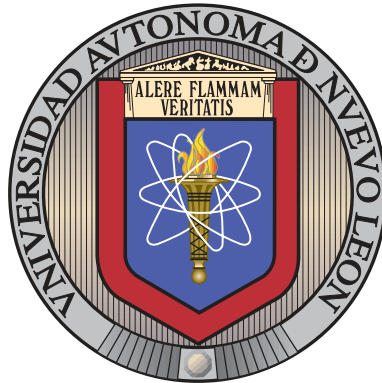


UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



MODELOS Y ALGORITMOS DE OPTIMIZACIÓN  
PARA EL PROCESO DE PREPARACIÓN DE  
PEDIDOS EN UN ALMACÉN

POR

MLYCS JOHANNA BOLAÑOS ZUÑIGA

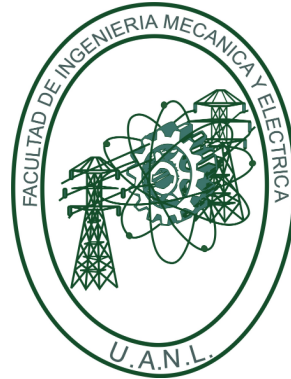
COMO REQUISITO PARA OBTENER EL GRADO DE  
DOCTORADO EN INGENIERÍA DE SISTEMAS

AGOSTO 2023

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



MODELOS Y ALGORITMOS DE OPTIMIZACIÓN  
PARA EL PROCESO DE PREPARACIÓN DE  
PEDIDOS EN UN ALMACÉN

POR

MLYCS JOHANNA BOLAÑOS ZUÑIGA

COMO REQUISITO PARA OBTENER EL GRADO DE  
DOCTORADO EN INGENIERÍA DE SISTEMAS

AGOSTO 2023

**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN**  
**Facultad de Ingeniería Mecánica y Eléctrica**  
**Posgrado**

Los miembros del Comité de Evaluación de Tesis recomendamos que la Tesis “Modelos y algoritmos de optimización para el proceso de preparación de pedidos en un almacén”, realizada por la estudiante Johanna Bolaños Zuñiga, con número de matrícula 1883900, sea aceptada para su defensa como requisito parcial para obtener el grado de Doctorado en Ingeniería de Sistemas.

**El Comité de Evaluación de Tesis**

Dra. María Angélica Salazar Aguilar  
Director

Dra. Jania Astrid Saucedo Martínez  
Co-director

Dra. Iris Abril Martínez Salazar  
Revisor

Dr. Vincent André Lionel Boyer  
Revisor

Dr. Igor Litvinchev  
Revisor

Dra. Edith Lucero Ozuna Espinosa  
Revisor

Dra. Diana Lucía Huerta Muñoz  
Revisor

Vo.Bo.

  
\_\_\_\_\_  
Dr. Simón Martínez Martínez  
Subdirector de Estudios de Posgrado



Institución 190001

Programa 557620

Acta Núm. 327

Ciudad Universitaria, a agosto 18 de 2023.

# ÍNDICE GENERAL

---

<b>Agradecimientos</b>	<b>VII</b>
<b>Resumen</b>	<b>IX</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Descripción del problema . . . . .	4
1.2. Hipótesis . . . . .	8
1.3. Objetivo general . . . . .	8
1.3.1. Objetivos específicos para el problema de asignación de ubica- ción de almacenamiento y enrutamiento del recolector . . . . .	8
1.3.2. Objetivos específicos para el problema de procesamiento de pedidos por lotes y enrutamiento del recolector . . . . .	9
1.4. Justificación . . . . .	9
1.5. Estructura de la tesis . . . . .	11
<b>2. Antecedentes</b>	<b>12</b>
<b>3. Asignación de ubicación de almacenamiento y enrutamiento del re- colector</b>	<b>18</b>
3.1. Revisión de literatura . . . . .	18
3.2. Reformulaciones matemáticas propuestas . . . . .	26
3.3. Metaheurística propuesta . . . . .	28
3.3.1. Fase constructiva . . . . .	29
3.3.2. Fase de mejora . . . . .	31

---

3.3.3. Grasp reactivo - RGRASP . . . . .	32
3.4. Experimentación computacional y Resultados . . . . .	33
3.4.1. Datos . . . . .	33
3.4.2. Recursos . . . . .	34
3.4.3. Calibración de parámetros de entrada para el RGRASP . . . . .	34
3.4.4. Criterios de paro para los modelos reformulados . . . . .	36
3.4.5. Análisis de los resultados obtenidos con las reformulaciones propuestas . . . . .	36
3.4.6. Desempeño RGRASP . . . . .	39
3.4.7. Análisis MIPStart . . . . .	41
3.4.8. Comparación de la situación real y la solución del RGRASP . . . . .	44
<b>4. Procesamiento de pedidos por lotes y enrutamiento del recolector</b>	<b>47</b>
4.1. Revisión de literatura . . . . .	47
4.2. Formulaciones matemáticas propuestas . . . . .	51
4.2.1. Procesamiento de pedidos por lotes (PPL) – Primera etapa . . . . .	52
4.2.2. Enrutamiento del recolector (PER) - Segunda etapa . . . . .	55
4.2.3. Generador de instancias . . . . .	57
4.3. Experimentación computacional y Resultados . . . . .	61
4.3.1. Recursos . . . . .	61
4.3.2. Procesamiento de pedidos por lotes (PPL) - Primera etapa . . . . .	62
4.3.3. Enrutamiento del recolector (PER) - Segunda etapa . . . . .	71
<b>5. Conclusiones</b>	<b>79</b>
5.1. Contribuciones . . . . .	82
5.2. Trabajo a futuro . . . . .	83

# ÍNDICE DE FIGURAS

---

1.1. Diseño del almacén del caso de estudio basado en Bolaños-Zuñiga <i>et al.</i> (2020) . . . . .	5
3.1. Gap de optimalidad reportado por CPLEX con el MIPStart. . . . .	43
3.2. Ubicación de los productos en el almacén. . . . .	45
4.1. Descripción gráfica de las instancias generadas para el modelo PPL. . . . .	63
4.2. Promedio tiempo total de viaje. . . . .	77

# ÍNDICE DE TABLAS

---

3.1. Notación utilizada en el modelo SLAUPR . . . . .	21
3.2. Resumen de los trabajos relacionados con la combinación del problema de asignación de ubicación de almacenamiento y el enrutamiento del selector. . . . .	25
3.3. Descripción de las instancias. . . . .	34
3.4. Resumen de resultados reportados en la literatura (SLAUPR) y los obtenidos con las reformulaciones propuestas. . . . .	37
3.5. Gap entre la mejor solución factible conocida y el RGRASP propuesto.	41
3.6. Resumen resultados modelos con MIPStart. . . . .	42
4.1. Descripción de las instancias para el modelo PPL. . . . .	63
4.2. Resumen de resultados reportados por el modelo PPL con $Q = 231$ . . .	65
4.3. Resumen de resultados reportadas por el modelo PPL con $Q = 252$ y $Q = 273$ . . . . .	68
4.4. Resumen de resultados reportados por el modelo PPL para la variante propuesta con $Q = 231$ y $Q = 252$ . . . . .	69
4.5. Descripción de las instancias para el modelo PER con $Q = 231$ . . . .	72
4.6. Resumen resultados modelo PER con $Q = 231$ . . . . .	74

# AGRADECIMIENTOS

---

Primeramente, gracias a **Dios** por otorgarme la vida, ser mi fuente de fortaleza en los momentos desafiantes y permitirme culminar este significativo logro.

Al **Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT)**, por la beca otorgada, la cual hizo posible la exitosa culminación de este doctorado.

A la **Universidad Autónoma de Nuevo León y a la Facultad de Ingeniería Mecánica y Eléctrica**, por brindarme la valiosa oportunidad de formar parte de su ilustre institución educativa y ser acogida en su comunidad. De igual manera, al **Posgrado en Ingeniería de Sistemas (PISIS)** por aceptarme y darme la oportunidad de realizar mis estudios de doctorado.

A mi directora de tesis **Dra. María Angélica Salazar Aguilar** y co-directora **Dra. Jania Astrid Saucedo Martínez**, quienes con sus consejos, paciencia y disposición hicieron posible que culminara este logro tan importante en mi vida.

A mis **revisores de tesis**, por sus valiosos aportes y sugerencias para el desarrollo del presente trabajo.

A mis amados padres **José John Jaiver Bolaños Martínez y Ruby Zuñiga Macca** por ser mi fuente de inspiración y apoyo incondicional.

A mi amado esposo **Diego Alexander González Casamachin** por brindarme su amor y compañía a lo largo de este viaje que emprendimos de la mano de Dios.

A toda mi **hermosa familia**, hermanos, tíos, tías, primos, primas, sobrinos y sobrinas, por transmitirme todo su apoyo y cuidar de mis padres.



Finalmente, a todos **mis amigos** por hacerme parte de sus vidas y por su apoyo constante.

¡Muchas gracias!

# RESUMEN

---

MLyCS Johanna Bolaños Zuñiga.

Candidato para obtener el grado de Doctorado en Ingeniería de Sistemas.

Universidad Autónoma de Nuevo León.

Facultad de Ingeniería Mecánica y Eléctrica.

Título del estudio: MODELOS Y ALGORITMOS DE OPTIMIZACIÓN PARA EL PROCESO DE PREPARACIÓN DE PEDIDOS EN UN ALMACÉN.

Número de páginas: 92.

**OBJETIVOS Y MÉTODO DE ESTUDIO:** En la actualidad, las empresas se ven en la necesidad de gestionar de forma efectiva su cadena de suministro para poder satisfacer las demandas del mercado y ofrecer un servicio de calidad a sus clientes. En este contexto, los almacenes pueden ofrecer una ventaja competitiva al permitir mayor agilidad y flexibilidad. Sin embargo, el costo del almacenamiento puede llegar a representar hasta el 20% de los costos logísticos de una empresa, lo que hace que la gestión de almacenes sea una de las actividades más importantes para el éxito empresarial. En este sentido, cualquier iniciativa que permita reducir los costos logísticos contribuirá a mejorar el rendimiento de la cadena de suministro.

Dentro de los almacenes, el proceso de preparación de pedidos es el más costoso. Por esta razón, una de las principales maneras para mejorar este proceso es la reducción del tiempo de viaje del recolector a través del almacén, el cual se puede llevar a cabo optimizando operaciones tales como el procesamiento de pedidos por lotes, la asignación de almacenamiento y el enrutamiento del recolector.

El presente trabajo se centra en estudiar una problemática identificada en el almacén de exportación de una empresa panificadora ubicada en la ciudad de Monterrey, México, la cual presenta problemas con la entrega de los pedidos a tiempo y el enrutamiento del recolector a través del almacén. Aunque en la mayoría de las investigaciones previas se han establecido modelos y algoritmos basados en el cumplimiento de la demanda, pocos han considerado otros factores como, las fechas de

compromiso de entrega de los pedidos, el inventario disponible y el peso del producto, los cuales son criterios importantes para cumplir en tiempo y forma con lo solicitado. Además de ser aspectos claves en el problema de estudio.

De acuerdo con la problemática presentada, se estudiaron dos variantes. La primera, es la combinación de los problemas de la asignación de ubicación de almacenamiento y el enrutamiento de recolector. La segunda, la combinación del problema de procesamiento de pedidos por lotes y el enrutamiento de recolector.

Los métodos de solución propuestos para la primera variante consistieron en una propuesta alternativa para mejorar los resultados de una investigación previa, en la que se propuso un modelo matemático de programación lineal entera mixta (MILP, por sus siglas en inglés de, *Mixed Integer Linear Programming*) que considera el peso del producto en la recolección de pedidos. Sin embargo, debido a la complejidad no lograron resolver a optimalidad varias de las instancias y para otras incluso, no encontraron solución factible.

Por otro lado, para resolver el problema de procesamiento de pedidos por lotes y el enrutamiento de recolector, se propusieron modelos de optimización que contemplan las fechas de cumplimiento de entrega de los pedidos, el inventario disponible, el cual está compuesto por el inventario inicial y en tránsito. Con el fin de garantizar una buena presentación de los pedidos de los clientes, se contempla el peso de los productos en la recolección. Se utiliza el concepto de procesamiento de pedidos por lotes para generar cada día dentro de un horizonte de planeación, el agrupamiento de las tarimas (lotes) que deberán ser recolectadas en aras de cumplir con la fecha de compromiso de entrega asociada al respectivo pedido al que están vinculadas. Es importante mencionar que se conocen los productos contenidos en cada tarima.

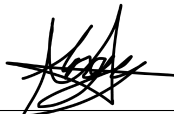
En este contexto, se emplea el enfoque del procesamiento de pedidos por lotes para crear diariamente un horizonte de planificación. Esto implica la agrupación de tarimas en lotes, en los cuales se encuentran identificados los productos que los componen. Estos lotes deben ser recolectados en aras de cumplir con la fecha de compromiso de entrega asociada al respectivo pedido al que están vinculados

**CONTRIBUCIONES Y CONCLUSIONES:** En este trabajo, se presentan dos formulaciones matemáticas proponiendo desigualdades válidas y eliminación de subciclos basadas en flujo para mejorar el desempeño computacional del modelo estudiado en la literatura para resolver simultáneamente las decisiones de asignación de espacios de almacenamiento y el enrutamiento del recolector, considerando el peso de los productos y las características propias de un caso de estudio. Los resultados de la experimentación computacional muestran una mejora significativa en la cantidad de soluciones factibles y óptimos alcanzados, principalmente, en los tamaños pequeñas y medianas y en los escenarios menos complejos. Como alternativa, se presenta una

metaheurística tipo GRASP reactivo (RGRASP) para generar soluciones de buena calidad para instancias de tamaño y escenario más complejos. Los resultados de la experimentación muestran una reducción en promedio de al menos el 29% en el tiempo de viaje reportado en la literatura. En consecuencia, al utilizar la solución obtenida por el RGRASP, como solución inicial, y resolver el problema con CPLEX, se logró obtener información de la brecha de optimalidad de las soluciones reportadas por la metaheurística propuesta.

Para la variante del procesamiento de pedidos por lotes y el enrutamiento de recolector se propusieron dos modelos matemáticos para resolver de forma independiente cada problema. Los resultados computacionales muestran un excelente rendimiento de estos modelos, permitiendo resolver instancias de tamaño real. Con base en la revisión de literatura se pudo identificar que es la primera vez que se propone un modelo matemático que brindan soluciones de calidad con base en los tiempos de ejecución para resolver el problema del procesamiento de pedidos por lotes considerando un almacén con forma general, fechas compromiso de entrega de pedidos, nivel de inventario disponible, capacidad máxima de tarimas a recolectar diariamente y demanda estable por periodos.

Firma de la directora: \_\_\_\_\_



Dra. María Angélica Salazar Aguilar

Firma de la co-directora: \_\_\_\_\_



Dra. Jania Astrid Saucedo Martínez

## CAPÍTULO 1

# INTRODUCCIÓN

---

Las empresas se ven obligadas a tener una correcta gestión de la cadena de suministro para poder cumplir con las exigencias de los consumidores y brindar un servicio de calidad, por lo cual, todas las actividades que están involucradas deberán estar alineadas a este fin manteniendo un equilibrio entre costos y eficiencia. De acuerdo con lo anterior, con base en la investigación de Bartholdi y Hackman (2019) y Davarzani y Norrman (2015), los almacenes son componentes críticos de las cadenas de suministro con la responsabilidad de recibir, almacenar, recoger, empacar y enviar productos entre proveedores y clientes. Además, contribuye con aproximadamente el 20 % de los costos logísticos de las empresas (Azadnia *et al.*, 2013) constituyendo, por ende, que su gestión es una de las actividades más importantes para el funcionamiento de una organización y, cualquier mejoramiento que permita disminuir los costos en los que incurra, contribuirán en la eficiencia y eficacia de la cadena de suministro.

De acuerdo con Goetschalckx y Ratliff (1988), desde 1984 se identificó el proceso de la preparación de pedidos (conocido como *Order Picking Process*) como una de las principales áreas de oportunidad en la gestión de almacenes ya que representa alrededor del 60 % de sus costos operativos (Theys *et al.*, 2010; Bartholdi y Hackman, 2019; Casella *et al.*, 2023). Por lo tanto, cualquier mejora que se realice en esta actividad reducirá los costos operativos de los almacenes. Este proceso implica una serie de actividades que van desde programación de la recolección de los pedidos (*Order Batching Problem*), la asignación de espacios y áreas de almacenamiento

para los productos (*Storage Location Assignment*), la recolección de los perdidos (*Picker Routing Problem*) hasta el despacho de estos, en respuesta a las solicitudes de sus clientes.

Por otro lado, Goetschalckx y Ashayeri (1989) mencionan que el nivel de servicio de una empresa contempla diversos factores tales como la integridad y la precisión del producto, el promedio y la variación del tiempo de entrega de lo demandado. Por lo tanto, un vínculo crucial entre la preparación de pedidos y el nivel de servicio es que cuanto más rápido se realice la recolección de lo solicitado, más rápido estará disponible para enviarla al cliente. De lo contrario, es posible que se incurra en un atraso de la entrega provocando una insatisfacción en la prestación del servicio ofrecida al cliente. No obstante, la empresa podría incurrir en trabajos adicionales para entregar a tiempo, elevando con ellos los costos de esta operación.

Según van Gils *et al.* (2018b), una de las principales formas para mejorar el proceso de la preparación de pedidos es mediante la optimización de aquellas operaciones que permitan minimizar el tiempo de viaje del recolector a través del almacén (conocido como *picking time* o *travel time*). Entre estas operaciones destacan el procesamiento de pedidos por lotes, la asignación de almacenamiento y el enrutamiento del recolector. La correcta ubicación de los productos en el almacén y un agrupamiento adecuado de los pedidos pueden facilitar y agilizar la recolección de los productos solicitados. Además, la flexibilidad para manejar cambios tardíos en los pedidos con un tiempo de recolección corto también es posible. Por lo tanto, al considerar estos problemas en conjunto, los gerentes pueden tomar decisiones estratégicas que resulten en operaciones más eficientes y rentables y, en consecuencia, un mejor servicio al cliente.

Aunque el tema del mejoramiento de la planeación de las actividades en la preparación de pedidos ha tenido un aumento constante de interés en los últimos 30 años (Winkelhaus *et al.*, 2021), con base en Dekker *et al.* (2004); Žulj *et al.* (2018); Chabot *et al.* (2016) y Zunic *et al.* (2017) es poco frecuente contemplar factores como el peso de los productos, el cual es un criterio muy importante cuando se realiza la

recolección de los productos de los pedidos ya que previene afectaciones al producto final y con ello se mantiene la satisfacción del cliente cuando recibe una buena presentación del producto. Otra de las razones es el incremento en los costos en la contratación de personal adicional para evitar entregas tardías por los desvíos en las rutas que pudiera tener el recolector para conservar el buen estado del producto final, ya que según lo menciona Pinto y Nagano (2022), las decisiones que se toman basadas únicamente en experiencias pueden resultar en una pérdida de tiempo y costo para los almacenes.

Otro aspecto que ha experimentado un incremento de interés es la manera más efectiva de abordar las fechas de compromiso de entrega de los pedidos a los clientes (Pinto y Nagano, 2022). Con base en Scholz *et al.* (2017), el incumplimiento de las fechas de entrega de los pedidos puede acarrear demoras en el envío posterior y/o los procesos de producción, generando como resultado una insatisfacción del cliente y costos elevados. Asimismo, los autores mencionan que esta situación puede atribuirse a inconsistencias en la agrupación de los pedidos de los clientes en lotes, la asignación y secuenciación de dichos lotes a los recolectores y la definición de la ruta que cada uno de estos debe seguir para recolectar los artículos de cada lote. Sin embargo, en la práctica, el incumplimiento de las fechas de entrega de los pedidos también puede ser atribuido a problemas relacionados con el nivel de inventario, ya que existen diversas variables externas que pueden obstaculizar la llegada del producto terminado a los almacenes (producto en tránsito).

Según lo expuesto previamente, el presente trabajo está enfocado a la optimización de las operaciones que conducen a la minimización del tiempo de viaje del recolector, tomando en cuenta aspectos como las fechas de cumplimiento de los pedidos, nivel de inventario, el peso de los productos, entre otros, dentro de un almacén donde se recolectan los productos destinados a ser enviados al exterior. En la Sección 1.1 se aborda con mayor detalle la problemática específica que enfrenta la empresa caso de estudio.

## 1.1 DESCRIPCIÓN DEL PROBLEMA

Este trabajo está motivado por una situación presentada en el almacén de productos enviados al extranjero (al cual llamaremos almacén de exportación) de una empresa del sector de alimentos ubicada en Monterrey, México, la cual cuenta con una cadena de suministro integrada por plantas de producción, un almacén general (CEDIS) y clientes nacionales e internacionales. La empresa es bien conocida por su excelente servicio de entrega y presentación de los pedidos de los clientes, y mantener este estándar es muy importante.

El diseño de almacén de exportación tiene una forma general, de acuerdo con los tipos de diseño señalados en la investigación de Masae *et al.* (2019). La demanda de sus productos es estable por periodos dependiendo, principalmente, de las estaciones del año. Por ende, la empresa no tiene problema en cambiar de ubicaciones los productos de manera periódica. En la Figura 1.1 se puede observar las 185 ubicaciones disponibles para almacenamiento dividido en cuatro áreas A, B, C y D, en las cuales la ubicación de los productos o SKUs (por sus siglas en inglés de, *Stock Keeping Unit*) se realiza con base en la frecuencia de solicitud de los mismos y se asigna una única ubicación a cada uno. La frecuencia de solicitud de un producto es la cantidad de veces que se solicita el producto durante un período determinado. También se puede observar que se cuenta con un depósito, definido como la ubicación inicial y final para los recorridos en la recolección.

Cada SKU representa una caja con el mismo tipo de producto (por ejemplo, galletas, pan). En las áreas A, B y C los SKUs son almacenados en tarimas las cuales contienen las cajas de un producto determinado y la capacidad de almacenamiento de tarimas es diferente en cada área. De igual manera, la cantidad de cajas por tarima difiere dependiendo del tipo de producto. El área D, el almacenamiento es por cajas. Aunque las ubicaciones de almacenamiento tienen diferentes capacidades físicas, estas son suficientes para almacenar la demanda de cualquier producto y se conocen en términos de cajas, dependiendo del tipo de producto asignado. Para este trabajo,



aunque en algunas ubicaciones el almacenamiento es por tarimas, la recolección se lleva a cabo en términos de cajas.

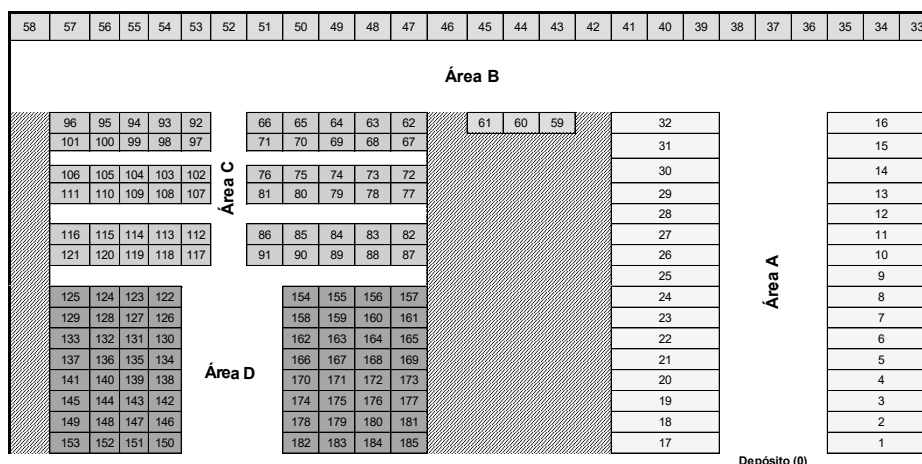


Figura 1.1: Diseño del almacén del caso de estudio basado en Bolaños-Zuñiga *et al.* (2020)

Para el manejo del almacén, la empresa cuenta con un WMS (por sus siglas en inglés de, *Warehouse Management System*). De igual manera, realiza la planeación de la recolección de sus pedidos en un horizonte de tiempo determinado (generalmente una semana).

Dentro del horizonte de planeación, las plantas de producción de la empresa envían con una frecuencia determinada, diferentes tipos de productos al CEDIS, haciendo que el inventario varíe de un día para otro. Así, al principio es posible conocer la cantidad de productos disponibles en el almacén (inventario inicial) y cuántas unidades llegarán (inventario en tránsito) para cada día en el horizonte de planeación. Cabe mencionar que una fábrica puede enviar diferentes cantidades de un mismo producto en más de un período dentro del horizonte y que el producto en tránsito se considera disponible el día posterior a su llegada al almacén.

Las solicitudes de pedidos de clientes internacionales (demanda estable) se reciben con anticipación. Cada pedido especifica la cantidad de productos requeridos y la fecha límite de entrega establecida por la empresa. El WMS contiene un módulo que permite la división de cada pedido de un cliente en diferentes tarimas para su

recolección, mostrando la cantidad de cajas de los productos a recolectar en cada tarima, asegurándose de no exceder la capacidad máxima de cajas que se pueden paletizar. Sin embargo, el envío solo se puede realizar una vez que se han recolectado todos los productos de las tarimas en las que se dividió el pedido del cliente, es decir, hasta que el pedido esté completo.

Es importante destacar que no se permite la recolección parcial de las cajas de los productos de una misma tarima, por lo que cada tarima debe ser recolectada por un único trabajador. La empresa puede entregar el pedido antes o durante la fecha de compromiso. No obstante, en ocasiones no es posible entregar los pedidos debido a que algunos de los productos demandados no están disponibles en *stock*. Por lo tanto, es crucial determinar para cada día del horizonte de planeación qué pedidos se deben recolectar (preparación de pedidos por lotes). Cabe mencionar también que los costos de inventario no son considerados en el presente trabajo.

Cabe señalar que las decisiones que se toman en el CEDIS no determinan cuáles son los planes de producción ni los pedidos solicitados ya que otros departamentos son los encargados de estos procesos. En general, para cumplir con la demanda total, el CEDIS trabaja con la información de los pedidos que deben entregarse durante el horizonte de planeación y la fecha de llegada del producto en tránsito. Sin embargo, dado que la empresa produce sus propios productos, la llegada del producto en tránsito depende de varios factores, tales como una programación inadecuada de la fabricación del producto, fallas en la línea de producción, daños del producto por la manipulación, entre otros. Por ende, el producto no está disponible en el momento en que se necesita.

Por otro lado, el WMS también genera la ruta o secuenciamiento que deben seguir los recolectores a través del almacén para recuperar los SKUs solicitados en cada tarima de un determinado pedido. No obstante, en la planeación de estas rutas, el sistema no considera el peso de las cajas de los SKUs (a pesar de que se conoce esta información), por lo cual los empleados recurren a rutas alternas para mantener una buena presentación del pedido, lo que ocasiona un incremento en el tiempo

de recolección. En consecuencia, para mitigar este incremento la empresa se ve en la necesidad de contratar personal extra. Adicionalmente, no existe una ubicación de almacenamiento apropiada de los productos en el almacén. Cabe resaltar que la recolección de los pedidos en el almacén es completamente manual y se recolecta por cajas.

Según Azadnia *et al.* (2013), minimizar el tiempo de viaje es un factor importante para reducir costos y retrasos dentro del proceso de preparación de pedidos. En la investigación de Bolaños-Zuñiga *et al.* (2020) propusieron un modelo matemático MILP, llamado SLAUPR (por sus siglas en inglés de, *Storage Location Assignment Unique and Picker Routing*) con el propósito de minimizar el tiempo de recolección en el almacén de exportación considerando de manera simultánea las decisiones de asignación de ubicación única de almacenamiento y el enrutamiento de recolector, contemplando el peso de los productos. Sin embargo, debido a la complejidad del problema, existen algunas instancias de tamaño real en las que no se encontró una solución factible. Por lo tanto, en esta investigación, se proponen reformulaciones matemáticas al SLAUPR y un algoritmo basado en GRASP reactivo para mejorar los resultados reportados en la literatura. Lo anterior, se realiza con el fin de proponer un algoritmo capaz de encontrar una solución aproximada con la cual se pueda determinar la ubicación única de los productos con base en las características del caso de estudio tales como la frecuencia de solicitud, demanda, peso, y a su vez que sirva de apoyo a la toma de decisiones en la práctica.

Adicionalmente, se propone un modelo matemático MILP para resolver el problema de la programación de pedidos durante un horizonte de planeación dado, considerando las fechas de cumplimiento de entrega y el inventario disponible, el cual está compuesto por el inventario inicial y en tránsito. Con el fin de garantizar una buena presentación de los pedidos de los clientes, para el modelo del enrutamiento del recolector se contempla el peso de los productos en la recolección.

## 1.2 HIPÓTESIS

Con el desarrollo de algoritmos exactos y metaheurísticos, la programación de los pedidos, la ubicación de productos y la recolección de los pedidos, con base en las fechas de entrega comprometidas con los clientes, nivel de inventario, demanda, frecuencia de solicitud y el peso de las cajas de los productos, se disminuirán las entregas tardías y mejorará el tiempo de recolección de los pedidos.

## 1.3 OBJETIVO GENERAL

El objetivo de este trabajo doctoral está enfocado en proponer y comparar diferentes formulaciones matemáticas, así como en desarrollar algoritmos eficientes con el fin de obtener soluciones de calidad en un tiempo favorable, considerando criterios tales como las fechas de entrega de los pedidos, nivel de inventario, demanda, frecuencia de solicitud y el peso de las cajas de los productos para minimizar las entregas tardías y el tiempo de viaje en la recolección de los pedidos.

Para el cumplimiento del objetivo general, en este trabajo se propuso solucionar la combinación de dos problemas. La primera combinación está relacionada con los problemas de asignación de ubicación de almacenamiento y el enrutamiento de recolector. La segunda está relacionada con el problema del procesamiento de pedidos por lotes y el enrutamiento de recolector. A continuación, se describen los objetivos específicos de cada variante estudiada.

### 1.3.1 OBJETIVOS ESPECÍFICOS PARA EL PROBLEMA DE ASIGNACIÓN DE UBICACIÓN DE ALMACENAMIENTO Y ENRUTAMIENTO DEL RECOLECTOR

- Analizar las características o métodos utilizados en trabajos relacionados.
- Proponer una reformulación del modelo matemático propuesto en la literatura

con el fin de disminuir la complejidad presentada.

- Diseñar y desarrollar un algoritmo basado en metaheurísticas para resolver el problema de forma eficiente.
- Determinar la eficiencia de los algoritmos propuestos mediante la ejecución de diversos experimentos computacionales con las instancias existentes en la literatura y analizar los resultados.

### 1.3.2 OBJETIVOS ESPECÍFICOS PARA EL PROBLEMA DE PROCESAMIENTO DE PEDIDOS POR LOTES Y ENRUTAMIENTO DEL RECOLECTOR

- Analizar las características o métodos utilizados en trabajos relacionados.
- Diseñar y proponer una formulación matemática para resolver el problema.
- Generar instancias para el problema.
- Determinar la eficiencia del modelo propuesto mediante la ejecución de diversos experimentos computacionales con las instancias propuestas y analizar los resultados.

## 1.4 JUSTIFICACIÓN

De acuerdo con de Koster *et al.* (2007), la preparación de pedidos es una actividad crítica en la cadena de suministro, ya que cualquier bajo desempeño en este proceso puede generar un servicio insatisfactorio al cliente y aumentar los costos operativos del almacén, así como de toda la cadena de suministro. Por tanto, resulta indispensable contar con un diseño y control robusto de este proceso para garantizar su eficiente operación.

A pesar de que la empresa cuenta con un software especializado que les permite obtener la dosificación de los pedidos en tarimas para su recolección y de generar

las rutas para llevarla a cabo, cuando no se considera el peso de los productos o se determina recolectar pedidos del cual no se tienen producto suficiente para cumplir con lo demandado, la ruta de recolección especificada por el sistema no es favorable, por lo tanto, implementar una herramienta que ayude en la ejecución eficiente de este proceso ayudará a los tomadores de decisiones a determinar las rutas óptimas que los operarios deben realizar a través del almacén y con ello disminuir el costo operacional. De acuerdo con de Koster *et al.* (2007), tener tiempos cortos en la recolección de pedidos implica una gran flexibilidad en el manejo de cambios tardíos en los pedidos, de modo que, minimizar el tiempo de recuperación de pedidos es una necesidad para cualquier sistema de preparación de pedidos.

En este trabajo, se propone el desarrollo de modelos matemáticos como una de las herramientas para representar la problemática de la empresa caso de estudio, ya que a través de formulaciones matemáticas y considerando una serie de criterios de un problema presentado, se analizan todas las posibles soluciones y se encuentra la mejor de ellas. Posteriormente, con base en los resultados, se puede llevar a cabo la toma de decisiones correspondiente. Estos modelos se ajustan perfectamente a esta finalidad ya que son una herramienta fundamental de la Investigación de Operaciones (IO).

De acuerdo con Hillier y G. (2010), la IO es una rama de las matemáticas que se enfoca en aplicar métodos científicos para abordar problemas relacionados con la gestión de recursos y la toma de decisiones. Su objetivo principal es encontrar soluciones óptimas que permitan alcanzar objetivos específicos, como maximizar ganancias o minimizar costos, considerando restricciones y limitaciones en la disponibilidad de recursos, como tiempo, personal e inventarios. La IO involucra la colaboración de grupos interdisciplinarios, como empresas, gobiernos, hospitales e industrias, para lograr soluciones que se adapten mejor a la organización en su conjunto y no solo a un área en particular.

Asimismo, los modelos matemáticos y el algoritmo metaheurístico propuesto podrán ser utilizados por cualquier empresa que cumpla con las características con las que

serán desarrollados.

## 1.5 ESTRUCTURA DE LA TESIS

El contenido restante del presente trabajo está estructurado de la siguiente forma: En el capítulo 2 se detallan los antecedentes del trabajo, los cuales abarcan los conceptos de cadena de suministro, logística, gestión de almacenes y preparación de pedidos.

El capítulo 3 está enfocado al estudio de la variante relacionada con la combinación de los problemas de asignación de ubicación de almacenamiento y el enrutamiento del recolector. Mientras que el capítulo 4, está enfocado al estudio de la combinación de los problemas de procesamiento de pedidos por lotes y el enrutamiento del recolector. En ambos capítulos se presenta la revisión de literatura, metodología, experimentación computacional y el análisis de los resultados de cada variante.

Finalmente, en el capítulo 5, se muestran las conclusiones generales, contribuciones y posible trabajo a futuro de la presente investigación.

## CAPÍTULO 2

# ANTECEDENTES

---

Los términos de cadena de suministro, la logística y el almacenamiento están estrechamente relacionados ya que el primero es un sistema integrado de proveedores, fabricantes, distribuidores y clientes que producen y entregan bienes y servicios de manera eficiente y rentable (Chopra y Meindl, 2008; Lambert, 2018; Waters, 2003; Simchi-Levi *et al.*, 2008). Mientras que la logística, se encarga de la planificación, implementación y control de la eficiencia en la gestión de la cadena de suministro, movimientos de materiales, la distribución y la entrega de bienes. Por otro lado, el almacenamiento, se encarga de mantener y proteger los bienes y materiales hasta que sean necesarios para su uso. Por ende, es un componente clave de la logística que ayuda a mantener el flujo de bienes y servicios en la cadena de suministro, por lo cual, algunas empresas mantienen instalaciones de almacenamiento en sus redes de distribución con el fin de mejorar su capacidad de respuesta y brindar un servicio eficiente a sus clientes.

En los centros de distribución, la recepción y almacenamiento de productos se lleva a cabo en grandes volúmenes, sin embargo, es común que los clientes soliciten pequeñas cantidades de diferentes productos en sus pedidos. Debido a esta demanda, surge la necesidad de llevar a cabo el proceso de preparación de pedidos. La capacidad de los almacenes para llevar a cabo este proceso con rapidez y precisión es un factor determinante en su éxito. Por lo tanto, la elección meticulosa de los sistemas de preparación de pedidos resulta crucial.



De acuerdo con la investigación realizada por Goetschalckx y Ashayeri (1989) y complementada por de Koster *et al.* (2007), los sistemas de preparación de pedidos están influenciados por una variedad de factores externos e internos. Entre los factores externos se encuentran los canales de comercialización, los patrones de demanda y reabastecimiento, los niveles de inventario y el estado de la economía actual. Por otro lado, los factores internos se dividen en dos niveles: estratégico y políticas.

Dentro del nivel estratégico o características del sistema, se consideran los tipos de mecanización, la disponibilidad de información (estática o dinámica) y la dimensionalidad del almacén. Por lo general, los problemas de decisión relacionados con estos factores suelen estar relacionados con la etapa de diseño. Por ende, de acuerdo con de Koster *et al.* (2007), tanto en la literatura como en la práctica se resaltan los siguientes tipos de mecanización o sistemas de recolección:

- Totalmente manuales y que involucran personas en la recolección: son aquellos donde el recolector se desplaza hasta la ubicación donde se encuentra el producto a recolectar, es conocido como *picker-to-parts*. Este sistema se divide en dos niveles: bajo y alto. En el nivel bajo, se debe caminar por los pasillos del almacén recolectando los productos solicitados. Mientras que, en nivel alto, el recolector se traslada sobre una grúa que se detiene automáticamente en la ubicación del producto que se desea recolectar.
- Semiautomatizados: son aquellos en los cuales la movilidad se realiza mediante tarimas o contenedores (unidad de carga) hacia los recolectores, conocidos como *parts-to-picker*. Esto se lleva a cabo a través de los sistemas automatizados de almacenamiento y recuperación (AS/RS por sus siglas en inglés de *Automated Storage and Retrieval System*). Una vez que el operario ha desmontado o surtido (modo de operación: simple, doble o múltiple) las cantidades requeridas, los contenedores son devueltos a su ubicación por medio de este sistema.
- Totalmente automatizados o robóticas: se utilizan en situaciones especiales, tales como en la manipulación de artículos pequeños, delicados y de alto valor.

Elbert *et al.* (2016) señalan que la diversidad de elementos, procesos y altos costos asociados con la adquisición de sistemas de mecanización automatizados constituyen un gran desafío para las pequeñas y medianas empresas (PyMEs) que buscan confiar e invertir plenamente en ellos. Por otro lado, en los sistemas manuales de bajo nivel, las mejoras en la eficiencia se pueden obtener mediante diseño y desarrollo de modelos matemáticos. Además, en la investigación realizada por de Koster *et al.* (2007), los autores mencionan que este sistema es el comúnmente utilizado en la industria.

Por otro lado, dentro del nivel de políticas, el cual está relacionado con las políticas de organización y funcionamiento, se presentan diferentes enfoques. Según el trabajo de Rouwenhorst *et al.* (2000), los autores plantean que las empresas toman decisiones a un nivel táctico u operativo. Por su parte, en el estudio realizado por van Gils *et al.* (2018a), identifican que se pueden distinguir cuatro problemas principales:

- Procesamiento de pedidos por lotes (táctico y operativo): consiste en agrupar los pedidos de los clientes en pedidos de recolección (lotes). Los lotes se pueden generar con la finalidad de o minimizar la distancia de viaje o reducir el tiempo total de viaje en la recolección de los productos, satisfaciendo la fecha límite de entrega de los pedidos (Gademann y Van de Velde, 2005; Henn, 2015).
- Asignación de ubicación de almacenamiento (táctico y operativo): consiste en determinar una ubicación específica para cada producto dentro del almacén, con el fin de asegurar un acceso rápido y eficiente a los productos, así como para optimizar los espacios de almacenamiento.
- Enrutamiento del recolector (operativo): determina la secuencia y la ruta en que los productos del pedido deben ser recolectados por el recolector, teniendo en cuenta que el producto ya cuenta con una ubicación de almacenamiento. Existen dos formas de llevarlo a cabo: individual, donde se recolecta un pedido a la vez; o por lotes, cuando se recolectan varios pedidos en un mismo recorrido.
- Zonificación (táctico y operativo): es cuando se divide el almacén en áreas más pequeñas. La asignación de los productos a estas áreas se puede realizar con

base en el tamaño, peso, temperatura del producto o también con base en el tipo de cliente y la frecuencia de solicitud de los productos.

De acuerdo con lo anterior, en Bartholdi y Hackman (2019) los autores señalan que el tiempo de recolección de pedidos es la principal área de oportunidad en el proceso de preparación de pedidos. En consecuencia, en la investigación llevada a cabo por van Gils *et al.* (2018b), los autores establecieron que la combinación de problemas más efectiva para mejorar este tiempo son el procesamiento de pedidos por lotes y el enrutamiento del recolector, seguido de la asignación de ubicación de almacenamiento y el enrutamiento del recolector. Aunque en de Koster *et al.* (2007) mencionan que las decisiones tomadas en los diferentes problemas son fuertemente interdependientes, gran parte de las investigaciones se centran en estudiar las decisiones de las anteriores combinaciones de forma independiente. No obstante, optimizar cada problema por separado puede conducir a una solución subóptima para todo el almacén. Dado que las nuevas tendencias en la industria logística requieren operaciones de preparación de pedidos aún más eficientes, algunos de estos problemas deben ser considerados simultáneamente para ser competitivos en el mercado (van Gils *et al.*, 2018b).

De igual manera, en la investigación de Bartholdi y Hackman (2019), los autores identifican cuatro actividades principales para determinar el tiempo total en la recolección de pedidos: configuración, búsqueda, recolección y viaje. Donde se identifica al viaje como la actividad que comprende la mayor parte del tiempo total (55%) empleado en la recolección de pedidos. Asimismo, de acuerdo con la investigación de Henn y Schmid (2013) el viaje representa la única parte variable ya que depende de la distancia o duración total del recorrido (enrutamiento) realizado por el recolector (Scholz *et al.*, 2016), mientras que el resto de las actividades se pueden considerar como constantes.

En la mayoría de los trabajos encontrados en la literatura, donde combinan el enrutamiento de recolector con otros problemas tales como el procesamiento de pedidos por lotes (Matusiak *et al.*, 2014; Scholz *et al.*, 2017; Attari *et al.*, 2021) o la asigna-

ción de ubicaciones de almacenamiento (Dekker *et al.*, 2004; Dijkstra y Roodbergen, 2017; Kordos *et al.*, 2020; Cai *et al.*, 2021), se enfocan en desarrollar los métodos de solución en un almacén con un diseño tradicional o de bloques. No obstante, existen otros diseños no tradicionales, como la espina de pescado utilizada en (Çelk y Süral, 2014; Wan y Liu, 2022; Zhou *et al.*, 2022), forma de U en Glock *et al.* (2019), un diseño inusual presentado en Zunic *et al.* (2017), uno irregular utilizado en Castier y Martínez-Toro (2023), un diseño para los almacenes de RMFS (por sus siglas en inglés de *Robotic Mobile Fulfillment System*) mostrado en Cai *et al.* (2021); Keung *et al.* (2022), o el almacén con diseño general como el presentado en el trabajo de Bolaños-Zuñiga *et al.* (2020). Los diferentes diseños de almacenes pueden ser consultados en la investigación de Masae *et al.* (2019).

Por otro lado, de acuerdo con Theys *et al.* (2010); Scholz *et al.* (2016) y van Gils *et al.* (2018b), el problema de enrutamiento del recolector de pedidos pertenece a la clase de problemas NP-duro ya que se puede interpretar como un caso especial del TSP (por sus en inglés de, *Travelling Salesman Problem*) o del VRP (por sus en inglés de, *Vehicle Routing Problem*) dependiendo de las características del problema. Por lo tanto, resolver de manera combinada cualquier proceso con el enrutamiento del recolector también podría generar problemas difíciles de resolver. Por lo anterior, se han propuesto una combinación de técnicas matemáticas y algoritmos para solucionar este tipo de problemas. Sin embargo, la elección del método adecuado depende de las necesidades específicas del problema y de los requisitos del tiempo y precisión de los resultados. Por ende, de acuerdo con de Koster *et al.* (2007) y van Gils *et al.* (2018b), ya que el diseño del almacén es una decisión a nivel táctico también es un importante factor que considerar para el desarrollo de los métodos a proponer.

De acuerdo con lo anteriormente mencionado, con el fin de proponer una solución a la problemática expuesta en la sección 1.1 y dado que la empresa cuenta con un nivel de mecanización manual de bajo nivel y lleva a cabo la recolección de pedidos de forma individual en un almacén con un diseño general, en esta investigación

---

nos enfocaremos en proponer herramientas para la integración de la asignación de ubicación de almacenamiento con el enrutamiento y el procesamiento de pedidos por lotes con el enrutamiento del recolector bajo estos criterios.

## CAPÍTULO 3

# ASIGNACIÓN DE UBICACIÓN DE ALMACENAMIENTO Y ENRUTAMIENTO DEL RECOLECTOR

---

Este capítulo está enfocado en el estudio de la primera combinación o variante del presente trabajo, la cual está relacionada con los problemas de asignación de ubicación de almacenamiento y el enrutamiento del recolector. En la Sección 3.1, se presenta la revisión de literatura y trabajos relacionados con la variante mencionada. En la Sección 3.2, se describen las reformulaciones matemáticas y, en la Sección 3.3, el algoritmo metaheurístico propuesto. La sección 3.4 muestra los resultados experimentales y la discusión sobre el rendimiento tanto de las reformulaciones como del algoritmo metaheurístico.

### 3.1 REVISIÓN DE LITERATURA

Como se mencionó en el capítulo 2, una de las mejores estrategias para mejorar el proceso de preparación de pedidos es reducir el tiempo total de viaje considerando resolver la combinación del problema de asignación de ubicación de almacenamiento (nivel táctico) con el problema de enrutamiento del recolector (nivel operativo). Aunque gran parte de las investigaciones se centra en estudiar ambas decisiones de

forma independiente, según van Gils *et al.* (2018b,a), optimizar cada problema por separado puede conducir a una solución subóptima para todo el almacén. Dado que las nuevas tendencias en la industria logística requieren operaciones de preparación de pedidos aún más eficientes, estos problemas deberían ser considerados de manera simultánea o integrada con el fin de obtener una solución competitiva en el mercado.

De acuerdo con Silva *et al.* (2020), aunque estudiar ambos problemas simultáneamente no es un problema nuevo, algunos investigadores se oponen a este método y consideran que el modelo integrado puede no ser útil debido a los diferentes horizontes planeación de ambos problemas, ya que el enrutamiento es un problema operativo con horizontes de tiempo cortos, mientras que la ubicación de almacenamiento tiene un período más largo. Sin embargo, en Salhi y Nagy (1999), resolver los problemas de manera simultánea presentó mejores resultados que resolverlos de manera independiente. De igual manera, Silva *et al.* (2020) mencionaron que integrar las decisiones de ubicación de almacenamiento y enrutamiento de los recolectores es una buena práctica que ha demostrado sus ventajas para mejorar la eficiencia de la recolección, principalmente cuando se conoce la demanda de cada producto y los productos contenidos en cada pedido, es decir, la frecuencia de solicitud del producto. De acuerdo con lo anteriormente mencionado, se han propuesto diferentes métodos de solución para resolver dicha combinación tanto de manera independiente como en conjunto.

En este sentido, en trabajos previos como el realizado por Dekker *et al.* (2004), se centran en proponer diferentes estrategias de asignación de almacenamiento y heurísticas de enrutamiento para dar solución a un caso de estudio. De igual manera, en Žulj *et al.* (2018) proponen diferentes estrategias de asignación de almacenamiento y un algoritmo de solución exacta (modificación del algoritmo de (Ratliff y Rosenthal, 1983)) basándose en el concepto de programación dinámica, para resolver este tipo de problemas en una empresa de productos para el hogar. Aunque en estos trabajos una vez los productos están asignados a los espacios de almacenamiento, posteriormente, estos son recolectados, ambas investigaciones consideran

restricciones de precedencia con base en el peso de los productos en ambos procesos. Restricciones que también son de importancia para la investigación realizada por Chabot *et al.* (2016) y Matusiak *et al.* (2014).

Por otro lado, algunos de los trabajos donde se considera resolver ambos procesos de manera simultánea se pueden encontrar en Daniels *et al.* (1998), Kordos *et al.* (2020), Silva *et al.* (2020) y Bolaños-Zuñiga *et al.* (2020). Aunque en el primer trabajo los autores proponen un modelo matemático y un algoritmo metaheurístico, estos no consideran el peso de los productos al momento de hacer la recolección y los productos están asignados a múltiples ubicaciones dentro del almacén. Mientras que en el segundo trabajo los autores no consideran el criterio del peso de los productos al momento de asignarlos a las ubicaciones y para la recolección contemplan el concepto de la ruta más corta en los algoritmos genéticos propuestos. Por otro lado, en el trabajo de Silva *et al.* (2020), aunque proponen diferentes modelos no lineales y sus respectivas linealizaciones, así como una metaheurística GNV (por sus siglas en inglés de, *General Variable Neighborhood Search*), los autores no consideran en las políticas de enrutamiento el peso de los productos. Finalmente, en el trabajo de Bolaños-Zuñiga *et al.* (2020), a pesar de que los autores proponen un modelo matemático MILP donde se consideran restricciones de precedencia con base al peso de los productos en la recolección y una ubicación única para cada producto en un almacén con un diseño general, no se logra encontrar las soluciones óptimas para todas las instancias analizadas, dejando la posibilidad de proponer algoritmos más eficientes para dar solución a este problema. Por ende, ya que este último trabajo aborda las características de la empresa caso de estudio, el modelo matemático llamado por los autores SLAUPR, se utiliza como base en la presente investigación. A continuación, se realiza una descripción de dicho modelo:

En la formulación del modelo matemático SLAUPR, se utilizó la notación presentada en el Tabla 3.1.



Tabla 3.1: Notación utilizada en el modelo SLAUPR

---

<b>Conjuntos</b>	
$I$	Ubicaciones incluido el depósito
$K$	Productos disponibles en el almacén
$P$	Pedidos solicitados
$X \subseteq K$	Productos a asignar*
<b>Parámetros</b>	
$d_{kp}$	Demanda del producto $k \in K$ en el pedido $p \in P$
$Q_{jk}$	Capacidad física de la ubicación $j \in I$ para almacenar cualquier producto $k \in K$
$w_k$	Peso del producto $k \in K$
$t_{ij}$	Matriz de tiempo de viaje entre ubicaciones ( $i, j \in I$ )
$M_{pjk}$	$\min \{d_{kp}, Q_{jk}\}$
$M$	$\max_{k \in K} \{w_k\}$
<b>VARIABLES DE DECISIÓN</b>	
$u_{pij} =$	$\begin{cases} 1, & \text{si en el pedido } p \in P \text{ se visita la ubicación } j \text{ después de la ubicación } i \text{ (} i, j \in I \text{)} \\ 0, & \text{caso contrario} \end{cases}$
$e_{jk} =$	$\begin{cases} 1, & \text{si la ubicación } j \in I \text{ almacena el producto } k \in K \\ 0, & \text{caso contrario} \end{cases}$
$v_{pjk}$	Cantidad de producto $k \in K$ , del pedido $p \in P$ , recolectado en la ubicación $j \in I$
$h_{jp}$	Variables auxiliares para los subciclos. Seguimiento de la posición en la que se visita la ubicación $j \in I$ al recolectar el pedido $p \in P$

---

\*Los productos a asignar  $k \in X$  son aquellos donde  $d_{kp} > 0 \quad \forall k \in K, p \in P$ .

La representación del modelo SLAUPR es el siguiente:

$$\min \sum_{i \in I} \sum_{j \in I} \sum_{p \in P} t_{ij} u_{pij} \quad (3.1)$$

$$s.a : \sum_{j \in I} v_{pjk} \geq d_{kp} \quad \forall k \in X, p \in P \quad (3.2)$$

$$\sum_{p \in P} v_{pjk} \leq Q_{jk} \quad \forall j \in I, k \in X \quad (3.3)$$

$$\sum_{j \in I \setminus \{0\}} u_{p0j} = 1 \quad \forall p \in P \quad (3.4)$$

$$\sum_{k \in X} e_{jk} \leq 1 \quad \forall j \in I \quad (3.5)$$

$$\sum_{j \in I \setminus \{0\}} e_{jk} = 1 \quad \forall k \in X \quad (3.6)$$

$$v_{pjk} \leq M_{pjk} \sum_{i \in I} u_{pij} \quad \forall k \in X, p \in P, j \in I \quad (3.7)$$

$$v_{pjk} \leq M_{pjk} e_{jk} \quad \forall k \in X, p \in P, j \in I \quad (3.8)$$

$$\sum_{j \in I} u_{pij} \leq 1 \quad \forall i \in I, p \in P, i \neq j \quad (3.9)$$

$$\sum_{j \in I} u_{pji} \leq 1 \quad \forall i \in I, p \in P, i \neq j \quad (3.10)$$

$$h_{ip} - h_{jp} + |I|u_{pij} \leq |I| - 1 \quad \forall i, j \in I \setminus \{0\}, p \in P, i \neq j \quad (3.11)$$

$$u_{pii} = 0 \quad \forall i \in I, p \in P \quad (3.12)$$

$$\sum_{i \in I} u_{pij} = \sum_{i \in I} u_{pji} \quad \forall j \in I, p \in P, i \neq j \quad (3.13)$$

$$\sum_{k \in X} e_{ik} w_k - \sum_{l \in X} e_{jl} w_l \geq M(u_{pij} - 1) \quad \forall i, j \in I \setminus \{0\}, p \in P \quad (3.14)$$

$$u_{pij} \in \{0, 1\} \quad \forall i, j \in I, p \in P \quad (3.15)$$

$$e_{jk} \in \{0, 1\} \quad \forall j \in I, k \in X \quad (3.16)$$

$$h_{jp} \geq 0 \quad \forall j \in I, p \in P \quad (3.17)$$

$$v_{pjk} \in Z_0^+ \quad \forall k \in X, p \in P, j \in I \quad (3.18)$$

La función objetivo (3.1) minimiza el tiempo de viaje en la recolección de los pedidos. Las restricciones (3.2) y (3.3) garantizan la satisfacción de la demanda del producto por pedido y la capacidad de las ubicaciones de almacenamiento, respectivamente. Las tipo (3.4) aseguran que cada pedido inicie en el punto de origen, ubicación

correspondiente al depósito. Las restricciones (3.5) y (3.6) aseguran que un producto sea asignado a un único espacio y que a cada espacio se le asigne al menos un producto. Las (3.7) y (3.8) son las restricciones inclusivas. Relacionan las variables de recolección con las variables de asignación y enrutamiento. Las restricciones (3.9) y (3.10) garantizan que cada ubicación se visite como máximo una vez en cada ruta. Las (3.11) y (3.12) corresponden a las restricciones de eliminación de subciclos con base en la formulación MTZ propuestas por Miller *et al.* (1960). Estos conjuntos de restricciones se consideran dado que algunos productos tienen el mismo peso. Las restricciones del tipo (3.13) garantizan la conservación del flujo por pedido. Las (3.14) son las restricciones de precedencia basadas en el peso de los productos, a mayor peso, mayor prioridad en la recolección. Finalmente, las restricciones (3.15)–(3.18) son la naturaleza de las variables.

De acuerdo con Shetty *et al.* (2020), el VRP basado en las restricciones de eliminación de subciclos de MTZ tiene una complejidad computacional de  $O(2^n)$  y es considerado NP-duro. Debido a esto, es un problema difícil de resolver, especialmente para un almacén grande ya que este puede llegar a generar cientos o miles de variables. Para detalles sobre el desafío computacional que plantea el VRP, se puede consultar la investigación de Toth y Vigo (2014). Con base en lo anterior, una de las limitantes para el SLAUPR son el tipo de restricciones de subciclos empleadas. Profundizando en este aspecto, se encontró la investigación de Palomo-Martínez *et al.* (2017), en el cual los autores realizan un análisis comparativo sobre las diferentes formulaciones para este tipo de restricciones, demostrando que la formulación con mejor desempeño son las propuestas por Gavish y Graves (1978), los cuales utilizan el concepto de flujo de red que involucra un solo tipo de producto básico denominado como (*commodity*).

Con base en lo anteriormente expuesto, aunque se han propuesto diferentes métodos de solución para resolver de manera conjunta o independiente la asignación de ubicación de almacenamiento y el problema de enrutamiento del recolector con diferentes variantes como se puede ver en las investigaciones de Cortés *et al.* (2017); Wang *et al.* (2017); Eydi *et al.* (2022) y en la Tabla 3.2, hasta donde sabemos, no hay ningún

trabajo en la literatura propuesto para mejorar los resultados mostrados en la investigación de Bolaños-Zuñiga *et al.* (2020). Por lo tanto, con el fin para mejorar el rendimiento computacional del SLAUPR, la principal contribución de esta investigación es el desarrollo de reformulaciones matemáticas considerando desigualdades válidas y restricciones de flujo para la eliminación de subciclos, así como el diseño de un Procedimiento de Búsqueda Adaptativa Aleatoria Codiciosa Reactiva (RGRASP por su siglas en inglés, *Reactive Greedy Randomized Adaptive Search Procedures*), como método de aproximación para resolver instancias de tamaño real en un tiempo computacional corto ya que con base en diversas investigaciones como las de Boudia *et al.* (2007) y Cantu-Funes *et al.* (2018), el RGRASP es una metaheurística relativamente simple, la cual consta de dos fases principales: construcción y mejora, que requiere pocos parámetros para ser ajustados y presenta un buen desempeño en muchos problemas de optimización combinatoria (Feo y Resende, 1995; Prais y Ribeiro, 2000).

Es importante mencionar que en el presente trabajo se determina resolver de manera integrada el problema de asignación de ubicaciones y el enrutamiento del recolector en respuesta a las necesidades específicas presentadas por la empresa caso de estudio. Además, basándonos en las investigaciones previas realizadas por van Gils *et al.* (2018b) y Silva *et al.* (2020), se ha evidenciado que abordar estos problemas de manera simultánea conlleva a una notable mejora en la eficiencia del proceso de recolección, en comparación con enfoques que los tratan de manera independiente.

En la Tabla 3.2 se muestra un resumen de los trabajos relevantes relacionados con la combinación del problema de asignación de ubicación de almacenamiento y el enrutamiento del recolector.

Tabla 3.2: Resumen de los trabajos relacionados con la combinación del problema de asignación de ubicación de almacenamiento y el enrutamiento del selector.

Trabajos	Sistema de recolección	Diseño del almacén	Solución simultánea de cada problema	Múltiples ubicaciones para un producto	Factores considerados	Propuestas
Daniels <i>et al.</i> (1998)	Bajo nivel	-	✓	✓	Demanda	Modelos matemáticos. Metaheurística (Tabu search)
Dekker <i>et al.</i> (2004)	Bajo nivel	Múltiples bloques	X	X	Demanda Tipo de producto	Algoritmos heurísticos
Dijkstra y Roodbergen (2017)	Bajo nivel	Un bloque	X	X	Demanda	Programación dinámica
Žulj <i>et al.</i> (2018)	Bajo nivel	Un bloque	X	X	Peso de los productos	Algoritmos exactos
Bolaños-Zuñiga <i>et al.</i> (2020)	Bajo nivel	General	✓	X	Demanda Peso de los productos	Modelos matemáticos
Kordos <i>et al.</i> (2020)	-	-	✓	X	Demanda	Algoritmo genético
Silva <i>et al.</i> (2020)	-	Un bloque	✓	X	Demanda	Modelos matemáticos. Metaheurística (GVNS)
Cai <i>et al.</i> (2021)	Robótico	RMFS	✓	X	Demanda	Modelos matemáticos
Keung <i>et al.</i> (2022)	Robótico	RMFS	X	X	Demanda	Modelos deterministas y estocásticos. Heurística ruta más corta
Xu y Ren (2022)	Bajo nivel	Un bloque	✓	✓	Correlación de la demanda. Congestión de recolectores	Modelos matemáticos. Algoritmos heurísticos
Zhou <i>et al.</i> (2022)	Bajo nivel	Espina de pescado	X	X	Demanda	Modelo estocástico
Lee (2022)	Bajo nivel	Uno y dos bloques	✓	X	Frecuencia y relación de los SKUs	Modelos analíticos
<b>Este trabajo</b>	<b>Bajo nivel</b>	<b>General</b>	<b>✓</b>	<b>X</b>	<b>Demanda Frecuencia y peso de los productos</b>	<b>Modelos matemáticos. Metaheurística (RGRAPS)</b>

## 3.2 REFORMULACIONES MATEMÁTICAS PROPUESTAS

De acuerdo con la revisión de literatura, se puede determinar que llevar a cabo de manera correcta la actividad de la preparación de pedidos, representa un ahorro significativo en los almacenes de distribución y a su vez en la cadena de suministro, permitiéndole tener a las empresas una capacidad de respuesta rápida a los clientes.

También se observó que este problema ha sido estudiado a lo largo de los años y se han empleado diferentes métodos tanto exactos como heurísticos para su solución, ya que debido a la competitividad que existe en los mercados se requiere que la toma de decisiones sea más rápida.

Como se mencionó en la descripción del problema, se cuenta con un modelo matemático (SLAUPR) producto de la investigación de Bolaños-Zuñiga *et al.* (2020). Sin embargo, para algunas de las instancias analizadas no fue posible encontrar soluciones óptimas e incluso la primera solución entera factible. Con el fin de mejorar el rendimiento computacional del SLAUPR, y analizando la formulación descrita en la Sección 3.1, se proponen dos reformulaciones para SLAUPR, en las que se derivaron desigualdades válidas y se reformularon las restricciones de eliminación del subciclos de la siguiente manera:

- Las restricciones (3.3) y (3.8) se fusionaron ya que es posible garantizar con una sola restricción se recolecte el producto solicitado sin exceder la capacidad de la ubicación de almacenamiento correspondiente.

$$\sum_{p \in P} v_{pjk} \leq Q_{jk} e_{jk} \quad \forall j \in I, k \in X. \quad (3.19)$$

- Se eliminaron las restricciones redundantes (3.10).
- Las restricciones (3.11) fueron reemplazadas por:

$$h_{ip} - h_{jp} + (M_p + 1) u_{pij} \leq M_p \quad \forall i, j \in I \setminus \{0\}, p \in P, i \neq j, \quad (3.20)$$

donde  $M_p = \sum_{k \in X} a_{kp} \quad \forall p \in P$ , tal que  $a_{kp}$  es 1 si el producto  $k \in K$  es requerido en el pedido  $p \in P$ ; 0 en otro caso. Nótese que  $M_p$  reemplaza a

$|I|$  sin perder la factibilidad de las restricciones (3.11), dado que la cantidad de ubicaciones a visitar por pedido, sin contemplar el depósito, es igual a la cantidad de productos diferentes que hay en este.

- En las restricciones (3.14), el parámetro  $M$  se estableció como:  $M = \max_{k \in \mathbf{X}} \{w_k\}$  para acotar el valor de este parámetro.

Considerando las modificaciones anteriores, la primera reformulación propuesta, a la cual se denomina **SLAUPR\_V2**, está compuesta por las expresiones (3.1), (3.2), (3.4) – (3.7), (3.9), (3.12)–(3.18), (3.19), y (3.20). Por lo tanto, SLAUPR\_V2 tiene dos conjuntos de restricciones menos que SLAUPR.

No obstante, con base los resultados empíricos del trabajo de Palomo-Martínez *et al.* (2017) el uso de la formulación de las restricciones para la eliminación de subciclos propuesta por Gavish y Graves (1978), presentaron un mejor desempeño que las otras formulaciones analizadas en dicho trabajo. Asimismo, en trabajos como el de Scholz *et al.* (2016), mencionan que las restricciones de eliminación de subciclos que presentan mejores desempeños son las que se realizan con base en el flujo.

De acuerdo con lo anteriormente mencionado y a la estructura del SLAUPR, se planteó reemplazar la restricción (3.20) y realizar una reformulación utilizando las restricciones propuestas por Gavish y Graves (1978). Para llevar a cabo la formulación de estas restricciones, se adiciona la variable  $g_{pij}$  para describir el flujo entre la ubicación  $i \in I$  y  $j \in I$  en cada pedido  $p \in P$ . Las restricciones para la eliminación de subciclos propuestas son las siguientes:

$$g_{pij} \leq M_p u_{pij} \quad \forall i, j \in I, p \in P, j \neq 0, \quad (3.21)$$

$$\sum_{j \in I} g_{pji} - \sum_{j \in I \setminus \{0\}} g_{pij} = \sum_{j \in I \setminus \{0\}} u_{pji} \quad \forall i \in I \setminus \{0\}, p \in P. \quad (3.22)$$

Las restricciones (3.21) garantizan el flujo correspondiente al número de productos a recuperar, y las restricciones (3.22) aseguran que en cada ubicación incluida en la ruta se deje una sola unidad del flujo.

Con base en lo anterior, la segunda reformulación propuesta, denominada **SLAUPR\_V3**,

está compuesta por las expresiones (3.1), (3.2), (3.4) – (3.7), (3.9), (3.12)–(3.18), (3.19), (3.21) y (3.22). Por lo tanto, el modelo propuesto SLAUPR\_V3 tiene un conjunto de restricciones menos que el SLAUPR, pero uno más que el SLAUPR\_V2.

### 3.3 METAHEURÍSTICA PROPUESTA

Es bien conocido en la literatura que el uso de heurísticas o metaheurísticas puede acelerar la convergencia de métodos exactos y proporcionar una solución temporal favorable a problemas complejos o de tamaño real. Por lo tanto, debido a la complejidad del SLAUPR y con el fin de mejorar los resultados reportados, también se diseñó una metaheurística RGRASP, la cual se basa en el GRASP (por sus siglas en inglés de *Greedy Randomized Adaptive Search Procedure*). Una de las principales ventajas es la facilidad de implementación y la construcción de soluciones de alta calidad que luego se mejoran, generalmente con un procedimiento de búsqueda local que apunta a la solución construida mediante la exploración iterativa de un vecindario definido. Además, el GRASP se ha empleado mucho debido a su eficiencia para resolver problemas complejos de rutas de vehículos, como se menciona en los trabajos de Chaovalitwongse *et al.* (2003); Villegas *et al.* (2011) y Haddadene *et al.* (2016).

El principal objetivo de esta etapa de la investigación se centró en el diseño de cada fase del GRASP para generar soluciones de alta calidad en un tiempo computacional corto. Además, para restringir la lista de candidatos (RCL, por sus siglas en inglés de *Restricted Candidate List*), en la fase constructiva, el valor del parámetro de calidad ( $\alpha$ ) se ajusta automáticamente (reactivo) durante el proceso de solución, dependiendo de la instancia disponible.

Un esquema general del GRASP se puede ver en el algoritmo 1, en el cual `iter_1` el número de iteraciones en las que se repetirá el ciclo de construcción-mejora; `instancia` es la instancia que se va a analizar;  $\alpha \in \{0, 1\}$  un parámetro de calidad;  $f(S)$  la función objetivo o tiempo total de recolección de los pedidos; e `iterBS`



y `tiempoBS`, las iteraciones consecutivas sin mejora y el tiempo límite de ejecución para la fase de mejora, respectivamente.

---

**Algoritmo 1** Procedimiento de búsqueda aleatorizada adaptativa y voraz propuesto

---

**Entradas:** `instancia`, `iter_1`, `iterBS`, `tiempoBS`,  $\alpha$

**Salida:**  $S^*$ : Mejor solución encontrada

```

1: Para  $i = 0$  hasta iter_1 hacer
2:    $S \leftarrow \text{Constructivo}(\text{instancia}, \alpha)$ 
3:    $S^* \leftarrow S$ 
4:    $\text{contadorBS} \leftarrow 0$ 
5:   Mientras  $\text{tiempoBusqueda} \leq \text{tiempoBS}$  &  $\text{contadorBS} \leq \text{iterBS}$  hacer
6:      $S' \leftarrow \text{Mejora}(S^*, \text{iterBS}, \text{tiempoBS})$ 
7:     Si  $f(S^*) > f(S')$  entonces
8:        $S^* \leftarrow S'$ 
9:        $\text{contadorBS} \leftarrow 0$ 
10:    De otro modo
11:       $\text{contadorBS} \leftarrow \text{contadorBS} + 1$ 
12:    Fin Si
13:  Fin Mientras
14:   $i \leftarrow i + 1$ 
15: Fin Para

```

---

A continuación, se proporciona una explicación más detallada de los componentes principales del RGRASP propuesto.

### 3.3.1 FASE CONSTRUCTIVA

El algoritmo propuesto para la fase de construcción consiste en una heurística de construcción aleatoria (HCA) compuesta de dos pasos: el primero está relacionado con la asignación de la ubicación de almacenamiento (ver Algoritmo 2). El segundo, está relacionado con la secuencia de recolección de pedidos y el tiempo total de recolección considerando las limitaciones del problema.

Inicialmente, para la primera etapa se genera una lista de candidatos (LC) con los valores de la función de evaluación, denotados como  $fa(k)$ , para cada producto  $k \in X$  con el fin de determinar el primer producto que se asignará. Posteriormente, se crea la RCL, en la cual los elementos que se incluirán están determinados por  $LC(k) \in \{LC_{min}, LC_{min} + \alpha(LC_{max} - LC_{min})\}$ , donde  $LC_{min}$  y  $LC_{max}$  son los valores mínimo y máximo de la LC, respectivamente y,  $\alpha$  es un parámetro de calidad que controla la aleatoriedad que se introduce en el proceso de construcción de la solución. Observe que si  $\alpha = 0$ , la RCL solo contiene el elemento  $LC_{min}$ . Mientras que si  $\alpha = 1$ , la RCL incluirá todos los elementos de la LC, lo que implica que la solución final sería una solución completamente aleatoria. Entonces, el valor  $\alpha$  ayuda a restringir la selección de los mejores elementos durante la construcción de la solución.

Para el caso de estudio se determinó como función de evaluación la ecuación 3.23:

$$fa(k) = \frac{w_k}{fs_k}, \quad (3.23)$$

donde  $fs_k$  es la frecuencia de solicitud del producto  $k \in X$ , es decir, el número de pedidos donde se demanda el producto  $k \in X$ .

Una vez que se determinan los elementos en la RCL, se elige al azar un producto y se elimina de la LC. Para decidir la ubicación de almacenamiento  $j$ , se usó el criterio del vecino más cercano. Con base en  $t_{ij}$  calculamos  $tiempo(i, j, 0) \forall i \in I, j \in I \setminus \{0\}$ , que es el tiempo de recorrido desde la ubicación  $i \in I$  hasta la ubicación  $j \in I \setminus \{0\}$  y regresando a la ubicación 0, definida como la ubicación inicial y final (depósito). Por lo tanto, la mejor ubicación de almacenamiento para asignar el primer producto será la que tenga el menor valor de  $tiempo(0, j, 0)$ , donde  $j \in I \setminus \{0\}$  es la ubicación adyacente libre más cercana a la ubicación 0.

Para que se asigne el siguiente producto, la RCL comienza de nuevo. Se elige un producto al azar y se elimina de la LC. Para decidir la asignación de ubicación, una ubicación adyacente libre a la última ubicación asignada  $i \in I$  es la que tiene el menor valor de  $tiempo(i, j, 0)$ . En el caso de un empate  $tiempo(i, j, 0)$  entre varias ubicaciones adyacentes libres, se elige una de ellas al azar. El proceso anterior se

repite hasta que todos los productos  $k \in X$  se asignan a una ubicación de almacenamiento única  $j \in I \setminus \{0\}$ , donde `AsignarProductoAEspacio` representa la relación de los productos con las ubicaciones de almacenamiento .

---

**Algoritmo 2** Primera etapa de la HCA: Asignación de los productos a una única ubicación

---

**Entradas:** Instancia,  $\alpha$ .

**Salida:** `AsignarProductoAEspacio`

- 1: Calcular  $\text{fa}(k) \forall k \in X$
- 2: Generar LC
- 3: Calcular  $\text{tiempo}(i, j, 0) \forall i \in I, j \in I \setminus \{0\}$
- 4:  $l \leftarrow 0$
- 5: **Mientras** LC  $\neq \emptyset$  **hacer**
- 6:     Crear RCL  $\leftarrow \{b \in LC \mid b \leq LC_{min} + \alpha(LC_{max} - LC_{min})\}$
- 7:     Seleccionar  $b^*$  al azar
- 8:     Eliminar  $b^*$  de LC
- 9:      $j \leftarrow \arg \min_{l \in I \setminus \{0\}} \text{tiempo}(l, j, 0)$
- 10:     `AsignarProductoAEspacio`( $b^*$ ,  $j^*$ )
- 11:      $l \leftarrow j^*$
- 12: **Fin Mientras**

Fuente: Elaboración propia

---

Después de asignar los productos a las ubicaciones de almacenamiento, el segundo paso del RCH consiste en determinar las rutas que debe seguir el recolector para retirar los productos de cada pedido. Los productos de cada pedido se recogen siguiendo un orden decreciente del peso de los productos. Finalmente,  $f(S)$  es la suma resultante del tiempo de selección de las rutas generadas.

### 3.3.2 FASE DE MEJORA

La fase de mejora propuesta es un procedimiento de búsqueda local, cuya solución inicial se obtiene del HCA. El vecindario se explora con base en la primera mejora

encontrada, es decir, se va a mover a la primera solución factible que mejore el valor de la solución que se tiene hasta el momento.

Para construir el vecindario, se propuso una estrategia que cambia la ubicación de almacenamiento de un par de productos con pesos similares para minimizar el tiempo total de recolección. Para determinar qué productos  $k_1$  y  $k_2$  se intercambiarán, se elige al azar el primero,  $k_1$ . Posteriormente, se identifica un producto  $k_2$  de igual o similar peso y se intercambian lugares de almacenamiento. GRASP se detiene cuando alcanza un número máximo de iteraciones consecutivas sin mejora (`iterBS`) o un límite de tiempo máximo (`tiempoBS`). El proceso se repite hasta que se cumple un criterio de parada y se informa la mejor solución encontrada.

### 3.3.3 GRASP REACTIVO - RGRASP

Para evitar la calibración manual del parámetro de calidad  $\alpha$  utilizado para crear la RCL, en la versión reactiva de GRASP, el valor de este parámetro se afina automáticamente de acuerdo con los resultados obtenidos en las iteraciones anteriores. Por lo tanto, el parámetro  $\alpha$  puede tomar cualquier valor de un conjunto discreto  $\Lambda = \{\alpha_1, \alpha_2, \dots, \alpha_m\}$  de posibles valores con una probabilidad  $p_i$ . Luego, en la primera iteración del RGRASP, cada posible valor de  $m$  tiene la misma probabilidad de ser seleccionado  $p_i = \frac{1}{m}, \forall i = 1, 2, \dots, m$

Con base en la investigación de Festa y Resende (2010) y Prais y Ribeiro (2000), después de un determinado número de iteraciones (`iter_2`), las probabilidades de selección para cada  $\alpha$  se reevalúan periódicamente, para un problema de minimización, mediante la ecuación 3.24:

$$p_i = \frac{q_i}{\sum_{j=1}^a q_j}, \quad (3.24)$$

donde  $q_i = \frac{1}{A_i}$  y  $A_i$  representa el valor promedio de todas las soluciones encontradas usando  $\alpha_i \in \Lambda$ . De acuerdo con la definición de  $q_i$ , se puede deducir que si el  $\alpha_i$  seleccionado ha encontrado buenas soluciones, entonces su  $p_i$  será mayor. La elección de parámetros para el RGRASP se describe en la Sección 3.4.3.

A continuación, se detalla la experimentación computacional que se llevó a cabo para validar las reformulaciones matemáticas (SLAUPR\_V2 y SLAUPR\_V3) y el algoritmo RGRASP propuesto.

## 3.4 EXPERIMENTACIÓN COMPUTACIONAL Y RESULTADOS

En esta sección se describen los datos, recursos y otros criterios necesarios que se consideraron para la experimentación computacional que se realizó para validar los métodos de solución propuestos. Asimismo, se realiza el análisis y discusión de los resultados.

### 3.4.1 DATOS

La validación y desempeño de los métodos de solución propuestos en las secciones 3.2 y 3.3 fueron evaluados con las instancias presentadas por Bolaños-Zuñiga *et al.* (2020). Las instancias se clasifican en cuatro tamaños diferentes según la cantidad de ubicaciones para asignar el producto y, en cuatro escenarios con base en el total de productos solicitados en todos o cada uno de los pedidos, considerando que puede existir productos comunes entre estos.

Las principales características de las instancias se muestran en la Tabla 3.3. La columna **Ubicaciones** ( $|I|$ ), representa el número de ubicaciones de almacenamiento disponibles. La columna **Productos solicitados**, muestra el rango del número total de productos solicitados. La columna **Productos a asignar** ( $|X|$ ), muestra el rango del número de diferentes productos a asignar. La columna **Pedido** ( $|P|$ ), muestra el rango de pedidos solicitados. Finalmente, la columna **Número de instancias**, representa el número de instancias disponibles en la literatura. Ver Bolaños-Zuñiga *et al.* (2020) para más detalles.

Tabla 3.3: Descripción de las instancias.

Tamaño	Ubicaciones ( $ I $ )	Productos solicitados	Productos asignados ( $ X $ )	Pedidos ( $ P $ )	Número de instancias
Pequeñas	10 – 16	4	1 – 16	2 – 4	90
Medianas tipo_1	32	8 – 16	5 – 32	1 – 5	144
Medianas tipo_2	61	15– 30	12 – 61	1 – 5	75
Grandes	121 – 185	30	25 – 104	1 – 6	25

### 3.4.2 RECURSOS

Para el desarrollo de los modelos propuestos SLAUPR\_V2 y SLAUPR\_V3, se utilizó el software GAMS versión 23.7.0 y Cplex 12 ya que se cuenta con licencia para su uso. Además, es el mismo software que se utilizó para el desarrollo del modelo SLAUPR. Mientras que el RGRASP propuesto fue implementado en el lenguaje C++.

Para la ejecución de todos los experimentos computacionales se utilizó un equipo de cómputo con procesador Intel Xenon de 2.4GHz, RAM de 64GB y disco duro con capacidad de 2 TB, bajo el sistema operativo Ubuntu versión 18.04.3 LTS.

### 3.4.3 CALIBRACIÓN DE PARÁMETROS DE ENTRADA PARA EL RGRASP

Para determinar los valores de `iter_1`, `tiempoBS`, `iterBS` y los valores para  $\lambda$  de RGRASP se realizó un diseño de experimentos con base en lo siguiente:

- Para fijar el criterio de paro por iteraciones consecutivas sin mejora en la búsqueda local (`iterBS`), se llevó a cabo la impresión del valor de la solución obtenida cada 100 hasta 1,000 iteraciones de cada una de las instancias. Mediante el análisis realizado por tipo de tamaño y escenario de las instancias se determinó, que después de 200 iteraciones en promedio, el valor de la solución

encontrada por la búsqueda local no mejoraba y se mantenía constante hasta las 1,000 iteraciones. Para determinar el porcentaje de mejora (al que denotamos con la letra  $\beta_{HB}$ ) del valor de la solución reportada por la búsqueda local ( $S_B$ ) respecto al valor de la solución de la heurística constructiva aleatorizada ( $S_H$ ) se utilizó la ecuación 3.25:

$$\beta_{HB} = 100 \times \frac{S_H - S_B}{S_H}. \quad (3.25)$$

- Para fijar el criterio de paro por límite de tiempo en la búsqueda local (`tiempoBS`), se utilizó la información del tiempo de ejecución de este algoritmo para reportar una solución, el cual fue menor a 60 segundos en todas las instancias analizadas.
- Para determinar qué conjunto de valores de alfa era el más favorable para todas las instancias en general, se ejecutó cada una de las instancias con los valores de:  $\Lambda = \{0.006, 0.007, 0.008, 0.009, 0.01, 0.05, 0.1, 0.15, 0.2, 0.25\}$ , se reportó el valor de la solución obtenida y después de realizar el análisis de los datos se determinó que sería apropiado utilizar valores de alfas mayores a 0.10.
- Finalmente, con el fin de encontrar el mejor valor posible (en nuestro caso, el menor valor de la solución) de la solución a cada instancia analizada mediante nuestro algoritmo propuesto, el último análisis para la calibración de datos fue determinar por medio de qué tipo de solución sería conveniente inicializar la búsqueda local, ya que mediante la heurística constructiva aleatorizada se pueden obtener buenas (menor valor) o malas (mayor valor) soluciones y, posteriormente, mejorarlas con la búsqueda local propuesta. Después de realizar el análisis de los datos reportados por tamaños y escenarios, se concluye que la búsqueda local encontraba mejores resultados sí esta era iniciada con una buena solución.

Después del análisis del diseño de experimentos realizado y de acuerdo con lo comúnmente utilizado en la literatura, para el RGRASP se utilizaron 10 posibles valores para  $\alpha$  determinados por el conjunto  $\Lambda = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$ , un valor para `iter_1` = 1,00, `iter_2` = 100, `tiempoBS` = 60, `iterBS` = 200.

### 3.4.4 CRITERIOS DE PARO PARA LOS MODELOS REFORMULADOS

Para llevar a cabo la experimentación computacional de los modelos reformulados, se consideran los mismos criterios de paro propuestos en la investigación de Bolaños-Zuñiga *et al.* (2020): 3,600 segundos de ejecución para las instancias de hasta 32 espacios, 7,200 segundos para las de 61 o menos espacios, 10,800 segundos las de 121 o menos espacios y, finalmente, 14,400 segundos para las instancias donde se contemplan todos los espacios disponibles para ubicar producto dentro del almacén.

### 3.4.5 ANÁLISIS DE LOS RESULTADOS OBTENIDOS CON LAS REFORMULACIONES PROPUESTAS

La Tabla 3.4 muestra un resumen de los resultados de los modelos reformulados propuestos. La columna **Número de instancias**, representa el número de instancias disponibles en la literatura. Las columnas **F / O** describen el número de soluciones factibles y el número de soluciones óptimas reportadas en la literatura (SLAPR) y obtenidas con SLAUPR\_V2 y SLAUPR\_V3, respectivamente. Las columnas **Gap promedio (%)** representan el porcentaje promedio del gap de optimalidad reportado por CPLEX de cada modelo.

De acuerdo con los resultados mostrados en Bolaños-Zuñiga *et al.* (2020), para las instancias pequeñas y medianas tipo\_2, el escenario 1 y el escenario 2 son los menos complejos. Mientras que, el escenario 3 y el escenario 4 son los escenarios que presentan mayor complejidad, excepto cuando sólo se debe recoger un pedido. Todos los escenarios son complejos para las instancias medianas tipo\_2 y grandes cuando se resuelven de manera óptima. Por lo tanto, Bolaños-Zuñiga *et al.* (2020) no reporta soluciones factibles para el escenario 4 en las instancias de tamaño mediano\_2 ni en el escenario 1 para las de tamaño grande.

La Tabla 3.4 muestra que las soluciones encontradas por las reformulaciones propuestas SLAUPR\_V2 y SLAUPR\_V3 son competitivas en comparación con los resultados del SLAUPR. Se puede notar que, para las instancias pequeñas, ambos



Tabla 3.4: Resumen de resultados reportados en la literatura (SLAUPR) y los obtenidos con las reformulaciones propuestas.

Tamaño instancia	Escenarios	Número de instancias	SLAUPR		SLAUPR_V2		SLAUPR_V3	
			F / O	Gap promedio (%)	F / O	Gap promedio (%)	F / O	Gap promedio (%)
Pequeñas	1	24	24 / 24	0.00	24 / 24	0.00	24 / 24	0.00
	2	24	24 / 24	0.00	24 / 24	0.00	24 / 24	0.00
	3	18	18 / 6	33.47	18 / 12	19.24	18 / 12	3.83
	4	24	24 / 6	35.92	24 / 12	23.27	24 / 12	8.48
<b>Totales</b>		<b>90</b>	<b>90 / 60</b>		<b>90 / 72</b>		<b>90 / 72</b>	
Medianas tipo_1	1	36	30 / 3	32.11	31 / 23	7.68	34 / 25	6.15
	2	36	30 / 16	22.62	33 / 24	13.13	36 / 24	15.63
	3	36	17 / 12	14.21	17 / 12	15.20	27 / 12	23.54
	4	36	6 / 0	63.49	14 / 0	74.43	19 / 0	54.98
<b>Totales</b>		<b>144</b>	<b>83 / 32</b>	<b>27.28</b>	<b>95 / 59</b>		<b>116 / 61</b>	
Medianas tipo_2	1	28	18 / 0	81.81	17 / 0	57.15	25 / 0	34.94
	2	28	24 / 0	88.51	26 / 0	73.17	25 / 0	57.53
	3	11	7 / 1	53.41	7 / 4	22.44	8 / 3	37.78
	4	8	0 / 0		0 / 0		1 / 0	81.04
<b>Totales</b>		<b>75</b>	<b>49 / 1</b>		<b>50 / 4</b>		<b>59 / 3</b>	
Grandes	1	4	0 / 0		0 / 0		0 / 0	
	2	15	3 / 0	94.68	6 / 0	99.29	0 / 0	
	3	6	2 / 0	83.16	1 / 0	98.23	4 / 0	87.55
<b>Totales</b>		<b>25</b>	<b>5 / 0</b>		<b>7 / 0</b>		<b>4 / 0</b>	
<b>Total general</b>		<b>334</b>	<b>227 / 87</b>		<b>242 / 135</b>		<b>269 / 136</b>	

modelos propuestos logran encontrar seis nuevos casos con solución óptima para los escenarios 3 y 4, respectivamente. De igual manera, el gap promedio alcanzado por el modelo SLAUPR\_V3 para las instancias de estos escenarios (3.83% y 8.48%, respectivamente) es mejor que el alcanzado por el SLAUPR\_V2 (19.24% y 23.27%, respectivamente), y que los reportados en la literatura (33.47% y 35.92%, respectivamente).

Para instancias medianas tipo\_1, ambos modelos encuentran mejores resultados que los reportados en la literatura. El SLAUPR\_V2 obtiene 12 nuevos casos con solución factible y 27 con solución óptima, mientras que el SLAUPR\_V3 encuentra 33 nuevos casos con solución factible y 29 con solución óptima. Asimismo, el gap de optima-

alidad promedio que logra el modelo SLAUPR\_V3 para las instancias de los cuatro escenarios (6.15 %, 15.63 %, 23.54 % y 54.98 %, respectivamente) es mejor que el promedio alcanzado por el SLAUPR\_V2 (7.68 %, 13.13 %, 15.20 % y 74.43 %, respectivamente), así como por los reportados en la literatura (32.11 %, 22.62 %, 14.21 % y 63.49 %, respectivamente). En el escenario 3, se observa que el SLAUPR\_V3 reporta un mayor valor promedio del gap de optimalidad ya que este modelo logra encontrar diez nuevos casos con solución factible, más que los encontrados con el SLAUPR\_V2 y los reportados en la literatura. Cabe señalar que, para este escenario, las instancias donde se reporta y se encuentra el óptimo son aquellas donde se solicita un solo pedido. Sin embargo, en el escenario 4, se puede observar que a pesar de que el SLAUPR\_V3 encuentra una mayor cantidad de nuevos casos con solución factible, el gap de optimalidad promedio es mejor que el alcanzado por el SLAUPR\_V2 y el reportado en la literatura, ya que los gaps de optimalidad mínimos y máximos alcanzados por estos modelos son mayores.

Para las instancias de tamaño medianas tipo.2, aunque el modelo SLAUPR\_V2 presenta un mejor desempeño para las instancias del escenario 3 ya que logra encontrar una mayor cantidad de nuevos casos con solución óptima y el gap de optimalidad promedio (22.44 %) más bajo de los tres modelos, el modelo SLAUPR\_V3, en general, encuentra diez nuevos casos con solución factible y dos con solución óptima, siendo la principal contribución poder obtener un caso, en el escenario 4, con solución factible.

Para las instancias de tamaño grande, a pesar de que los modelos propuestos no logran encontrar soluciones factibles para las instancias del escenario 1, en el escenario 2 el modelo SLAUPR\_V2 logra encontrar tres nuevos casos con solución factible y en el escenario 3 el modelo SLAUPR\_V3 encuentra dos nuevos casos con solución factible. No obstante, se continua, en promedio, con un gap de optimalidad por encima del 80 %. Cabe resaltar que se aún no se logra encontrar una solución factible para la instancia basada en los datos reales.

Finalmente, otro aspecto importante a resaltar de los resultados de los modelos

SLAUPR\_V2 y SLAUPR\_V3 propuestos es el tiempo de ejecución respecto al reportado por el SLAUPR, ya que, para las 90 instancias de tamaño pequeño y las 144 en las medianas tipo\_1, el modelo SLAUPR\_V3 redujo, en promedio, 48.47 % y 22.34 % el tiempo de ejecución, respectivamente. Mientras que, el SLAUPR\_V2 tuvo una reducción en promedio del 44.66 % y 26.35 %, respectivamente. Para las instancias de tamaño medianas tipo\_2 y grandes, la variación no fue significativa.

Considerando la información descrita anteriormente, si bien los modelos propuestos SLAUPR\_V2 y SLAUPR\_V3, en promedio, mejoran los resultados reportados en la literatura, aún no es posible encontrar todas las instancias con soluciones factibles en el tiempo de ejecución establecido. No obstante, el modelo SLAUPR\_V3 propuesto logra encontrar una mayor cantidad de casos con soluciones factibles, principalmente, en las instancias medianas tipo\_1 del 80.5 % (116/144) y en las tipo\_2 del 78.7 % (59/75), lo que sugiere que el uso de las restricciones de eliminación de subciclos basadas en el flujo propuestas por Gavish y Graves (1978) presenta un mejor rendimiento computacional en comparación con las restricciones MTZ propuestas por Miller *et al.* (1960) para este tipo de problema.

### 3.4.6 DESEMPEÑO RGRASP

Con el fin de evaluar la calidad de las soluciones encontradas por el RGRASP propuesto, se realizó una comparación de dichas soluciones contra las mejores soluciones conocidas hasta el momento. Entiéndase como la **mejor conocida** a la mejor solución reportada por CPLEX en los modelos SLAUPR, SLAUPR\_V2 o SLAUPR\_V3.

La Tabla 3.5 muestra los resultados de rendimiento del RGRASP propuesto. La columna **Mejor-conocidas F / O**, muestra el número de instancias con la mejor solución factible conocida y el número de soluciones óptimas encontradas. La columna **RGRASP, F / O** contiene el número de instancias resueltas de forma óptima por el RGRASP. Las columnas **Tiempo promedio (seg)**, representa el tiempo promedio de ejecución requerido por CPLEX y por el RGRASP propuesto, respectivamente.

La columna **Promedio Gap** $_{E,A}$  (%), es el gap promedio entre las mejores soluciones factibles conocidas y la mejor solución encontrada por el RGRASP. El  $\text{Gap}_{E,A}$  (%) se calcula de la siguiente manera:

$$\text{Gap}_{E,A}(\%) = 100 \times \frac{Z_E - Z_A}{Z_E}, \quad (3.26)$$

donde  $Z_E$  y  $Z_A$  representan el valor de la mejor solución conocida al resolver los modelos matemáticos con CPLEX y RGRASP, respectivamente. Un  $\text{Gap}_{E,A}$  negativo indica que los resultados de los modelos son mejores. De lo contrario, RGRASP reportó mejores resultados.

La Tabla 3.5 muestra que las soluciones encontradas por el algoritmo propuesto son competitivas en comparación con los resultados de las soluciones mejores conocidas. Cabe mencionar que el mejor desempeño del RGRASP fue en el conjunto de instancias pequeñas y medianas de tipo\_1 ya que se obtuvo el 100 % (72/72) y el 86.9 % (53/61) de las soluciones óptimas conocidas, respectivamente. Para las instancias medianas tipo\_2, aunque el RGRASP no encontró las soluciones óptimas, mejoró más de un 27 % y un 59 % de las soluciones factibles mejor conocidas en el escenario 2 y el escenario 4, respectivamente. Para instancias grandes, aproximadamente 78 % y 67 % de soluciones en el escenario 2 y escenario 3, respectivamente.

Una de las principales ventajas del RGRASP propuesto es que, por primera vez, se encontró una solución factible para las instancias más complejas reportadas en la literatura. Aunque el RGRASP no encontró todas las soluciones óptimas conocidas, este algoritmo mejora considerablemente más del 29 % de las soluciones factibles conocidas, además de encontrar la misma solución en más del 40 % de los casos. Para todas las instancias probadas, el tiempo de ejecución general promedio para reportar soluciones factibles es de 117 segundos.

Tabla 3.5: Gap entre la mejor solución factible conocida y el RGRASP propuesto.

Tamaño de instancia	Escenarios	Mejor-conocida		RGRASP		Promedio $\text{Gap}_{E,A}$ (%)
		F / O	Tiempo promedio (seg)	F / O	Tiempo promedio (seg)	
Pequeñas	1	24 / 24	0	24 / 24	8	0
	2	24 / 24	0	24 / 24	4	0
	3	18 / 12	1209	18 / 12	12	-2.46
	4	24 / 12	1833	24 / 12	16	0.16
<b>Totales</b>		<b>90 / 72</b>	<b>731</b>	<b>90 / 72</b>	<b>10</b>	
Medianas tipo_1	1	36 / 25	1145	36 / 24	19	-3.12
	2	36 / 24	1211	36 / 23	13	3.15
	3	27 / 12	2047	36 / 6	53	-5.57
	4	25 / 0	3600	36 / 0	78	27.42
<b>Totales</b>		<b>124 / 61</b>	<b>1855</b>	<b>144 / 53</b>	<b>41</b>	
Medianas tipo_2	1	26 / 0	7200	28 / 0	47	5.47
	2	28 / 0	7054	28 / 0	37	27.89
	3	9 / 5	5882	11 / 0	202	-7.91
	4	1 / 0	7200	8 / 0	496	59.87
<b>Totales</b>		<b>64 / 5</b>	<b>6951</b>	<b>75 / 0</b>	<b>114</b>	
Grandes	1	0 / 0	10800	4 / 0	107	-
	2	6 / 0	10800	15 / 0	121	78.72
	3	5 / 0	10800	6 / 0	114	66.92
<b>Totales</b>		<b>11 / 0</b>	<b>10800</b>	<b>25 / 0</b>	<b>117</b>	
<b>Total general</b>		<b>289 / 138</b>		<b>334 / 125</b>		

Fuente: Elaboración propia

### 3.4.7 ANÁLISIS MIPSTART

Para determinar la brecha de optimalidad de las soluciones reportadas por RGRASP y aumentar el número de instancias con soluciones óptimas, se utilizó la solución RGRASP como solución inicial (MIPStart) en el modelo reportado en la literatura (SLAUPR), y en los propuestos SLAUPR\_V2 y SLAUPR\_V3 en las mismas instancias analizadas en la sección anterior.

Los resultados del método MIPStart se muestran en la Tabla 3.6. La columna **Mejora promedio**  $\Delta\text{Gap}_{E,M}$  (%) se refiere a la mejora promedio de la solución entre el modelo original y el MIPStart. El  $\Delta\text{Gap}_{E,M}$  (%) se calcula de la siguiente manera:

$$\Delta\text{Gap}_{E,M}(\%) = 100 \times \frac{Z_E - Z_M}{Z_E}, \quad (3.27)$$

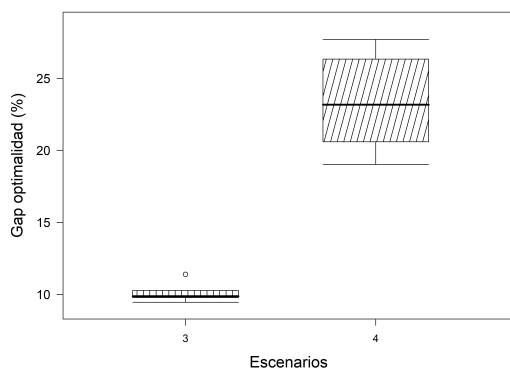
donde  $Z_E$  y  $Z_M$  representan el valor objetivo de la solución al resolver los modelos matemáticos sin y con solución inicial, respectivamente.

Tabla 3.6: Resumen resultados modelos con MIPStart.

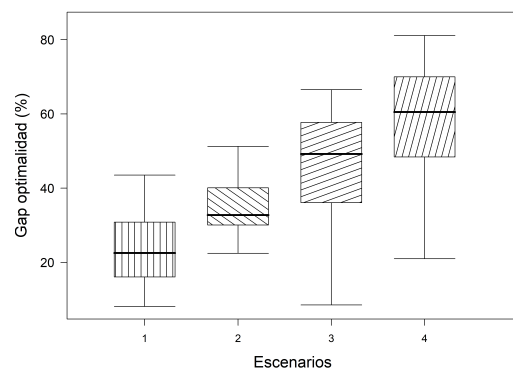
Tamaño instancia	Escenarios	Mejora promedio $\Delta\text{Gap}_{E,M}$ (%)		
		SLAUPR	SLAUPR_V2	SLAUPR_V3
<b>Pequeñas</b>	1	0.00	0.00	0.00
	2	0.00	0.00	0.00
	3	5.84	0.50	0.28
	4	2.72	1.67	0.80
<b>Medianas tipo_1</b>	1	8.8	0.41	1.98
	2	12.52	3.67	8.12
	3	1.42	2.83	10.75
	4	30.11	35.03	26.08
<b>Medianas tipo_2</b>	1	59.01	16.03	15.9
	2	74.1	35.87	35.95
	3	47.52	15.41	31.46
	4	-	-	59.87
<b>Grandes</b>	2	75.97	82.29	-
	3	60.13	72.01	73.69

La Figura 3.1 muestra el gap relativo reportado por CPLEX de la solución factible mejor encontrada (sin incluir las óptimas) al resolver SLAUPR, SLAUPR\_V2 o

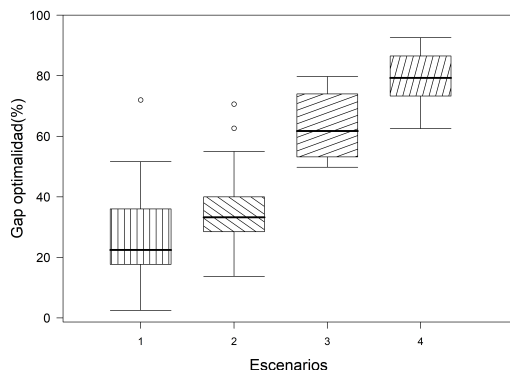
SLAUPR\_V3 con MIPStart.



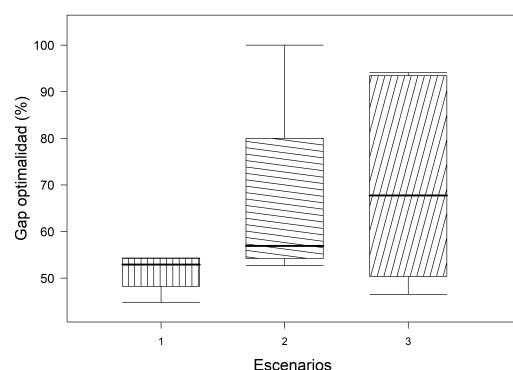
(a) Instancias Pequeñas



(b) Instancias Medianas tipo\_1



(c) Instancias Medianas tipo\_2



(d) Instancias Grandes

Figura 3.1: Gap de optimalidad reportado por CPLEX con el MIPStart.

Para las instancias pequeñas, el escenario 1 y el escenario 2 son los menos complejos, y como la solución inicial coincide con la solución óptima el MIPStart no se presentan mejoras (ver Tabla 3.6). Aunque este tamaño presenta una complejidad baja, los escenarios que han mostrado mayor complejidad son el 3 y el 4. Con base en la Figura 3.1(a), el gap de optimalidad se mantuvo por debajo del 30% en la mayoría de los casos, con máximos de 11% y 27%.

De acuerdo con la información de la Tabla 3.6, el MIPStart mejoró significativamente las soluciones encontradas para las instancias de tamaño medianas tipo\_1, tipo\_2 y

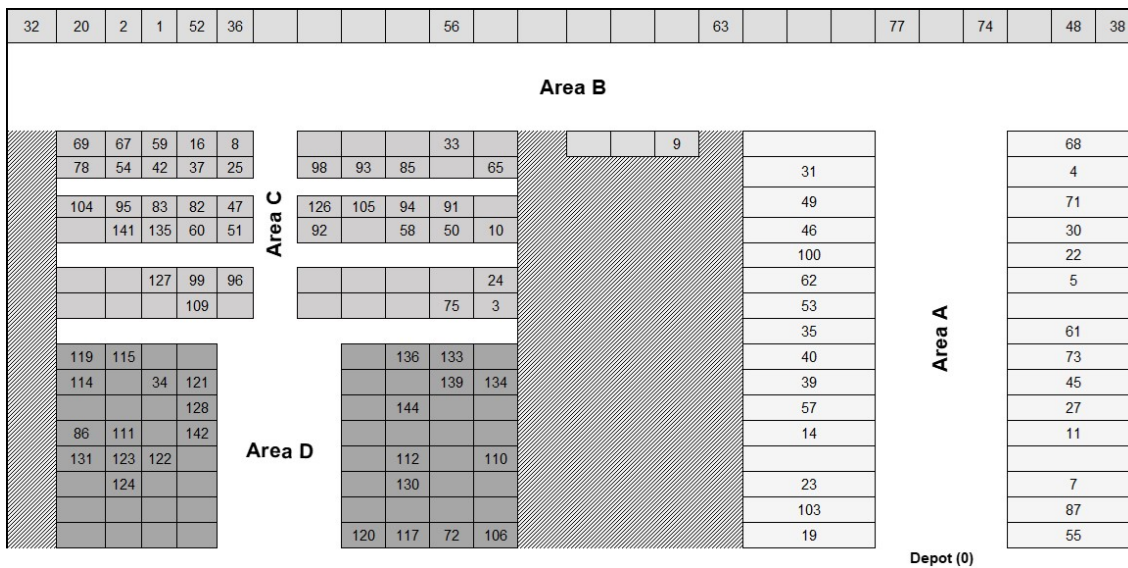
grandes en comparación con los modelos sin soluciones iniciales. El MIPStart del modelo SLAUPR presenta un mayor impacto en la convergencia del optimizador para las instancias de tamaño mediana tipo\_1 y tipo\_2. En cambio, para las grandes, el mayor porcentaje de mejora lo alcanzan el MIPStart del modelo SLAUPR\_V2 (82.29%) en el escenario 2 y el modelo SLAUPR\_V3 (73.69%) en el escenario 3. Dado que para el escenario 1 no logra encontrar casos con solución factible, no se reporta información de la mejora promedio con el MIPStart.

Con base en la solución MIPStart mejor encontrada, la Figura 3.1(b) muestra que, para los escenarios más complejos, en este caso, el escenario 3 y el escenario 4 de las instancias medianas tipo\_1 quedaron con una brecha de optimización máxima de 66% y 81% y con mínimo de 9% y 19%, respectivamente. Aunque para las instancias medianas tipo\_2, los cuatro escenarios siguen siendo muy complejos, la Figura 3.1(c) muestra que para el escenario 1, una brecha mínima informada por CPLEX del 3%, y para el 50% de las instancias probadas, oscila del 18% al 35%, y para el escenario 2, del 30% al 40%. Para el escenario 3, alrededor del 25% de las instancias presentaron un gap de optimalidad del 65% al 75%. Para el escenario 4, el gap mínimo fue de 60%. En el caso de las instancias más grandes, se continúa sin resolver a optimalidad las instancias de los tres escenarios analizados y, como se puede observar en la Figura 3.1(d), se reportó una brecha de optimalidad mínima del 45% y máximas superiores al 50%.

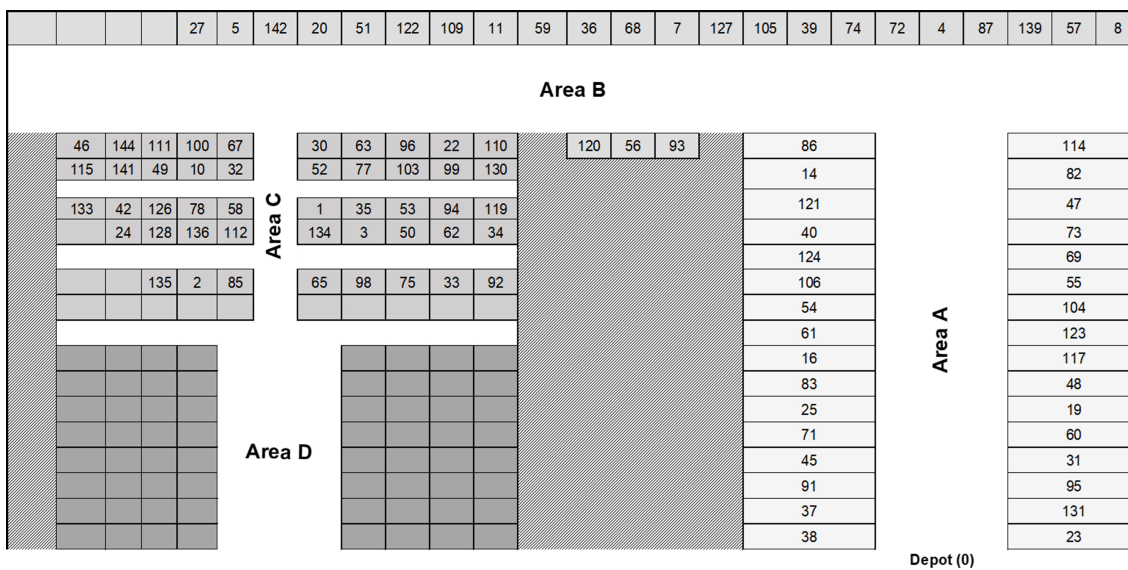
### 3.4.8 COMPARACIÓN DE LA SITUACIÓN REAL Y LA SOLUCIÓN DEL RGRASP

Para determinar las mejoras que aporta el RGRASP, se analiza una instancia basada en datos reales. En la Figura 3.2, se presenta una comparación de la ubicación de almacenamiento y los tiempos de viaje totales en función de la ubicación actual que se tiene en la empresa y la asignación obtenida con RGRASP. Los números de las casillas representan el ID de los productos.





(a) Asignación actual de la empresa - Tiempo de viaje: 4,274 seg.



(b) Asignación propuesta por RGRASP - Tiempo de viaje: 2,446 seg.

Figura 3.2: Ubicación de los productos en el almacén.

La Figura 3.2(a) muestra que, con la asignación de la empresa, hay una dispersión de los productos en el almacén. En contraste, la asignación obtenida por nuestro RGRASP presenta un grupo de productos más compacto (ver Figura 3.2(b)). Por lo tanto, el tiempo de viaje más corto lo logra el algoritmo propuesto ya que la reubicación de todos los productos analizados considera simultáneamente la asignación del mejor producto candidato a través de la función de evaluación descrita en la expresión (3.23) y la elección del producto más cercano.

El tiempo total de viaje en la ruta se reduce en más de 40 %, lo que permite realizar esta actividad más rápido sin afectar el producto final (pedido). Del mismo modo, los recolectores no tendrán que elegir rutas alternativas y no incurrirán en costos adicionales para cumplir con la recolección de los pedidos.

## CAPÍTULO 4

# PROCESAMIENTO DE PEDIDOS POR LOTES Y ENRUTAMIENTO DEL RECOLECTOR

---

Este capítulo está enfocado en el estudio de la segunda combinación o variante del presente trabajo, la cual está relacionada con los problemas de procesamiento de pedidos por lotes y el enrutamiento del recolector. En la Sección 4.1, se presenta la revisión de literatura y trabajos relacionados con la variante mencionada. En la Sección 4.2, se describen las formulaciones matemáticas propuestas. La Sección 4.3 se muestran los resultados experimentales y la discusión sobre el rendimiento de los modelos propuestos.

### 4.1 REVISIÓN DE LITERATURA

El problema de procesamiento de pedidos por lotes y el problema de enrutamiento del recolector son dos problemas importantes en las operaciones de almacén y están estrechamente relacionados. Conforme a lo mencionado por Gu *et al.* (2007), el problema del procesamiento por lotes radica en la necesidad de dividir una serie de solicitudes de clientes en lotes, los cuales se recolectarán y acumularán para su posterior empaque y envío durante un periodo de tiempo determinado o lote de

recolección. Cabe destacar que el tiempo necesario para la recolección de los artículos en cada lote no debe sobrepasar la ventana de tiempo establecida ni la duración prevista para el lote de recolección. Por otro lado, el problema del enrutamiento del recolector se enfoca en determinar las mejores rutas que deben seguir los recolectores mientras recolectan los productos en el almacén. Por ende, la combinación de estos problemas permite mejorar la eficiencia en el proceso de preparación de pedidos, ya que, al agrupar los pedidos de manera apropiada, el número de viajes que los recolectores deben realizar se reduce significativamente. Asimismo, al determinar las rutas óptimas para los recolectores, se minimiza la distancia total recorrida o el tiempo empleado para la recolección.

dado que el tiempo de recorrido de cada lote generado varía en función de cuáles son los productos que se van a recolectar.

De acuerdo con van Gils *et al.* (2018b), ya que esta combinación de problemas es la principal manera para mejorar el tiempo de recolección, en la literatura, se han propuesto diversos métodos exactos, heurísticas, metaheurísticas y algoritmos genéticos para resolver este tipo de problemas, de manera simultánea o independientes. Sin embargo, la mayoría de los trabajos proponen los métodos en almacenes con forma de bloque, mientras que para los almacenes con forma general analizado en presente trabajo, hasta donde sabemos, no hay una investigación que lo considere.

En el trabajo de Valle *et al.* (2016) se han propuesto diferentes técnicas heurísticas para el enrutamiento de múltiples recolectores y el procesamiento por lotes. En muchos casos, la distancia de enrutamiento se estima utilizando heurísticas de un único recolector mientras se resuelve el problema de procesamiento de pedidos por lotes. Se puede encontrar una investigación exhaustiva sobre los métodos empleados en este problema. No obstante, en estos trabajos no se ha considerado el factor de las fechas de cumplimiento de los pedidos, el cual, según Gu *et al.* (2007), es un criterio de desempeño al que también se encuentra sujeto el agrupamiento de pedidos. Dicho criterio resulta también de suma importancia en la presente investigación.

En cuanto a las investigaciones en las que solo se aborda el problema del procesa-

miento de pedidos por lotes y las fechas de entrega compromiso, algunos trabajos, como Tsai *et al.* (2008) y Elsayed *et al.* (1993) los autores incorporan penalizaciones por la finalización anticipada o tardía de los lotes para obtener los mejores planes posibles de recolección por lotes, ya que con base en Tsai *et al.* (2008), recolectar productos antes de la fecha de compromiso, genera acumulación de productos en el área destinada para despacho, lo que requiere mayor espacio y mano de obra para gestionarlos. Por lo tanto, se puede generar errores que pueden resultar en el incumplimiento de estas fechas. Cabe mencionar que en el trabajo de Tsai *et al.* (2008) en los algoritmos genéticos propuestos los productos de un único pedido pueden asignarse a diferentes lotes, mientras que en el trabajo de Elsayed *et al.* (1993), en la heurística propuesta cada pedido contiene un único producto y el almacén tiene un sistema de recolección semiautomatizado (AS/RS). Por ende, estos trabajos difieren de la presente investigación, ya que los pedidos contienen múltiples productos y no está permitido la recolección de los productos contenidos en cada tarima en diferentes recorridos. Además, los posibles problemas causados por una recolección anticipada de los pedidos no son considerados por la empresa caso de estudio.

Entre las investigaciones relacionadas con el presente estudio se encuentra el trabajo de van Gils *et al.* (2019), donde se propone para una empresa de repuestos de vehículos, un modelo matemático y un algoritmo heurístico para resolver de manera simultánea el procesamiento por lotes y el enrutamiento del recolector, incluyendo los tiempos de entrega de pedidos como restricciones duras en un almacén con sistema de recolección de nivel alto. Además, en este estudio se aborda el problema de programación del recolector, el cual no se considera en la presente investigación. Otra de los trabajos es el de Azadnia *et al.* (2013), en donde los autores propusieron algoritmos genéticos para resolver en la primera etapa, el procesamiento por lotes considerando las fechas de entrega de pedidos, y luego el enrutamiento en una segunda etapa, también considera el problema de secuenciación de pedidos, problema el cual no se aborda en el presente estudio. No obstante, en las investigaciones de Henn (2015); Schubert *et al.* (2018); Scholz *et al.* (2017) y Pinto y Nagano (2022) se puede consultar los trabajos relacionados con esta variante.

Por otro lado, investigaciones como las de Valle *et al.* (2016) y Valle *et al.* (2017) los autores proponen diferentes modelos de programación entera para resolver de manera conjunta el problema del procesamiento de pedidos por lotes y el enrutamiento del recolector. Sin embargo, estos modelos no consideran las fechas de compromiso de entrega de los pedidos.

Es importante mencionar que, en los trabajos comentados anteriormente, los autores han considerado una ubicación de almacenamiento ya establecida, criterio que también se considera en el presente trabajo. No obstante, se puede encontrar en los trabajos de van Gils *et al.* (2018a) investigaciones donde también se ha considerado el problema de asignación de almacenamiento.

De acuerdo con de Koster *et al.* (1999), es posible disminuir hasta en un 35 % el tiempo de viaje en la recolección de productos, siempre y cuando se diseñen cuidadosamente las rutas a seguir. Asimismo, se puede aumentar significativamente dichos beneficios si se toman en cuenta el procesamiento por lotes y el enrutamiento de manera simultánea, dado que el tiempo de recorrido de cada lote generado varía en función de cuáles son los productos que se van a recolectar. Sin embargo, con base en la problemática presentada en la empresa, se adoptará un enfoque independiente para resolver los problemas de procesamiento de pedidos por lotes y el enrutamiento del recolector en dos etapas. Lo anterior debido a que, el agrupamiento de los pedidos por lotes depende de la disponibilidad de los productos y las fechas de entrega de los pedidos y no sobre las decisiones de qué productos recolectar. Adicionalmente, ya que los contenidos de las tarimas son conocidos, el tiempo de recolección de sus productos se mantiene constante a lo largo del horizonte de planificación.

En este trabajo se utiliza el concepto de procesamiento de pedidos por lotes para generar diariamente dentro del horizonte de planeación, el agrupamiento de las tarimas (lotes) que deberán ser recolectadas para cumplir con la fecha de compromiso de entrega del pedido al cual pertenecen. Dicho lo anterior, en la primera etapa, nos enfocaremos en desarrollar un modelo matemático MILP para resolver el procesamiento por lotes con base en la fecha de compromiso de entrega de los pedidos y

el inventario disponible, con el fin de minimizar las entregas tardías. Cabe mencionar, que un pedido se considera completo cuando se han recolectado todas tarimas en las que fue dividido. Además, de existir una capacidad máxima de tarimas que se pueden recolectar por día. Aunque en investigaciones tales como Matusiak *et al.* (2014); Pinto y Nagano (2022) mencionan que el método más utilizado para resolver este tipo de problemas es por medio de heurísticos, en este trabajo no se descarta la posibilidad de desarrollar y formular un modelo matemático que se resuelva de manera exacta, ya que a diferencia de los trabajos previos, hasta donde sabemos, no existe un modelo que se adapte completamente a las particularidades del problema de estudio, principalmente considerando, un almacén con forma general, los productos contenidos en cada tarima a recolectar, fechas compromiso de entrega, nivel de inventario disponible, capacidad máxima de tarimas a recolectar diariamente, peso de los productos y demanda estable por periodos.

Teniendo en cuenta que la recolección de las tarimas en las que se divide un pedido se puede procesar por separado y que se cuenta con la asignación de ubicaciones de almacenamiento para cada uno de los productos en el almacén, en la segunda etapa del proceso se procede al enrutamiento de los recolectores, con el propósito de generar las rutas de recolección individual de las tarimas de los lotes generados en la primera etapa. En este enrutamiento se consideran las restricciones de precedencia del peso de los productos, con el objetivo de minimizar el tiempo de viaje en la recolección y garantizar una presentación adecuada de los pedidos de los clientes.

## 4.2 FORMULACIONES MATEMÁTICAS PROPUESTAS

Como se mencionó en la Sección 1.1, el proceso del agrupamiento de los pedidos por lotes y enrutamiento del recolector en el almacén de exportación, no se están realizando en los tiempos establecidos por la empresa. En ocasiones, no es posible entregar los pedidos antes o durante la fecha de compromiso debido a que se programa la recolección un pedido que incluye productos no disponibles en el inventario. Por otro lado, dado que en las rutas generadas por el WMS no se contempla el peso

del producto, los empleados realizan rutas alternas con base en este criterio, lo que incrementa el tiempo de viaje en la recolección. La empresa para tratar de mitigar la tardanza en la recolección ha requerido de más personal para llevar a cabo el proceso y, por consiguiente, se incrementado los costos por este concepto.

Para resolver los problemas del procesamiento de pedidos por lotes y enrutamiento del recolector se propone resolver cada problema en etapas que serán dependientes entre sí. En la primera etapa (ver Sección 4.2.1) se propone un modelo matemático MILP para generar los lotes de recolección con base en la fecha de compromiso de entrega de los pedidos y el inventario disponible, con el fin de minimizar las entregas tardías. Posteriormente, en la segunda etapa (ver Sección 4.2.2), con el objetivo de minimizar el tiempo de viaje en la recolección de los lotes formados en la primera etapa y garantizar una presentación adecuada de los pedidos de los clientes, se propone adaptar el modelo descrito en la Sección 2 para generar las rutas de recolección considerando de igual manera el peso de las cajas de los productos.

A continuación, se describen los modelos matemáticos propuestos.

#### 4.2.1 PROCESAMIENTO DE PEDIDOS POR LOTES (PPL) –

##### PRIMERA ETAPA

A continuación, se describen los supuestos, parámetros, variables y el modelo matemático propuesto con base en las características del caso de estudio. Este modelo tiene por nombre PPL.

##### **Supuestos:**

- La demanda del producto es estable y conocida. Debido a que las cantidades solicitadas son por número de cajas, entonces estas serán enteras.
- Se conoce el día en que los productos (SKU) llegan al almacén.
- Cada pedido se divide en tarimas, y se conocen el número de tarimas en cada pedido y los productos en cada tarima.



- El envío de un pedido solo se puede realizar una vez que se han recolectado todos los productos de las tarimas en las que fue dividido el pedido del cliente, es decir, hasta que el pedido esté completo.
- Se conoce la fecha de compromiso de entrega de cada pedido, la cual es la fecha límite de despacho por parte de la empresa, ya que al tratarse de productos que son enviados al exterior, existen riesgos en el transporte que no son considerados.
- El horizonte de planificación es de una semana (lunes a viernes).
- La solución se necesita una vez por semana.
- Cantidad de tarimas máximas recolectadas por día ( $Q$ ), considerando un total de 12 trabajadores.

La notación correspondiente al modelo propuesto PPL es:

**Conjuntos:**

$K$ : Denota el conjunto de los diferentes tipos de productos solicitados, indexado por  $k$ .

$O$ : Denota el conjunto de pedidos, indexado por  $o$ .

$\mathcal{P}$ : Denota el conjunto de tarimas, indexado por  $\rho$ .

$D$ : Denota el conjunto de días, indexado por  $d$ .

$\mathcal{P}_o \subseteq \mathcal{P}$ : Denota el conjunto de tarimas requeridas para cumplir con el pedido  $o \in O$ .

**Parámetros:**

$ini_k$ : Inventario inicial del producto  $k \in K$ .

$a_{kd}$ : Inventario en tránsito del producto  $k \in K$  en el día  $d \in D$ .

$dem_{k\rho o}$ : Cantidad de producto  $k \in K$  en la tarima  $\rho \in \mathcal{P}_o$  solicitado en el pedido  $o \in O$ .

$f_o$ : Fecha de compromiso de entrega del pedido  $o \in O$ .

$Q$ : Número máximo de tarimas a recolectar diariamente.

**Variables:**

$$r_{opd} = \begin{cases} 1, & \text{si la tarima } \rho \in \mathcal{P}_o \text{ del pedido } o \in O \text{ es recolectada en el día } d \in D \\ 0, & \text{caso contrario.} \end{cases}$$

$$y_o = \begin{cases} 1, & \text{si el pedido } o \in O \text{ está completo} \\ 0, & \text{caso contrario.} \end{cases}$$

$s_{kd}$ : Inventario disponible del producto  $k \in K$  en el día  $d \in D$ .

**Modelo PPL:**

$$\text{Min } |O| - \sum_{o \in O} y_o \quad (4.1)$$

$$s_{kd} = ini_k \quad \forall k \in K, d = 0 \quad (4.2)$$

$$s_{kd} = a_{k(d-1)} + s_{k(d-1)} - \left( \sum_{o \in O} \sum_{\rho \in \mathcal{P}_o} dem_{k\rho} r_{o\rho(d-1)} \right) \quad \forall k \in K, d \geq 1 \quad (4.3)$$

$$s_{kd} \geq \left( \sum_{o \in O} \sum_{\rho \in \mathcal{P}_o} dem_{k\rho} r_{o\rho d} \right) \quad \forall k \in K, d \in D \quad (4.4)$$

$$\sum_d r_{opd} \leq 1 \quad \forall o \in O, \rho \in \mathcal{P}_o \quad (4.5)$$

$$\sum_{\rho \in \mathcal{P}_o} \sum_{d \in D} r_{opd} = |\mathcal{P}_o| y_o \quad \forall o \in O \quad (4.6)$$

$$\sum_{o \in O} \sum_{p \in \mathcal{P}_o} r_{opd} \leq Q \quad \forall d \in D \quad (4.7)$$

$$r_{opd} = 0 \quad \forall p \in \mathcal{P}_o, o \in O, d > f_o \quad (4.8)$$

$$r_{opd} \in \{0, 1\} \quad \forall p \in \mathcal{P}_o, o \in O, d \in D \quad (4.9)$$

$$y_o \in \{0, 1\} \quad \forall o \in O \quad (4.10)$$

$$s_{kd} \geq 0 \quad \forall k \in K, d \in D. \quad (4.11)$$

La función objetivo (4.1) minimiza el número de pedidos con entregas tardías. Las restricciones del tipo (4.2) – (4.4) corresponden a restricciones de control de inventario. Las restricciones (4.5) garantizan que cada tarima de un pedido se asigne solo

un día. Las restricciones (4.6) garantizan que un pedido está completo, es decir, se recolectan todas las tarimas del pedido y además, garantiza que solo se asignen a recolección aquellas tarimas de un pedido completo. Las restricciones (4.7) aseguran que se cumpla la capacidad de tarimas recolectadas por día. Las restricciones (4.8) representan el vencimiento de la fecha de compromiso de entrega de cada pedido. Finalmente, las restricciones del tipo (4.9)–(4.11) son la naturaleza de las variables.

#### 4.2.2 ENRUTAMIENTO DEL RECOLECTOR (PER) - SEGUNDA ETAPA

Para esta etapa se cuenta con una ubicación de almacenamiento predeterminada para cada uno de los productos y la recolección de las tarimas de los lotes generados en la anterior etapa es individual (Menéndez *et al.*, 2017), es decir, cada operario recolecta todos los artículos incluidos en una tarima. Una vez que termina, continúa con otra tarima y así sucesivamente.

Continuando con la notación correspondiente en la sección anterior, se describen los supuestos, parámetros, variables y la adaptación del modelo propuesto. Este modelo tiene por nombre PER.

##### Supuestos

- Las cajas tienen las mismas medidas.
- Siempre hay producto disponible para cubrir la demanda de lotes generados en la primera etapa.
- Se conoce la ubicación de almacenamiento de cada producto, la cual es única para cada uno de estos.
- Se inicia y finaliza el recorrido en el depósito.
- Se conoce el peso de la caja del producto.
- El desempeño del recolector no afecta el tiempo de viaje entre ubicaciones.

**Conjuntos:**

$I$ : Denota el conjunto de ubicaciones de almacenamiento incluido el depósito, indexado por  $i, j$ .

$\mathcal{P}_d \subseteq \mathcal{P}$ : Denota el conjunto de tarimas asignadas a recolectar el día  $d \in D$ .

$I_\rho \subseteq I$ : Denota el conjunto de ubicaciones de almacenamiento a visitar por cada tarima  $\rho \in \mathcal{P}_d$ , incluido el depósito.

**Parámetros:**

$t_{ij}$ : Matriz tiempo de viaje desde la ubicación  $i$  hasta la ubicación  $j$  ( $i, j \in I$ ).

$w_j$ : peso del producto almacenado en la ubicación  $j \in I \setminus \{0\}$ .

$$M = \max_{j \in I_\rho \setminus \{0\}} \{w_j\}.$$

**Variables:**

$h_{j\rho}$  determina el orden en que son visitadas las ubicaciones  $j \in I$  para la tarima  $\rho \in \mathcal{P}_d$ .

$$u_{\rho ij} = \begin{cases} 1, & \text{si para la tarima } \rho \in \mathcal{P}_d \text{ se realiza el recorrido de } i \text{ a } j \text{ (} i, j \in I \text{)} \\ 0, & \text{caso contrario.} \end{cases}$$

**Modelo PER:**

$$\min \sum_{i \in I} \sum_{j \in I} \sum_{\rho \in \mathcal{P}_d | d \in D} t_{ij} u_{\rho ij} \quad (4.12)$$

$$s.a : \sum_{i \in I_\rho \setminus \{0\}} \sum_{j \in I_\rho \setminus \{0\}} u_{\rho ij} \geq |K_\rho| - 1 \quad \forall \rho \in \mathcal{P}_d, d \in D \quad (4.13)$$

$$\sum_{j \in I_\rho \setminus \{0\}} u_{\rho 0j} = 1 \quad \forall \rho \in \mathcal{P}_d, d \in D \quad (4.14)$$

$$\sum_{j \in I_\rho | j \neq i} u_{\rho ij} \leq 1 \quad \forall i \in I_\rho, \rho \in \mathcal{P}_d, d \in D \quad (4.15)$$

$$h_{ip} - h_{j\rho} + (|I_\rho| + 1)u_{\rho ij} \leq |I_\rho| \quad \forall i, j \in I_\rho \setminus \{0\}, \rho \in \mathcal{P}_d, \\ d \in D, i \neq j \quad (4.16)$$

$$u_{\rho ii} = 0 \quad \forall i \in I_\rho, \rho \in \mathcal{P}_d, d \in D \quad (4.17)$$

$$\sum_{i \in I_\rho} u_{\rho ij} = \sum_{i \in I_\rho} u_{\rho ji} \quad \forall j \in I_\rho, \rho \in \mathcal{P}_d, d \in D, i \neq j \quad (4.18)$$

$$w_i - w_j \geq M(u_{\rho ij} - 1) \quad \forall i, j \in I_\rho \setminus \{0\}, \rho \in \mathcal{P}_d, d \in D, i, j \geq 1 \quad (4.19)$$

$$u_{\rho ij} \in \{0, 1\} \quad \forall i, j \in I_\rho, \rho \in \mathcal{P}_d, d \in D \quad (4.20)$$

$$h_{j\rho} \geq 0 \quad \forall j \in I_\rho, \rho \in \mathcal{P}_d, d \in D \quad (4.21)$$

La función objetivo (4.12) minimiza el tiempo de viaje en la recolección de las tarimas. Las restricciones (4.13) aseguran que se visiten todas las ubicaciones que contienen los productos de cada tarima asignada en un determinado día. Las del tipo (4.14) aseguran que la ruta de cada tarima inicie en el punto de origen. Las (4.15) aseguran que se debe salir cuando mucho una vez de cada ubicación visitada en cada tarima. Las (4.16) y (4.17) son para de eliminación de subciclos con base en la formulación MTZ propuestas por (Miller *et al.*, 1960). Las restricciones del tipo (4.18) garantizan la conservación del flujo por tarima. Las (4.19) son las restricciones de precedencia basadas en el peso de los productos. Finalmente, las restricciones (4.20)–(4.21) son la naturaleza de las variables.

### 4.2.3 GENERADOR DE INSTANCIAS

Se tiene disponible información real sobre la composición de los pedidos realizados durante un mes, la demanda total por cada tipo de producto, así como la ubicación dentro del almacén de exportación donde se encuentra alojado cada producto. De acuerdo con el análisis de estos datos, se determinaron los siguientes parámetros para el generador de instancias

- Cantidad total de diferentes productos manejados por la empresa en el almacén de exportación ( $L$ ).
- Cantidad de días del horizonte de planeación ( $D$ ).
- Cantidad total de pedidos por semana ( $O$ ).

- Conjunto discreto  $QP = \{cp_0, cp_1, \dots, cp_n\}$  de posibles cantidades de productos diferentes que pueden ser asignados a un pedido por semana y su respectiva frecuencia ( $\mathbf{nProds}(cp)$ ).
- Capacidad mínima ( $\mathbf{minCajasT}$ ) y máxima ( $\mathbf{maxCajasT}$ ) de cajas en una tarima, cuando hay diferentes tipos de productos en esta.
- Cantidad de demanda mínima ( $\mathbf{minDem}$ ) y máxima de cada tipo producto ( $\mathbf{maxDem}$ ) en un pedido por semana.
- Capacidad máxima de cajas por cada tipo de producto en una tarima ( $\mathbf{cajasTProd}(l)$ ).
- Frecuencia de solicitud de cada tipo de producto ( $\mathbf{frecuenciaProd}(l)$ ) por semana.

Sea  $O = \{0, 1, \dots, o\}$  el conjunto total de pedidos a generar,  $L = \{0, 1, \dots, l\}$  el conjunto del total los diferentes tipos de productos disponibles en el almacén de exportación y  $D = \{0, 1, \dots, d\}$  el conjunto total de días del horizonte de planeación. Para la generación de los pedidos, inicialmente, se determina la cantidad de productos diferentes que se van a asignar a un pedido  $o \in O$ , para calcular dicha cantidad, se selecciona un valor aleatorio entre los valores del conjunto  $QP$ . Por ende, cada pedido  $o \in O$  tendrá una cantidad  $cp$  de productos diferentes. Para saber cuáles productos se asignarán en el pedido  $o \in O$ , se seleccionan de manera aleatoria  $cp$  productos diferentes entre los valores del conjunto  $L$ . Cabe mencionar que para evitar una selección totalmente aleatoria o determinista de un producto  $m \in L$ , este se selecciona aleatoriamente de una lista restringida de candidatos (LRC), donde los elementos a incluir en esa lista están determinados por  $\mathbf{frecuenciaProd}(l) \in \{\mathbf{frecuenciaProd}(l)_{max} - \Omega(\mathbf{frecuenciaProd}(l)_{max} - \mathbf{frecuenciaProd}(l)_{min}), \mathbf{frecuenciaProd}(l)_{max}\}$ , donde  $\Omega$  es un valor que ayuda a controlar el rango de los valores a incluir en la lista,  $\mathbf{frecuenciaProd}(l)_{min}$  y  $\mathbf{frecuenciaProd}(l)_{max}$  son los valores mínimo y máximo de la frecuencia de solicitud de cada producto  $l \in L$ , respectivamente. Una vez seleccionado el producto  $m \in L$  se le genera una  $\mathbf{demanda}(m)$  aleatoria entre  $\mathbf{minDem}$  y  $\mathbf{maxDem}$  y este se elimina como un producto posible a seleccionar en el mismo pedido

$o \in O$ . Para cada pedido  $o \in O$ , se actualiza la frecuencia de  $\mathbf{nProds}(cp)$  y para cada producto  $m \in L$  se actualiza  $\mathbf{frecuenciaProd}(m)$ . Dada la variabilidad de los valores de las frecuencias de solicitud de los productos  $l \in L$ , se determinó fijar  $\Omega = 0.1$ .

Una vez que se tienen determinados los diferentes tipos de productos  $m \in L$  en cada pedido  $o \in O$  y su respectiva  $\mathbf{demanda}(m)$ , se procede a determinar en cuantas tarimas se divide cada pedido  $o \in O$  para recolectar la demanda  $\mathbf{demanda}(m)$  generada. Para calcular la cantidad de tarimas ( $nTarimas_o$ ) se consideró el criterio de la cantidad máxima de cajas que se pueden paletizar en esta, la cual está limitada por la  $\mathbf{cajasTProd}(m)$  si solo coloca un tipo de producto en la tarima o un valor  $\mathbf{capacidadT}$  el cual es un número aleatorio generado entre  $\mathbf{minCajasT}$  y  $\mathbf{maxCajasT}$ , cuando en una tarima se coloca más de un tipo de producto. Por lo tanto, si para el pedido  $o \in O$  el  $cp = 1$  y la  $\mathbf{demanda}(m) \geq \mathbf{cajasTProd}(m)$ , entonces se necesita una tarima que contiene la cantidad de  $\mathbf{cajasTProd}(m)$  y otra tarima para la parte restante ( $\mathbf{demandaPend}(m)$ ), es decir,  $\mathbf{demanda}(m) - \mathbf{cajasTProd}(m)$ . En caso de que  $cp > 1$ ,  $\mathbf{demanda}(m) \geq \mathbf{cajasTProd}(m)$  y  $\mathbf{demanda}(m+1) \leq \mathbf{cajasTProd}(m+1)$ , se cumple que en cada tarima solo se puede colocar hasta  $\mathbf{cajasTProd}(m)$  de un solo tipo de producto y la  $\mathbf{demandaPend}(m)$ , se coloca en otra tarima tal que se cumpla la ecuación (4.22):

$$\sum_{m=1}^{cp} (\mathbf{demandaPend}(m) + \mathbf{demanda}(m+1)) \leq \mathbf{capacidadT} \quad (4.22)$$

En vista de que las decisiones tomadas en el CEDIS no tienen una influencia directa en la definición del plan de producción ni en las fechas de entrega de los pedidos por parte de la empresa, puesto que estos procesos son responsabilidad de otros departamentos, no se dispone de información precisa acerca de estos datos. Por consiguiente, los registros del inventario inicial del producto, el inventario en tránsito y las fechas de entrega de los pedidos fueron generados de forma aleatoria.

Dado que la empresa es responsable de la producción de sus propios productos, la llegada de los productos en tránsito está sujeta a diversos factores, tales como una

programación inadecuada en la fabricación, fallas en la línea de producción, daños del producto por la manipulación, entre otros. Por ende, ya que no se cuenta con la información real sobre las fechas de llegada del producto, ni tampoco cuanto producto podría llegar, con el fin de simular la situación de que el producto no llega en el momento en que se necesita, se generó de manera aleatoria cuándo y cuánto producto en tránsito (`transitoCubierto`) llega al CEDIS. De igual manera, se propone un porcentaje de inventario inicial (`inicialCubierto`) con base en la situación que sucede en la práctica, al menos para este tipo de empresas.

De acuerdo con lo mencionado anteriormente, se determinó que para cada producto diferente solicitado el inventario inicial más el inventario en tránsito es igual (o mayor) a la demanda total de ese producto, en el horizonte de planeación.

Por otro lado, se conoce que un producto puede llegar entre mínimo una vez y máximo tres veces por semana y que el producto en tránsito se considera disponible el día posterior a su llegada. Se determina como `minFecha` el primer día horizonte de planeación y `maxFecha` un día antes de finalizar dicho horizonte. A continuación, se describe la manera en que se generaron los datos correspondientes al producto en tránsito:

1. Generar el inventario total en tránsito.
2. Generar aleatorio ( $nDias$ ) entre uno y tres para saber cuántas veces se recibirá el producto  $l \in L$  durante el período de planeación.
3. Generar aleatoriamente entre `minFecha` y `maxFecha` en que llegará el producto  $l \in L$  (frecuencia de llegada del producto al almacén).
4. Si  $nDias = 1$ , se asigna el el valor del paso 1 al día generado en el paso 3.
5. Si  $nDias > 1$ , para cada día se genera un aleatorio  $pT_{nd}$  entre un valor  $\alpha$  y 1, para determinar qué porcentaje de la cantidad generada en el paso 1 se recibirá en cada uno de los días determinados en el paso 3, considerando siempre que



se cumpla la ecuación (4.23):

$$\sum_{nd=1}^{nDias} pT_{nd} = 1 \quad (4.23)$$

El valor  $\alpha$  fue propuesto con el fin de generar una cantidad heterogénea del producto en tránsito que se recibirá de acuerdo con la frecuencia de llegada al almacén. De igual manera, este valor ayuda a limitar el porcentaje mínimo del producto en tránsito que llegaría al almacén, es decir, si  $\alpha = 0$  se recibiría entre un 0% y 100% del producto en tránsito en los primeros días determinados en el paso 3, mientras que si  $\alpha = 30$ , se recibiría entre un 30% y 100%. Por ende, si  $\alpha$  es muy pequeño, hay una mayor probabilidad de que la cantidad del producto en tránsito llegue el último día del paso 3, mientras que, si es un valor muy grande, hay una mayor probabilidad de que se reciba la mayor cantidad del producto en tránsito los primeros días.

Finalmente, para establecer el día de entrega de cada pedido ( $f_o$ ), se genera un número aleatorio entre `minFecha` y el último día del horizonte de planeación.

Es importante mencionar que todos los números aleatorios generados en la ejecución del generador siguen una distribución uniforme.

## 4.3 EXPERIMENTACIÓN COMPUTACIONAL Y RESULTADOS

A continuación, se describen los datos, recursos y otros criterios necesarios que se consideraron para la experimentación computacional que se realizó para validar los modelos matemáticos propuestos. Asimismo, se realiza el análisis y discusión de los resultados.

### 4.3.1 RECURSOS

Para la implementación de los modelos matemáticos propuestos en las secciones 4.2.1 y 4.2.2 se utilizó el lenguaje de programación C++ y Cplex versión 22 como

optimizador.

Para la ejecución de todos los experimentos computacionales se utilizó un equipo de cómputo con procesador Intel Xenon de 2.4GHz, RAM de 64GB y disco duro con capacidad de 2 TB, bajo el sistema operativo Ubuntu versión 18.04.3 LTS.

### 4.3.2 PROCESAMIENTO DE PEDIDOS POR LOTES (PPL) -

#### PRIMERA ETAPA

##### 4.3.2.1 DATOS

Para la validación del modelo matemático propuesto para la primera etapa, es decir, para llevar a cabo el procesamiento de pedidos por lotes (PPL) considerando las fechas de compromiso de entrega de los pedidos y el inventario disponible (ver Sección 4.2.1), se utilizaron las instancias creadas mediante el generador propuesto en la Sección 4.2.3. Se proponen tres tamaños de instancias con base en la cantidad de pedidos por semana ( $O$ ), los cuales, de acuerdo con la base de datos, semanalmente se procesa un máximo de 160 pedidos. De igual manera se proponen los valores de 40 %, 50 % y 60 % como inventario inicial y el respectivo porcentaje del inventario en tránsito para cubrir el 100 %, 110 % y 120 % de la demanda total para cada uno de los diferentes productos ( $K$ ) en la instancia.

Los diferentes conjuntos de instancias propuestos se detallan en la Tabla 4.1 y en la Figura 4.1 una explicación gráfica sobre la misma.

Tabla 4.1: Descripción de las instancias para el modelo PPL.

Tamaño de instancia	Pedidos ( $ O $ )	Productos diferentes ( $ K $ )	Cantidad de productos en la instancia	Número de tarimas ( $ \mathcal{P}_o $ )	inicialCubierto (%)	transitoCubierto (%)
Pequeñas	70 – 90	104 – 128	2612 – 3928	(1 – 36)	40	[60, 70, 80]
					50	[50, 60, 70]
					60	[40, 50, 60]
Medianas	100 – 120	128 – 140	3803 – 5149	(1 – 35)	40	[60, 70, 80]
					50	[50, 60, 70]
					60	[40, 50, 60]
Grandes	130 – 160	138 – 146	5092 – 6157	(1 – 37)	40	[60, 70, 80]
					50	[50, 60, 70]
					60	[40, 50, 60]

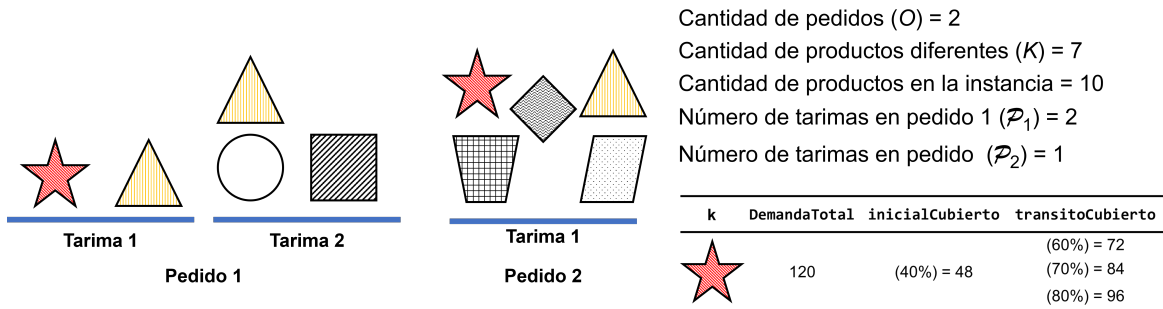


Figura 4.1: Descripción gráfica de las instancias generadas para el modelo PPL.

Posteriormente, con los resultados del modelo PPL, se generan las instancias para validar el desempeño del modelo matemático para el problema del enrutamiento del recolector propuesto en la Sección 4.2.2. El respectivo análisis de este modelo se presenta en la Sección 4.3.3.

#### 4.3.2.2 ANÁLISIS DE LOS RESULTADOS DEL MODELO PPL

Con el fin de evaluar el desempeño del modelo PPL propuesto, para cada conjunto de instancias de la Sección 4.3.2.1, se realizaron seis repeticiones y se consideraron dos criterios de paro para encontrar una solución exacta: el tiempo de cómputo fijado

en 60 segundos y la brecha o gap de optimalidad fijada en 0 %.

Considerando la información disponible, se estableció que el parámetro  $Q$  del modelo PPL descrito en la Sección 4.2.1 sea fijado en 231. Este valor representa el número máximo de tarimas que se recolectarán diariamente y fue determinado con base en la información de la base de datos, en la cual se encontró un total de 11 trabajadores con un horario normal de trabajo, de los cuales por persona durante la jornada laboral recolectan un máximo de 21 tarimas. En cuanto al horizonte de planeación, este fue establecido para 5 días.

La Tabla 3.4 muestra un resumen de los resultados del modelo PPL con un valor de  $\alpha = 20\%$  ya que este valor presentó los mejores resultados en promedio comparados con los valores de 0 %, 10 %, y 30 %. La columna **F / O** describe el número de soluciones factibles y el número de soluciones óptimas encontradas. La columna **No entregados promedio (%)** representa el porcentaje en promedio del total de pedidos que quedaron pendientes de entregar en el horizonte de planeación propuesto. Las columnas **Gap promedio (%)** y **Tiempo promedio (Seg)** representan el porcentaje promedio del gap informado por CPLEX y el promedio de tiempo de ejecución para cada conjunto de instancias. Finalmente, la columna **Entregados antes (%)** corresponde al porcentaje de los pedidos que se entregaron antes de la fecha de entrega.

El (%) de No entregados se calcula de la siguiente manera:

$$\text{No entregados (\%)} = 100 \times \frac{Z_{min}}{|O|}, \quad (4.24)$$

donde  $Z_{min}$  representa el valor de la solución informada por CPLEX.

La Tabla 4.2, se puede observar que para todas las instancias analizadas se logró encontrar una solución factible, y en general, en más del 96 % de los casos se alcanzó la solución óptima. Sin embargo, en aquellas instancias grandes en las que no se reportó la solución óptima en un tiempo de ejecución de 60 segundos, el gap de optimalidad en promedio reportado por CPLEX quedó muy bajo (0.60 %).

En cuanto al impacto del porcentaje de inventario inicial cubierto, para cada tamaño

Tabla 4.2: Resumen de resultados reportados por el modelo PPL con  $Q = 231$ .

Tamaño de instancia	inicialCubierto (%)	transitoCubierto (%)	F / O	No entregados promedio (%)	Entregados antes (%)	Gap promedio (%)	Tiempo promedio (seg)
Pequeñas	40	60	18 / 18	27.89	4.33	0.00	1.61
	40	70	18 / 18	27.83	5.19	0.00	1.22
	40	80	18 / 18	27.72	4.03	0.00	0.89
	50	50	18 / 18	20.33	4.18	0.00	1.00
	50	60	18 / 18	20.33	4.09	0.00	1.00
	50	70	18 / 18	20.22	4.00	0.00	0.89
	60	40	18 / 18	14.00	5.16	0.00	1.28
	60	50	18 / 18	14.00	4.11	0.00	1.17
	60	60	18 / 18	14.00	3.39	0.00	0.89
<b>Totales</b>			<b>162 / 162</b>	<b>20.70</b>	<b>4.27</b>	<b>0.00</b>	<b>1.10</b>
Medianas	40	60	18 / 18	26.11	4.37	0.00	3.44
	40	70	18 / 18	26.11	4.58	0.00	2.22
	40	80	18 / 18	26.11	3.83	0.00	2.22
	50	50	18 / 18	19.11	3.31	0.00	2.11
	50	60	18 / 17	19.06	3.24	0.00	1.56
	50	70	18 / 18	19.06	3.81	0.00	2.56
	60	40	18 / 18	13.11	3.20	0.00	2.50
	60	50	18 / 18	13.11	2.62	0.00	2.33
	60	60	18 / 18	13.11	3.95	0.00	1.72
<b>Totales</b>			<b>162 / 162</b>	<b>19.43</b>	<b>3.63</b>	<b>0.00</b>	<b>2.30</b>
Grandes	40	60	24 / 24	23.75	4.89	0.00	5.13
	40	70	24 / 24	23.75	4.22	0.00	4.88
	40	80	24 / 24	23.75	4.59	0.00	5.25
	50	50	24 / 22	17.21	4.20	0.21	7.25
	50	60	24 / 22	17.17	4.16	0.17	10.21
	50	70	24 / 22	17.13	3.57	0.17	7.58
	60	40	24 / 24	11.63	3.80	0.00	4.54
	60	50	24 / 24	11.58	3.77	0.00	5.46
	60	60	24 / 24	11.58	3.51	0.00	5.71
<b>Totales</b>			<b>216 / 210</b>	<b>17.50</b>	<b>4.06</b>	<b>0.06</b>	<b>6.22</b>

de instancia se observa que en promedio el porcentaje de pedidos no entregados disminuye aproximadamente en un 50 % a medida que se aumenta el porcentaje de inventario inicial cubierto. En las instancias de tamaño pequeño, el valor disminuye de un 27.89 % a un 14 %, en las medianas de un 26.11 % a un 13.11 %, y en las grandes de un 23.75 % a un 11.58 %. Mismo comportamiento que se puede apreciar en el porcentaje de los pedidos entregados antes de tiempo, al menos para las instancias de tamaño mediano y grande, se puede observar que este valor disminuye en aproximadamente un 40 %, mientras que para las instancias pequeñas no se observa un cambio significativo.

Por otro lado, al aumentar el porcentaje de inventario en tránsito para cubrir el 100 %, 110 % y 120 % de la demanda total de cada producto en las diferentes instancias, se observa que el porcentaje promedio de pedidos no entregados y el porcentaje de los pedidos entregados antes de tiempo, se mantienen bastante similares, respectivamente. Es importante destacar que las fechas de entrega de los pedidos se generaron aleatoriamente siguiendo una distribución uniforme, lo cual aseguró una distribución equitativa de las fechas de entrega en el horizonte de planeación. Por lo tanto, con base en los resultados mostrados en la Tabla 4.2, resulta beneficioso mantener una política de inventario inicial lo más alto posible para disminuir el porcentaje de pedidos no entregados. Mientras que, mantener un exceso del producto en tránsito, no impacta positivamente en la entrega de los pedidos antes de tiempo.

Aunque en promedio el porcentaje de pedidos no entregados a tiempo son relativamente altos en comparación con los estándares de la práctica, los cuales suelen estar por debajo del 5 %, es importante tener en cuenta que estos valores se ven afectados debido a la limitada disponibilidad de información real necesaria para ejecutar el modelo PPL propuesto. Sin embargo, utilizando la información real disponible (histórico de un mes de la composición de los pedidos realizados y la demanda total de cada tipo de producto) junto con información generada aleatoriamente (periodicidad y cantidad de llegada del producto en tránsito al CEDIS y fechas de entrega de los pedidos), el modelo PPL propuesto demuestra su capacidad para generar so-

luciones de alta calidad en menos de un minuto, incluso en las instancias de tamaño real.

Con el fin de determinar el impacto que tiene el parámetro  $Q$  en la minimización de los pedidos no entregados a tiempo, se ejecutaron nuevamente todas las instancias considerando aumentar hasta 1 y 3 trabajadores, es decir, fijar el parámetro  $Q$  en 252 y 273. Los resultados de la anterior experimentación son mostrados en la Tabla 4.3.

De acuerdo con la información mostrada en la Tabla 4.3, se puede observar que el porcentaje en promedio de pedidos no entregados se mantiene muy similar a pesar de aumentar más trabajadores. De igual manera, no hay cambios significativos en cuanto al comportamiento en el aumento del porcentaje de inventario inicial cubierto y porcentaje de inventario en tránsito cubierto. Por ende, de acuerdo con la información y los supuestos bajo los cuales se generaron los parámetros de entrada del modelo PPL, aumentar la cantidad de tarimas a recolectar por día, es decir, aumentar el número de trabajadores no es factor que impacta considerablemente en la minimización de los pedidos no entregados.

Para llevar a cabo un análisis más completo del modelo PPL, es importante considerar diferentes escenarios que pueden influir en el comportamiento de los pedidos no entregados. Hasta el momento, se ha estudiado el modelo bajo el supuesto de que los productos en tránsito no llegan en el momento requerido. Ahora, se analizará una variante en la que se asume que se tiene producto suficiente disponible, para lo cual se estableció la fecha de entrega de todos los pedidos el último día del horizonte de planeación. Se continua con los dos criterios de paro para encontrar una solución exacta: el tiempo de cómputo fijado en 60 segundos y la brecha o gap de optimalidad fijada en 0%. En la Tabla 4.4 se muestran los resultados de este variante considerando los valores de 231 y 252 para el parámetro  $Q$ .

De acuerdo con los resultados mostrados en la Tabla 4.4, se puede observar que el modelo PPL encuentra una solución factible para todas las instancias. Aunque, presenta una mayor dificultad en reportar una solución óptima, en un tiempo de

Tabla 4.3: Resumen de resultados reportadas por el modelo PPL con  $Q = 252$  y  $Q = 273$ .

Tamaño de instancia	inicialCubierto (%)	transitoCubierto (%)	Q = 252				Q = 273					
			F / O	No entregados promedio (%)	Entregados antes (%)	Gap promedio (%)	Tiempo promedio (seg)	F / O	No entregados promedio (%)	Entregados antes (%)	Gap promedio (%)	Tiempo promedio (seg)
Pequeñas	40	60	18 / 18	27.89	4.23	0.00	1.22	18 / 18	27.89	4.72	0.00	1.78
	40	70	18 / 18	27.83	4.42	0.00	0.94	18 / 18	27.83	4.52	0.00	1.28
	40	80	18 / 18	27.72	3.84	0.00	0.61	18 / 18	27.72	4.23	0.00	0.72
	50	50	18 / 18	20.33	4.70	0.00	0.94	18 / 18	20.33	3.83	0.00	1.17
	50	60	18 / 18	20.33	4.53	0.00	0.83	18 / 18	20.33	4.96	0.00	0.83
	50	70	18 / 18	20.22	3.74	0.00	0.39	18 / 18	20.22	3.91	0.00	0.67
	60	40	18 / 18	14.00	5.24	0.00	1.11	18 / 18	14.00	5.08	0.00	1.39
	60	50	18 / 18	14.00	5.00	0.00	0.94	18 / 18	14.00	5.32	0.00	1.00
	60	60	18 / 18	14.00	4.35	0.00	0.78	18 / 18	14.00	4.92	0.00	1.06
<b>Totales</b>			<b>162 / 162</b>	<b>20.70</b>	<b>4.47</b>	<b>0.00</b>	<b>0.86</b>	<b>162 / 162</b>	<b>20.70</b>	<b>4.63</b>	<b>0.00</b>	<b>1.10</b>
Medianas	40	60	18 / 18	26.11	4.99	0.00	2.67	18 / 18	26.11	3.83	0.00	3.28
	40	70	18 / 18	26.11	4.17	0.00	2.61	18 / 18	26.11	4.85	0.00	3.33
	40	80	18 / 18	26.11	3.21	0.00	2.17	18 / 18	26.11	4.17	0.00	2.06
	50	50	18 / 18	19.11	3.87	0.00	3.06	18 / 18	19.11	4.87	0.00	3.67
	50	60	18 / 18	19.06	3.68	0.00	2.72	18 / 18	19.06	3.49	0.00	1.67
	50	70	18 / 18	19.06	3.56	0.00	1.94	18 / 18	19.06	3.37	0.00	1.56
	60	40	18 / 18	13.11	3.60	0.00	2.72	18 / 18	13.11	3.55	0.00	2.61
	60	50	18 / 18	13.11	3.26	0.00	1.83	18 / 18	13.11	3.43	0.00	2.39
	60	60	18 / 18	13.11	3.66	0.00	1.67	18 / 18	13.11	2.73	0.00	1.83
<b>Totales</b>			<b>162 / 162</b>	<b>19.43</b>	<b>3.76</b>	<b>0.00</b>	<b>2.38</b>	<b>162 / 162</b>	<b>19.43</b>	<b>3.78</b>	<b>0.00</b>	<b>2.49</b>
Grandes	40	60	24 / 24	23.75	5.57	0.00	5.33	24 / 24	23.75	5.38	0.00	6.21
	40	70	24 / 24	23.75	4.55	0.00	4.25	24 / 24	23.75	5.12	0.00	5.29
	40	80	24 / 23	23.75	5.35	0.25	7.79	24 / 24	23.75	4.03	0.00	5.29
	50	50	24 / 24	17.17	3.22	0.00	7.63	24 / 23	17.17	3.26	0.17	7.83
	50	60	24 / 23	17.13	3.71	0.17	7.29	24 / 24	17.13	3.98	0.00	5.38
	50	70	24 / 24	17.13	3.92	0.00	3.38	24 / 23	17.13	3.71	0.17	6.25
	60	40	24 / 24	11.29	2.79	0.00	4.29	24 / 24	11.29	2.66	0.00	3.33
	60	50	24 / 23	11.29	2.85	0.25	8.13	24 / 24	11.25	3.24	0.00	4.08
	60	60	24 / 24	11.25	3.56	0.00	5.33	24 / 24	11.25	2.75	0.00	6.75
<b>Totales</b>			<b>216 / 213</b>	<b>17.39</b>	<b>3.89</b>	<b>0.07</b>	<b>5.94</b>	<b>216 / 214</b>	<b>17.38</b>	<b>3.74</b>	<b>0.04</b>	<b>5.60</b>



Tabla 4.4: Resumen de resultados reportados por el modelo PPL para la variante propuesta con  $Q = 231$  y  $Q = 252$ .

Tamaño de instancia	inicialCubierto (%)	transitoCubierto (%)	Q = 231					Q = 252				
			F / O	No entregados promedio (%)	Entregados antes (%)	Gap promedio (%)	Tiempo promedio (seg)	F / O	No entregados promedio (%)	Entregados antes (%)	Gap promedio (%)	Tiempo promedio (seg)
Pequeñas	40	60	18 / 18	0.00	14.44	0.00	36.72	18 / 18	0.00	12.15	0.00	29.83
	40	70	18 / 18	0.00	15.49	0.00	3.56	18 / 18	0.00	15.49	0.00	3.44
	40	80	18 / 18	0.00	14.38	0.00	3.00	18 / 18	0.00	15.21	0.00	3.28
	50	50	18 / 18	0.00	12.99	0.00	27.06	18 / 18	0.00	10.28	0.00	23.50
	50	60	18 / 18	0.00	14.86	0.00	2.72	18 / 18	0.00	11.18	0.00	2.67
	50	70	18 / 18	0.00	15.42	0.00	1.94	18 / 18	0.00	11.67	0.00	1.94
	60	40	18 / 18	0.00	12.64	0.00	30.22	18 / 17	5.00	11.70	5.56	28.61
	60	50	18 / 18	0.00	11.94	0.00	1.56	18 / 18	0.00	9.72	0.00	1.33
	60	60	18 / 18	0.00	11.67	0.00	1.33	18 / 18	0.00	12.57	0.00	0.94
<b>Totales</b>			<b>162 / 162</b>	<b>0.00</b>	<b>13.76</b>	<b>0.00</b>	<b>12.01</b>	<b>162 / 161</b>	<b>0.56</b>	<b>12.22</b>	<b>0.62</b>	<b>10.62</b>
Medianas	40	60	18 / 3	70.94	24.23	86.28	58.89	18 / 4	69.22	18.97	77.78	55.11
	40	70	18 / 8	5.22	23.25	46.94	39.28	18 / 14	0.22	21.26	22.22	26.06
	40	80	18 / 11	0.83	23.51	34.50	34.78	18 / 12	0.33	22.34	33.33	26.83
	50	50	18 / 9	21.39	20.84	49.94	49.67	18 / 15	9.94	17.20	16.67	44.83
	50	60	18 / 15	0.39	21.59	13.89	19.89	18 / 18	0.00	20.86	0.00	9.50
	50	70	18 / 18	0.22	21.66	0.00	8.78	18 / 18	0.00	20.20	0.00	7.22
	60	40	18 / 15	10.83	18.51	16.67	46.39	18 / 12	26.67	16.44	33.33	50.89
	60	50	18 / 18	0.22	19.74	0.00	6.56	18 / 18	0.00	18.38	0.00	5.78
	60	60	18 / 18	0.22	19.74	0.00	6.06	18 / 18	0.00	18.03	0.00	4.39
<b>Totales</b>			<b>162 / 115</b>	<b>12.25</b>	<b>21.30</b>	<b>27.58</b>	<b>30.03</b>	<b>162 / 129</b>	<b>11.82</b>	<b>19.46</b>	<b>20.37</b>	<b>25.62</b>
Grandes	40	60	24 / 12	6.75	29.03	8.17	41.67	24 / 8	22.58	28.14	34.46	49.50
	40	70	24 / 17	6.63	28.53	5.67	30.33	24 / 14	6.00	26.80	20.13	37.96
	40	80	24 / 16	6.67	29.20	7.13	30.58	24 / 15	7.83	27.53	18.00	35.88
	50	50	24 / 21	6.42	27.65	1.71	19.83	24 / 18	3.75	26.96	8.29	33.13
	50	60	24 / 22	6.38	28.13	1.63	19.67	24 / 17	3.58	26.80	11.33	29.08
	50	70	24 / 22	6.33	28.41	1.13	15.83	24 / 21	3.67	27.34	6.29	21.42
	60	40	24 / 24	6.29	24.12	0.00	10.75	24 / 23	7.46	25.25	4.17	24.08
	60	50	24 / 23	6.33	28.06	0.58	10.83	24 / 22	3.42	27.75	2.67	15.92
	60	60	24 / 24	6.29	28.08	0.00	6.96	24 / 23	3.33	26.06	2.08	13.67
<b>Totales</b>			<b>216 / 181</b>	<b>6.45</b>	<b>27.91</b>	<b>2.89</b>	<b>20.72</b>	<b>216 / 161</b>	<b>6.85</b>	<b>26.94</b>	<b>11.94</b>	<b>28.96</b>

ejecución de un minuto.

En cuanto al impacto del porcentaje de inventario inicial cubierto, para cada tamaño de instancia se observa que, en promedio hay una disminución de los pedidos entregados antes de tiempo, a medida que aumentan dichos porcentajes. Este comportamiento es consistente con lo observado bajo el supuesto de que los productos en tránsito no llegan en el momento requerido. Por lo tanto, mantener un exceso de producto en tránsito no tiene un impacto positivo en la entrega anticipada de los pedidos.

Para las instancias pequeñas, el modelo PPL logra encontrar una solución óptima para todas las instancias analizadas cuando el parámetro  $Q=231$ . De igual manera, el porcentaje de pedidos no entregados es de cero, lo cual indica que, para este tamaño de instancias el modelo PPL logra encontrar una solución óptima en la que todos los pedidos son entregados a tiempo. No obstante, para la instancia en la que no se encuentra la solución óptima cuando  $Q=252$ , el porcentaje en promedio de gap de optimalidad no es muy alto.

Para las instancias medianas, se observa que el modelo PPL presenta una mayor dificultad para alcanzar la solución óptima, en especial para aquellas instancias en donde se tiene un porcentaje de inventario inicial bajo y el porcentaje del producto en tránsito cubre el 100 %, de la demanda total de cada producto en las diferentes instancias analizadas. No obstante, a medida que se aumenta el porcentaje de inventario inicial cubierto y tránsito cubierto, contribuye a la minimización de los pedidos no entregados y aun mejor rendimiento del desempeño del modelo PPL, dado a que es posible encontrar una solución óptima con un porcentaje de pedidos no entregados menores al 1 %.

Finalmente, para las instancias de tamaño grande, de igual manera, no se logran alcanzar todas las soluciones óptimas de las instancias analizadas. No obstante, en aquellas instancias donde se logra alcanzar la optimalidad cuando  $Q=231$ , el porcentaje de pedidos no entregados es en promedio de un 6 %. Mientras que con  $Q=252$ , a pesar de que no se encuentran la solución óptima para todas las instancias, en

aquellas instancias donde se logra alcanzar la optimalidad el porcentaje de pedidos no entregados es de cero, es decir, el modelo PPL para algunas instancias logra encontrar una solución óptima en la que todos los pedidos son entregados a tiempo.

De acuerdo con lo anteriormente expuesto, bajo el enfoque de que se tiene producto suficiente disponible para cubrir la demanda semanal, aumentar la cantidad de tarimas a recolectar por día, es decir, aumentar el número de trabajadores es un factor que afecta tanto el desempeño del modelo PPL, así como la minimización de los pedidos no entregados. Por ende, se destaca la importancia de la generación de rutas de recolección eficientes. Aunque se disponga de suficiente producto para cubrir la demanda semanal, el tiempo necesario para recolectar las tarimas de los pedidos dentro de los plazos establecidos puede convertirse en un obstáculo.

### 4.3.3 ENRUTAMIENTO DEL RECOLECTOR (PER) - SEGUNDA ETAPA

#### 4.3.3.1 DATOS

Como se menciona en la Sección 4.2, la segunda etapa de la metodología propuesta consiste en el desarrollo de un modelo matemático para la generación de las rutas de recolección individuales óptimas con base en el peso de las cajas de los diferentes productos contenidos en cada tarima de los lotes obtenidos en la primera etapa. Los lotes representan las tarimas que deben ser recolectadas diariamente para llevar a cabo la entrega de los pedidos que fueron determinados como completos. Cabe mencionar que, en esta etapa, cada producto tiene asignada una ubicación de almacenamiento predeterminada, las cuales fueron determinadas con base en la información de la base de datos.

En la Tabla 4.5, se muestran las características de las instancias utilizadas en la ejecución del modelo PER. Dichas características se realizaron con base en los resultados del modelo PPL considerando el parámetro  $Q = 231$ . Los valores de las columnas

Tabla 4.5: Descripción de las instancias para el modelo PER con  $Q = 231$ 

Tamaño de instancia	inicialCubierto (%)	transitoCubierto (%)	Lotes diarios - Horizonte de planeación									
			1		2		3		4		5	
			Número de ubicaciones a visitar	Tarimas asignadas a recolectar	Número de ubicaciones a visitar	Tarimas asignadas a recolectar	Número de ubicaciones a visitar	Tarimas asignadas a recolectar	Número de ubicaciones a visitar	Tarimas asignadas a recolectar	Número de ubicaciones a visitar	Tarimas asignadas a recolectar
		$ I_p $	$ P_d $	$ I_p $	$ P_d $	$ I_p $	$ P_d $	$ I_p $	$ P_d $	$ I_p $	$ P_d $	
Pequeñas	40	60	9-10	19-218	7-11	31-116	8-11	39-100	7-13	45-138	9-13	80-198
	40	70	8-11	24-209	8-11	27-111	8-11	20-99	9-13	45-152	9-13	83-215
	40	80	8-12	40-212	8-11	39-152	8-11	26-112	8-13	48-150	9-13	75-164
	50	50	9-12	37-225	8-11	63-169	8-11	41-101	8-13	53-165	9-13	60-164
	50	60	9-12	128-220	8-12	56-161	8-12	35-119	8-13	43-162	9-13	56-169
	50	70	9-12	32-225	8-12	56-178	9-11	38-160	9-13	62-137	9-13	72-180
	60	40	9-12	141-230	9-11	83-218	8-11	42-155	9-13	41-168	9-13	66-161
	60	50	9-12	39-230	9-11	83-222	7-12	45-161	8-13	44-148	9-13	45-158
	60	60	9-12	42-223	9-11	79-191	9-11	51-158	8-13	41-127	9-13	52-162
Medianas	40	60	9-12	159-216	8-12	76-164	9-13	63-134	9-13	72-187	10-13	94-227
	40	70	9-12	103-223	9-11	88-177	9-13	56-169	9-13	45-220	10-13	116-230
	40	80	9-12	92-216	8-13	81-219	8-12	63-163	9-13	59-182	9-13	116-230
	50	50	10-13	172-226	9-12	114-225	8-13	79-168	9-13	62-188	10-13	116-218
	50	60	8-12	118-226	8-12	122-221	9-13	90-176	10-13	84-205	9-13	97-220
	50	70	9-13	106-222	9-12	126-225	9-13	85-187	9-13	78-198	9-13	80-214
	60	40	9-12	179-231	10-12	137-231	8-12	111-210	9-13	41-215	10-13	63-231
	60	50	9-12	148-231	9-12	163-231	9-13	104-216	9-13	85-209	10-13	102-231
	60	60	9-12	148-223	9-12	157-231	9-12	101-231	9-13	63-223	10-13	68-230
Grandes	40	60	9-13	152-226	8-12	122-227	9-12	89-199	10-14	103-228	10-13	119-231
	40	70	9-13	129-231	9-12	120-219	9-12	96-193	9-14	114-231	10-13	134-231
	40	80	9-13	127-229	9-12	124-216	8-12	69-227	9-14	93-229	10-13	160-231
	50	50	9-13	158-231	9-13	164-231	9-12	133-231	10-14	146-231	10-13	158-231
	50	60	9-13	173-231	9-12	153-231	10-12	118-231	10-14	137-231	10-13	157-231
	50	70	9-13	166-231	9-12	143-231	9-12	139-231	10-14	132-231	10-13	169-231
	60	40	9-13	193-231	9-12	209-231	9-13	175-231	10-14	147-231	10-13	190-231
	60	50	9-13	188-231	9-13	183-231	10-13	167-231	10-14	182-231	10-13	193-231
	60	60	10-13	194-231	9-13	201-231	10-13	187-231	10-14	111-231	10-13	170-231

**Número de ubicaciones a visitar**  $|I_\rho|$ , representan el rango de las ubicaciones de almacenamiento máximo que se visitó por cada tarima de los lotes generados. Los valores de las columnas **Tarimas asignadas a recolectar**  $|\mathcal{P}_d|$ , representan el rango de tarimas que fueron asignadas a recolectar por cada lote generado. Por ejemplo, para el conjunto de instancias de tamaño pequeño, con un (%) **inicialCubierto** = 40 y un (%) **transitoCubierto** = 60, para el lote generado en el día 1, las instancias que se analizaron tienen un mínimo de 19 y máximo 218 tarimas a recolectar y por cada tarima recolectada se visitó un máximo entre 9 y 10 ubicaciones de almacenamiento, es decir, cada tarima recolectada tenía asignado un máximo entre 9 y 10 productos diferentes.

#### 4.3.3.2 ANÁLISIS DE LOS RESULTADOS DEL MODELO (PER)

Con el fin de evaluar el desempeño del modelo PER propuesto se consideraron dos criterios de paro para encontrar una solución exacta: el tiempo de cómputo fijado en 60 segundos y la brecha o gap de optimalidad fijada en 0%.

En la Tabla 4.6, se muestra el resumen de los resultados del desempeño del modelo PER propuesto para los diferentes conjuntos de instancias presentados en Tabla 4.5. La columna **Promedio solución encontrada (seg)** representa la solución reportada por CPLEX para cada uno de los días en el horizonte de planeación. La columna **Promedio  $\Delta$  tarimas (%)** representa el porcentaje promedio de tarimas que contienen productos donde el peso de las cajas es igual. Las columnas **Gap promedio (%)** y **Tiempo promedio (Seg)** representan el porcentaje promedio del gap informado por CPLEX y el promedio de tiempo de ejecución para cada conjunto de instancias en general para los lotes generados. El  $\Delta$  tarimas (%) se calcula de la siguiente manera:

$$\Delta \text{tarimas}(\%) = 100 \times \frac{\text{sumaTarimas}}{|\mathcal{P}_d|}, \quad (4.25)$$

donde *sumaTarimas* es la suma de las tarimas que contienen productos con cajas de igual peso.

Tabla 4.6: Resumen resultados modelo PER con  $Q = 231$ .

Tamaño de instancia	inicialCubierto (%)	transitoCubierto (%)	Lotes diarios - Horizonte de planeación										Gap promedio (%)	Tiempo promedio (Seg)
			1		2		3		4		5			
			Promedio solución encontrada (seg)	$\Delta$ tarimas (%)	Promedio solución encontrada (seg)	$\Delta$ tarimas (%)	Promedio solución encontrada (seg)	$\Delta$ tarimas (%)	Promedio solución encontrada (seg)	$\Delta$ tarimas (%)	Promedio solución encontrada (seg)	$\Delta$ tarimas (%)		
Pequeñas	40	60	26,927.87	21.69	14,437.87	17.81	12,351.56	21.61	16,923.91	22.17	22,684.62	22.54	0	0
	40	70	24,599.38	21.50	13,582.86	19.07	12,048.36	20.33	18,136.44	22.31	23,473.50	23.02	0	0
	40	80	24,403.29	20.91	15,403.85	18.67	12,073.37	21.8	18,322.36	22.48	22,027.12	22.26	0	0
	50	50	32,816.19	21.11	21,046.53	18.63	13,368.42	21.67	17,320.16	23.3	22,020.01	22.87	0	0
	50	60	34,310.60	21.8	19,837.65	18.26	14,185.09	20.17	18,451.96	22.15	20,618.25	22.78	0	0
	50	70	32,275.68	21.65	21,347.04	18.76	15,515.90	21.13	17,805.26	21.46	20,052.47	23.35	0	0
	60	40	39,339.94	21.69	27,943.61	19.33	14,982.30	20.26	16,835.70	21.96	21,178.95	23.06	0	0
	60	50	38,651.15	22.06	27,591.96	19.20	17,030.05	19.65	17,182.47	22.33	19,806.15	21.74	0	0
60	60	36,366.26	21.44	27,420.57	19.30	18,429.32	20.50	17,691.90	22.15	20,028.11	22.06	0	0	
Medianas	40	60	35,853.08	21.07	22,963.01	17.93	16,895.75	20.43	24,851.84	20.61	31,796.44	23.24	0	0
	40	70	35,004.73	21.3	21,092.09	17.59	19,428.42	20.22	23,552.94	21.41	33,483.88	22.09	0	0
	40	80	31,612.27	20.54	25,081.18	18.26	20,029.74	19.74	24,757.23	20.67	31,301.35	22.57	0	0
	50	50	40,458.70	21.06	31,560.33	18.22	21,700.57	19.39	26,037.34	21.44	32,486.66	22.67	0	0
	50	60	38,802.76	20.26	31,434.36	17.98	23,962.56	21.04	27,152.28	21.39	31,260.16	21.7	0	0
	50	70	37,270.53	21.44	32,898.66	17.94	24,985.96	18.67	27,424.08	21.35	29,788.12	23.22	0	0
	60	40	42,883.95	20.48	39,075.22	19.94	31,024.74	20.13	26,734.42	20.00	31,398.88	21.35	0	0
	60	50	41,609.27	20.76	39,122.46	17.98	31,265.06	19.26	28,784.13	22.15	30,330.95	22.56	0	0
60	60	39,629.89	20.91	39,098.25	18.85	31,928.77	19.33	29,508.92	20.09	30,280.20	23.09	0	0	
Grandes	40	60	40,640.88	21.86	34,651.92	17.99	27,462.98	18.47	31,396.99	20.79	42,542.49	21.93	0	0
	40	70	39,162.76	21.56	32,875.34	17.51	28,988.15	18.85	33,709.34	21.63	41,803.53	21.39	0	0
	40	80	38,490.98	20.6	34,286.78	18.56	27,927.67	19.14	32,832.02	20.89	43,118.82	21.88	0	0
	50	50	42,629.80	21.24	40,723.97	19.86	37,269.26	18.21	37,468.18	20.65	44,382.12	21.65	0	0
	50	60	40,820.59	21.13	41,685.50	19.08	38,057.88	19.44	38,092.83	20.71	43,336.78	21.11	0	0
	50	70	41,766.07	21.51	39,672.80	19.43	37,215.81	17.96	38,752.08	20.50	44,215.92	21.99	0	0
	60	40	44,689.69	22.04	45,046.14	20.71	43,836.19	20.42	43,005.38	20.26	45,068.63	21.36	0	0
	60	50	44,722.90	21.92	44,407.30	20.83	43,929.22	20.85	43,535.11	19.69	45,129.09	21.35	0	0
60	60	44,309.99	21.54	45,146.19	19.96	44,020.21	21.44	42,644.99	20.21	45,546.86	21.35	0	0	

Los resultados presentados en la Tabla 4.6 muestran que todas las instancias analizadas para cada lote generado encontraron la solución óptima en un tiempo computacional inferior a un segundo, incluso en las instancias de tamaño real. De igual manera se observa que en promedio el porcentaje de tarimas que contienen productos con pesos idénticos en sus cajas es bajo, en general es aproximadamente menos del 24 %. Lo anterior implica que el proceso de optimización no se enfrenta a mayores dificultades al determinar cuál producto debe ser recolectado en primer lugar cuando se presenta un empate entre los pesos de las cajas de los productos contenidos en cada tarima. En consecuencia, se simplifica la toma de decisiones en el proceso de optimización.

Con el fin de evaluar las rutas generadas con el modelo PER y los tiempos de viaje obtenidos, se realiza una comparación de los resultados del tiempo de viaje total en función de dos métodos diferentes de recolección que se proponen con el fin de simular los recorridos que se realizan en la práctica. El primer método, llamado `Ruteo_Recolector` las rutas de recolección individual de cada tarima se generan con base en las rutas alternas que eligen los recolectores de manera empírica para preservar en buenas condiciones las cajas de los productos paletizados en las tarimas, es decir el recorrido se realiza considerando un orden decreciente del peso de las cajas de los productos contenidos en cada tarima. Para el segundo método, debido a que no se conoce el tipo de algoritmo que utiliza el WMS que maneja la empresa para generar las rutas de recolección individual de las tarimas, se propone un algoritmo basado en el criterio del vecino más cercano entre ubicaciones (sin contemplar el factor del peso de la caja del producto), como se describe en la Sección 3.3.1. En la presente sección se hace referencia a este método como `Ruteo_WMS`.

Dado que el comportamiento en promedio de los tiempos totales de viaje obtenidos bajo los tres métodos es muy similar para los diferentes conjuntos de instancias mencionadas en la Tabla 4.5, se seleccionó para cada tamaño el conjunto de instancias considerando (%) `inicialCubierto` = 60 y (%) `transitoCubierto` = 60 para mostrar la comparación de los resultados del promedio de tiempo de viaje

total de las rutas individuales generadas en función del modelo PER, el método `Ruteo_Recolector` y `Ruteo_WMS`. Dicha comparación se presenta en la Figura 4.2.

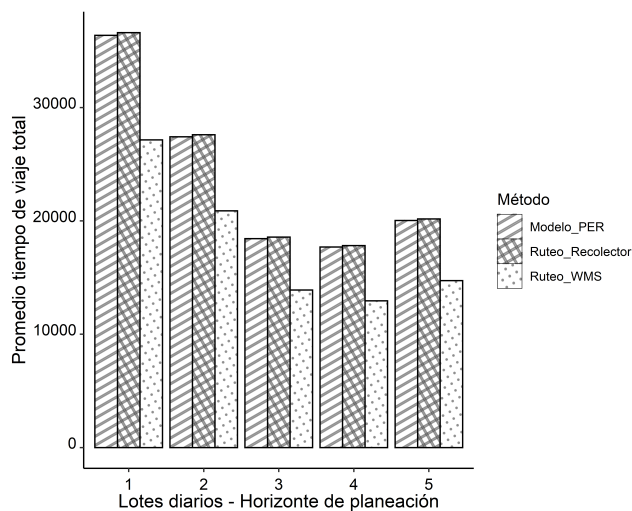
En la Figura 4.2, se puede observar que utilizar el modelo PER propuesto genera en promedio los mejores tiempos totales de viaje considerando el factor de peso del producto.

A pesar de que los tiempos de viaje totales promedio considerando el método del `Ruteo_Recolector` son muy similares a los obtenidos mediante el modelo PER, esto se debe principalmente al bajo porcentaje de tarimas que contienen productos con pesos idénticos en sus cajas, como se mencionó anteriormente. Por ende, en la mayoría de las tarimas a recolectar diariamente tanto el modelo PER como el enfoque del método `Ruteo_Recolector` generan rutas de recolección basadas en un orden decreciente del peso de las cajas de los productos a recolectar en cada tarima, lo que resulta en rutas similares.

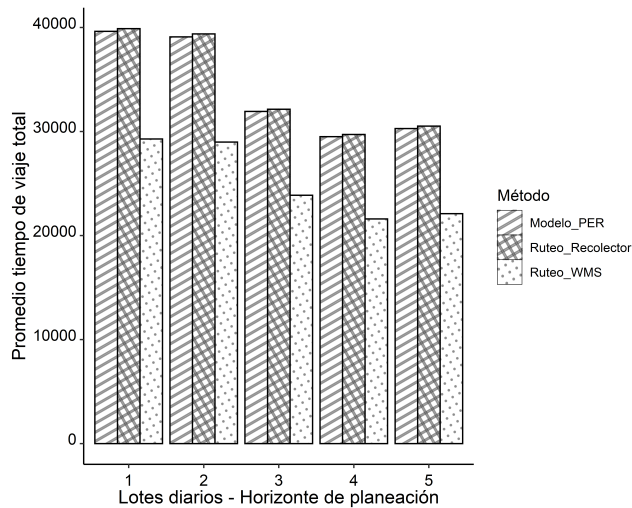
Sin embargo, en la práctica, el método `Ruteo_Recolector` es un proceso empírico que requiere experiencia y tiempo adicional para planificar la ruta de manera óptima. Por otro lado, el modelo PER ofrece un secuenciamiento óptimo por tarimas con un tiempo de ejecución significativamente menor. Aunque el porcentaje de mejora en los tiempos de viaje totales promedio con el modelo PER es aproximadamente del 1 % en comparación con el método `Ruteo_Recolector`, se logra obtener una solución óptima sin necesidad de experiencia adicional para generar las rutas de recolección.

En otras palabras, mientras que el método `Ruteo_Recolector` se basa en la experiencia de los recolectores y requiere tiempo adicional para su implementación óptima y de acuerdo Pinto y Nagano (2022), las decisiones que se toman basadas únicamente en experiencias pueden resultar en una pérdida de tiempo y costo para los almacenes, el modelo PER propuesto es un algoritmo capaz de generar soluciones óptimas en un tiempo de ejecución bajo. Esto significa que no se requiere experiencia adicional ni se invierte tiempo adicional en la generación de las rutas de recolección, lo que representa una ventaja significativa del modelo PER en términos de eficiencia y eficacia.

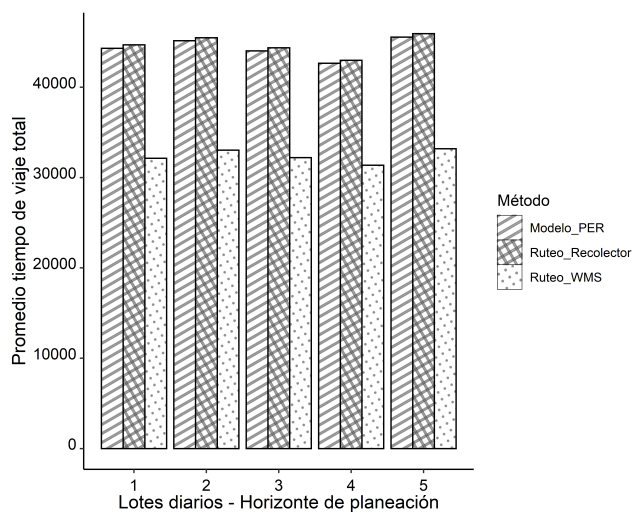




(a) Instancias pequeñas



(b) Instancias medianas



(c) Instancias grandes

Figura 4.2: Promedio tiempo total de viaje.

Por otra parte, al comparar los resultados del modelo PER con el método `Ruteo_WMS`, en el cual se generan las rutas de recolección por tarima considerando un secuenciamiento donde se visitan las ubicaciones (donde están alojados los productos contenidos en cada tarima) más cercanas con base en el tiempo de viaje más corto entre estas y el depósito, se observa que aunque el promedio del tiempo total de viaje aumenta aproximadamente en un 23 %, el método `Ruteo_WMS` no considera el peso de los productos.

Este aspecto es crucial, ya que el peso de los productos juega un papel fundamental en la preservación de buenas condiciones del producto final (pedido). Al no tener en cuenta este criterio, el método `Ruteo_WMS` permite un mayor margen de tiempo promedio para realizar la recolección, sin embargo, afecta el producto final. Por ende, el tiempo que se tiene destinado en promedio para realizar la recolección sin afectar el producto final se aumenta, lo que conlleva, en el caso de la empresa, a utilizar mano de obra adicional para cumplir con las entregas en tiempo y forma.

En conclusión, a pesar de que el método `Ruteo_WMS` ofrece tiempos de viaje más bajos, su falta de consideración del peso de los productos es claramente evidente en la necesidad de recurrir a más personal para cumplir con las entregas en tiempo y forma. Por lo tanto, se destaca la importancia del uso del modelo PER, que combina eficiencia en tiempo de ejecución, consideración del peso de los productos y generación de rutas de recolección óptimas.

## CAPÍTULO 5

# CONCLUSIONES

---

Las empresas se ven obligadas a tener una correcta gestión de la cadena de suministro para poder cumplir con las exigencias de los consumidores y brindar un servicio de calidad, por lo cual, todas las actividades que están involucradas deberán estar alineadas a este fin manteniendo un equilibrio entre costos y eficiencia. Para algunas empresas contar con almacenes, les proporciona agilidad, flexibilidad y competitividad al momento de satisfacer las demandas del mercado, lo que conlleva a que, en ocasiones, puede significar pérdidas o ganancias el contar con este tipo de estructuras ya que el almacenamiento contribuye con aproximadamente el 20 % de los costos logísticos de las compañías. Por ende, que su gestión es una de las actividades más importantes para el funcionamiento de una organización y cualquier mejoramiento que permita disminuir los costos en los que incurra, contribuye en la eficiencia y eficacia de la cadena de suministro.

De acuerdo con lo anterior, una estrategia clave para abordar el problema mencionado anteriormente es optimizar aquellas operaciones que permiten minimizar el tiempo de viaje del recolector dentro del almacén. Dentro de estas operaciones se destacan, el procesamiento de pedidos por lotes, la asignación de ubicaciones de almacenamiento y la planificación de la ruta del recolector. Al abordar estos desafíos en conjunto, ya sea de manera secuencial o simultánea, los gerentes pueden tomar decisiones que conducirán a operaciones más eficientes y rentables, lo que a su vez resultará en un mejor servicio al cliente.

En consecuencia, el objetivo de este trabajo doctoral está enfocado en minimizar las entregas tardías y el tiempo de viaje en la recolección de pedidos a través de diferentes formulaciones matemáticas y desarrollo de algoritmos eficientes, considerando criterios como las fechas de entrega de los pedidos, nivel de inventario, demanda, frecuencia de solicitud y el peso de las cajas de los productos para un caso de estudio. Para el cumplimiento del objetivo general, se propuso solucionar la combinación de dos problemas. La primera combinación está relacionada con los problemas de asignación de ubicación de almacenamiento y el enrutamiento de recolector. La segunda está relacionada con el problema del procesamiento de pedidos por lotes y el enrutamiento de recolector.

Los resultados de los métodos propuestos para solucionar la combinación relacionada con los problemas de asignación de ubicación de almacenamiento y el enrutamiento de recolector, resaltan la importancia de integración de ambas operaciones. Si bien en los trabajos previos generalmente han abordado estos problemas de manera independiente, asumiendo que las decisiones de ubicación de almacenamiento son a largo plazo y resistentes al cambio, las consideraciones prácticas en ciertas industrias y negocios pueden requerir enfoques más dinámicos. Factores como la estacionalidad y los ciclos de vida del producto pueden influir en la demanda del producto y requerir revisiones periódicas de las decisiones sobre la ubicación del almacenamiento de los productos. Aunque las preocupaciones sobre los costos de reubicación son un argumento para evitar reasignaciones periódicas de los productos a las ubicaciones de almacenamiento, los hallazgos experimentales demuestran los beneficios de ajustar las decisiones de almacenamiento y enrutamiento para minimizar el tiempo de viaje en el proceso de preparación de pedidos. Por ende, se considera importante que los tomadores de decisiones conozcan y consideren la interdependencia entre los dos problemas de manera simultánea.

Por otra parte, los resultados de las herramientas propuestas para abordar la combinación de los problemas de procesamiento de pedidos por lotes y enrutamiento del recolector resaltan la importancia de una adecuada gestión de las políticas de inven-

tario. Según los hallazgos experimentales, se observó que mantener un nivel inicial de inventario lo más alto posible resulta beneficioso para reducir el porcentaje de pedidos no entregados. Por otra parte, mantener un exceso de producto en tránsito no tiene un impacto positivo en la entrega temprana de los pedidos. Lo anterior, es un criterio importante que considerar puesto que, en la práctica, por lo general con el fin de mejorar el proceso de producción se tiende a producir en exceso, lo que puede resultar en desfases en los tiempos de entrega al CEDIS. Por otro lado, producir exactamente lo que se demanda implica un proceso costoso debido a los cambios de línea requeridos para satisfacer las cantidades exactas. Por ende, encontrar un equilibrio adecuado entre la cantidad de inventario a mantener y la gestión de los niveles de producción para garantizar tanto la eficiencia como la entrega puntual de los pedidos. Este desafío implica evaluar cuidadosamente los costos asociados y los beneficios esperados de producir más o producir exactamente lo necesario. Además, la llegada de los productos en tránsito está sujeta a diversos factores, como una programación inadecuada en la fabricación, fallas en la línea de producción y daños del producto debido a la manipulación

De igual manera, optimizar la generación de las rutas de recolección de los pedidos, es un factor importante, en la minimización de los pedidos no entregados a tiempo. Aunque se cuente con suficiente producto disponible para satisfacer los pedidos, el tiempo necesario para recolectar las tarimas de los pedidos dentro de los plazos establecidos puede convertirse en un factor limitante. Dado que la capacidad diaria de recolección de tarimas es limitada, optimizar las rutas de recolección permite aumentar la capacidad de recolección diaria sin requerir recursos adicionales y mantener en buenas condiciones el producto final (pedido). Por lo tanto, se destaca la importancia del uso del modelo PER, que combina eficiencia en tiempo de ejecución, consideración del peso de los productos y generación de rutas de recolección óptimas.

De acuerdo con lo anteriormente expuesto, el desarrollo de algoritmos exactos y metaheurísticos, la programación de los pedidos, la ubicación de productos y la recolección de los perdidos, con base en las fechas de entrega de los pedidos, nivel de

inventario, demanda, frecuencia de solicitud y el peso de las cajas de los productos, se minimiza las entregas tardías y mejorará el tiempo de recolección de los pedidos, dando como resultado la aprobación de la hipótesis planteada en el presente trabajo.

## 5.1 CONTRIBUCIONES

Para la combinación relacionada con los problemas de asignación de ubicación de almacenamiento y el enrutamiento de recolector, los resultados mostraron el excelente desempeño de los modelos y algoritmos propuestos. Con las reformulaciones propuestas se lograron encontrar 62 nuevas soluciones factibles y 51 nuevos casos con solución óptimas para las instancias existentes en la literatura, las cuales fueron obtenidas principalmente del modelo SLAUPR\_V3, lo que indica la buena eficiencia de las restricciones de eliminación de subciclos basadas en restricciones de flujo versus MTZ para este tipo de problemas. Por otro lado, mediante el algoritmo RGRASP se logró generar soluciones factibles para todas las instancias estudiadas en la literatura y mejora (más del 29 %) las mejores soluciones conocidas. Además, el RGRASP propuesto requiere un tiempo de ejecución general promedio de 117 segundos para reportar soluciones factibles.

Finalmente, comparando las asignaciones de los productos a las ubicaciones como los tiene la empresa frente a la asignación obtenida mediante el algoritmo RGRASP propuesto, se presenta una mejora (más del 40 %) en los tiempos de viaje totales. En consecuencia, con la metaheurística propuesta se mejoran los tiempos de recolección y ayudan a la toma de decisiones tácticas proporcionando la ubicación apropiada de los productos en el almacén. Por ende, el RGRASP propuesto es una excelente alternativa para cualquier empresa interesada en resolver de manera simultánea el problema de asignación de ubicación de almacenamiento y enrutamiento del recolector considerando el peso de los productos.

Es importante señalar que los tomadores de decisiones deben analizar el equilibrio entre soluciones óptimas y aproximadas para acelerar su implementación práctica, ya

que encontrar una solución óptima puede ser poco práctico debido a la complejidad del problema.

El desarrollo de modelos matemáticos, resueltos mediante el uso de un software de optimización de propósito general, brindan soluciones de calidad con base en los tiempos de ejecución para solucionar de manera independiente los problemas de procesamiento por lotes y el enrutamiento del recolector considerando particularidades del problema de estudio, principalmente, un almacén con forma general, fechas compromiso de entrega, nivel de inventario disponible, productos contenidos en cada tarima, capacidad máxima de tarimas a recolectar diariamente, pesos de los productos y demanda estable por periodos. Por ende, en el presente trabajo no se propuso el uso de algoritmos heurísticos o metaheurísticos para resolver esta combinación.

## 5.2 TRABAJO A FUTURO

Dentro de las áreas de oportunidad detectadas en el presente trabajo, se encuentra contemplar otro tipo de distribución de probabilidad en el generador de instancias lo cual afectaría el comportamiento del producto en tránsito.

Otro aspecto sería, considerar restricciones relacionadas con la duración de la jornada laboral de los recolectores en la variante del problema del procesamiento de pedidos por lotes y el enrutamiento del recolector.

De igual manera, se podrían estudiar diferentes variantes para el problema del procesamiento de pedidos por lotes entre las que se encuentra considerar incertidumbre en el parámetro del inventario en tránsito, con una demanda estable por periodos. Así como también la caducidad de los productos, prioridad de clientes, costo de inventario y la contenerización de los pedidos en los vehículos enviados a los clientes.

# BIBLIOGRAFÍA

---

- ATTARI, M. Y. N., A. E. TORKAYESH, B. MALMIR y E. N. JAMI (2021), «Robust possibilistic programming for joint order batching and picker routing problem in warehouse management», *International Journal of Production Research*, **59**(14), págs. 4434–4452.
- AZADNIA, A. H., S. TAHERI, P. GHADIMI, S. MAT, Z. MUHAMAD y K. Y. WONG (2013), «Order batching in warehouses by minimizing total tardiness: A hybrid approach of weighted association rule mining and genetic algorithms», *The Scientific World Journal*, **2013**, págs. 1–13.
- BARTHOLDI, J. J. y S. T. HACKMAN (2019), *Warehouse and Distribution Science Release 0.98.1*, Supply Chain and Logistics Institute Atlanta.
- BOLAÑOS-ZUÑIGA, J., J. A. SAUCEDO, T. E. SALAIS y J. A. MARMOLEJO (2020), «Optimization of the storage location assignment and the picker-routing problem by using mathematical programming», *Applied Science*, **10**(2), pág. 534.
- BOUDIA, M., M. LOULY y C. PRINS (2007), «A reactive GRASP and path relinking for a combined production–distribution problem», *Computers & Operations Research*, **34**(11), págs. 3402–3419.
- CAI, J., X. LI, Y. LIANG y S. OUYANG (2021), «Collaborative Optimization of Storage Location Assignment and Path Planning in Robotic Mobile Fulfillment Systems», *Sustainability*, **13**(10).
- CANTU-FUNES, R., M. A. SALAZAR-AGUILAR y V. BOYER (2018), «Multi-depot



- periodic vehicle routing problem with due dates and time windows», *Journal of the Operational Research Society*, **69**(2), págs. 296–306.
- CASELLA, G., A. VOLPI, R. MONTANARI, L. TEBALDI y E. BOTTANI (2023), «Trends in order picking: a 2007–2022 review of the literature», *Production & Manufacturing Research*, **11**(1), pág. 2191–115.
- CASTIER, M. y E. MARTÍNEZ-TORO (2023), «Planning and picking in small warehouses under industry-relevant constraints», *Production Engineering*, **17**, págs. 575 – 590.
- CHABOT, T., R. LAHYANI, L. C. COELHO y J. RENAUD (2016), «Order picking problems under weight, fragility and category constraints», *International Journal of Production Research*, **55**(21), págs. 6361–6379.
- CHAOVALITWONGSE, W., D. KIM y P. M. PARDALOS (2003), «Grasp with a new local search scheme for vehicle routing problems with time window», *Journal of Combinatorial Optimization*, **7**(2), págs. 179–207.
- CHOPRA, S. y P. MEINDL (2008), *Administración de la cadena de suministro*, cuarta edición, Pearson Prentice Hall, México, DF.
- CORTÉS, P., R. A. GÓMEZ-MONTOYA, J. MUÑUZURI y A. CORREA-ESPINAL (2017), «A tabu search approach to solving the picking routing problem for large- and medium-size distribution centres considering the availability of inventory and  $K$  heterogeneous material handling equipment», *Applied Soft Computing*, **53**, págs. 61–73.
- DANIELS, R. L., J. L. RUMMEL y R. SCHANTZ (1998), «A model for warehouse order picking», *European Journal of Operational Research*, **105**(1), págs. 1–17.
- DAVARZANI, H. y A. NORRMAN (2015), «Toward a relevant agenda for warehousing research: literature review and practitioners' input», *Logistics Research*, **8**(1), págs. 1–18.

- DE KOSTER, R. B. M., T. LE-DUC y K. J. ROODBERGEN (2007), «Design and control of warehouse order picking: A literature review», *European Journal of Operational Research*, **182**(2), págs. 481–501.
- DE KOSTER, R. B. M., K. J. ROODBERGEN y R. VAN VOORDEN (1999), «Reduction of Walking Time in the Distribution Center of De Bijenkorf», en *New Trends in Distribution Logistics. Lecture Notes in Economics and Mathematical Systems*, tomo 480, Springer Berlin Heidelberg, págs. 215–234.
- DEKKER, R., R. B. M. DE KOSTER, K. J. ROODBERGEN y K. H. VAN (2004), «Improving Order Picking Response Time at Ankor’s Warehouse», *Interfaces*, **34**(4), págs. 303–313.
- DIJKSTRA, A. S. y K. J. ROODBERGEN (2017), «Exact route-length formulas and a storage location assignment heuristic for picker-to-parts warehouses», *Transportation Research Part E: Logistics and Transportation Review*, **102**, págs. 38–59.
- ELBERT, R. M., T. FRANZKE, C. H. GLOCK y E. H. GROSSE (2016), «The effects of human behavior on the efficiency of routing policies in order picking: The case of route deviations», *International Journal of Production Research*, **111**, págs. 537–551.
- ELSAYED, E. A., M. K. LEE, S. KIM y E. SCHERER (1993), «Sequencing and batching procedures for minimizing earliness and tardiness penalty of order retrievals», *International Journal of Production Research*, **31**(3), págs. 727–738.
- EYDI, A., H. MOHAGHEGHI y S. GHASEMINEZHAD (2022), «Routing order pickers in warehouses considering congestion and aisle width», *Scientia Iranica*, **29**(6), págs. 3455–3469.
- FEO, T. A. y M. G. C. RESENDE (1995), «Greedy randomized adaptive search procedures», *Journal of Global Optimization*, **6**(2), págs. 109–133.
- FESTA, P. y M. RESENDE (2010), «GRASP: Basic components and enhancements», *Telecommunication Systems*, **46**(3), págs. 253–271.

- GADEMANN, N. y S. VAN DE VELDE (2005), «Order batching to minimize total travel time in a parallel-aisle warehouse», *IIE Transactions*, **37**(1), págs. 63–75.
- GAVISH, B. y S. GRAVES (1978), «The Traveling Salesman Problem and Related Problems», .
- GLOCK, C. H., E. H. GROSSE, H. ABEDINIA y S. EMDE (2019), «An integrated model to improve ergonomic and economic performance in order picking by rotating pallets», *European Journal of Operational Research*, **273**(2), págs. 516–534.
- GOETSCHALCKX, M. y J. ASHAYERI (1989), «Classification and design of order picking systems», *Logistics World*, **2**(2), págs. 99–106.
- GOETSCHALCKX, M. y H. RATLIFF (1988), «Order Picking In An Aisle», *IIE Transactions*, **20**(1), págs. 53–62.
- GU, J., M. GOETSCHALCKX y L. F. MCGINNIS (2007), «Research on warehouse operation: A comprehensive review», *European Journal of Operational Research*, **177**(1), págs. 1–21.
- HADDADENE, S. R. A., N. LABADIE y C. PRODHON (2016), «A GRASP × ILS for the vehicle routing problem with time windows, synchronization and precedence constraints», *Expert Systems with Applications*, **66**, págs. 274–294.
- HENN, S. (2015), «Order batching and sequencing for the minimization of the total tardiness in picker-to-part warehouses», *Flexible Services and Manufacturing Journal*, **27**, págs. 86–114.
- HENN, S. y V. SCHMID (2013), «Metaheuristics for order batching and sequencing in manual order picking systems», *Computers & Industrial Engineering*, **66**(2), págs. 338–351.
- HILLIER, F. y L. G. (2010), *Introducción a la investigación operativa*, McGraw Hill, México.

- KEUNG, K., C. LEE y P. JI (2022), «Industrial internet of things-driven storage location assignment and order picking in a resource synchronization and sharing-based robotic mobile fulfillment system», *Advanced Engineering Informatics*, **52**, pág. 101 540.
- KORDOS, M., J. BORYCZKO, M. BLACHNIK y S. GOLAK (2020), «Optimization of Warehouse Operations with Genetic Algorithms», *Applied Sciences*, **10**(14), pág. 4817.
- LAMBERT, D. (2018), *Supply Chain Management: Processes, Partnerships, Performance*, tercera edición, Supply Chain Management Institute, United States, Florida.
- LEE, H.-T. (2022), «Travel distance estimation of relation-based storage assignment policies in picker-to-part warehouses», *Journal of Business Administration*, **47**(1), págs. 17–41.
- MASAE, M., C. H. GLOCK y E. H. GROSSE (2019), «Order picker routing in warehouses: A systematic literature review», *International Journal of Production Economics*, **224**, pág. 107 564.
- MATUSIAK, M., R. DE KOSTER, L. KROON y J. SAARINEN (2014), «A fast simulated annealing method for batching precedence-constrained customer orders in a warehouse», *European Journal of Operational Research*, **236**(3), págs. 968–977.
- MENÉNDEZ, B., M. BUSTILLO, E. G. PARDO y A. DUARTE (2017), «General Variable Neighborhood Search for the Order Batching and Sequencing Problem», *European Journal of Operational Research*, **263**(1), págs. 82–93.
- MILLER, C. E., A. TUCKER y R. ZEMLIN (1960), «Integer Programming Formulation of Traveling Salesman Problems», *Journal of the Association for Computing Machinery*, **7**(4), págs. 326–329.
- PALOMO-MARTÍNEZ, P., M. SALAZAR-AGUILAR y V. ALBORNOZ (2017),

- «Formulations for the orienteering problem with additional constraints», *Annals of Operations Research*, **258**, págs. 503–545.
- PINTO, A. R. F. y M. S. NAGANO (2022), «A comprehensive review of batching problems in low-level picker-to-parts systems with order due dates: Main gaps, trade-offs, and prospects for future research», *Journal of Manufacturing Systems*, **65**(2022).
- PRAIS, M. y C. C. RIBEIRO (2000), «Reactive grasp: An application to a matrix decomposition problem in TDMA traffic assignment», *INFORMS Journal on Computing*, **12**(3), págs. 164–176.
- RATLIFF, H. D. y A. S. ROSENTHAL (1983), «Order-Picking in a Rectangular Warehouse: A Solvable Case of the Traveling Salesman Problem», *Operations Research*, **31**(3), págs. 507–521.
- ROUWENHORST, B., B. REUTER, V. STOCKRAHM, G. J. VAN HOUTUM, R. J. MANTEL y W. H. M. ZIJM (2000), «Warehouse design and control: Framework and literature review», *European Journal of Operational Research*, **122**(3), págs. 515–533.
- SALHI, S. y G. NAGY (1999), «Consistency and robustness in location-routing», *Studies in Locational Analysis*, (13), págs. 3–19.
- SCHOLZ, A., S. HENN, M. STUHLMANN y G. WÄSCHER (2016), «A new mathematical programming formulation for the Single-Picker Routing Problem», *European Journal of Operational Research*, **253**(1), págs. 68–84.
- SCHOLZ, A., D. SCHUBERT y G. WÄSCHER (2017), «Order picking with multiple pickers and due dates – Simultaneous solution of Order Batching, Batch Assignment and Sequencing, and Picker Routing Problems», *European Journal of Operational Research*, **263**(2), págs. 461–478.
- SCHUBERT, D., A. SCHOLZ y G. WÄSCHER (2018), «Integrated order picking and vehicle routing with due dates», *OR Spectrum*, **40**, págs. 1109—1139.

- SHETTY, N., B. SAH y S. CHUNG (2020), «Route optimization for warehouse order picking operations via vehicle routing and simulation», *SN Applied Sciences*, **2**(311), págs. 1–18.
- SILVA, A., L. C. COELHO, M. DARVISH y R. RENAUD (2020), «Integrating storage location and order picking problems in warehouse planning», *Transportation Research Part E: Logistics and Transportation Review*, **140**(102003), págs. 1–22.
- SIMCHI-LEVI, D., S. KAMINSKY y E. SIMCHI-LEVI (2008), *Designing and Managing the Supply Chain: concepts, Strategies and case studies*, tercera edición, McGraw-Hill Book Company.
- THEYS, C., O. BRÁYSY, W. DULLAERT y B. RAA (2010), «Using a TSP heuristic for routing order pickers in warehouses», *European Journal of Operational Research*, **200**(3), págs. 755–763.
- TOTH, P. y D. VIGO (2014), *Vehicle routing: problems, methods, and applications*, SIAM.
- TSAI, C.-Y., J. J. H. LIOU y T.-M. HUANG (2008), «Using a multiple-GA method to solve the batch picking problem: considering travel distance and order due time», *International Journal of Production Research*, **46**(22), págs. 6533–6555.
- VALLE, C. A., J. E. BEASLEY y A. S. DA CUNHA (2016), «Modelling and Solving the Joint Order Batching and Picker Routing Problem in Inventories», en *Combinatorial Optimization*, Springer International Publishing, Cham, págs. 81–97.
- VALLE, C. A., J. E. BEASLEY y A. S. DA CUNHA (2017), «Optimally solving the joint order batching and picker routing problem», *European Journal of Operational Research*, **262**(3), págs. 817–834.
- VAN GILS, T., A. CARIS, K. RAMAEKERS y K. BRAEKERS (2019), «Formulating and solving the integrated batching, routing, and picker scheduling problem in a real-life spare parts warehouse», *European Journal of Operational Research*, **277**(3), págs. 814–830.

- VAN GILS, T., K. RAMAEKERS, K. BRAEKERS, B. DEPAIRE y A. CARIS (2018a), «Increasing order picking efficiency by integrating storage, batching, zone picking, and routing policy decisions», *International Journal of Production Economics*, **197**, págs. 243–261.
- VAN GILS, T., K. RAMAEKERS, A. CARIS y R. B. M. DE KOSTER (2018b), «Designing efficient order picking systems by combining planning problems: State-of-the-art classification and review», *European Journal of Operational Research*, **267**(1), págs. 1–15.
- VILLEGAS, J. G., C. PRINS, C. PRODHON, A. MEDAGLIA y N. VELASCO (2011), «A GRASP with evolutionary path relinking for the truck and trailer routing problem», *Computers & Operations Research*, **38**(9), págs. 1319–1334.
- WAN, Y. y Y. LIU (2022), «Integrating Optimized Fishbone Warehouse Layout, Storage Location Assignment and Picker Routing», *IAENG International Journal of Computer Science*, **49**(3), págs. 957–974.
- WANG, W., J. GAO, T. GAO y H. ZHAO (2017), «Optimization of Automated Warehouse Location Based on Genetic Algorithm», en *Proceedings of the 2017 2nd International Conference on Control, Automation and Artificial Intelligence (CAAI 2017)*, Atlantis Press.
- WATERS, D. (2003), *Logistics: An Introduction to Supply Chain Management*, tercera edición, Palgrave Macmillan, United States, New York.
- WINKELHAUS, S., E. H. GROSSE y S. MORANA (2021), «Towards a conceptualisation of Order Picking 4.0», *Computers Industrial Engineering*, **159**(107511), págs. 1–21.
- XU, X. y C. REN (2022), «A novel storage location assignment in multi-pickers picker-to-parts systems integrating scattered storage, demand correlation, and routing adjustment», *Computers Industrial Engineering*, **172**(108618), págs. 1–15.

- ZHOU, L., J. ZHAO, H. LIU, F. WANG, Y. JIANGLONG y S. WANG (2022), «Stochastic models of routing strategies under the class-based storage policy in fishbone layout warehouses», *Scientific Reports*, **12**(12876), págs. 1–17.
- ZUNIC, E., A. BESIREVIC, R. SKROBO, H. HASIC, K. HODZIC y A. DJEDOVIC (2017), «Design of optimization system for warehouse order picking in real environment», en *2017 XXVI International Conference on Information, Communication and Automation Technologies (ICAT)*, IEEE, págs. 1–6.
- ÇELK, M. y H. SÜRAL (2014), «Order picking under random and turnover-based storage policies in fishbone aisle warehouses», *IIE Transactions*, **46**(3), págs. 283–300.
- ŽULJ, I., C. H. GLOCK, E. H. GROSSE y M. SCHNEIDER (2018), «Picker routing and storage-assignment strategies for precedence-constrained order picking», *Computers & Industrial Engineering*, **123**, págs. 338–347.