Algoritmo genético multiobjetivo para el diseño de una cadena de suministro de circuito cerrado para la industria de neumáticos con demanda estocástica

Paula Andrea Castellanos Peña y Johan Sebastián Hernández Ortiz

Trabajo de Grado para Optar el título de Ingeniero Industrial

Director
Julio Cesar Camacho Pinto
Magister en Ingeniería Industrial

Codirector
Edwin Alberto Garavito
Magister en Ingeniería Industrial

Universidad Industrial de Santander
Facultad de Ingenierías Fisicomecánicas
Escuela de Estudios Industriales y Empresariales
Bucaramanga
2022

Agradecimientos

A Dios por darme la fortaleza y perseverancia para culminar mi carrera profesional.
A mis padres y hermana por ser mi apoyo, motivación y ejemplo.
A mis amigas de universidad por su compañía y aprendizajes vividos durante todos estos años.
A mis profesores, en especial al director del proyecto, Julio Camacho, por su asesoría y paciencia.
A mi compañero Johan por el esfuerzo, dedicación y aporte para culminar este proyecto.

PAULA ANDREA CASTELLANOS PEÑA

A Dios por permitirme culminar esta etapa.
A mis padres y a mi hermano por su apoyo incondicional durante mi formación profesional.
A la Universidad Industrial de Santander por ser mi segundo hogar y a todos mis profesores por enseñarme y formarme.
A nuestro director Julio Camacho por su ayuda y guía durante la elaboración del proyecto.
A mi compañera Paula por todo su esfuerzo, compromiso y dedicación.

JOHAN SEBASTIÁN HERNÁNDEZ ORTIZ

Tabla de Contenido

Lista de Tablas

Lista de Figuras

Lista de Apéndices

Resumen

Título: Algoritmo genético multiobjetivo para el diseño de una cadena de suministro de circuito cerrado para la industria de neumáticos con demanda estocástica.

Autores: Paula Andrea Castellanos Peña y Johan Sebastian Hernandez Ortiz
Palabras Clave: cadena de suministro de bucle cerrado, multiobjetivo, programación estocástica, problema de localización y asignación, NSGAII.

Descripción:
La producción y el consumo responsable es uno de los objetivos de desarrollo sostenible de las naciones unidas el cual busca principalmente desvincular el crecimiento económico de la degradación medioambiental y aumentar la eficiencia de recursos, por esta razón el nuevo reto para las industrias consiste en coordinar y administrar efectivamente la gestión de la recolección y tratamiento de los productos desechos. El presente trabajo de investigación tiene como finalidad proponer y evaluar una solución alterna a un modelo matemático estocástico y multiobjetivo planteado de una cadena de suministro de circuito cerrado de una empresa iraní fabricante de llantas.

El modelo se caracteriza por integrar de manera simultánea el enfoque económico, ambiental y a su vez evaluar la capacidad de respuesta de la cadena. El componente estocástico del modelo se considera en la demanda. La solución propuesta para el modelo es el uso de un algoritmo genético de ordenación no dominada NSGA II adaptado al caso de estudio y busca mejorar los resultados y evidenciar que este tipo de solución se puede emplear para la solución de problemas de mayor complejidad. Para validar el método de solución propuesto se realizó un diseño experimental de un factor para tres factores con el fin de determinar su influencia en el algoritmo, además de la calidad y cantidad de las soluciones halladas.

Facultad de Ingeniería Fisicomecánica. Escuela de estudios industriales y empresariales. Director: Julio Cesar Camacho Pinto, Magister en Ingenieria industrial. Co-director: Edwin Alberto Garavito, Magister en Ingenieria industrial.

Abstract

Title: Multi-objective genetic algorithm for the design of a closed-loop supply chain for the tire industry with stochastic demand.*

Authors: Paula Andrea Castellanos Peña y Johan Sebastian Hernandez Ortiz

Key Words: closed loop supply chain, multiobjective, stochastic programming, location and allocation problema, NSGAII.

Description:

Responsible production and consumption is one of the sustainable development goals of the united nations which mainly seeks to decouple economic growth from environmental degradation and increase resource efficiency, for this reason the new challenge for industries is to effectively coordinate and manage the collection and treatment of waste products. The purpose of this research work is to propose and evaluate an alternative solution to a stochastic and multi-objective mathematical model of a closed-loop supply chain of an Iranian tire manufacturing company.

The model is characterized by simultaneously integrating the economic and environmental approach while evaluating the responsiveness of the chain. The stochastic component of the model is considered on the demand side. The proposed solution for the model is the use of a non-dominated sorting genetic algorithm NSGA II adapted to the case study and seeks to improve the results and demonstrate that this type of solution can be used to solve problems of greater complexity. To validate the proposed solution method, a one-factor experimental design for three factors was carried out to determine their influence on the algorithm, as well as the quality and quantity of the solutions found.

Faculty of Physicomechanical Engineering. School of Industrial and Business Studies. Director: Julio Cesar Camacho Pinto, Master in Industrial Engineering. Co-director: Edwin Alberto Garavito, Master in Industrial Engineering.

Introducción

La producción y el consumo responsable es uno de los objetivos de desarrollo sostenible de las naciones unidas el cual busca desvincular el crecimiento económico de la degradación medioambiental, aumentar la eficiencia de recursos y promover estilos de vida sostenibles; así como contribuir a la mitigación de la pobreza y a la transición hacia economías verdes y con bajas emisiones de carbono. (Unidas, 2030)

En la actualidad, la mayoría de las cadenas de suministro no están diseñadas para realizar la recolección durante el post-consumo mediante un flujo inverso. Por lo tanto, el nuevo reto para las industrias consiste en coordinar y administrar efectivamente la gestión de la recolección y tratamiento de los productos desechos. Las dificultades durante este proceso se deben primero, a que los productos devueltos se generan en cantidades, calidades y tiempos inciertos y segundo, a que las empresas no se enfocan en diseños fáciles de desensamblar o remanufacturar. Dadas estas limitaciones surge el concepto de CLSC, el cual busca reincorporar materiales, productos y empaques una y otra vez en el ciclo productivo. (Ortiz, 2004)

Dentro de la ingeniería industrial existe una rama enfocada en la toma de decisiones mediante el modelo matemático de funciones objetivo denominada investigación de operaciones, la cual busca representar mediante ecuaciones matemáticas la realidad con el fin de buscar sino la mejor un conjunto de buenas soluciones para tomar decisiones. Esta rama requiere de un estudio detallado de variables, factores y relaciones presentes de la realidad que se desea modelar, lo cual implica la revisión de modelos afines y la caracterización del problema en escenarios específicos.

Teniendo en cuenta lo anterior, el presente trabajo tiene como finalidad plantear una solución alterna a la problemática existente en un caso práctico en el que Seyed Babak Ebrahimi (2018) realiza el diseño de una red de cadena de suministro de bucle cerrado teniendo en cuenta los aspectos de sostenibilidad y los descuentos por cantidad bajo incertidumbre en un caso de estudio situado en Irán y propone un modelo de optimización estocástica multiobjetivo para la selección de los proveedores y los problemas de localización-asignación-ruta que luego soluciona con el método de las restricciones, mientras que en esta investigación se propone como método de

solución un algoritmo genético (AG) basado en el NSGA-II y adaptado a las necesidades del problema de programación.

El diseño de una cadena de suministro integrada hacia adelante y hacia atrás es un tema muy reciente y relativamente inexplorado en Colombia, por lo que se desea con investigaciones de este tipo despertar el interés en estos temas a nivel nacional.

Este documento se encuentra estructurado de la siguiente forma: en el capítulo 4 se encuentra la revisión de literatura relacionada con el objeto de estudio y en el capítulo 5 se presenta el marco teórico; en el capítulo 6 se encuentra la descripción del problema y la metodología utilizada, en el capítulo 7 está el algoritmo de solución propuesto y su aplicación. Finalmente, en el capítulo 8 está la validación del algoritmo propuesto, en los capítulos 9 y 10 se encuentran las conclusiones y recomendaciones respectivamente.

Tabla 1.

Cumplimiento de los objetivos

Objetivos específicos	**Cumplimiento**
Identificar un modelo de programación lineal entera mixta, estocástico y multiobjetivo para el problema de localización, asignación y ruteo en cadenas de suministro de circuito cerrado a partir de una revisión de literatura.	Numeral 6
Desarrollar un algoritmo genético NSGA-II para la solución del modelo matemático del problema de la cadena de suministro de circuito cerrado.	Numeral 7 Apéndice B
Evaluar la eficiencia del algoritmo propuesto con una instancia de la literatura.	Numeral 8
Elaborar un artículo de carácter publicable en un medio de difusión académica, con base en los resultados obtenidos en la investigación.	Apendice D

1. Planteamiento del problema

En la actualidad, la mayoría de las cadenas de suministro no están diseñadas para realizar la recolección durante el post-consumo mediante un flujo inverso. Por lo tanto, el nuevo reto para las industrias consiste en coordinar y administrar efectivamente la gestión de la recolección y tratamiento de los productos desechos. Además de implementar la logística inversa en las redes que suministran productos al consumidor, las empresas deben asegurar que se produzca el mínimo impacto ambiental debido a la operación realizada durante las actividades de devolución, recolección y tratamiento. Finalmente, para que la gestión directa e inversa se dé de forma eficiente y eficaz es imperativo aumentar la capacidad de respuesta de toda la red en su flujo directo e inverso.

Los países miembros de la Organización de Cooperación y Desarrollo Económicos (OCDE) deben contar con políticas gubernamentales internas que aseguren la buena gestión de los residuos sólidos que se generen, promoviendo la aplicación del principio de Responsabilidad Extendida del Productor (RPE) (Banguera et al., 2018), por esta razón, los fabricantes de neumáticos han tenido que diseñar sistemas de gestión de logística inversa que les permitan disminuir los residuos generados.

Las dificultades durante este proceso se deben primero, a que los productos devueltos se generan en cantidades, calidad y tiempos inciertos y segundo, a que las empresas no se enfocan en diseños fáciles de desensamblar o remanufacturar. Dadas estas limitaciones surge el concepto de Cadena de Suministro de Ciclo Cerrado (siglas en inglés CLSC), el cual busca reincorporar materiales, productos y empaques una y otra vez en el ciclo productivo. (Ortiz, 2004)

La intención inicial se basaba en diseñar un modelo matemático que permitiera modelar a través de una cadena de suministro de circuito cerrado el ciclo de vida de los neumáticos en Colombia, pero debido a la escasez de información divulgada, se evidenció la falta de interés por este tema en el país. Por lo anterior, se recurrió a abordar el problema basado en un modelo ya propuesto para una situación específica en Irán, siendo el propósito actual comparar dos métodos de solución, con el fin de sugerir el más adecuado para este tipo de modelo.

El autor en su artículo sugiere para investigaciones futuras, resolver el modelo original a través de métodos exactos como programación por objetivo, LP-métricas o por medio del desarrollo de heurísticas y metaheurísticas. Por consiguiente y basados en la revisión de la literatura, se decide resolver el problema planteado a través de un algoritmo genético de clasificación no dominado (NSGA-II), debido a que ofrece una mayor flexibilidad, sencillez y una robusta aplicación, además porque es apropiado para resolver problemas de grandes instancias.

2. Justificación del proyecto

Actualmente el desarrollo sostenible y amigable con el ambiente juega un papel importante en la economía. Las empresas no sólo se preocupan por ganar dinero sino también por el efecto que genera el post-consumo sobre el entorno. Por esto se han desarrollado procesos logísticos inversos para disminuir el efecto adverso. La industria responsable de la fabricación de neumáticos es una gran generadora de residuos peligrosos debido a la naturaleza del producto. Por tal razón, el diseño de sus redes inversas debe tener en cuenta procesos de recolección y transformación que permitan recuperar el producto una vez se termina su vida útil, para ofrecerlo al mercado nuevamente o tener otra disposición.

Durante este proceso inverso se debe garantizar el mínimo impacto ambiental, el mínimo costo y el máximo impacto social. Por lo tanto, en necesario desarrollar modelos matemáticos que modelen la situación y que tengan presente la incertidumbre de cada uno de los escenarios, así como métodos de solución que permitan a los obtener el mejor resultado de acuerdo a sus intereses primordiales.

3. Objetivos

3.1. Objetivo General

Implementar un algoritmo genético multiobjetivo para el diseño de una cadena de suministro de circuito cerrado para la industria de neumáticos con demanda estocástica.

3.2. Objetivos Específicos

Identificar un modelo de programación lineal entera mixta, estocástico y multiobjetivo para el problema de localización, asignación y ruteo en cadenas de suministro de circuito cerrado a partir de una revisión de literatura.

Desarrollar un algoritmo genético NSGA-II para la solución del modelo matemático del problema de la cadena de suministro de circuito cerrado.

Evaluar la eficiencia del algoritmo propuesto con una instancia de la literatura.

Elaborar un artículo de carácter publicable en un medio de difusión académica, con base en los resultados obtenidos en la investigación.

4. Revisión de la literatura

4.1. Análisis bibliométrico

Para esta revisión literaria se realizó una búsqueda dirigida a la gestión de una cadena de suministro de bucle cerrado considerando la localización y asignación de las instalaciones, así como las rutas necesarias para el funcionamiento de la cadena, haciendo uso de bases de datos multidisciplinarias disponibles en la Universidad Industrial de Santander, concretamente se consultó en Scopus y ISI Web of Science (WoS).

Se proponen dos ecuaciones de búsqueda que incluyen términos considerados como claves, pues representan directamente los aspectos de mayor interés que se pretenden abarcar en esta investigación (Ver Figura 1).

Figura 1.

Ecuación de búsqueda

Scopus: TITLE-ABS-KEY (("closed loop" OR "closed-loop" OR "supply chain" OR "network design" OR "reverse logistics") AND (multiobjective OR "multi-objective" OR "stochastic programming" OR "linear programming") AND ("routing problem" OR "location and allocation problem" OR "location allocation problem"))
Web of Science: TI: (("closed loop" OR "closed-loop" OR "supply chain" OR "network design" OR "reverse logistics") AND (multiobjective OR "multi-objective" OR "stochastic programming" OR "linear programming") AND ("routing problem" OR "location and allocation problem" OR "location allocation problem"))

En ambas ecuaciones se requiere encontrar artículos relacionados con el tema principal, cadenas de suministro de circuito cerrado (CLSC), razón por la cual se incluyen los términos “closed loop”, “supply chain”, “network desing” y “reverse logistics”.

A su vez, en la estructura de la ecuación se limita a una búsqueda de artículos que cuenten con los siguientes términos “multiobjective”, “stochastic programming”, “linear programming”

“routing problema”, “location and allocation problema”, “location allocation problema” los tres primeros con el fin de definir la estructura del modelo matemático y las tres siguientes con el propósito de enfocar el modelo a los problemas que se pretende abordar en esta investigación.

En la estructura de las ecuaciones se utiliza la función comillas (“”) para incluir términos compuestos y a su vez garantizar que se encuentren las palabras literalmente, también se hace uso de palabras compuestas repetidas ya que pueden ser escritas por algunos autores de manera diferente, ejemplo “closed-loop” y “closed loop”. De igual modo, se usaron sinónimos en aras de ampliar los resultados en la búsqueda.

En las dos ecuaciones ingresadas a las dos bases de datos se limitó la búsqueda para obtener resultados que incluyan los términos establecidos en su título, resumen o palabras claves, para Scopus el código de campo usado fue “TITLE-ABS-KEY” y para WoS fue “TOPIC”

El total de resultados obtenidos se puede observar en la tabla 1, entre estos se incluyen artículos, capítulos de libros, documento y revisiones de conferencias. Esta búsqueda se realizó con todos los años disponibles en cada base de datos (2001-2020 WoS y 1976-2020 Scopus).

Tabla 2.

Número de artículos encontrados con las ecuaciones

Base de datos	Scopus	Web of Science	Resultado total
Resultados encontrados	273	259	532

Al realizar un análisis superficial de los resultados encontrados en las dos bases de datos, a criterio de los autores, se encontraron publicaciones que no tenían relación alguna con el tema de interés de esta investigación, obteniendo así los resultados que se presentan en la tabla 2.

Tabla 3.

Número de artículos encontrados después de depuración

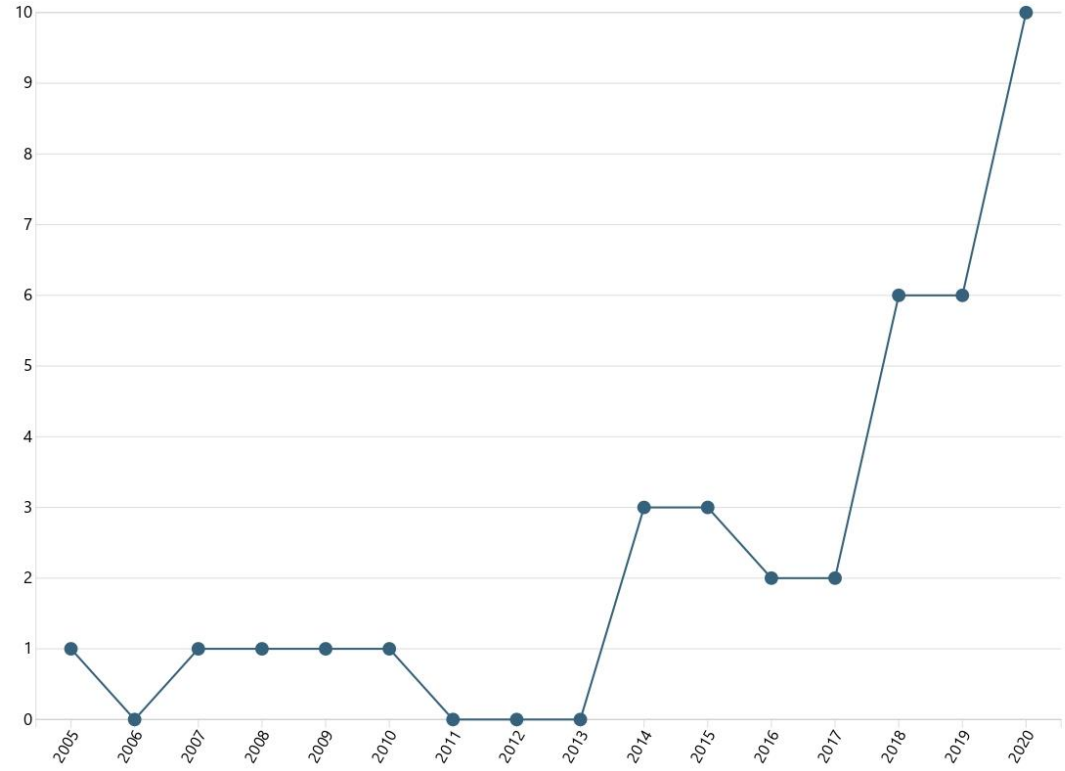
Base de datos	Scopus	Web of Science	Resultado total
Resultados obtenidos	25	24	49

Se obtuvo un total de 49 artículos, de los cuales 10 artículos se identificaron en ambas bases de datos, dejando un total de 39 artículos seleccionados para realizar el análisis presentado a continuación utilizando el software VantagePoint.

En primer lugar, se realizó un análisis de las publicaciones hechas entre 2005 y 2020. En la *figura 2* se puede apreciar un comportamiento volátil con tendencia estacional hasta el año 2016, a partir del 2017 se evidencia un crecimiento en el número de publicaciones con esta temática de estudio.

Figura 2.

Publicaciones por año

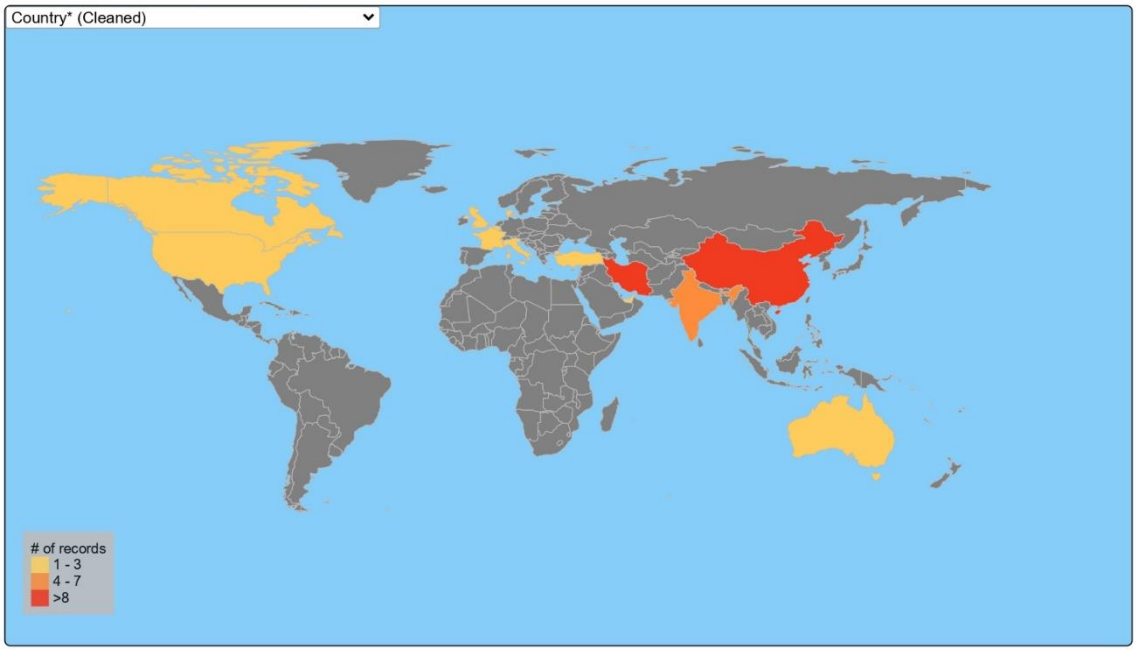


Nota. Gráfico obtenido con el Software Vantagepoint.

En la *figura 3*, se aprecian los países donde se han realizado aportes acerca del tema de estudio. Se puede observar que en Irán y China se concentra el mayor número de producciones científicas, con 16 y 14 publicaciones respectivamente, seguido por India con cuatro. Finalmente, en Estados Unidos, Francia, Dinamarca y Reino Unido se registran de a dos aportes; y Canadá, Italia, Turquía, Australia y Emiratos Árabes Unidos con sólo una elaboración. Por otra parte, cabe resaltar que no se han realizado estudios en Latino América acerca de este tópico.

Figura 3.

Países con mayor número de publicaciones

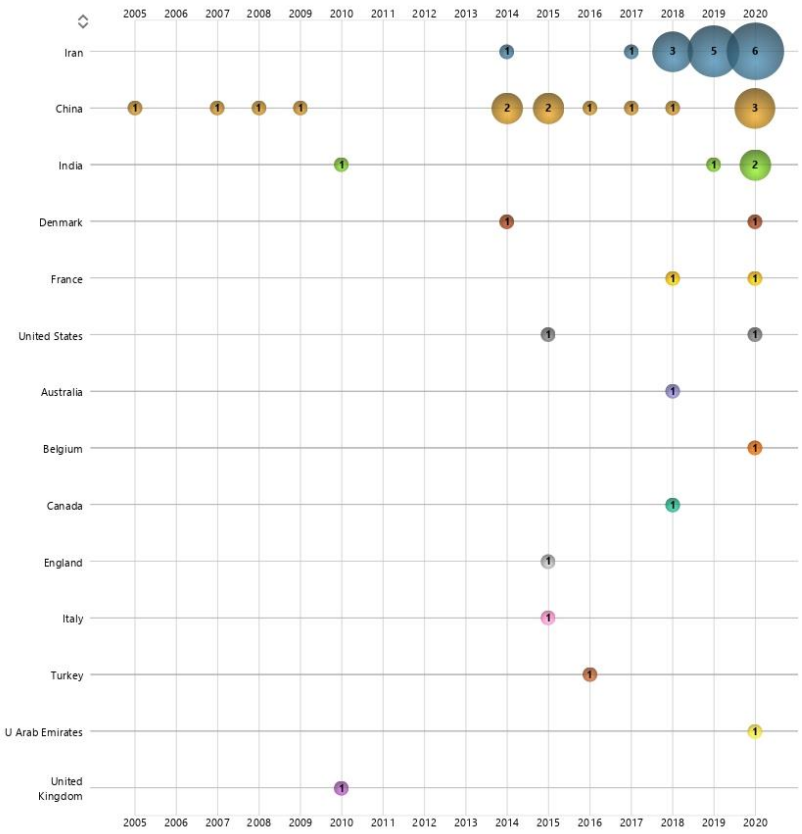


Nota. Gráfico obtenido con el Software Vantagepoint.

En la *figura 4*, se confirma como China, ha venido realizando paulatinamente estudios desde el año 2005, con una producción científica constante hasta el 2020. Por su parte Irán, publicó en 2014 su primer estudio y desde el 2018 hasta la actualidad ha aumentado el número de publicaciones anualmente.

Figura 4.

Cantidad de publicaciones por año y por país

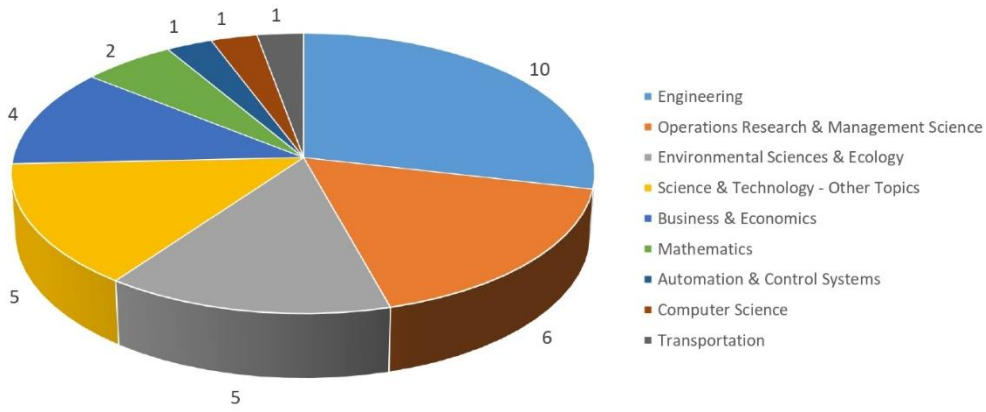


Nota. Gráfico obtenido con el Software Vantagepoint.

En esta revisión literaria los artículos estudiados pertenecen es su mayoría al área de Ingeniería y a Investigación de operaciones y Ciencias de la Gestión, como se evidencia en la *figura 5.* Lo cual se debe a que el tema de estudio es de gran interés y tiene un gran aporte tanto en la ingeniería como en la investigación de operaciones. Para este análisis es importante resaltar que un artículo puede pertenecer a más de una categoría.

Figura 5.

Publicaciones por categoría

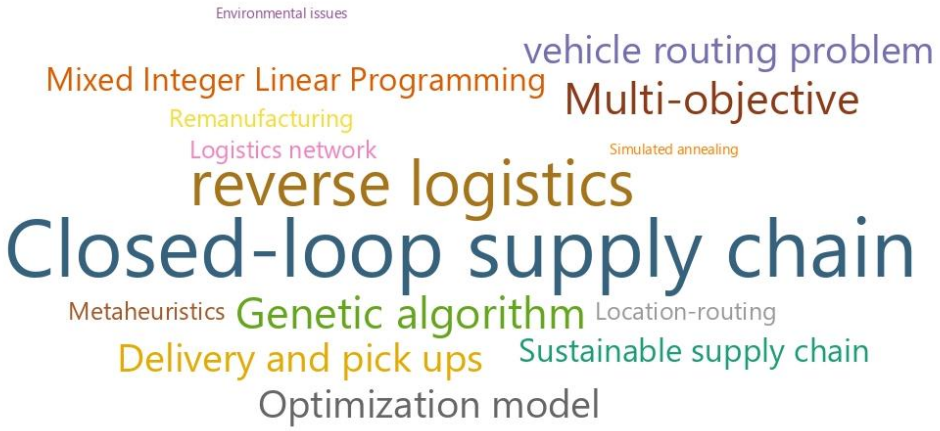


Nota. Gráfico adaptado del Software Vantagepoint.

Continuando con el análisis bibliométrico, encontramos que las palabras clave para el tema de estudio son 15, como se muestra en la nube de palabras de la *figura 6*. Entre las cinco palabras más destacadas se encuentran: cadena de suministro de circuito cerrado, logística inversa, multi-objetivo, algoritmo genético y programación lineal entera mixta.

Figura 6.

Nube de palabras clave



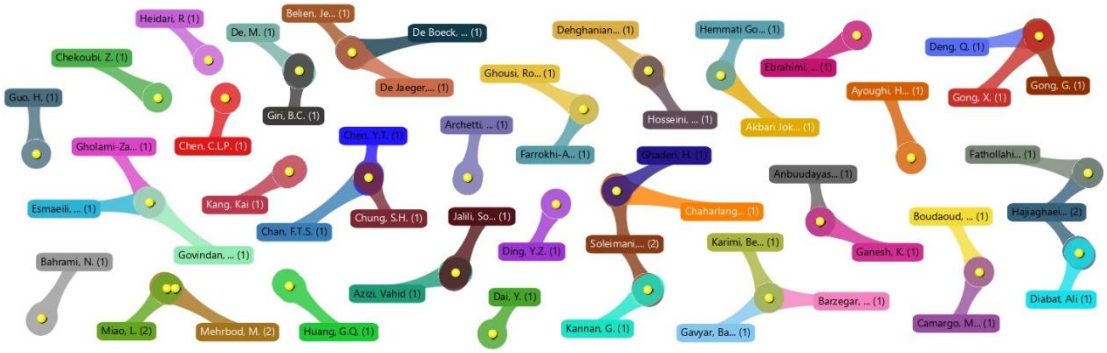
Nota. Gráfico adaptado del Software Vantagepoint.

Se realiza el gráfico de aduna para analizar la conexión entre los autores, el cual permite observar colaboraciones en el desarrollo de estudios. En la *figura 7*, se puede observar que, en su

gran mayoría, se presentan colaboraciones entre no más de tres autores, también se evidencian algunos trabajos individuales. Esta aduna incluyó el 50% de los autores debido a que los trabajos en su mayoría eran individuales.

Figura 7.

Colaboraciones entre autores

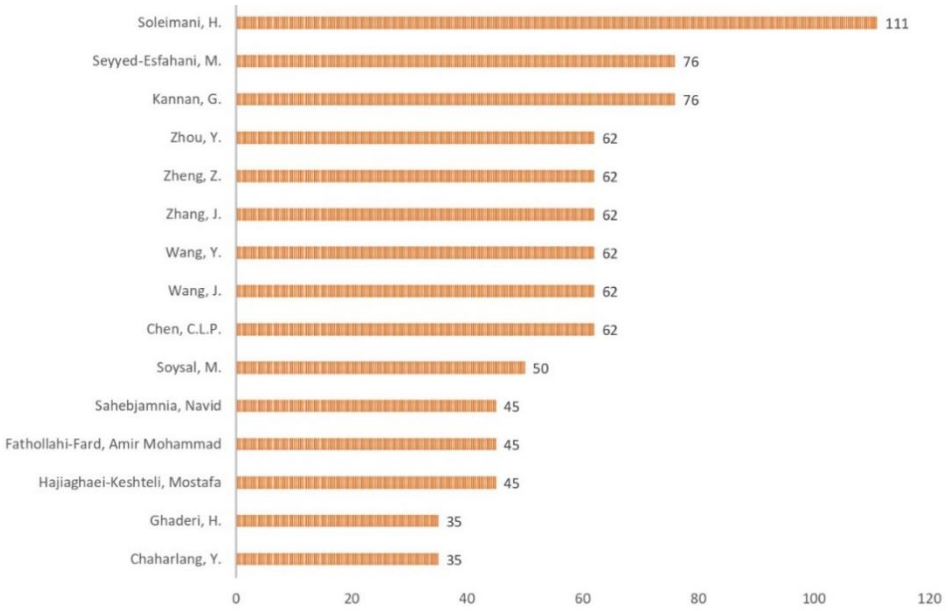


Nota. Gráfico adaptado del Software Vantagepoint.

Por último, en la *figura 8* se observan los autores con mayor número de citaciones. Entre los más citados, se encuentra Soleimani, Seyyed-Esfahani y Kannan.

Figura 8.

Número de citaciones por autor



Nota. Gráfico adaptado del Software Vantagepoint.

4.2. Análisis preliminar de la literatura

Las cadenas de suministro inicialmente se entendían como un conjunto de actividades, instalaciones y medios de distribución, ya sea por una o varias empresas, que intervienen en el proceso de transformación de materia prima y transporte de producto terminado a través de distintas fases, hasta llegar a satisfacer los requerimientos del cliente o consumidor final. (J. Ortiz, 2004).

Recientemente, debido a las crecientes preocupaciones ambientales y sociales, un número cada vez mayor de empresas se ha centrado en la logística inversa además de la logística directa (Mehrbod, Tu, y Miao, 2015). Por tal motivo, las empresas y los gobiernos ejerciendo una visión de responsabilidad compartida y entendiendo que el proceso de recuperación de productos y no es una tarea puramente logística, surge el concepto de Cadena de Suministro de Ciclo Cerrado.

Según Ortiz (2004), la Cadena de Suministro de Ciclo Cerrado buscan facilitar la continua reincorporación de los materiales, productos y empaques una y otra vez en el proceso productivo o darles una disposición final amigable con el ambiente.

El estudio de las Cadenas de Suministro de Ciclo Cerrado o CLSC, por sus siglas en inglés, considera la unificación de varios problemas clásicos como, la localización, la asignación y el ruteo, además de las variantes propias de cada problema en específico. Por tal motivo, a continuación, se definen algunos modelos, métodos de solución y resultados encontrados en la literatura.

4.2.1. Módelos matemáticos

Ya Peng y Zhong (2007); Ya Peng y Zhong (2008) y Yapeng (2009) analizan el diseño de la red logística integrada para el sistema de fabricación verde. Se propone un modelo de optimización para el diseño de la red de logística integrada que maneja devoluciones de productos sobre la base de un enfoque de programación lineal de enteros mixtos. Según el modelo, se puede decidir el número de varias instalaciones, sus ubicaciones y la asignación de los flujos logísticos correspondientes y configurar la estructura de red logística adecuada para minimizar el costo total de inversión y operación.

Soleimani, Seyyed-esfahani y Kannan (2014) consideran un problema de ubicación-asignación en una cadena de suministro de circuito cerrado (CLSC) con dos extensiones: primero, la demanda y los precios de productos nuevos y devueltos se consideran parámetros no deterministas y, segundo, la función objetivo se desarrolla a partir de la ganancia esperada a tres tipos de riesgo medio. Su artículo aborda el problema de diseño y planificación de un CLSC en una estructura estocástica de dos etapas. Además, los criterios de riesgo se consideran mediante el uso de tres tipos de medidas de riesgo populares y de buen comportamiento: desviación media absoluta, valor en riesgo y valor condicional en riesgo (CVaR).

Chen, Chan y Chung (2015) formulan un modelo de programación lineal, dicho modelo se basa en un estudio de caso real sobre la entrega y el reciclaje de cartuchos de tinta y tóner en Hong Kong. El modelo propuesto contiene ocho socios en la red CLSC: proveedores, fabricantes, almacenes, minoristas, clientes, puntos de recogida, centros de reciclaje y centros de eliminación de residuos.

Mehrbod, Xue, Miao y Lin (2015), en su trabajo, consideran una red logística de ciclo cerrado multiproducto de múltiples períodos con respecto a la expansión de instalaciones como un problema de ubicación-asignación de instalaciones. Además, desarrollan una formulación de programación no lineal entera mixta de objetivos múltiples para minimizar el costo total, el tiempo de entrega del producto y el tiempo de recolección del producto usado.

Soysal (2016) propone un modelo probabilístico de programación lineal de enteros mixtos para el problema de enrutamiento de inventario de circuito cerrado (CIRP) que da cuenta de las operaciones de logística hacia adelante y hacia atrás, el consumo explícito de combustible, la incertidumbre de la demanda y múltiples productos.

Wang et al. (2016) estudian una variante práctica del problema de enrutamiento de vehículos (VRP), llamado VRP con entrega y recolección simultáneas y ventanas de tiempo (VRPSDPTW), en la industria de la logística. En su artículo, definen un VRPSDPTW multiobjetivo general (MO-VRPSDPTW) con cinco objetivos, usando instancias del mundo real.

Soleimani, Chaharlang y Ghaderi (2018) abordan el Problema de Ruteo de Vehículos Ecológicos (GVRP) considerando la distribución de productos originales y remanufacturados (entrega y recogida), es decir, el distribuidor entrega los productos de primera mano y recoge los productos de segunda mano del mercado para repararlos. Posteriormente, el distribuidor vuelve a entregar al mercado los productos de primera mano y los de segunda mano reparados (a menor

precio). Los autores proponen un modelo no lineal multiobjetivo en el que se busca reducir los costos asociados a la distribución, mientras se minimizan las emisiones producidas por los vehículos involucrados en el sistema de distribución.

Chekoubi, Trabelsi y Sauer (2018) abordaron una extensión del Problema Integrado de Producción-Inventario-Enrutamiento (IPIRP) en el cual consideraron la recuperación y refabricación de productos al final de su vida útil como un aspecto sostenible. Los autores proponen una formulación de programación matemática para el problema considerado, donde el objetivo es determinar las cantidades óptimas de bienes (tamaños de lote) a fabricar, remanufacturar y almacenar cumpliendo con las solicitudes de los clientes (entregas y recogidas) con un coste total mínimo.

Gong et al. (2018) en su trabajo construyen una red de logística de cadena de suministro de circuito cerrado de problemas de rutas de vehículos con recolecciones y entregas simultáneas (VRPSPD) dominada por remanufacturadores, en la que los clientes se dividen en tres tipos: distribuidores, recicladores y proveedores. Además, el consumo de combustible se agrega a los objetivos de optimización del VRPSPD propuesto.

Sadeghi y Mina (2019) desarrollan un modelo de programación lineal de enteros mixtos (MILP) para un diseño de red de cadena de suministro de circuito cerrado (CLSC). El modelo propuesto es de múltiples períodos y productos que cubre la ubicación de las instalaciones y el enrutamiento de flotas de vehículos heterogéneos, simultáneamente. La función objetivo del problema minimiza siguientes costos: contactar al proveedor, establecer los centros, fabricación y procesamiento del producto en los centros, transporte de materias primas y productos entre los niveles, compra de vehículos y su consumo de combustible.

Zhang, Pratap, Zhao, Prajapati y George (2020) abordan un problema de generación de rutas para vehículos con recolección y entrega simultáneas con ventanas de tiempo desde múltiples depósitos (MVRPSPDTW) en un horizonte de tiempo en el sistema de logística de comercio electrónico B2C. Para ello, se desarrolla y prueba un modelo de programación no lineal de enteros mixtos (MINLP). El modelo propuesto aborda muchas características de relevancia práctica, tales como, un entorno de varios depósitos, una flota heterogénea de vehículos, varios períodos, costos operativos variables y una función objetivo de costo mínimo.

Hemmati y Akbari (2020) proponen el problema de producción-inventario-enrutamiento con recolección y entrega simultáneas, considerando un flujo inverso de productos. Los autores

plantean dos modelos en Programación Lineal Entera Mixta (MILP) para reducir el costo total del sistema; uno determinista y otro en la que la demanda de los minoristas, la cantidad de productos defectuosos recolectados y los costos de producción manejan condiciones de incertidumbre.

De y Giri (2020) estudian una cadena de suministro de circuito cerrado (CLSC) que se centra en la gestión, programación y enrutamiento de problemas para lograr la sostenibilidad económica y ambiental. El objetivo consiste en minimizar el costo total de transporte, incluida la emisión de carbono de una flota heterogénea con capacidad limitada. En su trabajo, los autores proponen un modelo que aborda cuatro políticas ambientales para las regulaciones de emisiones de carbono. El modelo se formula en el marco de la programación no lineal de enteros mixtos (MINLP).

4.2.2. Métodos de solución

Chen, Chan y Chung (2015) proponen un algoritmo genético modificado de dos etapas (GA), en la cual, la decisión de ruta y decisión de volumen de carga se encargan de codificar las soluciones iniciales del algoritmo. La etapa de decisión de ruta puede decidir la ruta de entrega entre cada nivel de la red de la cadena de suministro. Luego, el volumen de carga en cada ruta de entrega seleccionada se determina en la etapa de decisión del volumen de carga.

Mehrbod, Xue, Miao y Lin (2015) proponen un algoritmo genético basado en prioridades para resolver su problema. En dicho artículo comparan el rendimiento de los algoritmos genéticos basados en prioridad directa y basados en prioridad inicial, demostrando que el primero suele ser más eficiente en términos de la calidad de las soluciones.

Wang et al. (2016) diseñan, implementan y comparan dos algoritmos, búsqueda local multiobjetivo (MOLS) y algoritmo Memético multiobjetivo (MOMA) para la resolución de MO-VRPSDPTW.

Soleimani, Chaharlang y Ghaderi (2018) en su modelo multiobjetivo, usan variables de tipo binario y entero para linealizar la función objetivo y sus restricciones. Mientras que los subtours se eliminaron usando Ciclos Hamiltonianos con el mínimo peso en un gráfico totalmente ponderado.

Chekoubi, Trabelsi y Sauer (2018) plantean un modelo MILP para el problema, el cual es resuelto para instancias pequeñas usando un método de optimización exacta usando el software IBM ILOG CPLEX 12.7.

Gong et al. (2018) proponen un algoritmo evolutivo de abejas (BEG-NSGA-II) con un mecanismo de optimización de dos etapas diseñado para resolver el modelo VRPSPD propuesto con tres objetivos de optimización: consumo mínimo de combustible, tiempo mínimo de espera y distancia de entrega más corta. En la primera etapa, se utiliza un algoritmo genético evolutivo de abejas mejorado para optimizar la población inicial, mientras que, en la segunda etapa, se utiliza un NSGA-II mejorado. En su trabajo, los autores manejan una población auxiliar la cual les permite evolucionar a las soluciones no factibles.

Sadeghi y Mina (2020) plantean un modelo en programación no lineal, el cual, mediante el uso de variables auxiliares se encargan de transformar las expresiones no lineales del modelo propuesto. Su modelo es resuelto mediante el uso de la herramienta GAMS.

Zhang, Pratap, Zhao, Prajapati y George (2020) en su modelo matemático, el cual minimiza el costo total de transporte de la red y la penalización por entrega tardía en el punto de destino. Para el escenario evaluado, utilizaron dos enfoques; primero, el enfoque de optimización exacta, CPLEX, y segundo, el enfoque metaheurístico, mediante el uso de los algoritmos Evolutivo Diferencial (DE), Evolutivo Diferencial Paralelo (Par-DE), Genético (GA) y Genético Basado en Bloques (BBVA) para abordar la complejidad del problema.

Hemmati y Akbari (2020) en su trabajo, resuelven los modelos de programación lineal entera mixta mediante la herramienta GAMS, usando instancias aleatorias. Sin embargo, considerando la caracterización NP Hard del problema, los autores plantean dos algoritmos metaheurísticos, el Algoritmo Genético y el Algoritmo de Recocido Simulado. El ajuste de parámetros de realiza mediante el método Taguchi.

De y Giri (2020) transforman el modelo MINLP en un modelo MILP sustituyendo el producto de las variables por una nueva variable asumida. Para solventar los conflictos presentados por los objetivos incompatibles usan el método Lp-metrics para resolver los modelos propuestos a través del modelo multiobjetivo convertido en un solo objetivo. Finalmente, el problema es resuelto mediante el software IBM CPLEX 12.6.

4.2.3. Resultados obtenidos

Chen, Chan y Chung (2015), revisando la aplicación del modelo reciente, utilizaron la programación de enteros para resolver las instancias de prueba y compararla con el algoritmo propuesto. Los resultados muestran que el GA propuesto puede obtener una solución casi óptima en un tiempo de cálculo mucho más corto.

Soysal (2016) aplicó su modelo sobre las operaciones de distribución de una empresa de refrescos. Los resultados sugieren que el modelo propuesto puede lograr ahorros significativos en el costo total y, por lo tanto, ofrece un mejor apoyo a los tomadores de decisiones.

Wang et al. (2016) presentaron resultados de simulación en instancias del mundo real propuestas e instancias tradicionales, mostrando que MOLS supera a MOMA en la mayoría de los casos. Sin embargo, la superioridad de MOLS sobre MOMA en instancias del mundo real no es tan marcada en los casos de la vida real.

Soleimani, Chaharlang y Ghaderi (2018), con el fin de validar el modelo propuesto en circunstancias reales, identificaron un estudio de caso del mundo real que cumplió con los objetivos y limitaciones de su investigación. El estudio de caso comprende un sistema de distribución para uno de los periódicos iraníes más conocidos, Hamshahri. Los resultados obtenidos indican que el modelo matemático propuesto es capaz de reducir el costo de combustible, el costo de instalación del centro de distribución y el suministro de vehículos, así como minimizar la contaminación del aire.

Chekoubi, Trabelsi y Sauer (2018) en su trabajo obtienen que limitados a un tiempo máximo de 3600 segundos, no es posible encontrar soluciones optimas en algunas instancias, por lo tanto, aseguran que es necesario implementar métodos aproximados que permitan obtener soluciones factibles en tiempos computacionales relativamente cortos.

Gong et al. (2018) determinan que, el algoritmo propuesto de BEG-NSGA-II muestra un mejor desempeño en la optimización multiobjetivo además de cumplir con la optimización de objetivo único. Además, proponen profundizar su aplicación en problemas de optimización multiobjetivo, especialmente en el campo de la distribución logística.

Sadeghi y Mina (2020) implementan el modelo propuesto en la cadena de producción y distribución de una empresa de fabricación de repuestos para automóviles en Irán. Los resultados determinan la cantidad óptima de compra a los proveedores, las ubicaciones óptimas a establecer,

las rutas óptimas y la cantidad de productos transferidos entre los niveles. Los resultados de la ejecución del modelo en el mundo real indican la función exacta y precisa del modelo propuesto.

Zhang, Pratap, Zhao, Prajapati y George (2020) en su caso de estudio, utilizaron datos de dos meses (octubre de 2016) de una empresa de comercio electrónico para resolver el modelo y probar en un escenario práctico. Según los resultados obtenidos CPLEX es capaz de obtener soluciones optimas, sin embargo, dado el alto tiempo computacional requerido, algunas instancias no son evaluadas por este método. El algoritmo diferencial paralelo (Par-DE) demuestra un mejor desempeño que los algoritmos DE, GA y BBGA en todos los casos considerados.

Hemmati y Akbari (2020), proponen tres instancias, pequeña, mediana y grande. Para cada tamaño, se consideraron tres estados diferentes: el estado estándar, el estado con alto costo de transporte y el estado con alto costo de inventario; en donde el tamaño se define por el número de clientes. Respecto al desempeño de los algoritmos se determina que el algoritmo genético resulta tener mayor eficiencia que el de Recocido Simulado.

De y Giri (2020) verifican los resultados obtenidos mediante ejemplos numéricos, donde la red CLSC incluye tres fabricantes, dos depósitos, tres centros de distribución / acopio, diez clientes diferentes en un mercado para dos tipos de producto y 3 tipos de camiones según su capacidad de carga. Además, presentan un análisis integral para monitorear la tasa de devolución de productos y varios parámetros de transporte de la flota.

5. Marco teórico

A continuación, se presentan varios temas de relevancia relacionados con el proyecto.

5.1. Optimización combinatoria

“La optimización combinatoria es una rama de la optimización en matemáticas aplicadas y la ciencia de la computación, la cual estudia los problemas que se caracterizan por presentar una cantidad finita de soluciones factibles y trabajar con variables discretas”. (Blum & Roli, 2003)

Blum y Roli (2003) definen un problema de optimización combinatoria como:

Sea $P = (S, f)$ un problema de optimización combinatoria

- Un conjunto de variables $X = \{x_1, \dots, x_n\}$;
- Dominio de las variables $D_1, \dots, D_n$;
- Restricciones entre variables;
- Una $función\ objetivo\ f$ a ser minimizada, donde $f: D_1 \times \dots \times D_n \rightarrow \mathbb{R}^+$;

El conjunto de todas las posibles asignaciones factibles es:

$$S = \{s = \{(x_1, v_1), \dots, (x_n, v_n)\} \mid v_i \in D_i, s \text{ satisface todas las restricciones}\}$$

S es el espacio de búsqueda o espacio de solución, donde cada elemento del conjunto puede ser visto como una solución candidata. Para resolver un problema de optimización combinatoria, debe encontrarse una solución $s^* \in S$ con el mínimo valor de función objetivo, $f(s^*) \leq f(s)\ \forall s \in S$. s^* es llamado una solución óptima global de (S, f) y el conjunto $S^* \subseteq S$ es llamado el conjunto de soluciones óptimas globales.

Algunos ejemplos de problemas de optimización combinatoria son el problema del Vendedor Viajero (TSP), el problema de Asignación Cuadrática (QAP) y problemas de Programación.

5.2. Complejidad computacional

Dentro de la teoría de la computación, una de las áreas fundamentales es la teoría de la complejidad, la cual se enfoca en clasificar los problemas computacionales de acuerdo a su complejidad y la relación entre las distintas clases de complejidad, es decir la cantidad de recursos computacionales que requiera una solución sin importar el algoritmo que se esté utilizando. Para ello, considera 2 tipos de recursos requeridos durante el cómputo para resolver el problema:

- Tiempo: número de pasos de ejecución de un algoritmo para resolver un problema.
- Espacio: Cantidad de memoria computacional utilizada para resolver el problema.

La complejidad de un algoritmo se expresa como una función del tamaño de la entrada del problema, T(n): ratio de tiempo de ejecución y E(n): ratio de espacio de almacenamiento necesario.

Según Sanchis, Ledezma, Iglesias, García, y Alosnso (s. f.) para determinar el nivel de complejidad de un problema de decisión se utiliza una Máquina de Turing (Determinista o No Determinista) y los clasifica como:

- Clase P: Son los problemas de decisión que una Máquina de Turing Determinista puede resolver en tiempo polinómico. Estos problemas son tratables, es decir, en la práctica pueden resolverse en un tiempo razonable.
- Clase NP: Conjunto de problemas de decisión que una Maquina de Turing No Determinista puede resolver en un tiempo polinómico. Se caracterizan porque son el tipo de problemas en los cuales las instancias donde la respuesta es afirmativa tienen demostraciones verificables por cálculos determinísticos en tiempo polinómico. El tiempo de cómputo que se requiere para resolver este tipo de problemas se incrementa conforme crece el tamaño del problema presentando una dependencia funcional que no admite ser acotada por un polinomio. Los problemas NP pueden dividirse en problemas NP-complete y NP-hard.

El problema NP-complete se caracteriza por ser un problema NP, todos los problemas NP pueden reducirse a un NP-complete en tiempo polinómico. El problema NP- hard no tiene un algoritmo polinómico que lo resuelva, es así como se utilizan algoritmos que ofrezcan una solución cercana a la óptima.

5.3. Métodos de solución

Los modelos de optimización combinatoria han sido objeto de investigación dada la complejidad computacional intrínseca a estos y a sus diversas aplicaciones al mundo real. Las técnicas utilizadas de forma general se pueden clasificar en dos grupos: algoritmos exactos y algoritmos aproximados.

5.3.1. Algoritmos exactos

Los algoritmos exactos son métodos que, debido a su estrategia de búsqueda, exploran todo el espacio de soluciones, pudiendo demostrar la optimalizad de una solución. Sin embargo, debido a las tasas de crecimiento exponencial de los espacios de búsqueda para los problemas tipo NP hard, su aplicación a menudo solo es adecuada para instancias muy pequeñas. Algunos algoritmos exactos se mencionan a continuación.

5.3.1.1. Método Simplex. es un procesamiento desarrollado por George Dantzing en 1947, para dar soluciones numéricas a problemas de programación lineal. Es un procedimiento iterativo que permite mejorar la solución de la función objetivo con cada paso sucesivo; se parte de una solución básica inicial en un vértice cualquiera, el método consiste en buscar sucesivamente otro vértice que mejore la anterior solución y finaliza cuando no es posible continuar mejorando, es decir, se alcanza la solución óptima. La búsqueda se hace siempre a través de los lados del polígono de soluciones factibles o de las aristas de la región solución. (Hillier y Lieberman, 2010).

5.3.1.2. Branch and Bound. Este algoritmo es una de las herramientas más utilizadas a la hora de resolver problemas de optimización discreta de tipo NP-hard. El método parte de un conjunto se soluciones candidatas del espacio de búsqueda, estas soluciones constituyen un árbol que inicialmente estaba compuesto solamente por una raíz, en cada iteración un nodo es evaluado construyendo ramas en los nodos que tienen mejores soluciones que el nodo actual (Clausen, 1999).

5.3.2. Algoritmos aproximados

Dadas las limitaciones de los algoritmos exactos para encontrar soluciones en los problemas tipo NP-hard, los investigadores han definido una serie de métodos que, sin garantizar el óptimo global, logran alcanzar buenas soluciones a problemas de elevada complejidad en tiempos de cómputo razonables. Los algoritmos aproximados pueden clasificarse en dos grupos: heurísticas y metaheurísticas.

5.3.2.1. Heurísticas. “Una heurística es una técnica de búsqueda directa que utiliza reglas favorables prácticas para localizar soluciones mejoradas. La ventaja de la heurística es que en

general determina buenas soluciones con rapidez utilizando reglas de solución simples" (Taha, 2012, p. 336).

Existen métodos heurísticos de diversa naturaleza, sin embargo, estos pueden clasificarse en dos categorías:

5.3.2.1.1. Heurísticas constructivas. Son conocidas por la rapidez en encontrar soluciones de buena calidad de los problemas de localización. Estas determinan las soluciones mediante un procedimiento que incorpora iterativamente elementos a una estructura inicialmente vacía, este tipo de heurísticas se encargan de construir una solución factible paso a paso respectando las restricciones del problema y considerando el costo de la solución.

5.3.2.1.2. Heurísticas de búsqueda local. estas heurísticas parten de una solución factible y a partir de ella se hacen cambios de forma sistemática para mejorarla. Los cambios que hace el algoritmo dependen de la estrategia utilizada, múltiples estrategias de búsqueda local han sido utilizados en la literatura. Algunas de ellas son: Primera mejora (first improvement), examina la vecindad y selecciona el primer movimiento que produce una mejora en la solución actual; Mejor mejora (best improvement) , examina la vecindad y todos los posibles movimientos seleccionando el mejor movimiento de todos los posibles; y randomized; en la que se selecciona aleatoriamente soluciones en la vecindad (C. Ortiz, 2010).

5.3.2.2. **Metaheuristicas**. Son algoritmos aproximados de optimización y búsqueda de alto nivel que guían una o varias heurísticas subordinadas combinando de forma inteligente distintos conceptos para explorar y explotar adecuadamente el espacio de búsqueda (Herrera, 2009).

Los métodos metaheurísticos aplicados a los problemas de horarios en general, según Lewis (2007) pueden clasificarse en tres grupos:

(1) Algoritmos de optimización de una etapa: donde las restricciones duras y restricciones suaves son satisfechas simultáneamente.

(2) Algoritmos de optimización de dos etapas: donde la satisfacción de las restricciones suaves solo se considera una vez que un horario factible haya sido encontrado.

(3) Algoritmos que permiten relajación: las violaciones de las restricciones duras no se consideran al inicio del problema al relajar alguna característica del problema. Luego se intentan satisfacer las restricciones suaves a la vez que se van eliminando estas relajaciones.

Los algoritmos metaheurísticos según la cantidad de soluciones que manejen pueden clasificarse en dos tipos: metaheurísticas basadas en una solución única o basadas en poblaciones; dentro del primer grupo se encuentran los algoritmos de búsqueda tabú, recocido simulado, gran diluvio, VNS, GRASP, entre otros; en el segundo grupo, algunos algoritmos documentados el algoritmo genético, la optimización de colonia de hormigas, el algoritmo artificial de colonia de abejas y el algoritmo de búsqueda armónica.

A continuación, se presentan algunos métodos metaheurísticos.

5.3.2.2.1. Búsqueda tabú (TS). Esta metaheurística fue introducida por Fred Glover en 1986, es una técnica heurística global que trata de evitar caer en el óptimo local creando una lista especial llamada tabú. La búsqueda tabú hace uso del concepto de memoria, inducido por la Inteligencia Artificial y lo implementa mediante estructuras simples con el objetivo de dirigir la búsqueda teniendo en cuenta los antecedentes (Riojas, 2005). Esto minimiza la posibilidad de que se formen ciclos en la misma solución y, por lo tanto, crea más oportunidades de mejora al pasar a áreas no exploradas del espacio de búsqueda.

5.3.2.2.2. Recocido simulado (SA). Este método es basado en la analogía del simulado de recocido de los metales. El término de recocido se refiere a un proceso físico en el que un sólido es calentado hasta que pasa a una fase liquida y luego es enfriado lentamente con el fin de que posea menos energía.

El recocido simulado es uno de los primeros algoritmos metaheurísticos muy hábil para escapar de los óptimos locales, permitiendo que soluciones de mala calidad sean aceptadas (con alguna probabilidad).

5.3.2.2.3. Algoritmo de gran diluvio (GD). El algoritmo del gran diluvio es un enfoque metaheurístico propuesto por Dueck en 1993 y está inspirado en el comportamiento que podría surgir cuando alguien busca un terreno más alto para evitar el aumento del nivel del agua durante una lluvia constante.

5.3.2.2.4. Variable búsqueda del vecindario. (VNS por sus siglas en inglés) Esta metaheurística fue VNS fue presentada por Mladenović y Hansen (1997). Utiliza métodos de búsqueda local mediante el uso de varias estructuras de vecindario, las cuales actúan sistemáticamente dependiendo del estado actual de la solución. DE manera que, el VNS permite explorar una variedad de posibilidades y saltar a una nueva solución si es necesario.

5.3.2.2.5. Procedimientos de búsqueda adaptativa aleatoria codiciosa. (GRASP por sus siglas en inglés). Propuesto por Feo y Resende en 1994, es un algoritmo metaheurístico que se basa en una técnica de muestreo aleatorio en el cual por cada iteración proporciona una solución al problema. Esta ténica tiene como objetivo resolver problemas difíciles en el campo de la optimización combinatoria. Para este procedimiento existen dos fases en cada iteración, la fase de construcción y la fase de mejora. En la primera se construye una solución inicial a partir de una función voraz adaptiva aleatoria y la segunda aplica un procedimiento de búsqueda local sobre la solución inicial con el fin de encontrar una solución mejor.

5.3.2.2.6. Algoritmo genético (GA). Los algoritmos genéticos fueron popularizados por Holland en 1975. Esta metodología emplea operadores conocidos como operadores genéticos (como la selección, el cruce y la mutación) que manipulan soluciones individuales (denominadas cromosomas) en una población durante varias generaciones para mejorar la función objetivo (a menudo llamada la aptitud).

Un cromosoma se representa como una cadena de longitud fija, también puede tener varias dimensiones, donde cada posición se denomina gen y contiene una unidad de información de solución. El algoritmo funciona partiendo de una población inicial de soluciones, de las cuales son seleccionadas algunas de ellas para ser los padres de la siguiente generación, con cada generación la población se va actualizando mediante los procesos de cruce y mutación que pueden presentarse con cierta probabilidad. Luego de cierto número de iteraciones se espera tener una población de soluciones diversas con buena calidad.

5.3.2.2.7. Optimización de colonia de hormigas. (ACO por sus siglas en inglés). La optimización de colonia de hormigas es una metaheurística basada en la población propuesta por

Dorigo et al. (1996). La idea básica detrás de esta técnica de optimización se basa en la observación de que las hormigas encuentran su camino hacia una fuente de alimento y regresan a su nido. A medida que avanzan, se deposita un rastro de sustancia química (llamada feromona) que ayudará a otras hormigas a encontrar la misma fuente de alimento.

5.3.2.2.8. Colonia artificial de abejas. (ABC por sus siglas en inglés). Es un algoritmo metaheurístico basado en la naturaleza propuesto por Karaboga en 2005 para optimizar los problemas numéricos. Fue motivado por el comportamiento de búsqueda inteligente de las abejas melíferas. El modelo comprende tres fundamentos vitales: abejas forrajeras empleadas y desempleadas, fuentes de alimentos y distancia a la colonia. las abejas forrajeras empleadas y desempleadas buscan fuentes de alimentos ricos, en las cercanías de la colmena.

5.3.2.2.9. Métodos híbridos. El objetivo del enfoque híbrido es tomar la mejor idea de un enfoque e incorporarla con otra idea buena (o mejor) de otro enfoque, es decir, combina varios algoritmos heurísticos o metaheurísticos con el objetivo de mejorar el desempeño de los algoritmos aprovechando las ventajas que tiene cada enfoque. Cabe mencionar que muchos de los algoritmos ya descritos presentan métodos híbridos (en mayor o menor medida) que han sido aplicados al problema de programación de cursos universitarios.

5.3.3. Cadena de Suministro de Ciclo Cerrado

El concepto de cadena de suministro de ciclo cerrado es una extensión de la cadena de suministro tradicional, en la cual se integran los flujos hacia adelante y hacia atrás que intervienen desde la adquisición de los materiales para la elaboración de un producto hasta su disposición final.

> “Puede ser definida como el diseño, control y funcionamiento de un sistema para maximizar la creación de valor sobre el ciclo de vida de un producto con recuperación dinámica del valor de diferentes tipos de retornos a lo largo del tiempo” (Guide y van Wassenhove, 2006).

La gestión se refiere a la integración y al tratamiento de las actividades tanto internas como externas a la empresa, por tanto, una cadena de suministro intenta cerrar los ciclos de materiales y prevenir pérdidas en la cadena utilizando los costes mínimos para conseguir el máximo valor (Wang y Hsu, 2010).

5.3.4. Incertidumbre en las Cadenas de Suministro

El estudio de la incertidumbre asociada a las cadenas de abastecimiento, se inicia desde la descripción de esta, su percepción y por qué el conocimiento asociado al uso de tecnologías, sistemas de información y manejo de modelos matemáticos pueden ser fundamentales para su tratamiento. La incertidumbre se da cuando no se puede estimar el resultado de un evento o la probabilidad de su ocurrencia debido a que se carece de información o no es posible predecir el efecto de las acciones de control en la cadena. (Arango, Adarme y Zapata, 2010)

La existencia de parámetros de producción efectivos, considerando la incertidumbre en los problemas de la cadena de suministro es imperativa. Además, la optimización en términos de incertidumbre tanto en la teoría como en la práctica aumentó rápidamente y ha sido objeto de muchos estudios. Hay muchos enfoques para lidiar con la incertidumbre. Los más famosos de estos enfoques, que se han utilizado ampliamente en la literatura, son la programación difusa, la programación estocástica y la programación estocástica difusa, la optimización robusta. En la mayoría de los trabajos recientes sobre diseño de redes bajo incertidumbre, la incertidumbre se ha modelado en base a una planificación probabilística basada en escenarios. (Ebrahimi, 2018)

5.3.5. Problema de Localización-Asignación- Ruteo

El problema de Localización, Asignación y Ruteo consiste básicamente de tres problemas intrínsecamente relacionados.

5.3.5.1. **Problema de Localización de instalaciones.** Este problema busca decidir la ubicación de una instalación en un espacio geográfico de optimizar la solución, ya sea maximizar utilidades o minimizar costos de transportes. Estos modelos se usan frecuentemente para tomar decisiones de ubicación rápidamente sin necesidad de hacer un análisis robusto. Algunos ejemplos

típicos de problemas de localización de instalaciones son la instalación de una estaciones de policías, nuevas bodega de almacenamiento, sucursales, cadenas de supermercados, etc.

5.3.5.2. Problema de asignación (PA). Es uno de los problemas más interesantes de la Programación lineal. El problema consiste en asignar ciertos recursos disponibles (maquinas u personas) para la realización de determinadas actividades al menor costo, suponiendo que cada recurso se asigna a una sola actividad y que cada actividad es desarrollada por un único recurso. (Lopez y Conde, s. f.). Por ejemplo en el contexto del problema de Localización—Ruteo, la asignación, determina qué depósitos deben asignarse a cada fábrica o que clientes deben asignarse a qué depósito.

5.3.5.3. Problema de Ruteo de Vehículos (VRP). El VRP fue propuesto por primera vez por Dantzig y Ramzer (1959). Inicialmente fue diseñado con un algoritmo de solución para el problema de la entrega de gasolina a las estaciones de servicio. Un VRP clásico, por ejemplo, determina una ruta óptima para un conjunto de vehículos ubicados en un depósito que debe atender varios clientes. La mayoría de los problemas que se encuentran en la vida real suelen ser complicados con la adición de varias restricciones que dan lugar a muchas variantes del VRP. Tales limitaciones incluyen la capacidad del vehículo, las ventanas de tiempo de las entregas (o recogidas), la cantidad de depósitos, la cantidad de viajes permitidos, la naturaleza de las demandas, la naturaleza del servicio y muchas otras. Cuando se introducen una o más de estas limitaciones con un número creciente de clientes a los que atender, el problema resulta ser NP-difícil porque no pueden resolverse en tiempo polinomial.(Olawale, 2017)

5.3.6 NSGA-II (Non-Dominated Sorting Genetic Algorithm II)

El algoritmo genético de ordenación no dominada en su segunda versión fue propuesto por Deb y sus estudiantes, 2000. En el cual como lo menciona (Corral Sastre, 2018) en este algoritmo, la diversidad no se genera principalmente en las primeras generaciones, debido a que en esas etapas existen muchos frentes que se mantienen durante el paso de las generaciones, pero a medida que el algoritmo progresa muchos individuos van a formar parte del mejor frente, incluso haciendo que dicho frente se tenga que filtrar, por lo que se hace importante que las soluciones no rechazadas

sean buenas y escogidas mediante un método que garantice la diversidad dentro de un frente de dominancia.

La corrida del algoritmo inicia con la generación de una población P de un tamaño N, posterior a esto se ordena la población mediante la dominancia y se evalúan las distancias de apilamiento para todos los individuos, después se realiza la selección de la población; es justamente en esta parte que el algoritmo se diferencia de los demás algoritmos genéticos, dicha selección se realiza por tipo torneo, el cual consiste en comparar dos soluciones y elegir uno como ganador ya esa escogiendo al que tenga una mayor distancia de apilamiento o de manera aleatoria (MARTÍNEZ RUIZ, 2019); los pasos siguientes son el cruce y mutación, después se reúnen padres y descendientes y se clasifican mediante frentes de Pareto, por último se determinan los individuos que pasaran a la siguiente generación seleccionando los mejores frentes. (Corral Sastre, 2018).

Las ventajas de este algoritmo en su segunda versión es que tiene un algoritmo de ordenado más eficiente computacionalmente e incorpora elitismo, además es altamente competitivo en convergencia de Pareto. Inconvenientes que tiene son que parece no rendir bien cuando la población está representada mediante una representación binaria y tiende a tener problemas exploratorios a medida que se incrementa el número de objetivos, pero es un problema común entre los algoritmos actuales.(Corral Sastre, 2018).

6. Definición del problema y modelo matemático

6.1. Definición del problema

El problema que se trata en el presente trabajo está basado en un caso de estudio en Irán. Consiste en una red de cadena de suministro de circuito cerrado de una empresa que fabrica neumáticos. (Ebrahimi, 2018) Hoy en día, los neumáticos son uno de los productos más utilizados en distintos tipos de vehículos como carros, camiones, motocicletas y autobuses con una vida útil muy corta y sí estos entran en el medio ambiente, muchas criaturas del ecosistema serán destruidas. (Ebrahimi, 2018). La acumulación de neumáticos de desecho puede crear potencialmente efectos adversos para el medio ambiente, amenazas para la salud y la seguridad pública ya sea por

convertirse en un caldo de cultivo para mosquitos y roedores, generar incendios, emisiones a la atmósfera de los neumáticos como fuente de combustible y lixiviados de los componentes de los neumáticos (orgánicos e inorgánicos) al suelo y al agua. (Oyola-Cervantes & Amaya-Mier, 2019)

Las estadísticas demuestran que cada año se desechan más de 1.000.000 de toneladas de neumáticos usados en todo el mundo; por ejemplo, más de 600.000 toneladas al año en Alemania (Lebreton y Tuma, 2006) y el equivalente a 250.000 toneladas al año en Francia (Pedram et al., 2017) Sin embargo, sólo un pequeño porcentaje se reutiliza en sistemas de suministro de circuito cerrado (Serumgard, 1998), probablemente debido a la falta de alternativas de eliminación económica para los neumáticos residuales. (Oyola-Cervantes & Amaya-Mier, 2019) Con el uso prolongado, la superficie del neumático o de la banda de rodadura se desgasta y tiende a desinflarse como resultado del contacto con la superficie de la carretera. Esto hace que el neumático sea inadecuado para su uso en carretera debido a la reducción del rendimiento de los frenos y la adherencia. La carcasa de los neumáticos devueltos tiene menos probabilidades de sufrir daños significativos, por lo que es posible sustituir la banda de rodadura desgastada por una nueva y reducir los residuos. Además, una evaluación del ciclo de vida de la producción de neumáticos nuevos y del recauchutado pone de manifiesto que la producción de neumáticos nuevos consume cuatro veces más materiales que la producción de neumáticos recauchutados. Además, el uso de energía para producir un neumático nuevo es tres veces mayor que el de un neumático recauchutado. Por lo tanto, el recauchutado es la clave para reducir los residuos y el consumo de materias primas no renovables. (Pedram et al., 2017)

Los neumáticos de desecho deben considerarse una fuente y no un material de desecho; los neumáticos de desecho triturados, astillados o enteros pueden quemarse de forma respetuosa con el medio ambiente; y los neumáticos de desecho tienen un importante contenido térmico, y en condiciones controladas este calor puede extraerse para un uso beneficioso. Además, el uso adecuado de los neumáticos de desecho reduce los costos de salud pública al minimizar los posibles criaderos de insectos y roedores, que a menudo conducen al desarrollo de enfermedades, como el dengue, la fiebre amarilla, la leptospirosis y la malaria (Oyola-Cervantes & Amaya-Mier, 2019).

Para la construcción de un neumático se utilizan diversos materiales químicos, naturales y minerales para cada una de sus partes como son el diseño y dibujo de la banda de rodadura, la pared lateral, el cinturón, los cordones del cuerpo radical, el hilo del talón y el vértice especial de

alta rigidez como se ve en la figura 9. Por lo tanto, cada uno de estos materiales desempeña un papel especial en el rendimiento del neumático y durante su producción. Para el caso de estudio los materiales principales del neumático son los siguiente: caucho, hollín, hilo, alambre y agentes químicos. Cabe señalar que la cantidad de uso de cada uno de estos componentes depende del tipo y la estructura del neumático, las condiciones de la carretera, las condiciones climáticas y otros factores diversos.

Figura 9.

Estructura del neumático

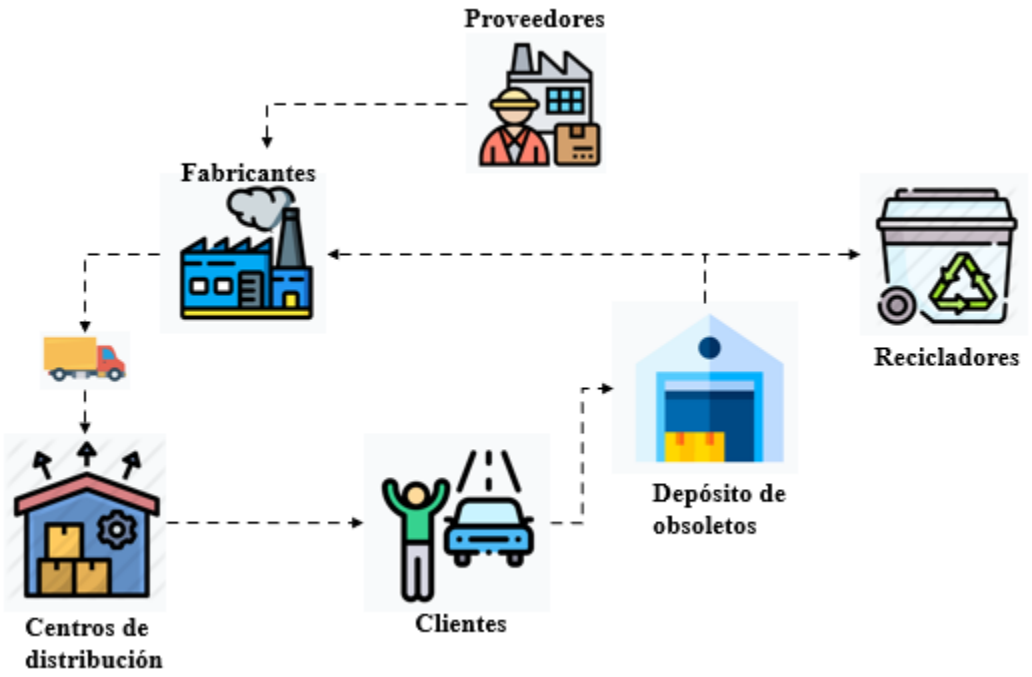
Nota. Figura adaptada de (Ebrahimi, 2018)

El proceso de la cadena de suministro de circuito cerrado del caso de estudio se ilustra en la figura 10. En el cual, se cuenta con ocho proveedores opcionales los cuales ofrecen o no un descuento por cantidad, de estos se seleccionan máximo seis para que suministren las cinco materias primas a tres fabricantes. Los fabricantes producen cuatro tipos de neumáticos (camión, coche, camioneta y tractor), y estos productos se envían a un máximo de cinco centros de distribución seleccionados entre siete candidatos a través de una de las dos rutas disponibles. Los centros de distribución envían estos productos almacenados a diez clientes. Los clientes devuelven algunos de estos neumáticos utilizados a un máximo de tres centros obsoletos de los cuatro centros existentes. Por último, algunos de estos productos pueden ser recuperados y se envían a los

fabricantes para introducirse nuevamente a la red tras un proceso de recuperación y el resto se envía a dos centros de reciclaje para su descomposición final.

Figura 10.

Cadena de suministro de circuito cerrado del modelo



La solución del problema consiste en determinar cuántos proveedores se deben seleccionar, cuantos centros de distribución y de obsolescencia abrir, cuanta materia prima comprar y los flujos de cada tipo de neumático que serán transportados en la cadena. Lo anterior se plantea mediante tres funciones objetivo que son: (1) la minimización de los costos fijos de apertura y de pedidos, (2) la minimización de los efectos de las emisiones ambientales y (3) la maximización de la capacidad de respuesta de la red integrada hacia adelante y hacia atrás.

6.2. Obtención de datos

Los datos utilizados para la resolución del problema se pueden encontrar en el Apéndice A, estos fueron obtenidos de (Ebrahimi, 2018) y provienen del caso de estudio mencionado con anterioridad que consiste en el diseño de una cadena de suministros integrada hacia adelante y hacia atrás de un fabricante iraní que produce neumáticos.

En el trabajo de (Ebrahimi, 2018) no se diferencia los tipos de descuentos por cantidad y por cantidad incremental que ofrecen los proveedores, razón por la cual y con base en el análisis de sensibilidad en cuatro condiciones de modelización realizada por el autor, para el desarrollo del presente trabajo se van a manejar según la segunda condición, Modelo sin considerar el descuento por cantidad y considerando el problema de las rutas, y se comparan los resultados respecto a esta condición.

6.3. Formulación del modelo matemático

6.3.1. Índices

i Conjunto de materiales, i=1,…,I

s Conjunto de proveedores, s=1,…,S

j Conjunto de rutas, j=1,…,J

m Conjunto de fábricas, m=1,…,M

p Conjunto de productos, p=1,…,P

d Conjunto de centros de distribución potenciales, d=1,…,D

c Conjunto de clientes, c=1,…,C

o Conjunto de depósitos obsoletos potenciales, o=1,…,O

r Conjunto de recicladores, r=1,…,R

ϕ Conjunto de escenarios, ϕ=1,…,Φ

6.3.2. Variables de decisión:

Q^{φ}_{ism} Cantidad de materia prima i comprada por la fábrica m al proveedor s en el escenario φ

H^{φ}_{pm} Cantidad de producto p producido por la fábrica m en el escenario φ

E^{φ}_{pmdj} Cantidad de producto p enviado por la fábrica m al centro de distribución d en la ruta j en el escenario φ

V^{φ}_{pdc} Cantidad de producto p enviado del centro de distribución d al cliente c en el escenario φ

U^{φ}_{pco} Cantidad de producto p enviado por el cliente c al depósito obsoleto o en el escenario φ

T^{φ}_{pom} Cantidad de producto p devuelto del depósito obsoleto o a la fábrica m en el escenario φ

B^{φ}_{por} Cantidad de producto p devuelto del depósito obsoleto o al reciclador r en el escenario φ

Y_d $\begin{cases} = 1, & \text{si el centro de distribución } d \text{ está ubicado en el lugar potencial} \\ = 0, & \text{DLC} \end{cases}$

X_o $\begin{cases} = 1, & \text{si el deposito obsoleto o está ubicado en el lugar potencial} \\ = 0, & \text{DLC} \end{cases}$

W_s $\begin{cases} = 1, & \text{si se selecciona el proveedor } s \\ = 0, & \text{DLC} \end{cases}$

γ_{mdj} $\begin{cases} = 1, & \text{si se selecciona la ruta } j \text{ de la fábrica } m \text{ al centro de distribución } d \\ = 0, & \text{DLC} \end{cases}$

6.3.3. Parámetros:

FS_S Costo fijo del pedido asociado al proveedor *s*

PC_{is} Costo de compra de la materia prima *i* al proveedor *s*

FD_d Costo de apertura del centro de distribución *d*

FO_o Costo de apertura del depósito obsoleto *o*

TR_i Costo te transportar la materia prima *i* por km

TP_p Costo de trasportar el producto *p* por km

Dis_{sm} Distancia entre el proveedor *s* y la fábrica *m*

Dis_{mdj} Distancia entre la fábrica *m* al centro de distribución *d* en la ruta *j*

Dis_{dc} Distancia entre el centro de distribución *d* y el cliente *c*

Dis_{co} Distancia entre el cliente *c* y el depósito obsoleto *o*

Dis_{or} Distancia entre el depósito obsoleto *o* y el reciclador *r*

Dis_{om} Distancia entre el depósito obsoleto *o* y la fábrica *m*

PRC_p Costo de producción del producto *p*

A_p Ahorro en costos del producto *p* (debido a la recuperación del producto)

Dem^{φ}_{cp} Demanda del cliente *c* del producto *p* en el escenario φ

B_p Porcentaje del producto *p* que es recuperable

VC_{pd} Costo de retención variable del producto *p* por el centro de distribución *d*

RE_p Tasa de devolución del producto *p* por los clientes

$CAPm_{mp}$ Capacidad de la fábrica *m* para el producto *p*

$CAPd_{dp}$ Capacidad del centro de distribución *d* para el producto *p*

$CAPo_{op}$ Capacidad del depósito obsoleto *o* para el producto *p*

$CAPs_{si}$ Capacidad del proveedor *s* para la materia prima *i*

$CAPr_{rp}$ Capacidad del reciclador *r* para el producto *p*

BOM_{ip} Cantidad de materia prima *i* para la fabricación del producto *p*

eis_{sm} Impacto ambiental por unidad embarcada desde el proveedor *s* a la fábrica *m*

eis_{mdj} Impacto Ambiental por unidad embarcada desde la fábrica *m* al centro de distribución *d* por la ruta *j*

eis_{dc} Impacto Ambiental por unidad embarcada desde el centro de distribución *d* al cliente *c*

eis_{co} Impacto ambiental por unidad embarcada desde el cliente *c* al depósito obsoleto *o*

eis_{om} Impacto Ambiental por unidad embarcada desde el depósito obsoleto *o* a la fábrica *m*

eis_{or} Impacto Ambiental por unidad embarcada desde el depósito obsoleto *o* al reciclador *r*

es_s Impacto Ambiental relacionado con el proveedor *s*

eo_d Impacto ambiental relacionado con abrir el centro de distribución *d*

eo_o Impacto Ambiental relacionado con abrir el depósito obsoleto *o*

P_{ξ}^{φ} Probabilidad de ocurrencia del escenario φ

λ Factor de ponderación para la capacidad de respuesta hacia adelante

$1\text{-}\lambda$ Factor de ponderación para la capacidad de respuesta hacia atrás

6.3.4. Función objetivo

La función objetivo que modela la situación planteada para la cadena de suministro de bucle cerrado está compuesta por tres objetivos. Cada grupo reúne los parámetros y variables de decisión necesarios para asegurar la disminución en los costos, la disminución en los impactos al ambiente generados por los procesos y el aumento en la capacidad de respuesta de la red integrada hacía delante y hacía atrás.

Objetivo 1: Min (F1)

$$Min\ F1 = \sum_S FSs * Ws + \sum_d FDd * Yd + \sum_o FOo * Xo + \sum_\varphi P\xi\varphi * Z1\varphi$$

$$*Z1\varphi = \sum_i \sum_s \sum_m PCis * \text{Qφism} + \sum_i \sum_s \sum_m (TRi * Dissm) * \text{Qφism} + \sum_p \sum_m \sum_j \sum_d (PRCp + TPp) Dismdj * \text{Eφpmdj} * \gamma\text{mdj} + \sum_p \sum_d \sum_c (VCpd + TPp) * Disdc * \text{Vφpdc} +$$

$$\sum_p \sum_c \sum_o TPp * Disco * \text{Uφpco} + \sum_p \sum_o \sum_m (-Ap + TPp) * Disom * \text{Tφpom} + \sum_p \sum_o \sum_r TPp * Disor * \text{Bφpor}$$

El primer objetivo tiene el propósito de minimizar el costo total de la cadena de suministro. La función relaciona todos los costos contemplados en el modelo. Los costos fijos de realizar las órdenes de materia prima a cada uno de los proveedores seleccionados. Los costos fijos de apertura de las instalaciones que se abrirán para los centros de distribución y para los centros de obsolescencia. Además de un promedio de costos asociados a diferentes escenarios, en los que se incluyen los rangos de descuento según la cantidad comprada al proveedor, costos de transporte de materia prima y de producto final, costos de producción, costos de almacenamiento en los centros de distribución, ***Cost-saving*** (costo de los productos devueltos) de los clientes a los centros de obsolescenia y el costo de transporte del producto recuperado a los recicladores o a las plantas según el caso.

Objetivo 2: Min (F2)

$$Min\ F2 = \sum_S ESs * Ws + \sum_d EOd * Yd + \sum_o EOo * Xo + \sum_\varphi P\xi\varphi * Z2\varphi$$

$$* Z2\varphi = \sum_i \sum_s \sum_m EISsm * \text{Qφism} + \sum_j \sum_p \sum_m \sum_d EISmdj * \text{Eφpmdj} + \sum_p \sum_d \sum_c EISdc * \text{Vφpdc} + \sum_p \sum_c \sum_o EISco * \text{Uφpco} + \sum_p \sum_o \sum_m EISom * \text{Tφpom} + \sum_p \sum_o \sum_r EISor * \text{Bφpor}$$

El objetivo 2 busca disminuir el efecto ambiente producido por la selección de los proveedores ***(primera parte de la ecuación)***, efecto ambiental de construir los centros de distribución y los centros de obsolescencia. Además de la relación de los impactos ambientales asociados a los traslados de la materia prima y los productos fabricados y devueltos de un lugar a otro.

Objetivo 3: Max (F3)

$$Max\ F3 = \sum_\varphi P\xi\varphi * \left(\lambda * \left(\sum_p \sum_d \sum_c \frac{\text{Vφpdc}}{Dem\varphi cp} \right) + (1-\lambda) * \left(\sum_p \sum_c \sum_o \frac{\text{Uφpco}}{REp * Dem\varphi cp} \right) \right)$$

El objetivo 3 la cual relaciona los factores de importancia hacía adelante y hacía atrás según la razón de respuesta de la demanda del consumidor y a la respuesta de recogida de los productos devueltos de acuerdo con cada uno de los productos contemplados en el modelo y el escenario en el que se encuentre la demanda.

6.3.5. Restricciones

Compra de materia prima

$$\sum_{p} H_{pm}^{\varphi} * BOM_{ip} = \sum_{s} Q_{ism}^{\varphi} \quad \forall\, i, m, \varphi \quad (1)$$

La restricción (1) muestra que la cantidad de materias primas necesarias para producir los productos manufacturados por las plantas debe ser igual a las materias primas compradas al proveedor.

Balance de flujo

$$H_{pm}^{\varphi} + \sum_{0} T_{pom}^{\varphi} = \sum_{j} \sum_{d} E_{pmdj}^{\varphi} * \gamma_{jmd} \quad \forall\, p, m, \varphi \quad (2)$$

La restricción (2) confirma que la suma de los diferentes productos fabricados por las plantas y los productos devueltos por los clientes que son recuperables sea igual la cantidad de productos enviada por las plantas a los centros de distribución bajo una ruta específica.

$$\sum_{j} \sum_{m} E_{pmdj}^{\varphi} \geq \sum_{c} V_{pdc}^{\varphi} \quad \forall\, p, d, \varphi \quad (3)$$

La restricción (3) confirma que la cantidad de productos enviados desde las plantas hasta los centros de distribución usando una ruta específica, sea mayor o igual a la cantidad de productos que son enviados a los clientes desde los centros de distribución.

$$\sum_{d} V_{pdc}^{\varphi} = Dem_{cp}^{\varphi} \quad \forall\, c, p, \varphi \quad (4)$$

La restricción (4) asegura que la cantidad de producto enviado de los centros de distribución a cada cliente sea igual a la demanda de cada consumidor según cada producto.

$$\sum_{o} U_{pco}^{\varphi} = RE_{p} * Dem_{cp}^{\varphi} \quad \forall\, c, p, \varphi \quad (5)$$

La restricción (5) asegura que la cantidad de productos devueltos por los clientes a los centros de obsolescencia sea igual a la tasa de devolución de producto de la demanda de cada consumidor según cada producto.

$$\sum_{m} T_{pom}^{\varphi} = \beta_p * \sum_{c} U_{pco}^{\varphi} \quad \forall\, p, o, \varphi \quad (6)$$

La restricción (6) asegura que la cantidad de productos devueltos que se envía de los centros de obsolescencia a las plantas sea igual a la cantidad de productos que son recuperables y que fueron devueltos por los clientes a los centros de obsolescencia.

$$\sum_{r} B_{por}^{\varphi} = \left(1 - \beta_p\right) * \sum_{c} U_{pco}^{\varphi} \quad \forall\, p, o, \varphi \quad (7)$$

La restricción (7) asegura que la cantidad de productos devueltos que se envía de los centros de obsolecencia a reciclaje sea igual a la cantidad de productos que no son recuperables y que fueron devueltos por los clientes a los centros de obsolescencia.

Restricciones de capacidad

$$H_{pm}^{\varphi} + \sum_{o} T_{pom}^{\varphi} \leq CAPm_{mp} \quad \forall\, p, m, \varphi \quad (8)$$

La restricción (8) confirma que la suma de las unidades producidas por cada una de las plantas y la cantidad de productos que son recuperables devueltos por los clientes a los centros de obsolescencia sea menor o igual a la capacidad máxima que tiene cada planta para producir cada producto.

$$\sum_{j} \sum_{m} E_{pmdj}^{\varphi} \leq Y_d * CAPd_{dp} \quad \forall\, p, d, \varphi \quad (9)$$

La restricción (9) asegura que la cantidad de productos enviados desde las plantas hasta los centros de distribución usando una ruta específica sea menor o igual a la capacidad máxima que tiene cada centro de distribución dependiendo si este es abierto.

$$\sum_{c} U_{pco}^{\varphi} \leq X_O * CAPo_{op} \quad \forall\, p, o, \varphi \quad (10)$$

La restricción (10) muestra que la cantidad de productos devueltos por los clientes a los centros de obsolescencia debe ser menor o igual a la capacidad máxima que tienen los depósitos para retener cada producto si estos son abiertos.

$$\sum_{m} Q_{ism}^{\varphi} \leq W_s * CAPs_{si} \quad \forall\, i, s, \varphi \quad (11)$$

La restricción (11) muestra que la cantidad de cada materia prima comprada para cada planta debe ser menor o igual a la capacidad de los proveedores que son seleccionados.

Restricciones de selección

$$\sum_{s} W_S \leq Smax \quad (12)$$

$$\sum_{d} Y_d \leq Dmax \quad (13)$$

$$\sum_{o} X_o \leq Omax \quad (14)$$

Las restricciones (12), (13) y (14) muestran que la cantidad de proveedores seleccionados, centros de distribución abiertos y centros de obsolescencia abiertos, respectivamente, sean menores o iguales al número máximo permitido para cada uno.

Restricción de ruta

$$\sum_{j} Y_{mdj} = 1 \quad \forall m, d \quad (15)$$

La restricción (15) garantiza que entre cada planta y cada centro de distribución se use sólo una ruta para transportar los productos.

Restricciones de decisión

$$Y_d, X_o, W_s, \gamma_{mdj} \in \{0,1\} \quad (16)$$

$$Q_{ism}^{\varphi}, H_{pm}^{\varphi}, E_{pmdj}^{\varphi}, V_{pdc}^{\varphi}, U_{pco}^{\varphi}, T_{pom}^{\varphi}, B_{por}^{\varphi} \geq 0 \quad \forall i, j, s, p, m, d, c, o, r, \varphi \quad (17)$$

Las restricciones (16) y (17) garantizan la naturaleza de las variables binarias y las variables de decisión enteras positivas.

6.4. Supuestos

En el presente trabajo se definen los siguientes supuestos para la solución del problema:

El modelo utiliza una programación estocástica en la formulación en la que se tienen en cuenta varios escenarios para tratar la incertidumbre.

El modelo tiene seis niveles: el proveedor, fabricantes, centro de distribución, cliente, depósitos obsoletos y reciclador.

Los flujos entre niveles se consideran en el escenario φ y su probabilidad

Los factores medioambientales se tienen en cuenta a la hora de seleccionar un proveedor.

Se tienen en cuenta los factores ambientales para el transporte del producto.

Cada proveedor, fabricante, centro de distribución, reciclador y centro de obsoletos tiene la capacidad máxima de almacenamiento

La demanda de los clientes para cada producto se considera bajo incertidumbre.

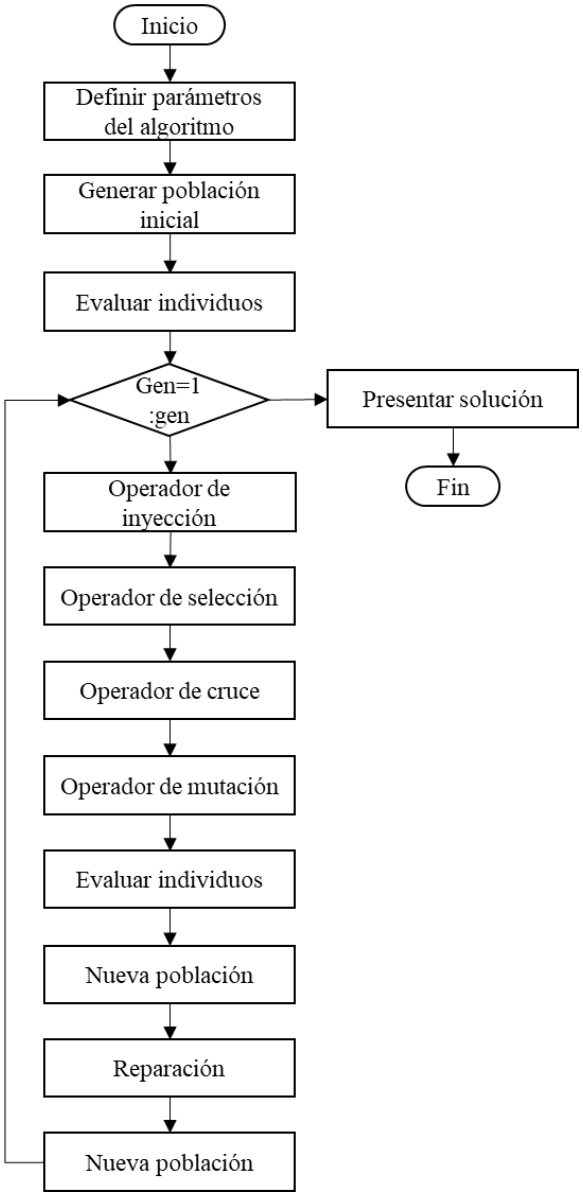
Para enviar los productos desde los fabricantes a los centros de distribución, se consideran varias rutas candidatas que una ruta debe ser seleccionada sobre la base del costo mínimo de transporte y el impacto de las emisiones.

7. Algoritmo de solución propuesto

El algoritmo propuesto para resolver el modelo matemático planteado por Seyed Babak Ebrahimi, es el NSGA-II (Nondominated sorting genetic algorithm II) propuesta por Deb et al, 2002 y desarrollado en MATLAB con base en el algoritmo propuesto en el proyecto de grado "Modelo de optimización difuso multiobjetivo multiperíodo para el diseño de una red logística inversa sostenible" (Caballero & Aguilar, 2019), realizado en la Escuela de Estudios Industriales y Empresariales de la Universidad Industrial de Santander. Este algoritmo evolutivo es una versión mejorada del NSGA (Nondominated sorting genetic algorithm) ya que aborda las limitaciones que contempla el NSGA, en cuanto a complejidad computacional, dado que no resultaba eficiente para poblaciones de gran tamaño, también la falta de elitismo, y finalmente, la necesidad de usar un parámetro *sharing* para mantener la distancia de separación entre las soluciones, como mecanismo para preservar la diversidad (Deb et al., 2002). El ciclo general de operaciones del algoritmo utilizado se presenta a continuación:

Figura 11.

Diagrama de flujo del algoritmo propuesto



Nota: Figura adaptada Deb et al., 2002

Este algoritmo sigue de manera general la estructura del NSGA-II clásico propuesto en Deb. et al (2002), con algunas modificaciones realizadas con el fin de la preservación de la diversidad. Las modificaciones son:

1. La inclusión de un operador de inyección bajo un criterio de convergencia.

2. El uso de un algoritmo de reparación que permite convertir soluciones no factibles en factibles.

Estas dos modificaciones se detallan en las secciones 7.8 y 7.9 respectivamente.

7.1. Codificación de las soluciones

Las soluciones al modelo matemático planeado en el capítulo 6, están representadas por medio de un cromosoma que codifica los valores que puede tener cada una de las variables decisión, las cuales están representadas por un gen específico. Es importante resaltar que la naturaleza del cromosoma permite dar cumplimiento a las restricciones relacionadas con la cantidad máxima de proveedores que pueden ser seleccionados y la cantidad máxima de centros de distribución y depósitos de obsolescencia que pueden abrirse. La estructura de la codificación también permite dar cumplimiento a la restricción que asegura que sólo se use una ruta para enviar los productos terminados de cada fábrica a los centros de distribución. En el siguiente apartado se profundiza acerca de la estructura del cromosoma y se proporciona un ejemplo para su entendimiento.

7.1.1. Cromosoma.

Las soluciones al modelo matemático planteado estarán representadas por un cromosoma que está compuesto por 571 genes para cada uno de los tres posibles escenarios que se contemplan. Los genes representan las once variables de decisión del problema y consideran cada uno de los niveles del modelo según los cinco tipos de materias primas, los seis posibles proveedores, las dos rutas de transporte de producto, las tres fábricas, los cinco centros de distribución candidatos, los diez clientes, los tres posibles depósitos para la recepción de productos devueltos, los dos recicladores, los tres escenarios y los cuatro tipos de productos.

A continuación, se muestra el cromosoma diseñado para representar los individuos que servirán como posible solución al modelo matemático con cada uno de los genes que representan las variables de decisión.

Tabla 4.

Representación del cromosoma resumido

Cantidad de materia prima i y producto terminado p en el escenario φ										
Si el proveedor s es seleccionado	Si el CD d es seleccionado	Si el depósito o es seleccionado	Si la ruta j se selecciona para ir de la fábrica m al CD d	Cantidad de materia prima i comprada para la fábrica m al proveedor s bajo escenario φ	Cantidad de producto p producido por la fábrica m bajo escenario φ	Cantidad de producto p enviado de la fábrica m al CD d bajo escenario φ usando la ruta j	Cantidad de producto p enviado de CD d al cliente c bajo escenario φ	Cantidad de producto p devuelto por el cliente c al depósito o bajo escenario φ	Cantidad de producto p devuelto del depósito o a la fábrica m bajo escenario φ	Cantidad de producto p devuelto del depósito o al reciclador r bajo escenario φ
S	D	O	J	Q(i,s,m,φ)	H(p,m,φ)	E(p,m,d,j,φ)	V(p,d,c,φ)	U(p,c,o,φ)	T(p,o,m,φ)	B(p,o,r,φ)

La estructura mostrada en la tabla 4 codifica la cantidad de materias primas que se compran a los proveedores seleccionados. La cantidad de productos según los tipos, que se producen en las fábricas. La cantidad de productos terminados que se envían de las fábricas a los centros de distribución seleccionados bajo una posible ruta. La cantidad de productos terminados que se envían a los clientes y la cantidad que ellos regresan a los depósitos de obsolescencia. Finalmente, se muestra la cantidad de productos devueltos que son enviados a las fábricas o a los centros de reciclaje.

Los valores de los genes que representan las variables de selección relacionadas con los proveedores *s*, los centros de distribución *d*, las rutas *j* y los depósitos de obsolescencia *o*, permanecen constantes en los tres escenarios para cada individuo durante la generación de la población inicial ya que no presentan un comportamiento estocástico. Para los operadores genéticos de cruce y de mutación se explica más adelante en esta sección el funcionamiento para este tipo de genes.

Con el fin de ilustrar un individuo real, en la tabla 5 se presenta un fragmento de la codificación de un cromosoma en donde se muestran los 11 tipos de genes y sólo se considera el escenario 3. Se analiza la materia prima 2, la fábrica 3, el tipo de producto 3, el cliente 6 y el reciclador 1. Se observa que se selecciona el proveedor 4 y que el centro de distribución y el depósito de obsolescencia abiertos son el 1 y 4, respectivamente. La ruta seleccionada para transportar el producto 3 es la ruta 1.

Al proveedor 4 se le compran 190 unidades de la materia prima 2 para la fábrica 3 en donde se producen 36 unidades del producto 3. Luego, se transportan 1934 unidades del producto 3 a través de la ruta 1 de la fábrica 3 al centro de distribución 1. El centro de distribución 1 entrega 1183 unidades del producto 3 al cliente 6, el cual regresa 372 unidades del producto 3 al depósito 4. El depósito envía 364 unidades de ese producto a la fábrica 3 y 140 unidades al reciclador 1. La situación se representaría en el cromosoma de la siguiente manera:

Tabla 5.

Representación de la ruta del ejemplo

Cromosoma bajo las condiciones especificadas										
S	D	O	J	Q(2,4,3,3)	H(3,3,3)	E(3,3,1,1,3)	V(3,1,6,3)	U(3,6,4,3)	T(3,4,3,3)	B(3,4,1,3)
4	1	4	1	190	36	1934	1183	372	364	140

Esta estructura se extiende horizontalmente para todos los niveles que contempla el modelo. La longitud dependerá de la cantidad de proveedores, centros de distribución y depósitos seleccionados según como se muestra en las restricciones 13, 14 y 15 durante el capítulo 6. En la figura 12, se muestra la totalidad de la codificación.

Figura 12.

Ejemplo de la codificación para las variables y escenarios que contempla el modelo**.**

	Si el proveedor s es seleccionado	Si el CD d es seleccionado	Si el depositos o es seleccionado	Si la ruta j se selecciona para ir de la fábrica m al CD d	Cantidad de materia prima i comprada para la fábrica m al proveedor s bajo escenario φ	Cantidad de producto p producido por la fabrica m bajo escenario φ	Cantidad de producto p enviado de la fabrica m al CD d bajo escenario φ usando la ruta j	Cantidad de producto p enviado de CD d al cliente c bajo escenario φ	Cantidad de producto p devuelto por el cliente c al depósito o bajo escenario φ	Cantidad de producto p devuelto del depósito o a la fabrica m bajo escenario φ	Cantidad de producto p devuelto del depósito o al reciclador r bajo escenario φ
Escenario 1	1…s …S	1…d …D	1…o …O	1…j…J	Q(1,1,1,1)…Q(i,s,m,1)	H(1,1,1)…H(p,m,1)	E(1,1,1,1,1)...E(p,m,d,j,1)	V(1,1,1,1)... V(p,d,c,1)	U(1,1,1,1)… U(p,c,o,1)	T(1,1,1,1)…T(p,o,m,1)	B(1,1,1,1)...B(p,o,r,1)
Escenario 2	1…s …S	1…d ...D	1…o …O	1…j…J	Q(1,1,1,2)…Q(i,s,m,2)	H(1,1,2)…H(p,m,2)	E(1,1,1,1,2)...E(p,m,d,j,2)	V(1,1,1,2)... V(p,d,c,2)	U(1,1,1,2)… U(p,c,o,2)	T(1,1,1,2)…T(p,o,m,2)	B(1,1,1,2)...B(p,o,r,2)
…											
Escenario e	1…s …S	1…d ...D	1…o …O	1…j…J	Q(1,1,1,e)…Q(i,s,m,e)	H(1,1,e)…H(p,m,e)	E(1,1,1,1,e)...E(p,m,d,j,e)	V(1,1,1,e)... V(p,d,c,e)	U(1,1,1,e)… U(p,c,o,e)	T(1,1,1,e)…T(p,o,m,e)	B(1,1,1,e)...B(p,o,r,e)

7.2. Parámetros del algoritmo

Además de los parámetros delimitados por el modelo matemático y los datos históricos del problema, se deben definir parámetros del algoritmo tales como la probabilidad de cruce y mutación. A continuación, se listan dichos parámetros con su respectiva explicación.

Tabla 6.

Parámetros del algoritmo con su respectiva explicación

Parámetro	Explicación
pob_num	Tamaño de la población (Número de individuos tomados al final de cada generación).
gen	Número total de iteraciones del algoritmo.
pc	Probabilidad de cruce.
pm	Probabilidad de mutación.
p_mut	Proporción de mutación

7.3. Generación de la población inicial

La población inicial se genera de forma aleatoria haciendo uso de diferentes funciones de MATLAB según la necesidad de cada gen. Como se mencionó al principio del capítulo 7, los genes que representan las variables de selección presentan un comportamiento diferente a los demás. Los

genes mencionados ocupan las posiciones 1 a la 29 en el cromosoma. Para el caso de proveedores, centros de distribución y depósitos los valores se generan en los vectores *S_S* y *D_S, O_S,* respectivamente, haciendo uso de la función *randperm* la cual genera permutaciones aleatorias de números enteros considerando un máximo según cada caso. Además, los valores para el gen que determina la ruta se generan por medio de la función *randi* que genera enteros pseudoaleatoria usando una función de distribución uniforme y se guardan en el vector *MDJ_S.*

Para los genes que ocupan las posiciones 30 a la 571 y almacenan las cantidades de materia prima y de producto terminado que circulan por la cadena, se generan los vectores IMS_S para la materia prima, PM_S para la cantidad de producto fabricado, PMDJ_S para la cantidad enviada de la fábrica al centro de distribución, PDC_S para el producto enviado de centros de distribución a clientes, PCO_S para la cantidad de vuelta por los clientes a los depósitos, POM_S para la cantidad que reingresa a la fábrica y POR_S para la cantidad enviada a los recicladores.

Cada uno de los genes que componen los cromosomas consideran un límite superior y un límite inferior de acuerdo con la naturaleza de cada gen. Por ejemplo, los genes que representan las cantidades producidas de cada tipo de producto en cada fabrica tienen como límite las capacidades de cada fábrica para producir cada tipo de producto. Estos vectores se generan haciendo uso de la función *randi* y el resultado generado se almacena en la matriz *cromosoma_pob,* la cual se compone de el número de individuos definidos para la población, según el parámetro *pob_num.*

7.4. Evaluación de los individuos

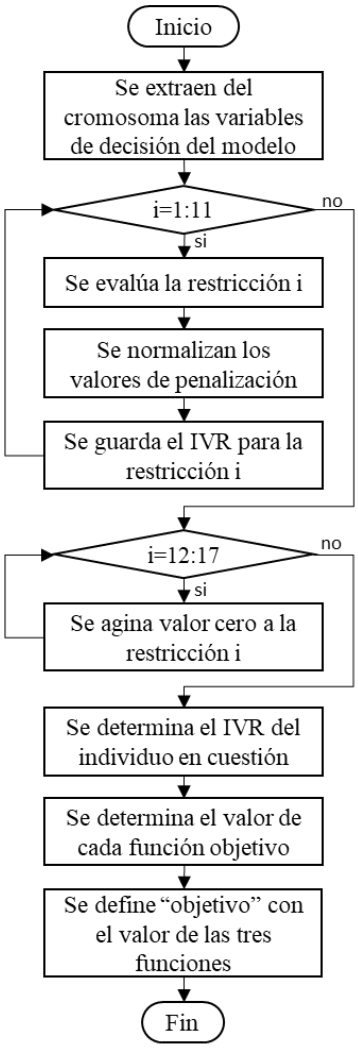
Para medir la calidad de los individuos generados en cada población con respecto al modelo, se mide el índice de violación de restricciones. El I.V.R. representa la suma de la valoración que se le otorgan a las restricciones para cada individuo (Deb et al. 2002). Para obtener un valor de índice equitativo es necesario normalizar los valores que se obtienen en cada restricción, con el fin de evitar los valores extremos que desvíen el índice. La normalización se realiza buscando el valor máximo por columna y dividiendo en este valor cada valor de la fila para

obtener un valor de restricción en términos de proporción que permita realizar agrupaciones y poder realizar una comparación justa.

Es importante resaltar que cuando se cumple una restricción, el valor de penalización asignado al individuo en cuanto a esa restricción será cero, pero cuando no se cumple se le asignará como valor de penalización el obtenido por la restricción. Cada valor permite comparar el nivel de violación por individuo con respecto a todas las restricciones. Un individuo se considera factible si el índice de violación es igual a 0, y no factible en caso contrario. La información obtenida se almacena en *restricción_pob* para cada generación. A continuación, se ilustra el flujo de para determinar el I.V.R. de cada individuo.

Figura 13.

Diagrama de flujo de la evaluación de individuos



Una vez se conocen los índices se procede a aplicar el ordenamiento rápido no dominado. Para las soluciones factibles se realiza el ordenamiento en el cual se obtiene el número de soluciones que dominan a cierta solución *a (Np)* y la cantidad de soluciones que domina *a (Sp)*. Las soluciones que no son dominadas por ninguna otra *(Np=0)* se organizarán en el primer frente. En el segundo frente se asignan las soluciones que estén dominadas únicamente por las soluciones del primer frente, de tal manera que el proceso de evaluación de soluciones se repite tantas veces como frentes de Pareto se generen.

En el caso de las no factibles se realiza el ordenamiento no dominado según el I.V.R. en donde cada individuo se asigna a un frente específico según el valor del índice, tendrán un mejor rango los individuos con menor índice.

Una solución *a* domina a una solución *b*, si alguna de las siguientes tres condiciones se cumple (Deb et al. 2002):

1. La solución a es factible y la solución b no lo es.
2. La solución a y b son no factibles, pero la solución a tiene un menor índice de violación de restricciones.
3. La solución a y b son factibles, pero la solución a domina a la solución b dado los valores de las funciones objetivos.

A continuación, se presenta el pseudo código para el ordenamiento rápido no dominado.

Figura 14.

Pseudo código del Ordenamiento No Dominado Rápido (Fast Non-Dominated Sorting). Adaptado de Deb et al., 2002

Ordenamiento rápido no dominado

Para cada individuo p que pertenece a la población P
    S_p=Ø
    n_p=0
    Para cada individuo q que pertenece a la población P
        Si p domina a q, entonces:
            S_p=S_p U {q}
            Se agrega q al conjunto de soluciones dominadas por p
        Si no, entonces:
            n_p=n_p+1
            Se incrementa el contador de dominancia de p
    Si p pertenece al primer frente n_p=0, entonces:
        Rango de p=1
        F_1=F_1 U {p}
i=1 (Inicializar el contador frontal)
Mientras $F_i \neq$ Ø
    Q= Ø
    Para cada individuo p que pertenece a Fi
        Para cada individuo p que pertenec a S_p
            n_q=n_q-1
            Si n_q=0, entonces:
                Rango de q=i+1
                Q = Q U {q}

$i=i+1$
$F_i=Q$

A cada individuo se le asigna una distancia de apilamiento de acuerdo con un orden descendente según cada valor de cada función objetivo. Para las soluciones que se encuentran en los extremos del rango de ordenamiento no dominada se les asignara una distancia de apilamiento infinita. A las soluciones intermedias se les asigna un valor de distancia igual a la diferencia absoluta normalizada de los valores de la función objetivo de las dos soluciones adyacentes. El valor total de la distancia de apilamiento será igual a la suma de los valores de distancia individuales correspondiente a cada objetivo (Deb, et al, 2002). La distancia entre soluciones nos permite comparar el grado de diversidad de estas. A continuación, se muestra el pseudo código para el cálculo de la distancia de apilamiento de soluciones que se encuentran agrupadas en el frente no dominado *I*:

Figura 15.

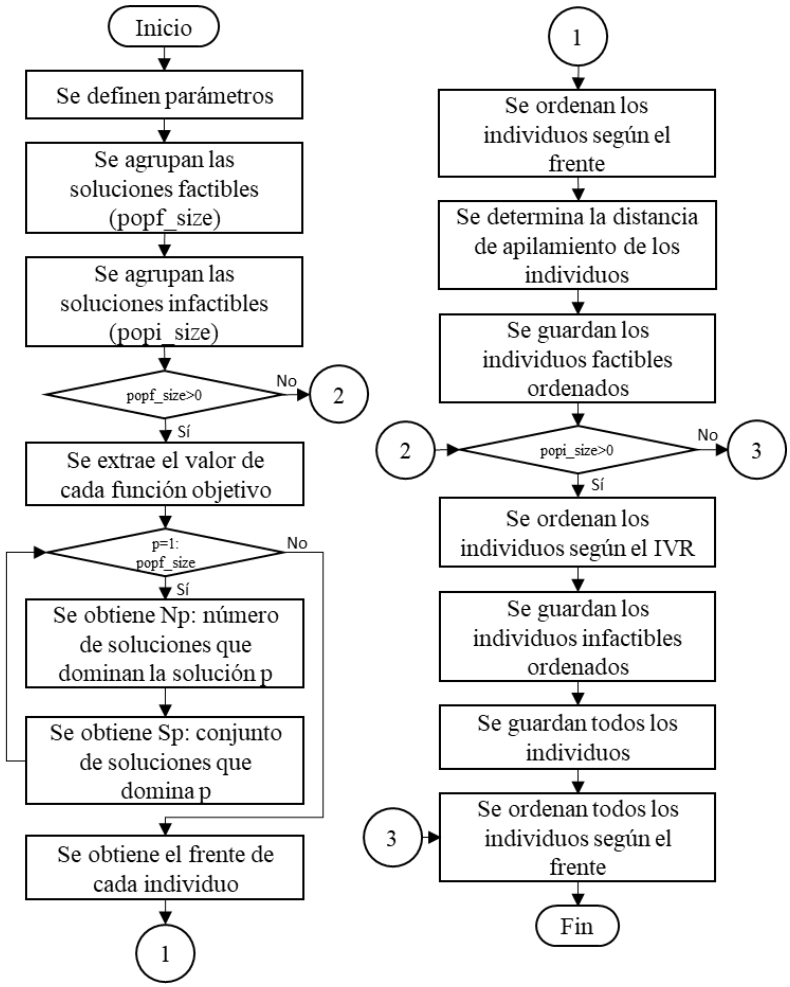
Pseudocodigo de la distancia de apilamiento. Adaptado de Deb et al., 2002

Asignación de la distancia de apilamiento

Se define la cantidad de soluciones en el frente no dominado *I*
$l=|I|$
Se inicializa la distancia:
Para cada *i*, $I[i]_{distance}=0$
Para cada función objetivo m:
    Ordenar los individuos del conjunto I usando cada función objetivo
    Se define la distancia de apilamiento para las soluciones de los límites
    $I[1]_{distance}=I[l]_{distance}=\infty$
    Se define la distancia de apilamiento para las soluciones intermedias
    for i=2 hasta $(l-1)$
        $I[1]_{distance}= I[l]_{distance}+(I(i+1).m - I(i-1).m)/ (f_m^{max} - f_m^{min})$

Figura 16.

Diagrama de flujo del ordenamiento no dominado rápido



7.5. Criterio de selección

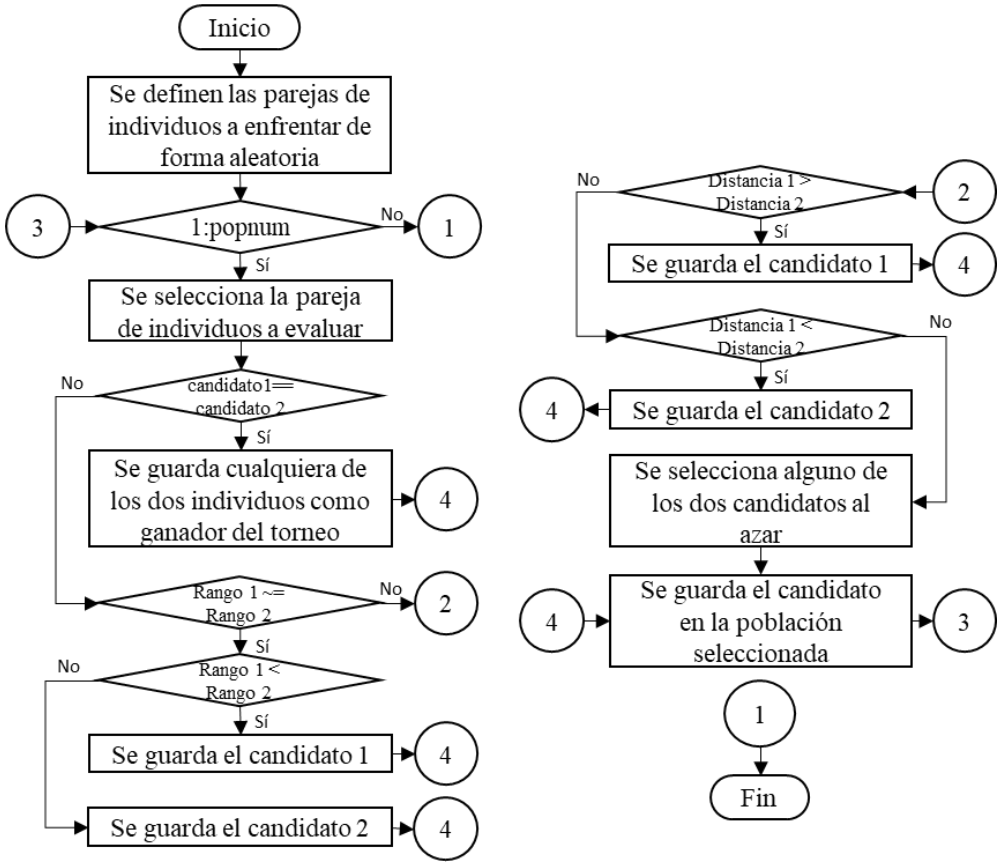
El método de selección usada para el algoritmo es selección por torneo binario, el cual consiste en escoger un par de individuos al azar y evaluarlos entre sí. Se genera una matriz auxiliar *pob_candidatos* con dos columnas que representan a los candidatos que se van a enfrentar en cada torneo. Los candidatos se generan por medio de la función *randperm* tomando como límite la cantidad de individuos definidos en *pob_num.* El proceso de selección valida si se están enfrentando dos individuos iguales para guardar uno solo sin usar ningún criterio para escoger entre ellos.

El criterio de selección para los individuos que son diferentes es el rango en el que se encuentran según su dominancia, se seleccionan los individuos del mejor rango, es decir, el menor.

Para individuos con el mismo rango se evalúa la distancia de apilamiento en donde se selecciona al individuo con mayor distancia, debido a que se encuentra en una región del espacio de búsqueda con un menor número de soluciones cercanas a él, es decir, una región menos poblada. Esto con el fin de mantener la diversidad genética de las poblaciones. Los individuos que ganan cada torneo se guardan en la matriz *poblacion_seleccionada.* En la figura 17 se muestra el flujo del operador genético.

Figura 17.

Diagrama de flujo de la selección de individuos



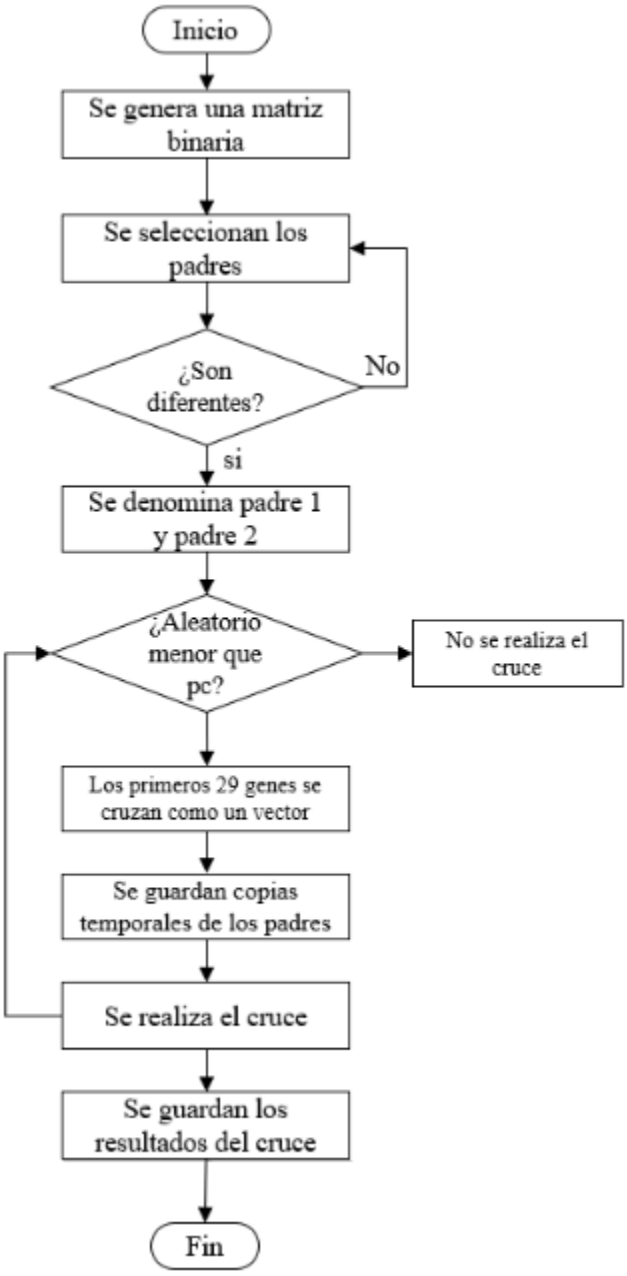
7.6. Operador de cruce

Para realizar el cruce se guarda toda la información de los individuos seleccionados en la matriz *cromosoma_seleccionado_pob*. Se genera un vector aleatorio llamado *rc* que define las parejas de padres que se van a cruzar, en donde el primer miembro de la pareja estará definido por

la posición 2*i-1 del vector *rc* y el segundo por la posición 2*i. Una vez se conocen los padres se procede a definir la probabilidad de que este operador genético suceda. Se genera un número aleatorio distribuido uniformemente que representa la probabilidad de cruce para los padres seleccionados. Se evalúa si este porcentaje es menor o igual a la probabilidad de cruce definida en *pc*, en dicho caso el cruce sucederá, de lo contrario, no.

Figura 18.

Diagrama de flujo del operador de cruce



Como se muestra en la figura 19. el método de cruce que se emplea para el algoritmo consiste en una máscara binaria del mismo tamaño del cromosoma que se genera a partir de valores aleatorios los cuales definen que genes serán transferidos de los padres a cada uno de los hijos. Para generar el hijo 1 los genes que se pasan del padre 1 serán los que coincidan con el valor 1 que determina la máscara binaria y de la misma forma los genes que pasan del padre 2 van a estar determinados por el valor 0 de la máscara. Los hijos se guardan en la matriz *poblacion_cruzada.* De esta forma el hijo 2 estará compuesto por los genes opuesto a los valores que determina la máscara, como se muestra a continuación:

Figura 19.

Operador de cruce utilizando el método propuesto

Padre 1

8	5	1	1	694247	5702	4929	28	1	3	5
8	5	1	1	379235	1841	1685	532	4	1	7

Padre 2

7	7	2	1	639875	5913	1726	933	2	1	1
7	7	2	1	277862	639	161	25	1	2	2

Máscara binaria

0	1	0	0	1	1	1	0	1	1	1
0	1	0	0	1	0	0	1	0	0	1

Hijo 1

8	7	1	1	639875	5913	1726	28	2	1	1
8	7	1	1	277862	1841	1685	25	4	1	2

Hijo 2

7	5	2	1	694247	5702	4929	933	1	3	5
7	5	2	1	379235	639	161	532	1	2	7

7.7. Operador de mutación

El proceso de mutación se realiza con el fin de complementar el proceso de cruce. Se altera en cierta medida a los individuos recreando las pequeñas modificaciones que ocurren en el mundo natural después del proceso de reproducción. La mutación se representa a través de un porcentaje que permite cambiar cierto número de genes del cromosoma. Se define un porcentaje *pm* que representa la probabilidad de que el individuo sea mutado, y también un porcentaje *p_mut* que permite conocer el número de genes a mutar. Una vez se definen estos parámetros se genera un individuo auxiliar con el cual se realizará el intercambio de genes.

Para los genes que representan las variables de decisión, la mutación se realiza cambiando el gen que corresponde a la variable a mutar en los tres escenarios, por los genes del individuo auxiliar bajo las mismas condiciones. Para los otros genes se realiza la mutación únicamente intercambiando el gen seleccionado según la proporción sin considerar el escenario. En la figura 21 se muestra un ejemplo del proceso de mutación.

Con este proceso se demuestran los posibles errores que se pueden presentar en el material genético de los individuos. El cruce y la mutación garantizan el recorrido por todo el espacio de búsqueda, asegurando que todas las zonas tendrán la misma probabilidad de ser exploradas. (MARTÍNEZ RUIZ, 2019). El proceso se ilustra a continuación para un individuo según la estructura definido para el cromosoma. En la figura 20 se presenta el flujo del operador genético:

Figura 20.

Diagrama de flujo del operador de mutación

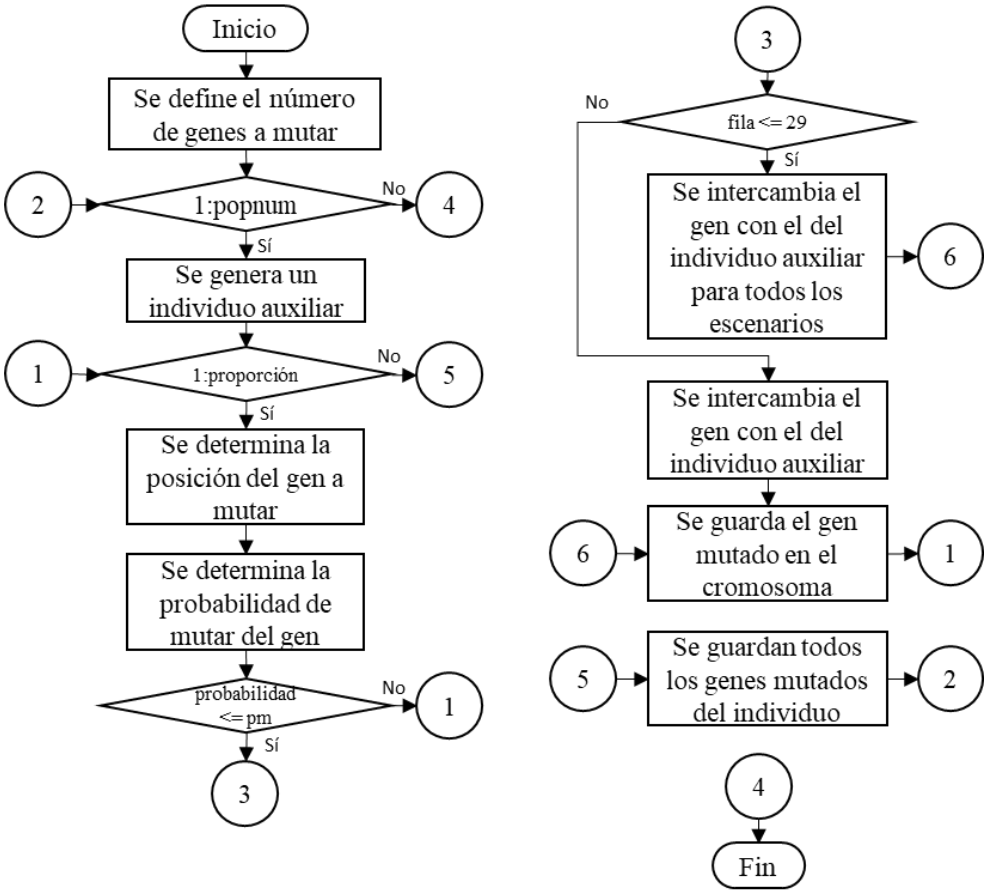


Figura 21.

Operador de mutación utilizando el método propuesto

Individuo

8	7	1	1	639875	5913	1726	28	2	1	1
8	7	1	1	277862	1841	1685	25	4	1	2

Individuo Auxiliar

6	5	1	2	456344	4455	4563	23	1	2	2
6	5	1	2	344325	1945	9875	34	2	2	1

Individuo mutado

8	7	1	1	639875	5913	1726	28	2	1	1
8	7	1	1	277862	1841	1685	25	4	1	2

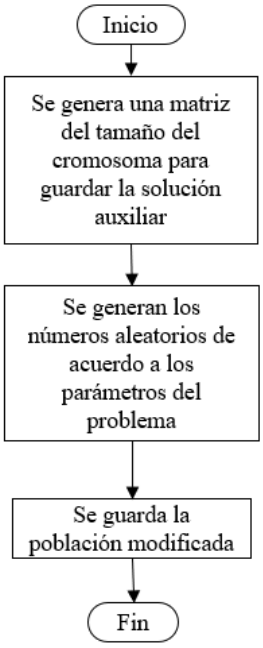
7.8. Operador de inyección

El operador de inyección consiste en sustituir un nuevo número aleatorio de la población para mantener la diversidad y evitar la convergencia prematura. Los mecanismos que preservan la diversidad pueden ayudar a la optimización de dos maneras. Una población diversa es capaz de tratar con funciones multimodales y puede explorar varias colinas en el paisaje de la aptitud simultáneamente. Los métodos de preservación de la diversidad pueden (Gupta & Ghafir, 2012).

En el presente trabajo se considera un operador de inyección el cual se aplica en caso de que cierta cantidad de individuos converja a mínimos locales, es decir, que todos tengan el mismo índice de violación de restricciones. Esta cantidad está determinada por un porcentaje aplicado a *pop_num*, el cual define el número mínimo de soluciones con igual I.V.R. aceptadas antes de realizar la inyección. En caso de que la cantidad de individuos con mismo índice sea igual o mayor a la cantidad definida anteriormente, se reemplazará el número de individuos iguales pero seleccionados de forma aleatoria por individuos generados bajo el mismo procedimiento como se generaron lo individuos de la población inicial. El proceso se detalla en la figura 22.

Figura 22.

Diagrama de flujo del operador de inyección



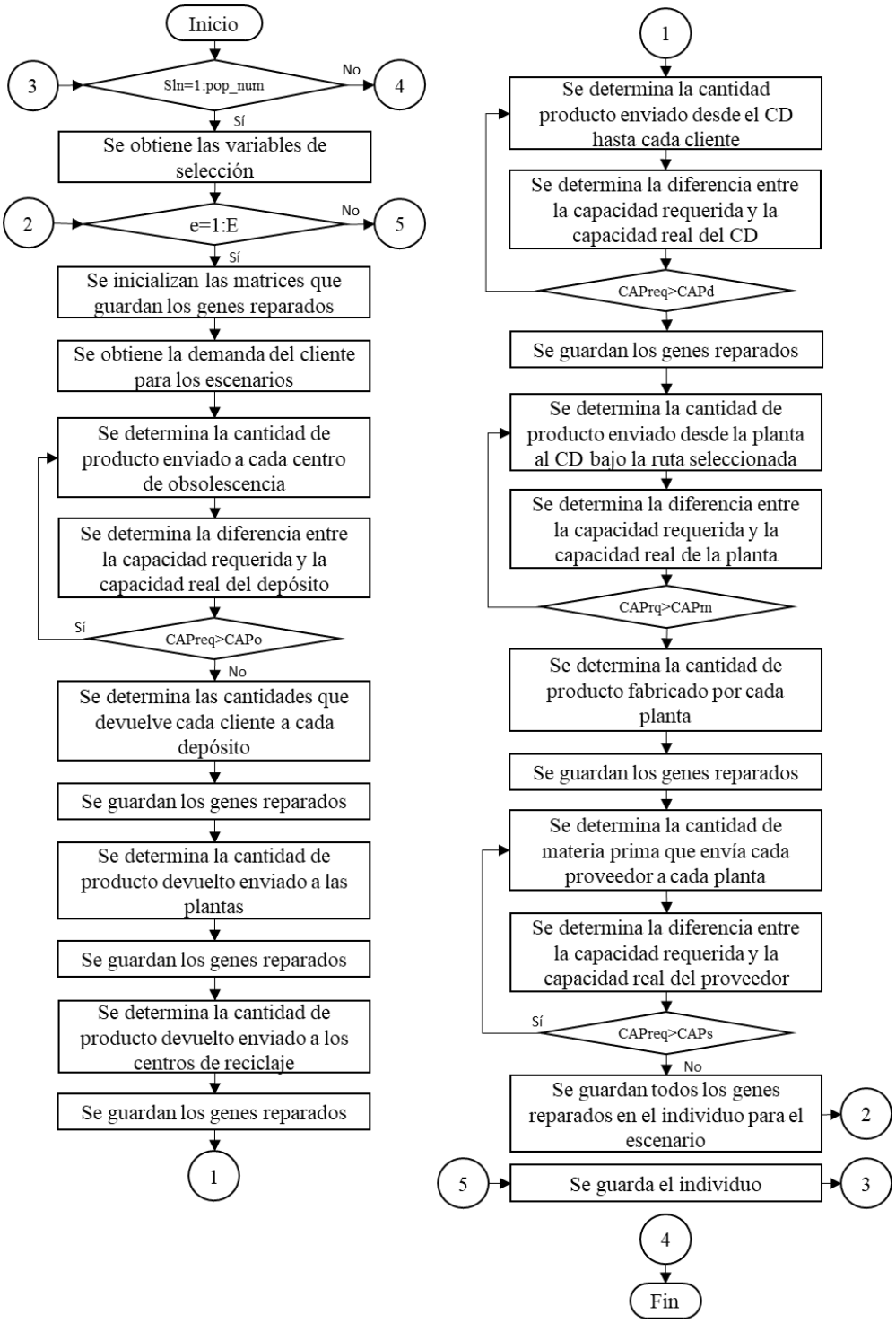
7.9. Reparación de las soluciones

Los algoritmos de reparación transforman soluciones no factibles en soluciones factibles por medio de la aplicación de procesos específicos. Para esto se presentan tres enfoques, el primero se llama Lamarkiano, el cual transforma cada individuo infactible en factible y se repite en todas las generaciones. El enfoque Baldwiniano, se repara la solución no factible solo para la evaluación, pero la población, el cromosoma del individuo o la solución no es modificada y por último en el enfoque Annealing las soluciones no factibles son aceptadas con cierta probabilidad que se reduce a medida que pasan las generaciones (Rasjido et al., 2014).

Para este trabajo se realizó un algoritmo de reparación bajo el enfoque Lamarkiano, en el cual se busca que el individuo cumpla con todas las restricciones del modelo matemático. Es decir, que se respeten las cantidades que se fabrican y se transportan frente a las capacidades de cada eslabón de la cadena, partiendo del supuesto que la demanda de cada cliente se satisface en su totalidad. Para reparar los individuos se extraen los valores iniciales de las variables de selección y con base en ellas se definen las demás variables. Una vez se obtiene el individuo después de la reparación se guarda en la matriz *poblacion_reparada*. En la figura 23 se muestra el flujo del proceso de reparación de las soluciones.

Figura 23.

Diagrama de flujo del algoritmo de reparación



7.10. Definición nueva población

A continuación, se describe el proceso de selección para definir la nueva población según la metodología propuesta por Deb, et al, 2002 que se ilustra en la figura 24.

La población que será introducida P_{t+1} en la siguiente iteración del algoritmo, va a estar definida por una población *Rt* que es la combinación de la población resultante de la iteración anterior P_t y la población de descendientes de la iteración actual Q_t después de aplicar los operadores genéticos. Las dos poblaciones tienen tamaño *pop_num* por lo que se obtendrá una población *Rt* de tamaño *2pop_num*.

Una vez se obtiene R_t se procede a ordenar la población según la no dominancia, para separar los individuos según los frentes F_1, F_2, F_3... F_t, en donde F_1 representan el mejor frente, F_2 el siguiente mejor frente y así sucesivamente. Los individuos del frente F_1 serán los mejores de ambas poblaciones por lo que deben tener mayor importancia que cualquier otro individuo de la población combinada.

El conjunto P_{t+1} estará conformado por la cantidad de individuos necesario para alcanzar *pop_num.* Si la cantidad de soluciones del frente F_1 es menor que *pop_num* se seleccionan todos los individuos de este frente para conformar P_{t+1}. Los miembros que falten para llenar la población serán seleccionados de los frentes siguientes a F_1 y continua hasta que se haya alcanzado el número requerido de individuos.

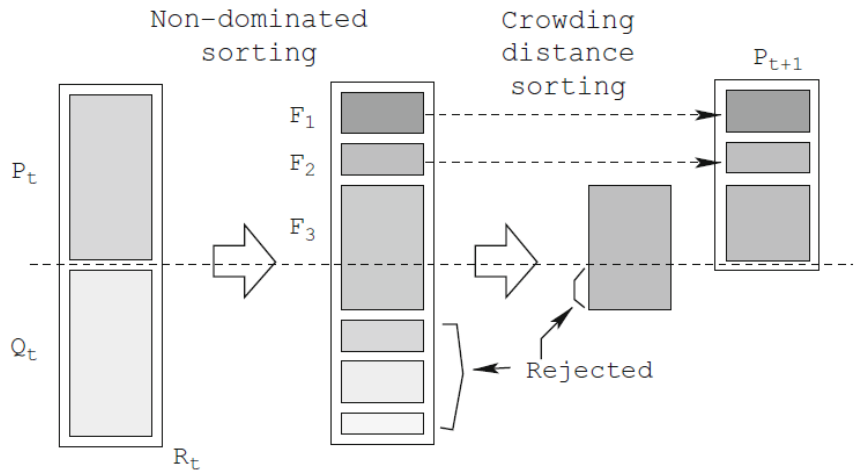
Supongamos que el frente F_t es el último conjunto de soluciones del cual ya no se podrán acomodar más individuos para alcanzar el tamaño requerido de P_{t+1}, es decir, que desde F_1 hasta F_t se supera *pop_num*. Entonces, de F_t se debe seleccionar ciertas soluciones para completar el conjunto P_{t+1} y lograr exactamente *pop_num.* Dado que estas soluciones pertenecen al mismo frente, es decir que tiene el mismo nivel de no dominancia, se deben ordenar de forma descendente según la distancia de apilamiento (de mayor a menor distancia) para escoger la cantidad de individuos que completarán la población que va a ser introducida en la siguiente iteración.

El proceso asegura por una parte el elitismo, dado que se tienen en cuenta los mejores individuos no dominados de la iteración anterior y la actual para definir los padres de la siguiente

iteración, y por otra parte la diversidad al determinar la distancia de apilamiento de las soluciones y seleccionar a las que se encuentran en espacios menos poblados. La población P_{t+1} será alterada por todos los operadores genéticos definidos para el algoritmo durante la próxima iteración. El proceso de selección descrito en esta sección se repetirá el número de veces especificado en *gen.*

Figura 24.

Procedimiento NSGA II, tomado de Deb et al., 2002



7.11. Criterio de selección de solución

Una vez termina el algoritmo, es decir, cuando se completa el número de iteraciones definido en *gen* se procede a determinar la solución que dará respuesta al modelo matemático planteado. En la matriz *ordenamiento_pob* se almacenan los mejores individuos obtenidos en cada iteración con sus respectivos valores para cada función objetivo, los índices de violación de restricciones, el rango de no dominancia y la distancia de apilamiento.

Para seleccionar el individuo solución se ordenan todas las soluciones de *ordenamiento_pob* según su nivel de no dominancia como se explicó en la sección 7.4 de este capítulo. Para dar respuesta al problema se seleccionan a los individuos que pertenecen al primer rango y de ellos se escoge el que responde mejor a los objetivos del modelo.

8. Validación del algoritmo

El algoritmo se implementó en MATLAB, y se corrió en la versión de MATLAB R2020b, utilizando un computador portátil ASUS con una memoria instalada (RAM) de 4 Gb y un procesador Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz. Los parámetros del modelo se obtienen del artículo de S.B. Ebrahimi y se encuentran recopilados en el Apéndice A.

El número de individuos por población se fija en 50 para el nivel bajo dado que el cromosoma tiene codificación real (DeJong, 1975) y los demás valores se fijan luego de experimentación empírica. En el Apéndice C se encuentran los valores que se utilizaron para la generación de los intervalos de confianza en MINITAB 20.

Tabla 7.

Niveles de los parámetros utilizados

Parámetro	Nivel bajo	Nivel alto
pob_num	50	100
gen	100	300
pc	70%	90%
pm	2%	10%
p_mut	50%	100%

Los valores anteriores se utilizaron durante el proceso de validación del algoritmo para conocer la combinación de niveles más apropiada. Se usó un diseño de experimento de 2^5 en el cual se tienen 5 réplicas para cada combinación de parámetros. Se usaron 160 tratamientos para correr el algoritmo. Los datos se analizaron en MINITAB 20 realizando el análisis ANOVA de un solo factor. Para la experimentación y posterior validación de las soluciones encontradas por el algoritmo, se usarán las siguientes métricas:

1. Resultados de las funciones objetivos
2. Cantidad de soluciones no dominadas encontradas.
3. Velocidad computacional.
4. Soluciones no dominadas por minuto.
5. Dominancia entre las soluciones no dominadas encontradas.

Los valores obtenidos para cada una de las funciones objetivo son los valores reportados en la solución del problema bajo el modelo sin considerar descuentos por cantidad y considerando el problema de ruteo realizado por (Ebrahimi, 2018) y es la medida que se usa para la validación del algoritmo. Se comparan también los resultados según la variación del factor de ponderación para la respuesta de la cadena de suministro, así como lo realiza el autor.

Las demás métricas se utilizan para comparar el desempeño y calidad de las soluciones al experimentar con algunos de los parámetros del algoritmo. La cantidad de soluciones no dominadas encontradas consiste en la cantidad de soluciones de Pareto diferentes encontradas por el algoritmo en cada corrida experimental. El tiempo de computación se mide respecto a una corrida experimental, sus unidades son segundos y es un valor calculado por MATLAB. Y las soluciones no dominadas encontradas por minuto, se calculan dividiendo las soluciones no dominadas encontradas por el tiempo computacional convertido de segundos a minutos, permitiendo medir en cierta forma el equilibrio entre las soluciones encontradas y el tiempo que toma encontrarlas.

La dominancia entre las soluciones no dominadas encontradas consiste en una comparación de las soluciones halladas con diferentes niveles de un parámetro, básicamente es comprobar si un conjunto de soluciones domina al otro; para ello, se extraen y unen los conjuntos no dominados finales de cada réplica experimental de cada nivel del factor y se realiza el ordenamiento no dominado rápido al conjunto resultante. Para la comparación se grafican todos los conjuntos en una misma figura. La dominancia, dada una gráfica de dos funciones de minimización, se puede evidenciar si alguna figura se encuentra más a la izquierda y más hacia abajo que el resto

Con base en los resultados de experimentación empírica realizada al probar el algoritmo en MATLAB y garantizar su correcto funcionamiento, se realiza diseño experimental para los parámetros mencionados a continuación.

8.1. Diseño experimental para el tamaño de la población

Para definir el tamaño de la población se tuvieron en cuenta las siguientes hipótesis a priori:

1. Un tamaño de población bajo puede generar mejores resultados para el algoritmo propuesto.
2. La relación entre el tamaño de la población y el tiempo computacional es directa: mayor tamaño de población implica mayor tiempo computacional.

A continuación se presentan los resultados experimentales para el tamaño de población:

Tabla 8.

Intervalos de confianzas para el factor tamaño de la población y la variable de estudio cantidad de soluciones no dominadas

Tamaño de Población	N	Media	Desv.Est.	IC de 95%
50	80	3,237	1,416	(2,927; 3,548)
100	80	2,888	1,396	(2,577; 3,198)

Figura 25.

Gráfica de intervalos de los niveles del tamaño de población para la variable respuesta cantidad de soluciones no dominadas encontradas

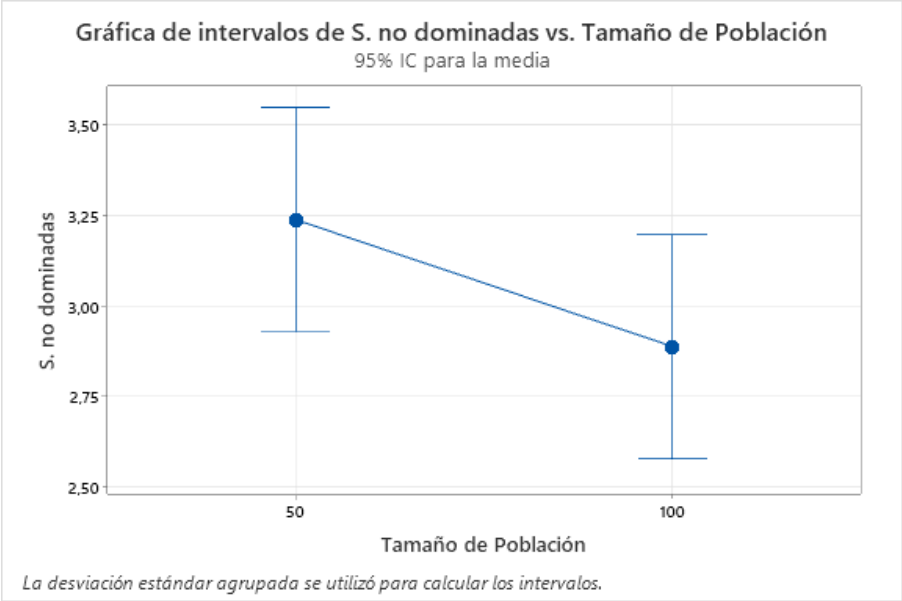


Tabla 9.

Intervalos de confianzas para el factor tamaño de la población y la variable de estudio tiempo computacional

Tamaño de Población	N	Media	Desv.Est.	IC de 95%
50	80	64,49	28,33	(54,76; 74,22)
100	80	111,30	55,52	(101,57; 121,03)

Figura 26.

Gráfica de intervalos de los niveles del tamaño de población para la variable respuesta tiempo computacional

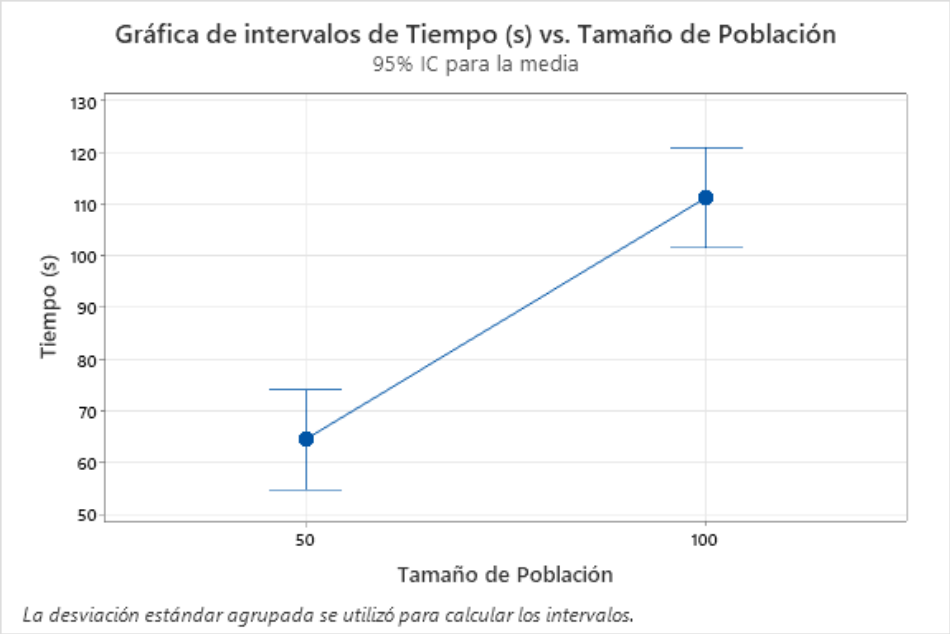


Tabla 10.

Intervalos de confianzas para el factor tamaño de la población y la variable de estudio cantidad de soluciones no dominadas por minuto

Tamaño de Población	N	Media	Desv.Est.	IC de 95%
50	80	3,515	2,085	(3,130; 3,899)
100	80	1,946	1,310	(1,561; 2,330)

Figura 27.

Gráfica de intervalos de los niveles del tamaño de población para la variable respuesta cantidad de soluciones no dominadas por minuto

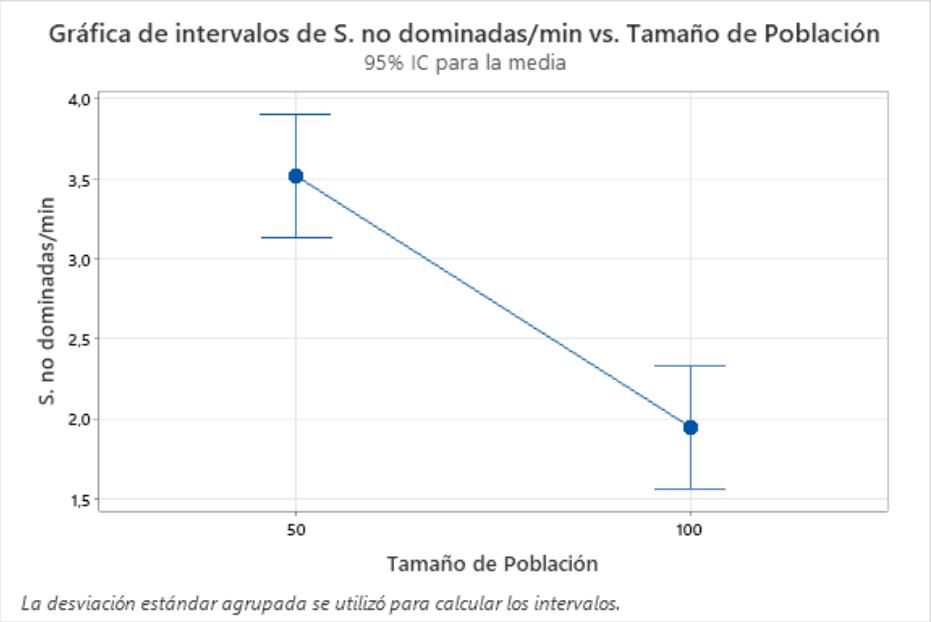
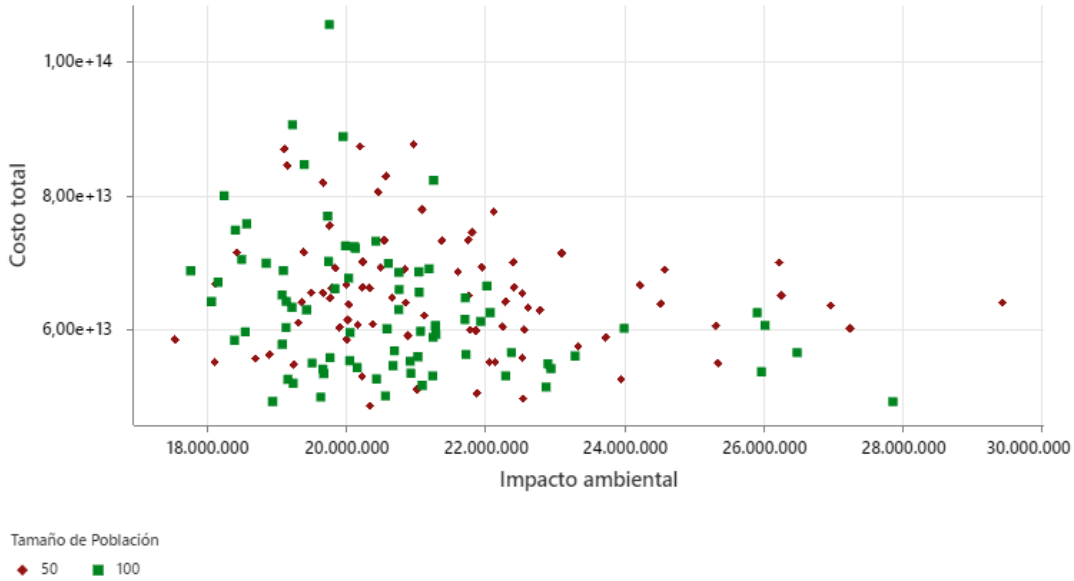


Figura 28.

Comparación de los conjuntos de mejores soluciones de todas las réplicas del experimento del factor tamaño de la población.



En la tabla 8, tabla 9 y tabla 10 se presenta el resultado de los intervalos de confianza para cada nivel del factor tamaño de población según cada una de las variables respuestas contempladas. El analisis se realiza para conocer la mejor cantidad de individuos según el comportamiento de

cada nivel según cada variable respuesta con la que se analiza. Con el fin de concluir usando los resultados de forma visual, en la figura 18, figura 19 y figura 20 se muestra la representación gráfica de estos intervalos de confianza para el tamaño de la población según el número de soluciones no domanadas, el tiempo de cómputo, y el número de soluciones no dominadas por minuto, respectivamente.

En la tabla 8, se analiza el comportamiento de la cantidad de soluciones no dominadas para cada uno de los niveles del tamaño de la población. Se evidencia que para poblaciones con 50 individuos la media de soluciones no dominadas es mayor a la media de soluciones alcanzadas para poblaciones de 100 individuos. Con lo cual se concluye que para tamaños de población de 50 individuos se logran un mayor número de soluciones no dominadas. La cantidad de soluciones se encuentran más dispersas en poblaciones de 50 individuos que en poblaciones de 100 individuos, lo que genera que esta última tenga intervalos de confianza menos amplios que la primera.

Lo anterior se puede evidenciar gráficamente en la figura 18 y se da un análisis similar para la variable respuesta soluciones no dominadas por minuto según la tabla 10 y la figura 20. Con respecto a la variable respuesta de tiempo computacional, en la tabla 9, se evidencia que para el nivel 50 del factor se logra un promedio de tiempo menor durante cada ejecución del algoritmo, por lo tanto se concluye que con un menor número de individuos se requiere menor tiempo computacional. Los datos del análisis se encuentran más densos y consistentes para el nivel 50 según el porcentaje de desviación estandar en comparación con el nivel 100.

En la figura 21, se presenta un diagrama de dispersión que muestra el comportamiento de las mejores soluciones no dominadas obtenidas durante cada corrida del algoritmo. Los cuadros verdes y rojos representan el nivel 100 y 50 del factor tamaño de la población, respectivamente. La finalidad del diagrama de dispersión es encontrar alguna tendencia en las corridas según el costo total y el impacto ambiental. Se observa que las soluciones obtenidas con un tamño de población de 100 tienden a estar más concentradas en valores menores para cada función objetivo mientras que las soluciones con tamaño de población 50 se encuentras más dispersas y se ubican en valores más grande para cada función objetivo.

Dado lo anterior se confirma la primera y segunda hipótesis, y se concluye que con un nivel de confianza de 95% el nivel 50 para el factor de tamaño de la población consigue un mayor número promedio de soluciones no dominadas y requiere menor tiempo promedio de cómputo que el nivel 100.

8.2. Diseño experimental para el numero de generaciones

Para definir el número de generaciones se tuvieron en cuenta las siguientes hipótesis a priori:

1. Un número de generaciones alto permite encontrar un mayor número de soluciones no dominadas.
2. El número de generaciones es diretamente proporcional al tiempo que se emplea para terminar cada iteración.

A continuación se presentan los resultados experimentales para el número de generaciones:

Tabla 11.

Intervalos de confianzas para el factor generaciones y la variable de estudio cantidad de soluciones no dominadas

Generaciones	N	Media	Desv.Est.	IC de 95%
100	80	2,737	1,290	(2,433; 3,042)
300	80	3,388	1,463	(3,083; 3,692)

Figura 29.

Gráfica de intervalos de los niveles de las generaciones para la variable respuesta cantidad de soluciones no dominadas

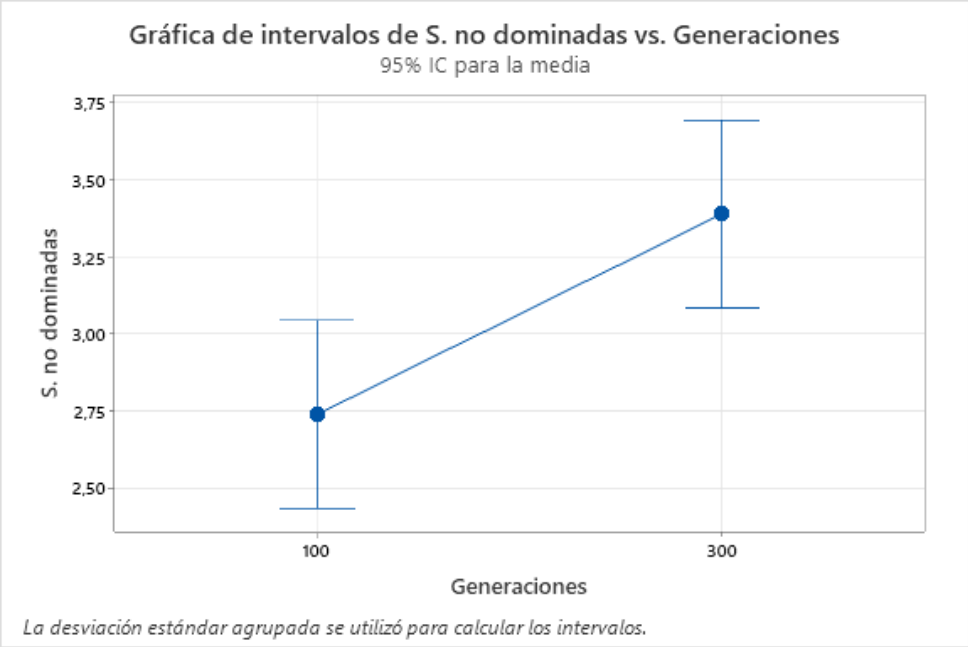


Tabla 12.

Intervalos de confianzas para el factor generaciones y la variable de estudio tiempo computacional

Generaciones	N	Media	Desv.Est.	IC de 95%
100	80	51,96	17,45	(44,34; 59,58)
300	80	123,83	45,54	(116,21; 131,44)

Figura 30.

Gráfica de intervalos de los niveles de las generaciones para la variable respuesta tiempo computacional

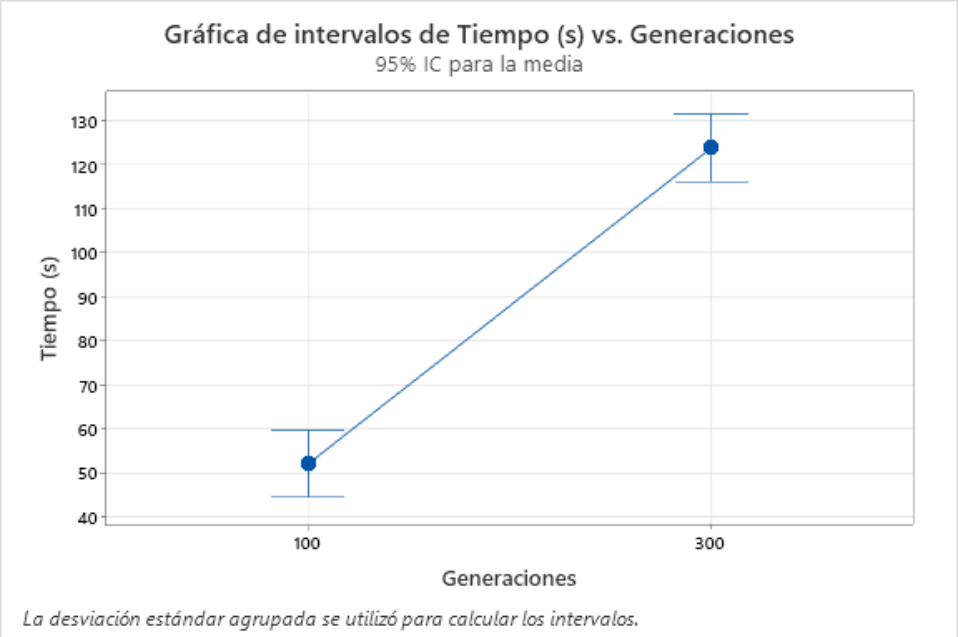


Tabla 13.

Intervalos de confianzas para el factor generaciones y la variable de estudio soluciones no dominadas por minuto

Generaciones	N	Media	Desv.Est.	IC de 95%
100	80	3,543	2,154	(3,161; 3,924)
300	80	1,918	1,154	(1,536; 2,299)

Figura 31.

Gráfica de intervalos de los niveles de las generaciones para la variable respuesta soluciones no dominadas por minuto

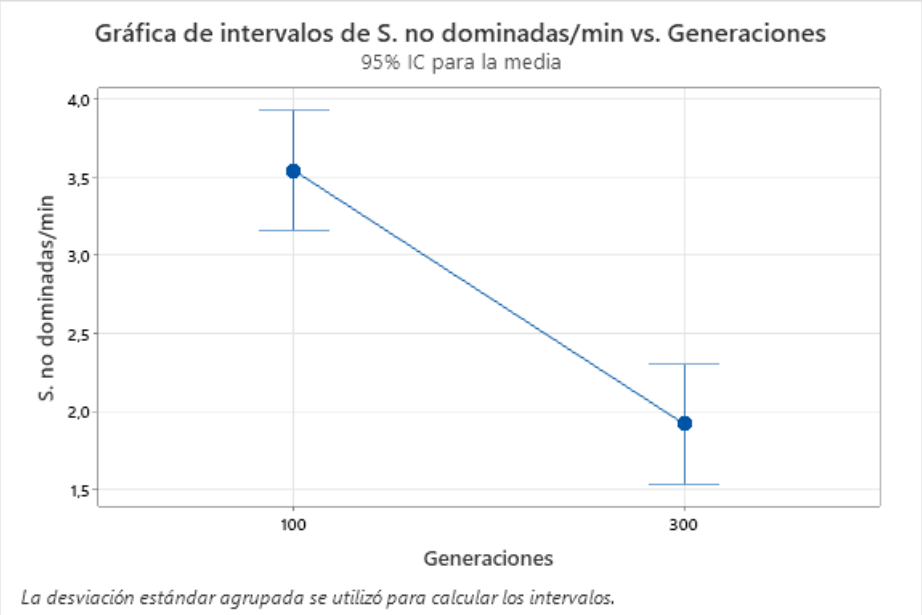
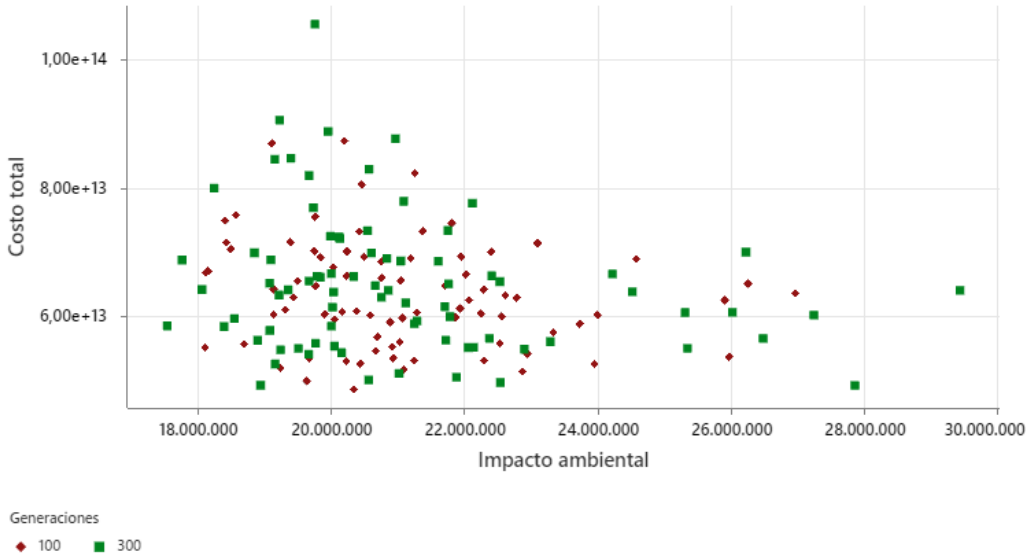


Figura 32.

Comparación de los conjuntos de mejores soluciones de todas las réplicas del experimento del factor generaciones.



En la tabla 11, tabla 12 y tabla 13, se presentan los resultados del análisis de un solo factor. Se analiza el comportamiento del factor número de generaciones según sus niveles, para cada una de las variables respuestas contempladas en el análisis. Se busca conocer el número adecuado de generaciones según el comportamiento de cada nivel del factor con cada variable respuesta. En la figura 29, figura 30 y figura 31 se muestra gráficamente el comportamiento antes mencionado

sujeto al número de soluciones no dominadas, tiempo de cómputo y número de soluciones no dominadas por minuto, respectivamente.

En la tabla 11, se analiza la interacción de cada nivel con el número de soluciones no dominadas. Se puede evidenciar que la media de soluciones no dominadas alcanzada por el nivel 300 de generaciones es mayor que la media de soluciones que logra el nivel 100. Se concluye entonces que para 300 generaciones se esperan más soluciones no dominadas. La cantidad de soluciones se encuentran más dispersas en el nivel 300 que en el nivel 100, lo que genera que esta última tenga intervalos de confianza menos amplios que la primera. El análisis anterior se puede apreciar gráficamente en la figura 29.

La tabla 12 muestra la interacción de cada nivel con el tiempo requerido para ejecutar el algoritmo, se observa que para un número mayor de generaciones el tiempo promedio empleado será mayor que el tiempo promedio empleado para un número menor de generaciones. Se concluye que con un menor número de generaciones se requiere menor tiempo computacional. Los datos del análisis se encuentran más densos y consistentes para el nivel 100 según el porcentaje de desviación estandar en comparación con el nivel 300.

En la tabla 13 se aprecia que para el nivel 100 del factor generaciones se consiguen en promedio un mayor número de soluciones no dominadas por minuto en comparación a las soluciones que logra el nivel 300 en el mismo lapso. De igual manera en la figura 31 se puede apreciar la situación antes descrita.

En la figura 32, se observa la dispersión de los dos niveles del factor según las mejores soluciones de cada corrida del algoritmo para el costo total y el impacto ambiental. Los cuadros verdes y rojos representan el nivel 300 y 100 del factor numero de generaciones, respectivamente. La finalidad del diagrama de dispersión es encontrar alguna tendencia en las corridas según el costo total y el impacto ambiental. Se observa que las soluciones obtenidas para ambos niveles presentan un comportamiento similar y se encuentran con el mismo nivel de dispersión. Ambos niveles tienden a estar concentrados en valores menores para cada función objetivo.

Según el análisis anterior se confirma la primera y segunda hipótesis planteada. El número de soluciones dominadas encontradas para el nivel 300 es mayor en comparación al nivel 100, aunque por minuto este último encuentra más soluciones. El tiempo de cómpito también es mayor para el nivel con mayor número de generaciones. Finalmente, se selecciona el nivel 100 debido a que encuentra más soluciones por minuto y tiene mejor rendimiento computacional.

8.3. Diseño experimental para la probabilidad de cruce

Para definir el valor de la probabilidad de cruce se tuvieron en cuenta las siguientes hipótesis a priori:

1. Un porcentaje de probabilidad de cruce mayor permite encontrar un mayor número de soluciones no dominadas.
2. El porcentaje de probabilidad de cruce no tiene efecto significativo sobre el tiempo computacional.

A continuación se presentan los resultados experimentales para la probabilidad de cruce:

Tabla 14.

Intervalos de confianza para el factor probabilidad de cruce y la variable de estudio soluciones no dominadas

Probabilidad cruce	N	Media	Desv.Est.	IC de 95%
0,7	80	3,025	1,599	(2,712; 3,338)
0,9	80	3,100	1,208	(2,787; 3,413)

Figura 33.

Gráfica de intervalos de los niveles de la probabilidad de cruce para la variable respuesta soluciones no dominadas

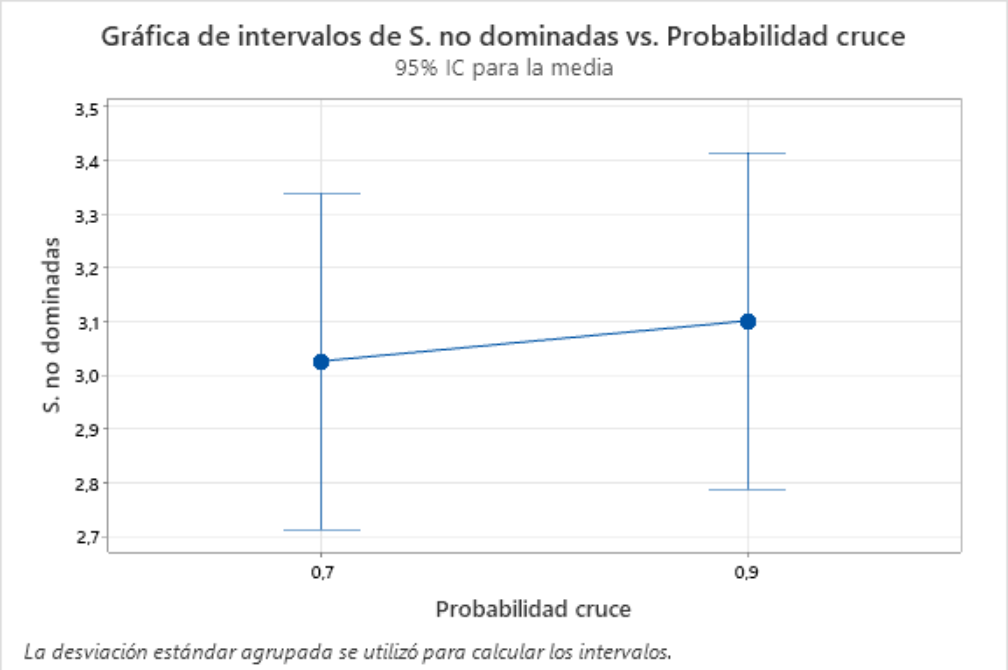


Tabla 15.

Intervalos de confianza para el factor probabilidad de cruce y la variable de estudio tiempo computacional.

Probabilidad cruce	N	Media	Desv.Est.	IC de 95%
0,7	80	87,13	47,99	(76,10; 98,17)
0,9	80	88,66	51,87	(77,62; 99,69)

Figura 34.

Gráfica de intervalos de los niveles de la probabilidad de cruce para la variable respuesta tiempo computacional

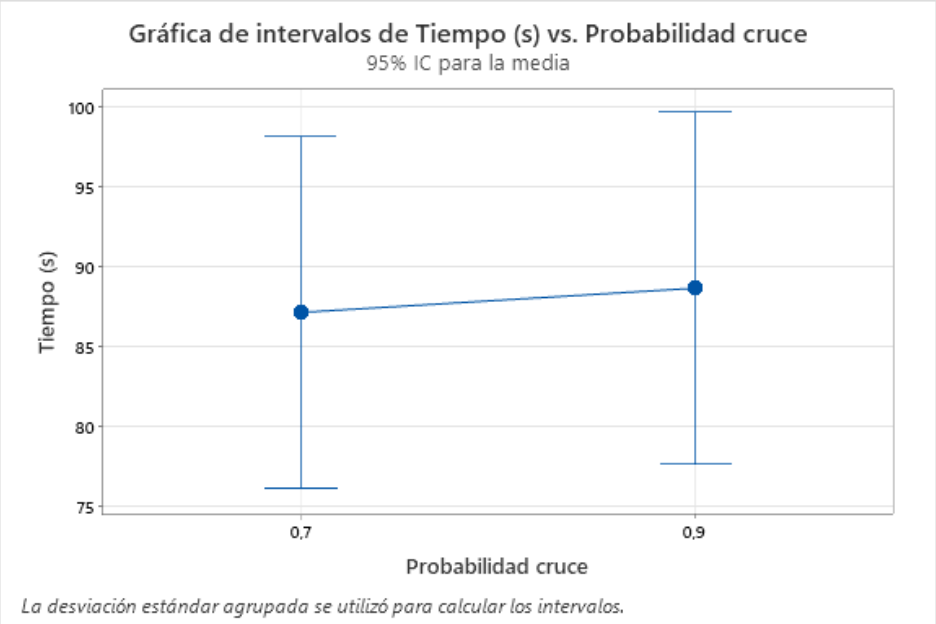


Tabla 16.

*Intervalos de confianza para el factor probabilidad de cruce y la variable de estudio soluciones no dominadas por minuto***.**

Probabilidad cruce	N	Media	Desv.Est.	IC de 95%
0,7	80	2,664	1,942	(2,243; 3,086)
0,9	80	2,796	1,878	(2,374; 3,218)

Figura 35.

Gráfica de intervalos de los niveles de la probabilidad de cruce para la variable respuesta soluciones no dominadas por minuto.

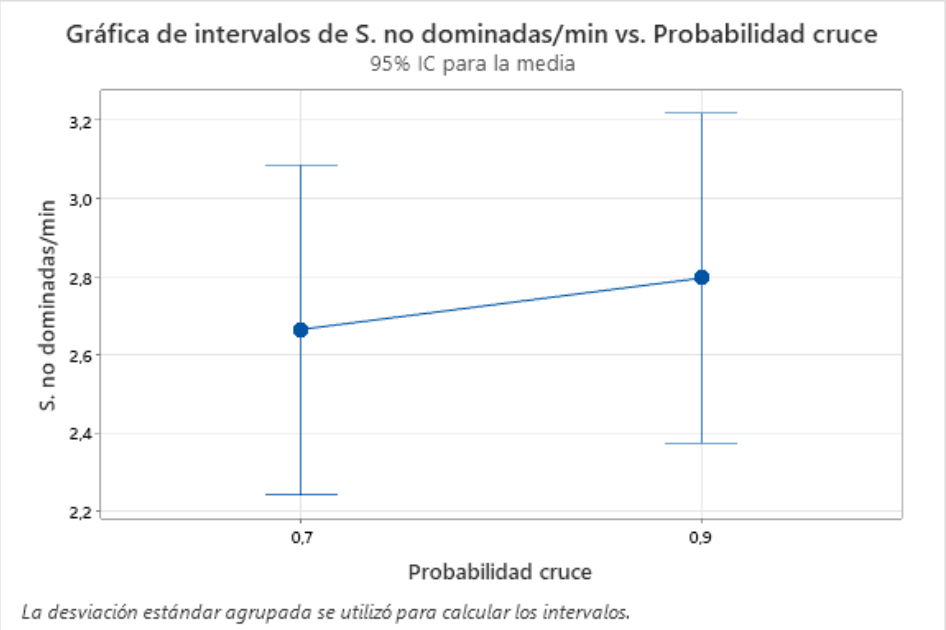
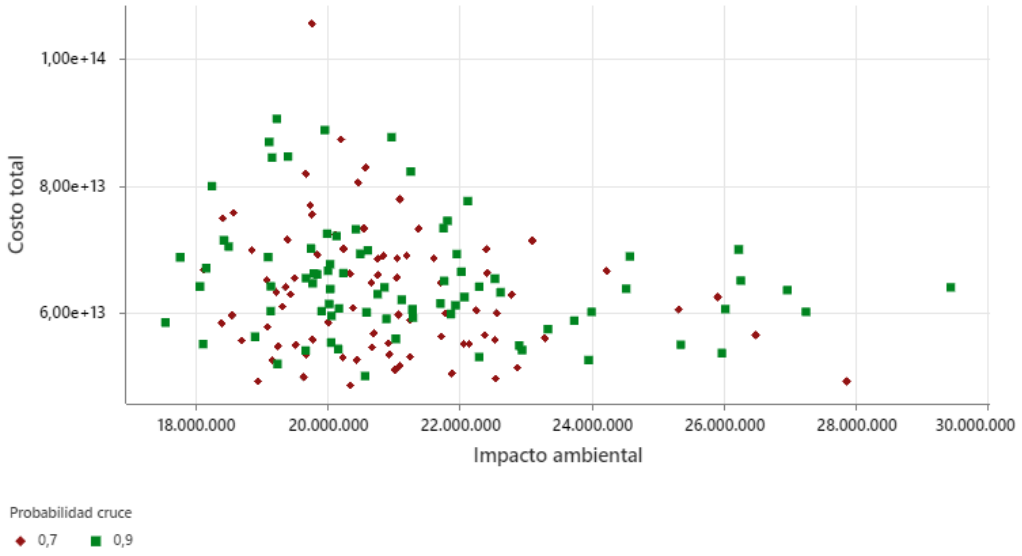


Figura 36.

Comparación de los conjuntos de mejores soluciones de todas las réplicas del experimento del factor probabilidad de cruce.



En los intervalos de confianza resultantes para los diferentes factores de estudio, presentadas en las Tabla 14, Tabla 15 y Tabla 16, se evidencia que no existe diferencia significativa entre la probabilidad de cruce del 70% y el 90%, para ninguna de las variables de estudio (cantidad de soluciones no dominadas encontradas, tiempo computacional en segundos y cantidad de

solucioens no dominadas encontradas por minuto). No obstante, la media del nivel de cruce 90% es mayor para todas las variables de estudio, tal cómo se puede observar en gráficas **¡Error! No se encuentra el origen de la referencia.**33, **¡Error! No se encuentra el origen de la referencia.**34 y **¡Error! No se encuentra el origen de la referencia.**35.

Dado que no existen diferencias significativas entra las medias de los niveles del factor porcentaje de cruce 70% y 90% para la variable de estudio tiempo computacional, no existe evidencia significativa para rechazar la segunda hipótesis planteada y por tanto se puede afirmar con un nivel de confianza del 95% que los diferentes niveles del factor porcentaje de cruce no tienen un efecto significativo sobre la variable respuesta tiempo computacional.

Dado que las medias para las otras variables de estudio tampoco presentan diferencias significativas, se puede concluir con un nivel de confianza del 95% que el factor porcentaje de mutación no ejerce un efecto significativo en ninguna variable respuesta sin embargo con una probabilidad de mutación del 90% si encuentra un mayor número de soluciones no dominadas, por lo tanto, se acepta la primera hipótesis.

En la gráfica 36, se presenta un diagrama de dispersión que muestra el comportamiento de las mejores soluciones no dominadas obtenidas durante cada corrida del algoritmo. Los cuadros verdes y rojos representan el nivel 90 y 70 del factor probabilidad de cruce, respectivamente. Se observa que las soluciones obtenidas con una probabilidad del 90% tienden a estar más concentradas en valores menores para cada función objetivo mientras que las soluciones con probabilidad del 70% se encuentran más dispersas y se ubican en valores más grandes para cada función objetivo.

8.4. Diseño experimental para la probabilidad de mutación

Para definir el valor de la probabilidad de mutación se tuvieron en cuenta las siguientes hipótesis a priori:

1. Un porcentaje de probabilidad de cruce menor al usualmente usado (10%) no genera mejores resultados para el algoritmo propuesto.
2. El porcentaje de probabilidad de mutación es diretamente proporcional al tiempo que se emplea para terminar cada iteración.
3. Un porcentaje de probabilidad de mutación menor al usualmente usado (10%) permite encontrar un número similar de soluciones no dominadas pero en menos tiempo.

A continuación se presentan los resultados experimentales para la probabilidad de cruce:

Tabla 17.

Intervalos de confianza para el factor probabilidad de mutación y la variable de estudio soluciones no dominadas.

Probabilidad mutacion	N	Media	Desv.Est.	IC de 95%
0,02	80	3,050	1,413	(2,737; 3,363)
0,10	80	3,075	1,421	(2,762; 3,388)

Figura 37.

Gráfica de intervalos de los niveles de la probabilidad de mutación para la variable respuesta soluciones no dominadas.

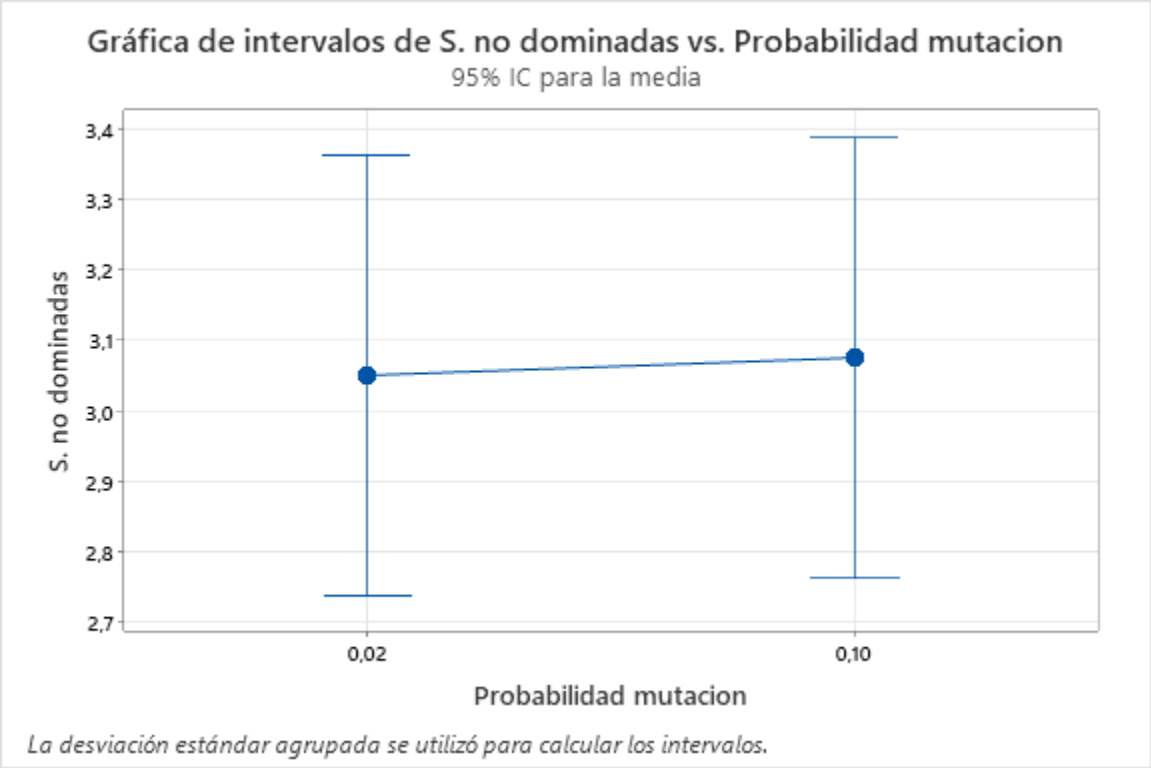


Tabla 18.

Intervalos de confianza para el factor probabilidad de mutación y la variable de estudio tiempo computacional.

Probabilidad mutacion	N	Media	Desv.Est.	IC de 95%
0,02	80	83,23	46,73	(72,25; 94,22)
0,10	80	92,56	52,59	(81,57; 103,54)

Figura 38.

Gráfica de intervalos de los niveles de la probabilidad de mutación para la variable respuesta tiempo computacional.

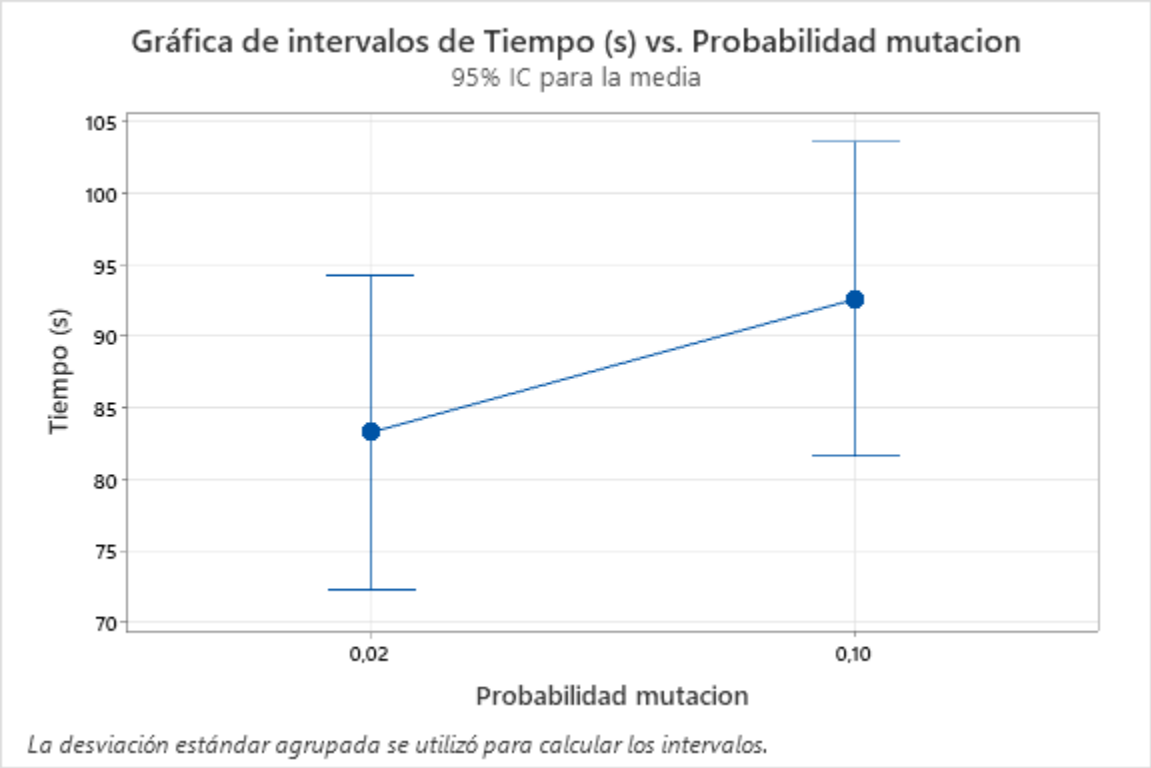


Tabla 19.

*Intervalos de confianza para el factor probabilidad de mutación y la variable de estudio soluciones no dominadas por minuto***.**

Probabilidad mutacion	N	Media	Desv.Est.	IC de 95%
0,02	80	2,816	1,945	(2,394; 3,237)
0,10	80	2,645	1,874	(2,223; 3,067)

Figura 39.

Gráfica de intervalos de los niveles de la probabilidad de mutación para la variable respuesta soluciones no dominadas por minuto.

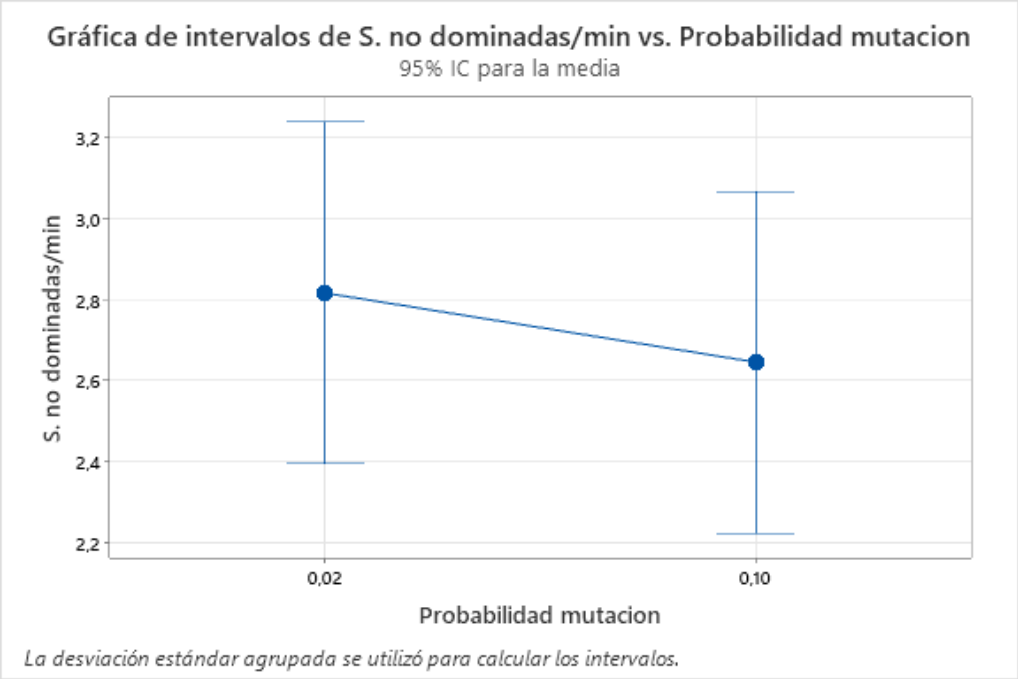
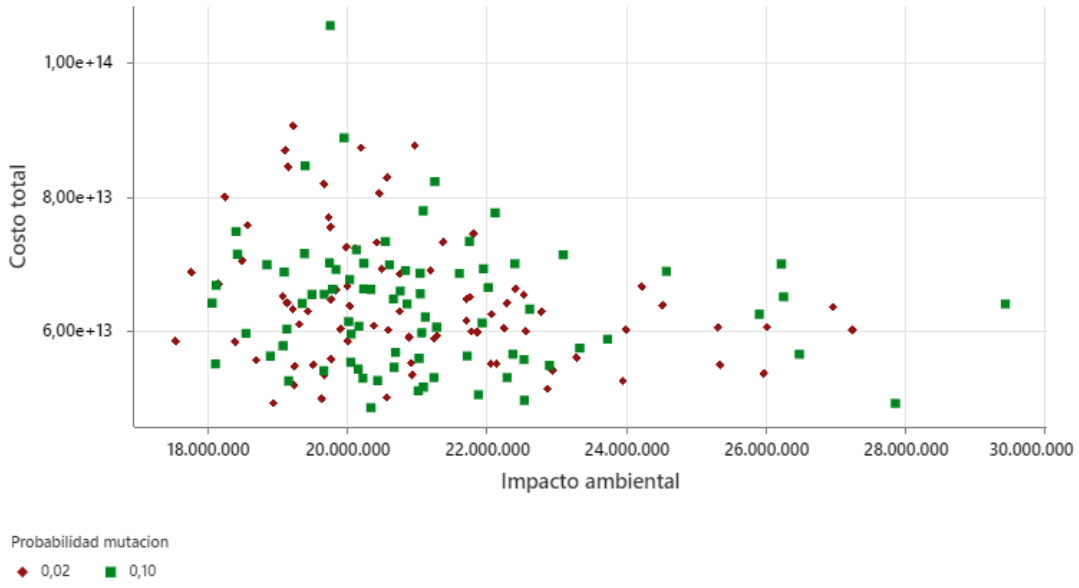


Figura 40.

Comparación de los conjuntos de mejores soluciones de todas las réplicas del experimento del factor probabilidad de mutación.



En los intervalos de confianza resultantes presentados en las Tabla 17, se evidencia que no existe diferencia significativa entre la probabilidad de mutación del 2% y el 10%, para la variable de estudio cantidad de soluciones no dominadas encontradas, como se puede observar en la figura

37, sin embargo una probabilidad de mutación del 2% presenta un menor número de soluciones no dominadas, por lo tanto, con un nivel de confianza del 95% se puede afirmar que un porcentaje de mutación inferior al usualmente usado no genera mejores soluciones al algoritmo propuesto.

Respecto al tiempo computacional en segundos, como se presenta en la tabla 18, para un porcentaje del 2% se presenta una mejor media de tiempo frente al 10%, representado graficamente en la figura 38. Dado que no existe evidencia significativa entre una probabilidad del 2% y el 10%, el número de soluciones no dominadas por minuto es mayor para un 2% como se detalla en la tabla 19 y se representa en la figura 39, por lo tanto, con un nivel de confianza del 95% se confirma que el porcentaje de mutación es directamente proporcional al tiempo computacional y que un porcentaje de probabilidad de mutación menor al usualmente usado (10%) permite encontrar un número similar de soluciones no dominadas pero en menos tiempo.

En la gráfica 40, se presenta un diagrama de dispersión que muestra el comportamiento de las mejores soluciones no dominadas obtenidas durante cada corrida del algoritmo. Los cuadros verdes y rojos representan el nivel 10 y 2 del factor probabilidad de mutación, respectivamente. Se observa que las soluciones obtenidas con una probabilidad del 2% tienden a estar más concentradas en valores menores para cada función objetivo mientras que las soluciones con probabilidad del 10% se encuentran más dispersas y se ubican en valores más grandes para cada función objetivo.

8.5. Diseño experimental para la proporción de mutación

Para definir la proporción de mutación se tuvieron en cuenta las siguientes hipótesis a priori:

1. Una proporción de mutación inferior al 100% puede generar mejores resultados para el algoritmo propuesto.
2. La relación entre la proporción de mutación y el tiempo computacional es directa: mayor proporción de mutación implica mayor tiempo computacional.

A continuación se presentan los resultados experimentales para el tamaño de población:

Tabla 20.

Intervalos de confianza para el factor proporción de mutación y la variable de estudio soluciones no dominadas.

proporción de mutación	N	Media	Desv.Est.	IC de 95%
0,5	80	3,025	1,273	(2,712; 3,338)
1,0	80	3,100	1,548	(2,787; 3,413)

Figura 41.

Gráfica de intervalos de los niveles de la proporción de mutación para la variable respuesta soluciones no dominadas.

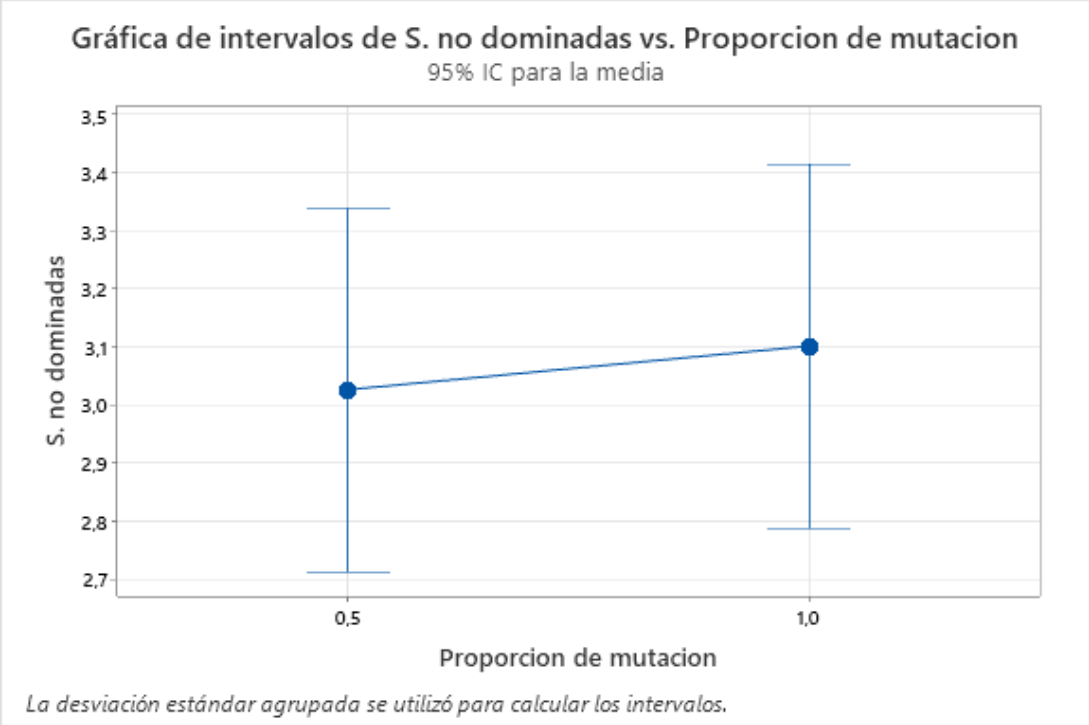


Tabla 21.

Intervalos de confianza para el factor proporción de mutación y la variable de estudio tiempo computacional.

proporción de mutación	N	Media	Desv.Est.	IC de 95%
0,5	80	85,28	47,26	(74,26; 96,30)
1,0	80	90,51	52,41	(79,49; 101,53)

Figura 42.

Gráfica de intervalos de los niveles de la proporción de mutación para la variable respuesta tiempo computacional.

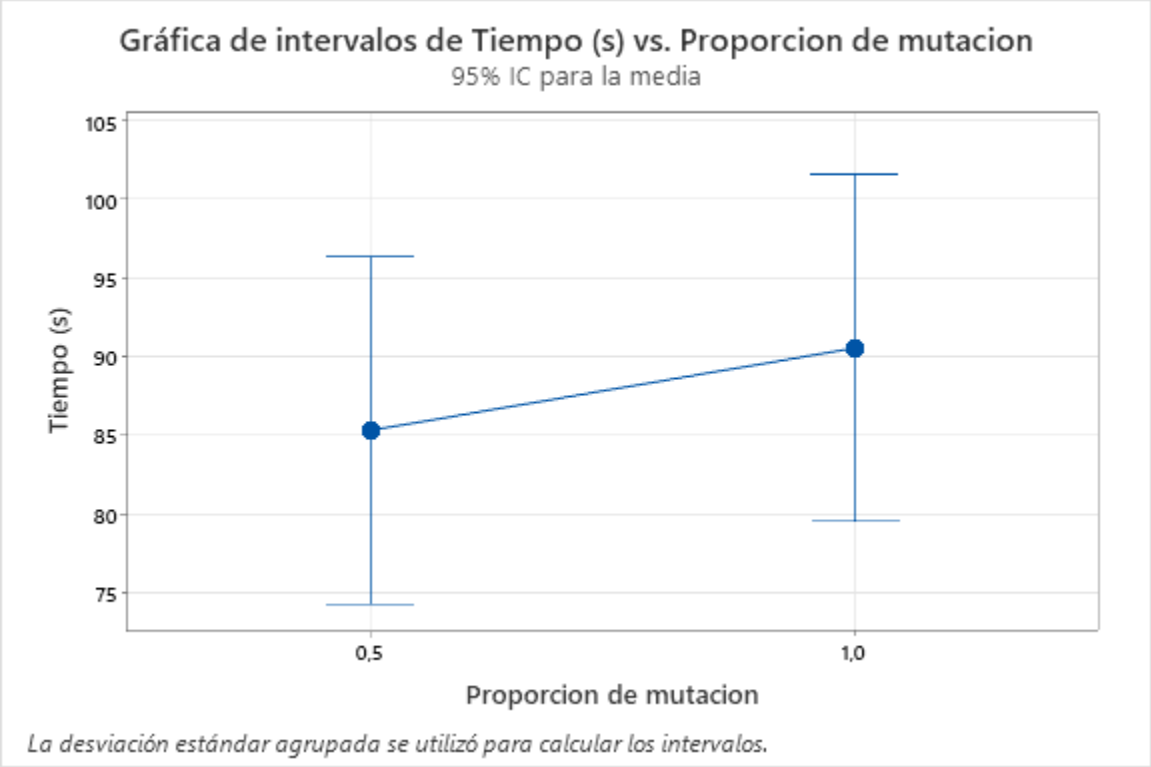


Tabla 22.

Intervalos de confianza para el factor proporción de mutación y la variable de estudio soluciones no dominadas por minuto.

Proporción de mutación	N	Media	Desv.Est.	IC de 95%
0,5	80	2,822	2,061	(2,400; 3,244)
1,0	80	2,638	1,745	(2,217; 3,060)

Figura 43.

Gráfica de intervalos de los niveles de la proporción de mutación para la variable respuesta soluciones no dominadas por minuto.

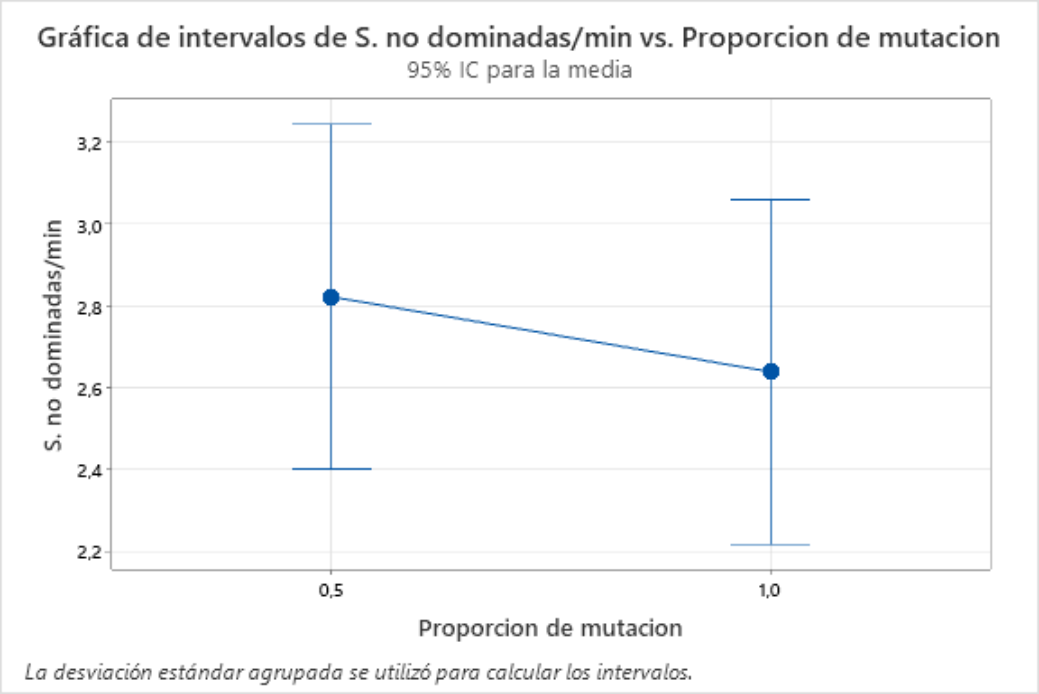
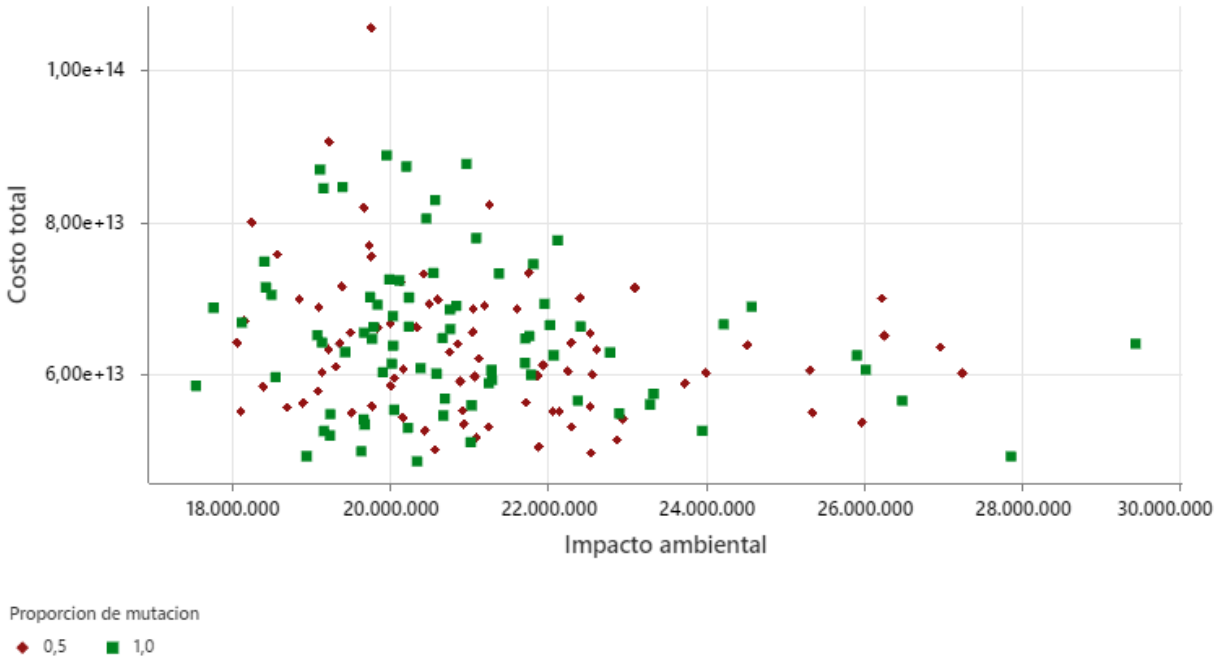


Figura 44.

Comparación de los conjuntos de mejores soluciones de todas las réplicas del experimento del factor proporción de mutación.



En la tabla 20, tabla 21 y tabla 22 se presenta el resultado de los intervalos de confianza para cada nivel del factor proporción de mutación para cada una de las variables respuestas contempladas en el análisis. Se busca conocer el mejor porcentaje de proporción de genes a ser mutados según el comportamiento de cada nivel para cada variable respuesta con la que se analiza. En la figura 41, figura 42 y figura 43 se puede visualizar el comportamiento antes mencionado sujeto al número de soluciones no dominadas, tiempo de cómputo y número de soluciones no dominadas por minuto, respectivamente.

En la tabla 20, se analiza el comportamiento de la cantidad de soluciones no dominadas para cada uno de los niveles del factor proporción de mutación. Se puede observar que la media de soluciones no dominadas conseguidas por el nivel 1 es levemente mayor a la media del nivel 0,5. Se puede concluir que se logra tener un número similar de soluciones con cualqueira de los dos niveles, por lo tanto no hay un impacto significativo dado un nivel de proporción específico. La dispersión de las soluciones en los dos niveles también es similar por lo cual ambos tendrán intervalos de confianza con límites que coinciden. Lo anterior se puede evidenciar gráficamente en la figura 41 en donde se aprecia una leve diferencia en cuanto al valor de soluciones no dominadas para el nivel 1.

En la tabla 21 se analiza la interacción de cada nivel de proporción con el tiempo requerido para ejecutar el algoritmo. Se evidencia que el nivel 1 emplea un mayor tiempo promedio de cómputo en comparación al nivel 0,5. Se concluye que el tiempo aumenta para una propoción mayor de mutación. Los datos del análisis se encuentran más densos y consistentes para el nivel 0,5 según el porcentaje de desviación estandar en comparación con el nivel 1. En la tabla 22 se aprecia que para el nivel 0,5 del factor se alcanza un número de soluciones no dominadas por minuto levemente mayor en comparación al nivel 1 en el mismo lapso, esta situación se puede apreciar en la figura 43.

.

En la figura 44, se muestra la dispersión de las soluciones obtenidas durante cada corrida del algoritmo. Los cuadros verdes y rojos representan el nivel 1 y 0,5 del factor proporción de mutación, respectivamente. La finalidad del diagrama de dispersión es encontrar alguna tendencia en las corridas según el costo total y el impacto ambiental. Se observa que las soluciones obtenidas para ambos niveles presentan un comportamiento similar y siguen el mismo patrón de dispersión. Los dos niveles tienden a estar concentrados en valores menores para cada función objetivo.

Dado lo anterior se confirma la primera y segunda hipótesis y se determina que la proporción de mutación seleccionada será el nivel 0,5 dado que alcanza un desempeño muy parecido al nivel 1 y lo hace en un menor tiempo computacional.

8.6. Contraste con instancia de la literatura

El problema fue resuelto en (Ebrahimi, 2018) por un método de solución diferente, considerando descuentos por cantidad, sin embargo para evaluar mejor el comportamiento del modelo, realiza un análisis de sensibilidad de los parámetros de peso en la tercera función objetivo para comparar los resultados obtenidos, Este parámetro ajusta el peso de la capacidad de respuesta en la logística directa e inversa, en el cual valida el modelo sin considerar descuentos por cantidad y considerando el problema de ruteo variando el parámetro de capacidad de respuesta de la cadena de suministro. Los datos obtenidos por el autor se mencionan en la tabla 23 y se constatarán con los resultados obtenidos en este trabajo.

Tabla 23.

Resultados obtenidos por (Ebrahimi, 2018)

Instancia	Z1	Z2	Z3
Λ=0,6	63.484.260.000.000	9411698	67.36
Λ=0,5	63.484.400.000.000	9411860	75.5
Λ=0,4	63.484.230.000.000	9411662	80.44

Los resultados obtenidos en este trabajo, considerando los parámetros del algoritmo definidos en el diseño de experimentos se presenta en la tabla 24. Dado que las primeras dos funciones objetivo son de minimización y la tercera es de maximización, en el diseño del algoritmo se consideró esta última negativa.

Tabla 24.

Resultados obtenidos

	Z1	Z2	Z3	Tiempo computacional (s)
$\Lambda=0,4$	64.436.678.639.099,70	28746181,31	-39,81	29,0793024
$\Lambda=0,5$	78.565.608.322.682,40	20407544,74	-39,84	29,879012
$\Lambda=0,6$	61.414.365.039.620,80	19575495,81	-39,87	78,9819882

De acuerdo con los resultados anteriores con un factor de respuesta de la cadena de 0,6 se obtiene un mejor resultado que el encontrado por el autor en los objetivos uno y tres.

9. Conclusiones

El interés generado por el diseño de cadenas de suministro de circuito cerrado viene de la problemática mundial de contaminación, generando así objetivos de desarrollo sostenible que buscan la producción y el consumo responsable buscando desvincular el crecimiento económico de la degradación medioambiental.

En la literatura hay pocas instancias de diseños de cadenas de suministro de circuito cerrado debido a las dificultades del proceso pues se deben tener en cuenta que los los productos devueltos se generan en cantidades, calidades y tiempos inciertos y segundo, a que las empresas no se enfocan en diseños fáciles de desensamblar o remanufacturar (Ortiz, 2004). De las instancias encontradas, la mayor concentración se encuentra en Irán seguido de China, por lo que este proyecto puede ser considerado como una referencia para futuras investigación y despertar el interes en colombia por el diseño de las cadenas de circuito cerrado para los residuos considerados peligrosos.

El problema tratado en el presente trabajo de investigación está relacionado con un problema que tienen todos los productores, en este caso, los productores de llantas, los cuales día a día deben aumentar su producción debido al crecimiento elevado del mercado y la baja cultura de reutilización. En Colombia la recolección de llantas se regula por la resolución 1326 de 2017 la cual busca establecer a cargo de los productores de llantas la obligación de formular, presentar, implementar y mantener actualizados los sistemas de Recolección Selectiva y Gestión Ambiental de Llantas Usadas, con el fin de prevenir y controlar la degradación del ambiente. (Ambiente, 2017)

Debido a que el problema solución del presente trabajo era de gran complejidad al momento de usar una metaheurística con valores aleatorios es muy probable que se violen restricciones del modelo, para lo cual se debió realizar un algoritmo de reparación el cual limita ciertos valores de las variables para que se sigan cumpliendo las condiciones del modelo.

De acuerdo con los resultados experimentales, se encontró que el algoritmo propuesto funciona mejor para tamaños de población pequeños para el problema planteado, siendo el mejor tamaño de población entre los evaluados 50. Se encontró que con un número de generaciones, una proporción y porcentaje de mutación menor (100, 50% y 2% respectivamente) se encuentran mayores soluciones no dominadas por minuto. También se encontró evidencia de que altos porcentajes de cruce pueden tener efectos benéficos para los algoritmos genéticos, dado que el algoritmo utilizado se desempeña mejor con un 90% respecto al 70%.

10. Recomendaciones

Con el ánimo de incentivar el estudio y aplicación de métodos heurísticos probabilísticos basados en población para resolver problemas de localización asignación y ruteo, se recomienda usar métodos de solución como optimización por colonia de hormigas o algoritmos meméticos para resolver el modelo matemático tratado a lo largo de este trabajo. Esto, para poder comparar el comportamiento del modelo según diferentes estrategias de solución y establecer relaciones en cuanto a la calidad de las soluciones y los métodos por los cuales se obtienen.

Puede considerarse también involucrar un objetivo que responda al impacto social que tendría implementar la red logística directa e inversa. El objetivo podría basarse en los tiempos de fabricación según los tipos de neumáticos abordados por el modelo y la demanda de los clientes. De esta forma calcular el número de personas requeridas para ejecutar las labores de producción. El objetivo pretende aumentar el número de personas contratadas en cada fábrica.

Se puede también evaluar la posibilidad de usar una codificación diferente a la propuesta para representar los individuos solución. Con el fin de lograr un cromosoma más eficiente en la medida de que se puedan usar un menor número de genes. Una representación más estratégica del cromosoma puede reducir los tiempos de cómputo por cada corrida del algoritmo.

Por último, se propone fomentar el interés de los estudiantes y profesores por abordar situaciones reales locales que puedan ser modeladas y permitan responder a necesidades nacionales para darle un sentido mayor a la ejecución de este tipo de proyectos de grado. Se recomienda a la Escuela de Estudios Industriales y Empresariales crear un banco de datos en el que se guarde el código diseñado para el algoritmo NSGA-II propuesto, que sirva como base para próximos trabajos que usen el mismo método de solución y que facilite la programación y el entendimiento del método en el software MATLAB.

Referencias bibliográficas

Ambiente, M. de M. (2017). Resolución No. 1326 de 2017. In *República de Colombia* (pp. 1–22).

Arango, M., Adarme, W., & Zapata, J. (2010). Gestión cadena de abastecimiento - logistica con indicadores bajo incertidumbre, caso aplicado sector panificador palmira. *Ciencia e Ingeniería Neogranadina*, *20*, 97–115.

Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: overview and conceptual comparison. *ACM Computing Surveys*, *35*(3), 189–213. https://doi.org/10.1007/s10479-005-3971-7

Caballero, J., & Aguilar, M. (2019). *Modelo de optimización difuso multiobjetivo multiperíodo para el diseño de una red logística inversa sostenible*.

Chekoubi, Z., Trabelsi, W., & Sauer, N. (2018). The Integrated Production-Inventory-Routing problem in the context of Reverse Logistics: the case of collecting and remanufacturing of EOL products. *The IEEE 4th International Conference on Optimization and Applications (ICOA)*. https://doi.org/10.1109/ICOA.2018.8370563

Chen, Y. T., Chan, F. T. S., & Chung, S. H. (2015). An integrated closed-loop supply chain model with location allocation problem and product recycling decisions. *International Journal of Production Research*, *53*(10), 3120–3140. https://doi.org/10.1080/00207543.2014.975849

Clausen, J. (1999). Branch and bound algorithms-principles and examples. *Department of Computer Science, University of …*, 1–30.

Corral Sastre, A. (2018). Desarrollo y Evaluación de Algoritmos Genóticos Multiobjetivo. Aplicación al problema del viajante. *Escuela Técnica Superior de Ingenieria Informética Universidad Politécnica de Valencia*, 2017–2018.

De, M., & Giri, B. C. (2020). Modelling a closed-loop supply chain with a heterogeneous fleet under carbon emission reduction policy. *Transportation Research*, *133*(May 2019), 101813.

https://doi.org/10.1016/j.tre.2019.11.007

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182–197. https://doi.org/10.1109/4235.996017

DeJong, K. A. (1975). *Analysis of the Behavior of a Class of Genetic Adaptive Systems*.

Ebrahimi, S. B. (2018). A stochastic multi-objective location-allocation-routing problem for tire supply chain considering sustainability aspects and quantity discounts. *Journal of Cleaner Production*, *198*, 704–720. https://doi.org/10.1016/j.jclepro.2018.07.059

Gong, G., Deng, Q., Gong, X., Zhang, L., Wang, H., & Xie, H. (2018). A Bee Evolutionary Algorithm for Multiobjective Vehicle Routing Problem with Simultaneous Pickup and Delivery. *Mathematical Problems in Engineering*, *2018*. https://doi.org/10.1155/2018/2571380

Gupta, D., & Ghafir, S. (2012). An Overview of methods maintaining Diversity in Genetic Algorithms. *International Journal of Emerging Technology and Advanced Engineering*, *2*(5), 56–60.

Hemmati, A., & Akbari, M. (2020). A robust optimization approach for the production-inventory-routing problem with simultaneous pickup and delivery. *Computers & Industrial Engineering*, *143*(March), 106388. https://doi.org/10.1016/j.cie.2020.106388

Herrera, F. (2009). *Introducción a los Algoritmos Metaheurísticos*. Soft Computing and Intelligent Information Systems.

Hillier, F., & Lieberman, G. (2010). *Introduccion a la investigacion de operaciones* (McGraw-Hil).

Lewis, R. (2007). Metaheuristics for University Course Timetabling (Tesis doctoral). In *Studies in Computational Intelligence* (Vol. 49, Issue January). https://doi.org/10.1007/978-3-540-48584-1

Lopez, D., & Conde, E. (n.d.). *El método Húngaro de Asignación: Aplicaciones*. Universidad de Sevilla.

MARTÍNEZ RUIZ, S. (2019). *MODELO DE EVALUACIÓN DE LA SOSTENIBILIDAD DE SOLUCIONES CONSTRUCTIVAS DE URBANIZACIÓN MEDIANTE ALGORITMOS GENÉTICOS*.

Mehrbod, M., Tu, N., & Miao, L. (2015). A hybrid solution approach for a multi-objective closed-

loop logistics network under uncertainty. *Journal of Industrial Engineering International*, 237–252. https://doi.org/10.1007/s40092-014-0089-z

Mehrbod, M., Xue, Z., Miao, L., & Lin, W. (2015). A STRAIGHT PRIORITY-BASED GENETIC ALGORITHM FOR A LOGISTICS NETWORK. *RAIRO Operations Research*, *49*, 243–264. https://doi.org/10.1051/ro/2014032

Olawale, A. (2017). *LOCATION-ALLOCATION-ROUTING APPROACH TO SOLID WASTE COLLECTION AND DISPOSAL By ADELEKE*. Covenant University.

Ortiz, C. (2010). *Algoritmos heurísticos y metaheurísticos para el problema de localización de regeneradores. (Tesis de pregrado)*. Universidad Rey Juan Carlos.

Ortiz, J. (2004). *Cadena de suministro de ciclo cerrado: Estructura conceptual y modelación*. Instituto Tecnológico y de Estudios Superiores de Monterrey.

Oyola-Cervantes, J., & Amaya-Mier, R. (2019). Reverse logistics network design for large off-the-road scrap tires from mining sites with a single shredding resource scheduling application. *Waste Management*, *100*, 219–229. https://doi.org/10.1016/j.wasman.2019.09.023

Pedram, A., Yusoff, N. Bin, Udoncy, O. E., Mahat, A. B., Pedram, P., & Babalola, A. (2017). Integrated forward and reverse supply chain: A tire case study. *Waste Management*, *60*, 460–470. https://doi.org/10.1016/j.wasman.2016.06.029

Rasjido, J. A., Pandolfi, D. R., & Villagra, N. A. (2014). Algoritmos de reparación para el manejo de restricciones en la planificación del mantenimiento de locaciones petroleras. *Informes Científicos Técnicos - UNPA*, *3*(2), 41–62. https://doi.org/10.22305/ict-unpa.v3i2.34

Riojas, A. (2005). *Conceptos, algoritmo y aplicación al problema de las N –reinas*. UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS.

Sadeghi, A., & Mina, H. (2019). A mixed integer linear programming model for designing a green closed-loop supply chain network considering location-routing problem A mixed integer linear programming model for designing a green closed-loop supply chain network considering location-routi. *International Journal of Logistics Systems and Management*, *August 2019*. https://doi.org/10.1504/IJLSM.2020.10019491

Sanchis, A., Ledezma, A., Iglesias, J., García, B., & Alosnso, J. (n.d.). *Complejidad Computacional*.

Soleimani, H., Chaharlang, Y., & Ghaderi, H. (2018). Collection and distribution of returned-remanufactured products in a vehicle routing problem with pickup and delivery considering

sustainable and green criteria. *Journal of Cleaner Production*, *172*, 960–970. https://doi.org/10.1016/j.jclepro.2017.10.124

Soleimani, H., Seyyed-esfahani, M., & Kannan, G. (2014). Incorporating risk measures in closed-loop supply chain network design. *International Journal OfProduction Research*, *52*(6), 1843-|867. https://doi.org/10.1080/00207543.2013.849823

Soysal, M. (2016). Closed-loop Inventory Routing Problem for returnable transport items. *Transportation Research Part D: Transport and Environment*, *48*(December 2015), 31–45. https://doi.org/10.1016/j.trd.2016.07.001

Taha, H. (2012). *Investigación de operaciones* (Pearson).

Unidas, N. (2030). *La Agenda 2030 y los Objetivos de Desarrollo Sostenible: una oportunidad para América Latina y el Caribe*.

Wang, J., Zhou, Y., Wang, Y., Zhang, J., Chen, C. L. P., & Zheng, Z. (2016). Multiobjective Vehicle Routing Problems With Simultaneous Delivery and Pickup and Time Windows : Formulation , Instances , and Algorithms. *IEEE Transactions on Cybernetics*, *46*(3), 582–594. https://doi.org/10.1109/TCYB.2015.2409837

Ya peng, Z. (2009). Optimization Design for Multi-echelon Logistics Network in Remanufacturing System. *Second International Conference on Intelligent Computation Technology and Automation*, *1*, 221–224. https://doi.org/10.1109/ICICTA.2009.520

Ya Peng, Z., & Zhong, D. Y. (2007). *Optimization Model for Closed-loop Logistics Network Design in Manufacturing and Remanufacturing System*. *3*, 7–10.

Ya Peng, Z., & Zhong, D. Y. (2008). Optimization Model for Integrated Logistics Network Design in Green. *International Conference on Information Management, Innovation Management and Industrial Engineering Optimization*, *5*, 138–141. https://doi.org/10.1109/ICIII.2008.167

Zhang, M., Pratap, S., Zhao, Z., Prajapati, D., & George, Q. (2020). Forward and reverse logistics vehicle routing problems with time horizons in B2C e-commerce logistics. *International Journal of Production Research*, *0*(0), 1–20. https://doi.org/10.1080/00207543.2020.1812749