c;

cSlack(NumeVar+S2)=1e9
```

```
vmax=30;
chi=25;
eta=1e3;
beta=1;
Deltat=2e-6;

ASlackT=ASlack';

NofN1=size(ASlack,1);

NofN2=size(ASlack,2);

Capas{1}=zeros(NofN1,1);

Capas{2}=vSlack;

Pesos{1}=[ASlack];
Pesos{2}=-ASlackT;

Bias{1}=-b;

Tt2=[];
Epoch=1;
vt2=[];
t2=[];
utslack=vtslack*0;
PenFun=[];

j=1;
dut=[];
fv=[];
Restricciones=[];
Restricciones2=[];

Energy=[];

for t=0:Deltat:NEpochs_max*Deltat,
    PenFun=[PenFun; ([ASlack*Capas{2}-b]'*[ASlack*Capas{2}-b])/2];
    Tt=beta*exp(-eta*t);
    Tt2=[Tt2;Tt];
    vt2=[vt2 Capas{2}];
    t2=[t2;t];

    Bias{2}=-Tt*cSlack;

    Capas{1}=Pesos{1}*Capas{2}+Bias{1};

    dut2=[Pesos{2}*Capas{1}+Bias{2}]';

    dut=[dut;dut2];
```

```
if j>1,
    ut=[trapz (dut)] '*Deltat;
else
    ut=[dut]' *Deltat;
end

Capas{2}=vmax./(1+exp(-chi*ut));

fv=[fv; (c'*Capas{2}(1:NVarSis))];
Res1=zeros(1,NRestricciones);
Res2=zeros(1,NRestricciones);
for JJ=1:NRestricciones,
    Res1(JJ)=(A(JJ,:)*Capas{2}(1:NVarSis));
    Res2(JJ)=(fv(j)-(A(JJ,1)*Capas{2}(1)))/A(JJ,2);
end
Restricciones=[Restricciones;Res1];
Restricciones2=[Restricciones2;Res2];
Energy=[Energy;abs(Tt*(fv(j)-vmax)+PenFun(j))];
Epoch=1+Epoch;
j=j+1;
end
```