



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Modelamiento eficiente de la preparación de pedidos en un almacén usando un metaheurístico de Búsqueda Tabú

Laura Vélez Calle

Universidad Nacional de Colombia
Facultad de Minas, Departamento Ingeniería de la Organización
Medellín, Colombia
2016

Modelamiento eficiente de la preparación de pedidos en un almacén usando un metaheurístico de Búsqueda Tabú

Laura Vélez Calle

Tesis presentada como requisito parcial para optar al título de:

Magister en Ingeniería Industrial

Director (a):

Alexander Alberto Correa Espinal, Ph.D.

Codirector (a):

Rodrigo Andrés Gómez Montoya, Ph.D.

Línea de Investigación:

Logística en la gestión de almacenes y Centros de Distribución

Grupo de Investigación:

Modelamiento para la Gestión de Operaciones (GIMGO)

Universidad Nacional de Colombia

Facultad de Minas, Departamento Ingeniería de la Organización

Medellín, Colombia

2016

A Dios, a mi familia y a todas las personas que creyeron en mí, ya que con su amor, su paciencia, su apoyo y su fe me permitieron alcanzar esta meta tan importante en mi vida.

Agradecimientos

A mi director de tesis Alexander Correa Espinal que es Profesor Titular de la Universidad Nacional de Colombia y a mi codirector Rodrigo Andrés Gómez Montoya quien es profesor de la Institución Universitaria Politécnico Jaime Isaza Cadavid por brindarme su tiempo, su paciencia, su experiencia y sus valiosas observaciones y sugerencias para la construcción de esta tesis de maestría.

A la Universidad Nacional de Colombia que me ha acompañado y formado durante todo mi proceso profesional, que me ha dado las bases para ser una gran persona y una excelente profesional.

Al Grupo Bancolombia que me facilitó el espacio para asistir a cada una de las actividades formativas de este proceso académico.

A mi familia, a José Daniel Hernández Vahos, a Sergio Cardona Fernández y a Miguel Ángel Maldonado por escucharme y acompañarme en todo este proceso formativo.

Resumen

En esta tesis se modela un metaheurístico de *Búsqueda Tabú* que facilita la resolución del *problema de programación de la preparación de pedidos* en un almacén, considerando los tiempos de entrega de las órdenes. Para lograrlo, se minimiza la tardanza total de las ordenes teniendo en cuenta para el cálculo de los tiempos de procesamiento, variables tales como: k equipos de manejo de materiales heterogéneos, n productos, m posiciones de almacenamiento y actualización del nivel de inventario. Al comparar los resultados obtenidos entre el metaheurístico desarrollado y la regla de priorización *Earliest Due Date (EDD)*, a través, de una prueba de hipótesis, se observa que el valor de la tardanza total promedio obtenida con el metaheurístico, es menor con un nivel de confianza del 95%, lo que nos permite inferir que se obtiene un mejor desempeño en el almacén al usar el modelo propuesto en esta tesis.

Palabras clave: Programación de operaciones, preparación de pedidos, reglas de despacho, almacén, metaheurísticos, optimización, tardanza.

Abstract

This thesis presents a Tabu Search metaheuristic facilitating problem resolution scheduling order picking in a warehouse, considering the time of delivery of orders is modeled. To achieve this, the total delay of orders is minimized taking into account for the calculation of processing times, variables such as: k handling equipment heterogeneous materials, products n , m storage locations and update inventory level. By comparing the results obtained between the metaheuristic developed and the rule of prioritization *Earliest Due Date (EDD)* through of a hypothesis test, it is observed that the value of the total average delay obtained with the metaheuristic is less with a level 95% confidence, which allows us to infer that perform better in store when using the model proposed in this thesis is obtained.

Keywords: Scheduling Operations, picking, dispatching rules, warehouse, metaheuristics, Optimization, tardiness.

Contenido

	Pág.
Resumen	IX
Lista de figuras	XIV
Lista de tablas	XV
Introducción	1
1. Revisión del estado del arte y antecedentes	5
1.1 Metodología empleada para la revisión del estado del arte	5
1.2 Justificación de la revisión	6
1.3 Formulación de las preguntas de investigación.....	7
1.4 Búsqueda de documentos	7
1.5 Selección de documentos.....	11
1.5.1 Criterios de inclusión o exclusión	11
1.6 Evaluación de la calidad	12
1.7 Extracción de datos	13
1.7.1 (Alonso-Ayuso et al., 2013)	13
1.7.2 (Azadnia et al., 2013)	14
1.7.3 (Bozer & Kile, 2008)	14
1.7.4 (Choy et al., 2014).....	15
1.7.5 (Henn, 2012)	15
1.7.6 (Ho et al., 2008)	15
1.7.7 (Henn, 2012)	16
1.7.8 (Henn & Schmid, 2013)	16
1.7.9 (Hsieh & Fan, 2010)	17
1.7.10 (Hsieh & Huang, 2011).....	17
1.7.11 (Rubrico et al., 2006).....	18
1.7.12 (Rubrico et al., 2011).....	18
1.7.13 (De Koster, Van der Poort, & Wolters, 1999).....	19
1.7.14 (Piroird & Dale, 1998).....	19
1.7.15 (Gómez, 2015)	20
1.8 Síntesis de datos	21
1.9 Objetivos.....	21
1.9.1 Objetivo General	21
1.9.2 Objetivos específicos	21
2. Marco teórico	23
2.1 Flujo del marco teórico.....	23

2.2	Cadena de suministro.....	24
2.3	CEDI (Centro de Distribución)	25
2.3.1	Elementos de un CEDI	26
2.3.2	El CEDI como un tipo de almacén	26
2.4	Operaciones en un almacén.....	27
2.5	Recursos en un Almacén	28
2.6	Programación de operaciones.....	28
2.7	Datos de un problema de la programación	29
2.8	Representación de un problema de programación	29
2.8.1	Ambientes de máquinas	29
2.8.2	Características y restricciones de procesamiento	30
2.8.3	Objetivos para el problema de programación.....	31
2.9	Heurísticos	32
2.10	Metaheurístico.....	32
2.11	Clasificación de las heurísticas y metaheurísticos	33
2.12	Herramientas de programación	33
2.12.1	Lenguaje y entorno de programación.....	34
2.12.2	Framework OpenTS®	34
2.12.3	Base de datos.....	35
2.13	Conceptos estadísticos y herramienta estadística	36
2.13.1	Prueba de hipótesis	37
2.13.2	Prueba <i>T</i> pareada.....	37
2.13.3	Herramienta estadística	38
3.	Planteamiento del problema de programación de preparación de pedidos y la selección del metaheurístico	39
3.1	Descripción del proceso de preparación de pedidos en un almacén	39
3.2	Tiempo de procesamiento de la orden	40
3.3	Planteamiento de las variables de decisión y la función objetivo	40
3.3.1	Variables de decisión para el problema tratado con intercambios consecutivos.....	41
3.3.2	Función objetivo para el problema tratado con intercambios consecutivos.....	41
3.3.3	Variables de decisión para el problema tratado con intercambios no consecutivos.....	44
3.3.4	Función objetivo para el problema tratado con intercambios no consecutivos.....	44
3.4	Razones para seleccionar un metaheurístico	47
3.5	Teorema <i>NFL</i> (<i>No Free Lunch</i>)	47
3.6	Búsqueda Tabú.....	48
3.7	Algoritmos Genéticos	49
3.8	Recocido Simulado	50
3.9	Optimización de enjambre de partículas (<i>PSO</i>).....	50
3.10	Colonia Hormigas.....	51
3.11	Metaheurístico seleccionado	52
4.	Programación del metaheurístico.....	55
4.1	Proceso de preparación de pedidos en un almacén	55
4.2	Descripción del Almacén seleccionado	56
4.3	Escenarios creados para el proceso de preparación de pedidos.....	57

4.4	Estructura básica de la Búsqueda Tabú	58
4.5	Estructura de Búsqueda Tabú adaptada al problema tratado en esta tesis....	59
4.6	Herramientas empleadas para el desarrollo del metaheurístico de Búsqueda Tabú	60
4.7	Selección de la configuración inicial.....	61
4.8	Conformación del vecindario.....	61
4.9	Evaluación Función Objetivo.....	62
4.10	Memoria de corto plazo (Lista Tabú).....	65
4.11	Criterio de parada	65
5.	Análisis de Resultados	67
5.1	Regla de despacho seleccionada para comparar los resultados.....	67
5.2	Prueba de hipótesis seleccionada	68
5.3	Obtención de datos iniciales	68
5.4	Calculo del tamaño de la muestra.....	70
5.5	Prueba de normalidad de las muestras.....	71
5.6	Resultados de la prueba <i>T</i> pareada	73
6.	Conclusiones y recomendaciones.....	77
6.1	Conclusiones	77
6.2	Recomendaciones y trabajos futuros	78
A.	Escenarios de aplicación.....	81
B.	Listado de los productos que se usan en el almacén seleccionado	83
	Bibliografía	85

Lista de figuras

	Pág.
Figura 1-1 Estructura de la revisión sistemática de la literatura (<i>SLR</i>) que se usa en esta tesis.	6
Figura 2-1 Fases del marco teórico.	24
Figura 2-2 Secuencia del <i>framework OpenTS</i> [®]	35
Figura 3-1 Ejemplo de una configuración donde se hace un intercambio consecutivo entre las órdenes de las posiciones <i>PIO</i> y <i>PID</i> . Las letras <i>A</i> , <i>B</i> , <i>C</i> , <i>D</i> y <i>E</i> corresponden a órdenes de un almacén.	42
Figura 3-2 Ejemplo de una configuración donde se hace un intercambio no consecutivo entre las órdenes de las posiciones <i>PIO</i> y <i>PID</i> . Las letras <i>A</i> , <i>B</i> , <i>C</i> , <i>D</i> y <i>E</i> corresponden a órdenes de un almacén.	45
Figura 4-1 Almacén de tipo <i>layout</i> multipasillo, de dos pisos, con códigos de las posiciones de almacenamiento y la ubicación del <i>Depot</i>	56
Figura 4-2 Metaheurístico básico Búsqueda Tabú.	59
Figura 4-3 Estructura de Búsqueda Tabú adaptada al problema tratado en esta tesis.	60
Figura 4-4 Intercambio posible entre el atributo x_2 con x_3 en una configuración x (solución) de ejemplo.	62
Figura 4-5 Pseudocódigo de los intercambios entre atributos consecutivos.	63
Figura 4-6 Pseudocódigo de los intercambios entre atributos no consecutivos.	64
Figura 4-7 Pseudocódigo con el cálculo de la mejor tardanza.	65
Figura 5-1 Prueba de normalidad para la muestra uno (Metaheurístico de Búsqueda Tabú), donde el valor p equivale a 0,14	72
Figura 5-2 Prueba de normalidad para la muestra dos (<i>Earliest Due Date (EDD)</i>), donde el valor p equivale a 0,06	72
Figura 5-3 (a) Resultado arrojado por la herramienta Minitab [®] al aplicar la prueba T pareada, (b) Caja de diferencias, (c) Valores individuales de diferencias y (d) Histograma de diferencias.	75

Lista de tablas

	Pág.
Tabla 1-1 Resultados de la primera búsqueda de documentos en las diferentes fuentes científicas, sin usar filtros.	8
Tabla 1-2 Resultados de la primera búsqueda de documentos en las diferentes fuentes científicas, usando filtros.	9
Tabla 1-3 Resultados de la segunda búsqueda de documentos en las diferentes fuentes científicas, sin usar filtros.	10
Tabla 1-4 Resultados de la segunda búsqueda de documentos en las diferentes fuentes científicas, usando filtros.	10
Tabla 1-5 Características que se buscan en los artículos revisados.	12
Tabla 1-6 Resultados de la revisión de literatura para la programación de la preparación de pedidos en almacenes.....	13
Tabla 3-1 Comparativo de metaheurísticos más utilizados en la literatura.....	52
Tabla 4-1 Atributos de las órdenes con valores de ejemplo.....	57
Tabla 4-2 Características de los equipos de manejo de materiales.	58
Tabla 4-3 Intercambios posibles entre atributos que forman el vecindario del ejemplo de la Figura 4-4. Se resalta en color azul y verde el intercambio realizado.	62
Tabla 5-1 Resultados de las tardanzas obtenidos por el metaheurístico de Búsqueda Tabú desarrollado en esta tesis (uno) y por la regla <i>Earliest Due Date (EDD)</i> (dos).....	68
Tabla 5-2 Diferencia entre la muestra uno y la muestra dos.....	69
Tabla 5-3 Parámetros de entrada ingresados en la herramienta Minitab® para el cálculo del tamaño de la muestra poblacional	70
Tabla 5-4 Resultados de Minitab® para el tamaño de la muestra poblacional que se debe emplear para el diseño experimental.....	71
Tabla B-1 Listado de productos en el almacén.....	83

Introducción

La *preparación de pedidos* es la operación con más altos costos que se realiza en un almacén, (Rosenwein, 1996), (Ruben & Jacobs, 1999), (Van Den Berg, 1999). Según (Ho, Su, & Shi, 2008), la optimización de las operaciones debe ser considerada en las políticas de las instalaciones, con la finalidad de ofrecer bajos costos y no incurrir en retrasos en la entrega de pedidos a los clientes. Dicho esto, se evidencia la importancia de investigar sobre el problema de la programación de la preparación de pedidos en almacenes.

El problema de la programación de la preparación de pedidos en almacenes ha sido abordado por diversos autores como: (Piroird & Dale, 1998), donde estudian el proceso de cumplimiento de pedidos particularmente con los plazos establecidos de entrega internos y externos en una fábrica de productos químicos especializados; (Alonso-Ayuso, Tirado, & Udías, 2013), que investigan cómo realizar la selección, la programación de un sistema de almacenamiento y la recuperación automática de productos; (Azadnia, Taheri, Ghadimi, Saman, & Wong, 2013), buscan minimizar la tardanza promedio de todos los pedidos en un almacén, usando cuatro fases: la minería de reglas de asociación, un modelo de programación binaria entera, un algoritmo genético integrado con el problema del vendedor viajero, un algoritmo genético para secuenciar los lotes construidos, con el fin de minimizar la tardanza promedio de todos los pedidos; (Henn, 2012), estudian el *problema de conformación de lotes* en la operación de *preparación de pedidos* en un sistema de ordenes en línea, donde buscan minimizar el máximo tiempo de realización de los pedidos de los clientes que van llegando al almacén; (Henn, 2015), estudian dos métodos, el descenso en vecindarios variables (*Variable Neighborhood Descent*) y la búsqueda en vecindarios variables (*Variable Neighborhood Search*) y muestran cómo se pueden aplicar, con el fin de minimizar la tardanza total de un determinado conjunto de pedidos; (Henn & Schmid, 2013), estudian cómo los metaheurísticos se pueden utilizar con el fin de minimizar la *tardanza* total para un conjunto de pedidos; (Rubrico, Ota, Higashi, & Tamura, 2006), se enfocan en investigar la operación de preparación de

pedidos en un almacén, para ello designan un número de agentes, donde cada uno tiene su propio conjunto de secuencias (rutas de recogida) y (Rubrico, Higashi, Tamura, & Ota, 2011), investigan una solución para un problema de reprogramación dinámica para la preparación de pedidos donde van llegando de manera aleatoria nuevas órdenes y donde las ordenes asignadas quedan estáticas en la programación.

De la revisión hecha con los autores antes descritos, se identifica la ausencia de modelos que incorporen en la solución del problema, variables tales como: k equipos de manejo de materiales heterogéneos, n productos, m posiciones de almacenamiento y actualización del nivel de inventario para obtener una *tardanza* mínima. Solo en el trabajo de (Gómez, 2015), se aborda el uso de dichas variables, pero sin tener en cuenta la tardanza, mostrando una limitación para modelar eficientemente la operación de preparación de pedidos en un almacén.

Por los motivos expuestos se justifica la importancia de resolver este problema de investigación, con el fin de contribuir al estado del arte y a la adecuada programación de la preparación de pedidos en un almacén de materia prima, incluyendo variables como: k equipos de manejo de materiales heterogéneos, n productos, m posiciones de almacenamiento y actualización del nivel de inventario.

El aporte que se pretende realizar con este trabajo consiste en modelar un metaheurístico de Búsqueda Tabú que facilite resolver el problema de programación de la preparación de pedidos en un almacén minimizando la tardanza. Para ello, fue necesario dividir el trabajo en varias etapas, la cuales se abordan como capítulos.

En el capítulo 1, se inició con un análisis del estado del arte de la programación de la preparación de pedidos, donde se identificó que no se cuenta con un modelo que minimice la tardanza de las ordenes de un almacén, teniendo en cuenta para el cálculo del tiempo de procesamiento de las ordenes, variables tales como: k equipos de manejo de materiales heterogéneos (diferente velocidad y capacidad), múltiples posiciones de almacenamiento, manejo de inventario y múltiples productos a recolectar, (Gómez, 2015).

En el capítulo 2, se muestran los conceptos y herramientas que se necesitaron para desarrollar esta tesis. En este, se abordan, los temas de: Cadena de Suministro, Centro de Distribución (CEDI), programación de operaciones, Heurísticos, Metaheurísticos, herramientas de programación (Lenguaje y Entorno de Programación, *Framework OpenTS*[®], Base de Datos), conceptos estadísticos y herramienta estadística (Pruebas de Hipótesis, Prueba *t* Pareada).

En el capítulo 3, se definió la función objetivo y las restricciones que se usarían para modelar correctamente la problemática identificada. Para ello, se usó el trabajo de (Gómez, 2015), para abordar la operación de la preparación de pedidos y para la minimización de la tardanza se usó el modelamiento básico expuesto por (Pinedo, 2009), los cuales fueron adaptados para que pudieran representar el problema tratado en esta tesis. Además, se explica el por qué se selecciona un metaheurístico para resolver el problema tratado en esta tesis y el por qué se decide escoger entre los metaheurísticos más usados en la literatura el metaheurístico de Búsqueda Tabú.

En el capítulo 4, se decide escoger el mismo tipo de almacén definido por (Gómez, 2015) para crear múltiples escenarios. Los pasos realizados fueron los siguientes: primero, se definieron los atributos que tendrían las órdenes del almacén seleccionado; segundo, se variaron los valores de los atributos de las órdenes para encontrar nuevos escenarios; y tercero, con los escenarios ya definidos, se programó el metaheurístico de Búsqueda Tabú, que permite encontrar la secuencia en que se deben programar las órdenes para minimizar su tardanza total en el almacén.

En el capítulo 5, se validaron los resultados empleando una prueba de hipótesis que permitió comparar los valores obtenidos con el metaheurístico de Búsqueda Tabú desarrollado, con respecto a la regla *Earliest Due Date (EDD)*. A partir de este análisis, se identificó que los mejores resultados se consiguen con el metaheurístico desarrollado en esta tesis.

Por último, en el capítulo 6, se listan las conclusiones obtenidas en esta tesis, las recomendaciones y trabajos futuros identificados.

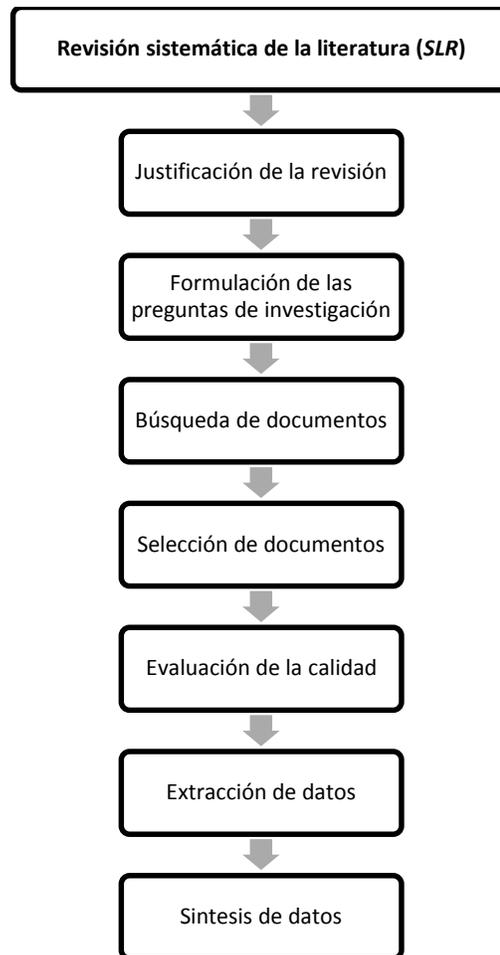
1. Revisión del estado del arte y antecedentes

En este capítulo, se presenta la revisión bibliográfica sobre la programación de la preparación de pedidos en almacenes. Para ello, se consultaron las bases de datos *Taylor and Francis*, *Elsevier*, *Emerald*, *Springer*, la herramienta bibliográfica, *Scielo*, *Redalyc*, y los repositorios de algunas Universidades afines a la ingeniería industrial relacionadas con el tema, lo que permitió analizar la literatura existente, con el fin de identificar las falencias y a partir de éstas identificar los posibles aportes que pueden ser generados con esta tesis, para luego enfatizar el tema de la programación de la operación de preparación de pedidos en los almacenes.

1.1 Metodología empleada para la revisión del estado del arte

Para la revisión del estado del arte de esta tesis, se utilizó la revisión sistemática de la literatura (*SLR*) descrita por (Velásquez, 2015a), (Velásquez, 2015b) y (Kitchenham et al., 2009). En la cual, se presentan los siguientes pasos, ver Figura 1-1, donde: primero, se realiza la justificación del por qué se selecciona el método de la revisión sistemática de la literatura (*SLR*); segundo, se formulan las preguntas de investigación de esta tesis; tercero, se hace la búsqueda de los documentos en las diversas fuentes de datos científicas y se estructura la ecuación de búsqueda a ser utilizada; cuarto, se seleccionan los documentos a utilizar en la tesis, para ello, se definen los criterios de inclusión o exclusión de documentos; quinto, se hace la evaluación de la calidad de los documentos con respecto a la investigación que se realiza con esta tesis; sexto, se hace la extracción de los datos donde se observan las similitudes y diferencias de los estudios; finalmente, se realiza la síntesis de los datos, donde se resume la información relevante de esta revisión con respecto a las preguntas investigativas.

Figura 1-1 Estructura de la revisión sistemática de la literatura (SLR) que se usa en esta tesis.



Fuente: Elaboración propia basada en (Velásquez, 2015a), (Velásquez, 2015b), (Kitchenham et al., 2009)

1.2 Justificación de la revisión

Para realizar el proceso de revisión de literatura de esta tesis, se decide utilizar una revisión sistemática de la literatura (SLR), ya que no se identifica revisiones recientes concernientes al tema de la programación de operaciones en almacenes, en las bases de datos *Taylor and Francis*, *Elsevier*, *Emerald*, *Springer*, la herramienta bibliográfica, *Scielo*, *Redalyc*, y repositorios de algunas Universidades afines a la ingeniería industrial

relacionas con el tema, tales que describan el estado actual de este tema. También, se utiliza *SLR*, ya que permite identificar oportunidades investigativas en la literatura.

1.3 Formulación de las preguntas de investigación

Para esta tesis, se plantean las siguientes preguntas investigativas (*Pi*) concernientes al tema de la programación de operaciones en los almacenes, que permitan identificar los vacíos que hay en la literatura con respecto a este tema:

Pi1. ¿Cuál operación es la de mayor interés investigativo en los almacenes?

Pi2. ¿Cuáles algoritmos, métodos o reglas se utilizan en la literatura científica para programar y secuenciar de manera eficiente la operación de mayor interés investigativo en los almacenes?

1.4 Búsqueda de documentos

Para realizar la búsqueda de documentos científicos que sirvan de insumo para esta tesis, se realiza: primero, una búsqueda general de la programación de operaciones en almacenes, para ir afinando la búsqueda por palabras claves, a la vez, de ir identificando cuál es la operación del almacén de mayor interés investigativo; y segundo, se realiza una búsqueda específica sobre la programación de dicha operación de interés, para terminar de responder las preguntas de investigación mencionadas en el numeral 1.3.

Para realizar la primera búsqueda, se consultan las siguientes bases de datos: *Taylor and Francis*, *Elsevier*, *Emerald*, *Springer*, la herramienta bibliográfica *Scopus* que tiene acceso a “más de 219.000 títulos de aproximadamente 5.000 editoriales” (Velásquez, 2015b), *Scielo*, *Redalyc*, y los repositorios de algunas Universidades afines a la ingeniería industrial, esto con el fin de cubrir la búsqueda de investigaciones realizadas en diferentes partes del mundo. También, se utilizan las siguientes palabras claves: *Scheduling Operations* o programación de operaciones, *Warehouse*. Adicionalmente, se realiza la búsqueda en todos los años disponibles en las bases de datos, ver Tabla 1-1. Después se realiza un refinamiento, donde se filtra entre los años de 1998 al 2016 para tener cubrimiento de los últimos dieciocho años y se selecciona las disciplinas de:

“engineering” or “business & management” or “logistics”, ver Tabla 1-2. A continuación, se muestra la ecuación de búsqueda resultante:

En Inglés: “Scheduling Operations” and “warehouse” LIMIT-TO (SUBJAREA, “Engineering”) OR LIMIT-TO (SUBJAREA, “Business, Management and Accounting”) OR LIMIT-TO (SUBJAREA, “logistics ”) AND PUBYEAR > 1997.

En Español: “Programación de operaciones” y “almacén” limitado a (disciplina: ingeniería) o limitado a (disciplina: Administración de empresas) o limitado a (disciplina: logística) y publicado > 1997.

Tabla 1-1 Resultados de la primera búsqueda de documentos en las diferentes fuentes científicas, sin usar filtros.

	Palabras clave: “Scheduling Operations” o “programación de operaciones”	Palabras clave: “Scheduling Operations” and “Warehouse” o “programación de operaciones” and “almacén”
<i>Taylor & Francis</i>	164.628	6,323
<i>Sciencedirect (Elsevier)</i>	272.144	10,411
<i>Emerald</i>	31.936	3.496
<i>Spinger</i>	244.719	13.220
<i>Scopus</i>	27.792	205
<i>Scielo</i>	60	2
<i>Redalyc</i>	147.398	0
<i>Bdigital: Repositorio institucional UN</i>	1.859	331

Fuente: Elaboración propia

Tabla 1-2 Resultados de la primera búsqueda de documentos en las diferentes fuentes científicas, usando filtros.

	<p>Palabras clave: “Scheduling Operations” and “Warehouse” o “programación de operaciones” and “almacén”</p> <p>Disciplina: <i>engineering or business & management or logistics.</i></p> <p>Años: 1998 y 2016</p>
<i>Taylor & Francis</i>	1,345
<i>Scencedirect (Elsevier)</i>	4.456
<i>Emerald</i>	68
<i>Spinger</i>	3.776
<i>Scopus</i>	105
<i>Scielo</i>	0
<i>Redalyc</i>	0
<i>Bdigital: Repositorio institucional UN</i>	146

Fuente: Elaboración propia.

De la primera búsqueda, se identifica que existen cuatro operaciones principales en un almacén: recepción, almacenamiento, preparación de pedidos y envío, y entre esas operaciones (Rosenwein, 1996), (Ruben & Jacobs, 1999), (Van Den Berg, 1999) y (Gómez, 2015), comentan que la preparación de pedidos es la más costosa de todas las operaciones que hay en un almacén, por esto, se escoge la operación de preparación de pedidos para realizar esta segunda búsqueda. Para esto, se utilizan las mismas fuentes científicas de la primera, se definen como palabras claves: “*scheduling*”, “*picking*”, “*warehouse*” y se busca en todos los años disponibles en las fuentes científicas, ver Tabla 1-3. Después se realiza un refinamiento, donde se filtra entre los años de 1998 y 2016 para tener cubrimiento de los últimos dieciocho años y se selecciona las disciplinas de: “*engineering*” or “*business & management*” or “*logistics*”, ver Tabla 1-4Tabla 1-2. A continuación, se muestra la nueva ecuación de búsqueda resultante:

En inglés: (TITLE-ABS-KEY (scheduling) AND TITLE-ABS-KEY (picking) AND TITLE-ABS-KEY (warehouse)) AND LIMIT-TO (SUBJAREA, "Engineering") OR LIMIT-TO (SUBJAREA, "Business, Management and Accounting") OR LIMIT-TO (SUBJAREA, "logistics ") AND PUBYEAR > 1997.

En Español: “Programación” y “preparación de pedidos” y “almacén” limitado a (disciplina: ingeniería) o limitado a (disciplina: Administración de empresas) o limitado a (disciplina: logística) y publicado > 1997.

Tabla 1-3 Resultados de la segunda búsqueda de documentos en las diferentes fuentes científicas, sin usar filtros.

	Palabras clave: “Scheduling” and “picking” and “Warehouse” o “programación” and “preparación de pedidos” and “almacén”
<i>Taylor & Francis</i>	1.665
<i>Scimedirect (Elsevier)</i>	2.941
<i>Emerald</i>	1.129
<i>Spinger</i>	4.190
<i>Scopus</i>	56
<i>Scielo</i>	0
<i>Redalyc</i>	0
<i>Bdigital: Repositorio institucional UN</i>	4

Fuente: Elaboración propia.

Tabla 1-4 Resultados de la segunda búsqueda de documentos en las diferentes fuentes científicas, usando filtros.

	Palabras clave: “Scheduling” and “picking” and “Warehouse” o “programación” and “preparación de pedidos” and “almacén” Disciplina: engineering or business & management or logistics. Años: 1998 y 2016.
<i>Taylor & Francis</i>	442
<i>Scimedirect (Elsevier)</i>	566
<i>Emerald</i>	122
<i>Spinger</i>	51
<i>Scopus</i>	33
<i>Scielo</i>	0
<i>Redalyc</i>	0

	<p>Palabras clave: “Scheduling” and “picking” and “Warehouse” o “programación” and “preparación de pedidos” and “almacén”</p> <p>Disciplina: engineering or business & management or logistics.</p> <p>Años: 1998 y 2016.</p>
Bdigital: Repositorio institucional UN	4

Fuente: Elaboración propia.

1.5 Selección de documentos

Al realizar la primera búsqueda de los documentos mencionados en el numeral 1.4, se encuentran 9.896 documentos. Luego, cuando se hace la segunda búsqueda se disminuye esa cantidad de documentos un 87.69%, quedando un total de 1.218 documentos. Con esta última búsqueda, se realiza un filtrado de documentos con base a unos criterios de inclusión o exclusión. Los cuales, permiten afinar la búsqueda de tal manera que solo queden los documentos de interés para esta tesis.

1.5.1 Criterios de inclusión o exclusión

Los documentos que se incluyen para su revisión, son aquellos que muestran qué algoritmos, métodos o reglas utilizan para programar y secuenciar de manera eficiente la preparación de pedidos en almacenes.

Ahora, los documentos que se excluyen de la revisión son porque cumplen alguna de las siguientes características:

- Documentos que no especifiquen bien, el método que utilizan para programar y secuenciar de manera eficiente la operación de preparación de pedidos.
- Documentos que no especifiquen qué variable están minimizando, bien sea la tardanza, los tiempos de ruta, etc.
- Documentos que no especifiquen que estudian las operaciones de los almacenes.

1.6 Evaluación de la calidad

Para realizar el análisis de calidad de esta tesis, se definen unos componentes o ítems, ver Tabla 1-5, los cuales, se buscan en los documentos seleccionados en la *SLR*, para así determinar en qué proporción los documentos, dan respuesta a las preguntas de investigación mencionadas en la sección 1.3.

Tabla 1-5 Características que se buscan en los artículos revisados.

#	Componentes o ítems de búsqueda
C1	Métodos <i>Scheduling</i>
C2	<i>Earliest Due Date (EDD)</i>
C3	Tardanza
C4	Conformación de lotes y ruteo
C5	<i>K</i> equipos heterogéneos
C6	Inventario
C7	<i>n</i> productos <i>i</i>
C8	<i>m</i> posiciones de almacenamiento <i>j</i>
C9	Metaheurísticos, heurísticos
C10	Reglas
C11	WMS o Sin información
C12	minimiza tiempo de ruta
C13	Tiene en cuenta la capacidad del vehículo o <i>picker</i>
C14	Tiene en cuenta la velocidad del vehículo o del <i>picker</i> (velocidad constante)

Fuente: Elaboración propia

Para determinar la proporción mencionada en el párrafo anterior, se usará la letra (C) cuando el documento cumple totalmente con el componente de la Tabla 1-5, de (P) cuando cumple de forma parcial y cuando el documento no cumple ni de forma completa ni de forma parcial los componentes, se entenderá que dicho documento no cumple (*espacio en blanco*), ver Tabla 1-6.

Tabla 1-6 Resultados de la revisión de literatura para la programación de la preparación de pedidos en almacenes

AUTOR	C1		C2		C3		C4		C5		C6		C7		C8		C9		C10		C11		C12		C13		C14			
	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P	C	P		
Alonso-Ayuso (2013)	x		X				x										x													
Azadnia <i>et al.</i> (2013)	x		X		x		x						X				x		x				x							
Bozer <i>et al.</i> (2008)		x					x										x						x		x					
Choy <i>et al.</i> (2014)	x				x									x	x				x						x		x			
Henn <i>et al.</i> (2012a)	x					x	x							x			x		x									x		
Henn <i>et al.</i> (2012b)	x		x		x		x			x			X		x		x									x		x		
Henn <i>et al.</i> (2013)	x		x		x		x			x			X		x		x									x		x		
Ho <i>et al.</i> (2008)	x						x								x				x						x		x			
Hsieh <i>et al.</i> (2010)		x					x						X			x			x						x		x			
Hsieh <i>et al.</i> (2011)		x					x						X		x				x						x		x			
Kulak <i>et al.</i> (2012)	x						x										x													
Rubrico <i>et al.</i> (2011)	x						x							x		x		x									x			
Rubrico <i>et al.</i> (2006)	x						x							X		x		x									x			
Rubrico <i>et al.</i> (2005)	x						x							X		x		x												
Piroird y Dale (1998)	x		x																x											
De Koster <i>et al.</i> (1999)	x						x									x		x								x		x		x

Fuente: Elaboración propia.

1.7 Extracción de datos

En esta sección, se describen las investigaciones realizadas en cada uno de los documentos seleccionados en la SLR, a la vez de que, se identifican las oportunidades de investigación.

1.7.1 (Alonso-Ayuso et al., 2013)

Investigan cómo realizar la selección y programación de un sistema de almacenamiento y la recuperación automática de productos. Su objetivo se basa en minimizar el tiempo de carga total de los vehículos que realizan el transporte, teniendo en cuenta las fechas de vencimiento. Para ello utiliza un heurístico que se basa en dos fases, donde para la primera usa programación binaria entera para seleccionar las cajas que serán extraídas

de los estantes de tal manera que se satisfaga la demanda del almacén y se obtenga un tiempo mínimo de ruta y lo combina con la segunda fase, que consiste en encontrar la secuencia de cajas que se transportaran en los camiones de tal manera que se minimice el tiempo medio de carga de los camiones, para esta segunda fase se hace uso de una regla de prioridad estática y otra regla de prioridad dinámica. En este artículo no se identifica el uso de manejo del inventario, el uso de k equipos heterogéneos, la capacidad del equipo de manejo de materiales, ni se identifica que se tenga en cuenta la minimización de la tardanza de las órdenes de los clientes.

1.7.2 (Azadnia et al., 2013)

Buscan minimizar la tardanza promedio de todos los pedidos en un almacén enfocándose en cuatro fases. La primera fase, usa la minería de reglas de asociación para calcular las asociaciones de las órdenes con su fecha de vencimiento; la segunda fase, usa un modelo de programación binaria entera para maximizar la cantidad de asociaciones entre las órdenes de cada lote; la tercera fase, usa un algoritmo genético integrado con el problema del vendedor viajero para encontrar la ruta de viaje más conveniente que permita desarrollar la operación de preparación de pedidos en el almacén; la cuarta, utiliza un algoritmo genético para secuenciar los lotes construidos con el fin de minimizar la tardanza promedio de todos los pedidos. En este artículo, no se tiene en cuenta el manejo del inventario, el uso de k equipos heterogéneos ni la capacidad del equipo de manejo de materiales.

1.7.3 (Bozer & Kile, 2008)

Investigan la conformación de lotes de pedidos. Para esto, proponen un modelo de Programación Entera Mixta (*MIP*) que les permita minimizar el tiempo de viaje más largo entre los lotes resultantes. Una desventaja que se evidencia en esta investigación es que la carga computacional es alta. Por eso, los autores sugieren hacer otros estudios empleando otro tipo de heurísticos que puedan hallar buenas soluciones teniendo presente el consumo computacional. Adicionalmente, no se identifica que usen k equipos heterogéneos, ni la capacidad del equipo de manejo de materiales.

1.7.4 (Choy et al., 2014)

Proponen un modelo de evaluación para el análisis de confiabilidad de un sistema de almacén. Ellos estudian el rendimiento de un almacén cuando aplican diferentes combinaciones de políticas dentro de cada operación en este. Para ello, diseñan una simulación que encierra las características de una empresa de Hong Kong de servicios logísticos con fines evaluativos. Entre sus conclusiones se encuentra que cuando se emplean reglas en la zonificación y conformación de lotes, aumenta la confiabilidad del almacén, así este tenga un volumen grande de órdenes a procesar en el día. Además, identifican que el rendimiento del almacén no permanece constante y depende en gran medida de las combinaciones de políticas que se utilizan en las operaciones. En este artículo los autores no tienen en cuenta el manejo del inventario, el uso de k equipos heterogéneos, la capacidad del equipo de manejo de materiales y el uso de metaheurísticos.

1.7.5 (Henn, 2012)

Estudian el problema de conformación de lotes en la operación de preparación de pedidos en un sistema de órdenes en línea donde buscan minimizar el máximo tiempo de realización de los pedidos de los clientes que van llegando al almacén. Para esto, se debe tener en cuenta que los pedidos solo se reciben en un tiempo determinado del día y que no se conoce lo que llegara. Los autores, plantean que al tener una ventana de tiempo limitada para la recepción de pedidos deben definir la cantidad de tiempo que un pedido de un cliente esperará para ser aceptado. Dicha definición, indican los autores, le permitirá saber si la orden será procesada o será tenida en cuenta al final del último pedido que se reciba por parte del cliente. En este trabajo, no se tiene en cuenta el uso de k equipos de manejo de materiales heterogéneos y el manejo de inventario.

1.7.6 (Ho et al., 2008)

Los autores se basan en el estudio de (Ho & Tseng, 2006), para investigar más métodos que les sirvan para la conformación de lotes. Los métodos de conformación de lotes que

proponen, son compuestos por una regla de selección de la orden semilla y una regla de selección de pedidos de acompañamiento. En total, ellos crean once reglas para la selección de la orden semilla y catorce reglas de la selección de pedidos de acompañamiento. Para probar el rendimiento de las reglas ellos emplearon una simulación, con la cual muestran que se puede minimizar la distancia recorrida por los operadores. En esta investigación, no se tiene en cuenta la fecha de vencimiento de los productos, la minimización de la tardanza, el uso de k equipos heterogéneos y el manejo de la disponibilidad de inventario.

1.7.7 (Henn, 2012)

Estudian dos métodos el *Variable Neighborhood Descent* y *Variable Neighborhood Search* y muestran cómo se pueden aplicar con el fin de minimizar la tardanza total de un determinado conjunto de pedidos. Los autores usan experimentos numéricos para medir el rendimiento de los dos métodos sobre diferentes tipos de problemas. Los resultados que obtienen les ayudan a mostrar que los métodos que formularon logran conseguir mejores soluciones, lo que facilita que la operación de preparación de pedidos sea más eficiente en el almacén. En esta investigación, los autores no tienen en cuenta el manejo del inventario y no especifican si los k equipos de manejo de materiales que utilizan son heterogéneos.

1.7.8 (Henn & Schmid, 2013)

Estudian cómo los metaheurísticos se pueden utilizar con el fin de minimizar la tardanza total de un conjunto de pedidos. Para ello, usan tres metaheurísticos, siendo el primero el de *Iterated Local Search*, el segundo, que fue inspirado en el *Attribute-Based Hill Climber* y el tercero que se basa en el principio simple de Búsqueda Tabú. Los autores usan experimentos numéricos para probar el rendimiento de estos metaheurísticos y demuestran que los métodos propuestos proporcionan soluciones que facilitan que la operación de preparación de pedidos sea más eficiente. Adicionalmente, indican que las soluciones que obtienen muestran mejoradas en un 46% en promedio, comparándolas con otras heurísticas constructivas estándar, tales como la regla de la Fecha de

Vencimiento Más Temprana. Ellos en su investigación no tienen en cuenta el manejo de inventario y no especifican si los equipos de manejo de materiales son heterogéneos.

1.7.9 (Hsieh & Fan, 2010)

Desarrollan una nueva heurística que les facilita la conformación de lotes en un sistema de preparación de pedidos en un centro de distribución. Ellos usan un mapa auto organizado (*SOM*) para agrupar los pedidos con mayor relación en un lote y así ayudar a reducir la distancia total recorrida en la operación de preparación de pedidos. Adicionalmente, los autores plantean que es muy importante saber el número de *SKU's* en una orden y el número de pasillos cubiertos por estas para poder realizar la conformación de lotes. También indican, que el aumento de *SKU's* en una orden afectan los pesos óptimos de la heurística *SOMB* que estudian. Para probar el rendimiento del heurístico *SOMB*, ellos formulan un experimento donde usan una combinación de 18 políticas para la asignación de almacenamiento, para la conformación de lotes y para el ruteo de los despachadores con las que buscan reducir la distancia recorrida en la preparación de pedidos e incrementar la utilización del vehículo de recogida. Los criterios que usaron para medir el rendimiento del heurístico en su estudio fueron la distancia total del viaje, el tiempo de corrida de la *CPU* y el promedio de la utilidad del vehículo de carga. Ellos en su investigación, no tienen en cuenta el uso de k equipos de manejo de materiales heterogéneos, el manejo de inventario, la fecha de vencimiento de los pedidos, ni la minimización de la tardanza.

1.7.10 (Hsieh & Huang, 2011)

Ellos estudian dos nuevas heurísticas para la conformación de lotes llamados *K-means batching* (*KMB*) y *Self-organisation Map Batching* (*SOMB*), con las cuales buscan minimizar la distancia total de recorrido de la operación de preparación de pedidos en un almacén. Ellos encuentran que tanto *KMB* como *SOMB* tienen un rendimiento superior en cuanto a la minimización de la distancia total del recorrido, a la utilidad media de los vehículos de despacho y a los tiempos de ejecución de la *CPU*. También, investigan el rendimiento global de los sistemas de preparación de pedidos que integran la asignación de almacenamiento, la conformación de lotes y el enrutamiento del recolector por medio

de la combinación adecuada de diferentes políticas. Ellos realizan un análisis de sensibilidad para encontrar la relación entre las estrategias que mejoran el rendimiento en las operaciones. Ahora, para la experimentación utilizan cuatro tipos de órdenes que difieren en el número de SKUs dentro de cada pedido, las cuales son pequeñas, medianas, grandes y mixtas. En su estudio, no dejan claro si usan k equipos de manejo de materiales heterogéneos y si la velocidad del equipo de manejo de materiales es constante o no; además no tienen en cuenta el manejo de inventario ni la minimización de la tardanza.

1.7.11 (Rubrico et al., 2006)

Ellos se enfocan en investigar la operación de preparación de pedidos en un almacén. Para ello designan un número de agentes, donde cada uno tiene su propio conjunto de secuencias (rutas de recogida). Ellos proponen crear un procedimiento de simulación basada en el despacho para resolver el problema. El objetivo es reducir los retrasos creados por las colas formadas por los agentes, para esto emplean tres reglas que buscan distanciar los agentes entre sí, dichas reglas son: la *No Common Sector (NCS)*, *Free Sector Start (FSS)* y *Least Cost Start (LCS)*. Antes de finalizar su investigación, los autores realizan varias simulaciones estadísticas sobre un almacén que demuestran que dicho procedimiento puede minimizar el tiempo promedio de operación de los agentes con respecto a cuándo no se tiene implementada dicha política de despacho para los agentes. En su investigación, los autores no tienen en cuenta el uso de k equipos de manejo de materiales heterogéneos y el manejo de inventario.

1.7.12 (Rubrico et al., 2011)

Ellos investigan una solución para un problema de reprogramación dinámica para la preparación de pedidos donde van llegando de manera aleatoria nuevas órdenes y donde las ordenes que ya hubieran sido asignadas quedan estáticas en la programación. Para esto, usan un procedimiento de búsqueda local donde proponen dos variaciones a un esquema de planificación estática: uno basado en *steepest descent insertion (OR1)*, y el otro en *multistage rescheduling (OR2)*. Luego, los autores realizan una serie de

experimentos estadísticos teniendo en cuenta datos reales tomados en un proceso de preparación de pedidos e indican que las metodologías que proponen son competitivos y muestran que al usar *OR1* se obtienen los mejores resultados en promedio y que al usar *OR2* se puede reducir el tiempo de recogida cuando el dinamismo va de bajo a moderado. En su investigación, ellos no tienen en cuenta k equipos heterogéneos y el manejo del inventario.

1.7.13 (De Koster, Van der Poort, & Wolters, 1999)

Ellos estudian el problema de la conformación de lotes en almacenes y observan que al utilizar la conformación por lotes en el proceso de preparación de pedidos el tiempo de viaje se reduce en comparación al uso de la política *First-come, First-Served (FCFS)*. En su trabajo, evalúan dos algoritmos el: *Seed algorithms* y el *Savings algorithms* y miden el rendimiento de estos usando dos estrategias de enrutamiento: el *S-Shape* y *Largest Gap* con los cuales comparan el tiempo de viaje y el número de lotes conformados. Ellos concluyen que *Seed algorithms* se comporta mejor si se usa con la estrategia *S-Shape* y se posee una buena capacidad del equipo de manejo de materiales e indican que el *Savings algorithms* se comporta mejor si se usa la estrategia de *Largest Gap* y se puede usar con una capacidad reducida del equipo de manejo de materiales.

1.7.14 (Piroird & Dale, 1998)

Ellos estudian el proceso de cumplimiento de pedidos particularmente con los plazos establecidos de entrega internos y externos en una fábrica de productos químicos especializados y encuentran que en la fábrica no cuenta con políticas para los plazos de entrega, no tienen manejo de inventario, no manejan negociación de plazos de entrega y no tienen control sobre el tiempo de revisión de los productos. Ellos plantean que para solucionar esto se debe tener en cuenta tres temas: el establecimiento de prioridades, la capacidad y la flexibilidad de los plazos de entrega. En este estudio, no se tiene en cuenta la conformación de lotes, el uso de k equipos heterogéneos y el uso de metaheurísticos.

1.7.15 (Gómez, 2015)

Estudia el problema de la conformación de lotes con ruteo para la operación de acomodo y preparación de pedidos considerando variables como: K equipos de materiales (*EMMateriales*) heterogéneos, n productos, m posiciones de almacenamiento y la condición de disponibilidad del inventario. Dicho estudio, produce como resultado un *WMS* que integra la conformación de lotes en la operación de acomodo, utilizando una regla de prioridad (*FCFS, First Come First Served*) y un metaheurístico de búsqueda de vecindarios inteligentes (*INS, Intelligent Neighborhood Search*). Adicionalmente, integra el ruteo en la operación de acomodo, utilizando variaciones del metaheurístico Búsqueda Tabú, tales como: una Búsqueda Tabú clásica (*RABTCLA*), una Búsqueda Tabú 2-Opt Intercambio, una Búsqueda Tabú *RABTDI* 2-Opt Inserción, una Búsqueda Tabú *RABTDI* 3-Opt Intercambio, una Búsqueda Tabú *RABTDI* 3-Opt Inserción y una búsqueda usando una matriz de intensificación y frecuencias (*RABTIN*). En cambio, para la conformación de lotes en la operación de preparación de pedidos, integra al *WMS*, un metaheurístico *PNS (Picking Neihgborhood Search)* y la regla *PEPR* (Primero en Entrar al CEDI, Primero a Recoger) y para el ruteo de la operación de preparación de pedidos, integra al *WMS* variaciones del metaheurístico de Búsqueda Tabú, tales como: una Búsqueda Tabú clásica (*RUPBTCLA*), una Búsqueda Tabú *RUPBTDI* 2-Opt Inserción, una Búsqueda Tabú *RUPBTDI* 3-Opt Intercambio, una Búsqueda Tabú *RUPBTDI* 3-Opt Inserción y una búsqueda usando una matriz de intensificación y frecuencias (*RUPBTIN*). Al realizar un diseño de experimentos, él determina que para la conformación de lotes, el metaheurístico *INS (Intelligent Neighbourhood Search)* y *PNS (Picking Neighbourhood Search)*, obtienen los mejores resultados tanto para la operación de acomodo como para la preparación de pedidos y que para el ruteo de estas operaciones, los mejores resultados son los que se obtienen por medio de los metaheurísticos híbridos *RABTIN* y *RUPBTDI* 3-Opt Inserción. En este estudio, no se identifica que se aborde el tema de la programación y secuenciación de las órdenes de los clientes en el almacén de tal manera que se minimice la tardanza de estas, por lo que se identifica una oportunidad investigativa.

1.8 Síntesis de datos

Con la revisión sistemática de la literatura y que se resumen en la Tabla 1-6, se da respuesta a las preguntas de investigación planteadas en tesis. Los siguientes son los resultados obtenidos: para la primera pregunta, se identifica que la operación de mayor interés investigativo en los almacenes corresponde a la de preparación de pedidos (Rosenwein, 1996), (Ruben & Jacobs, 1999), (Van Den Berg, 1999), (Gómez, 2015) y para la segunda pregunta, se identifica que existen diversos heurísticos, métodos y reglas en la literatura científica con los cuales se aborda el tema de la programación y secuenciación de productos en el almacén. Sin embargo, no se identifican modelos que incorporen en la solución del problema, el uso simultáneo de los siguientes elementos: n productos, k equipos heterogéneos, la restricción de disponibilidad del inventario y un almacén con m posiciones de almacenamiento que permitan obtener una tardanza mínima haciendo uso de un metaheurístico.

A partir de los resultados obtenidos en la revisión del estado del arte, es necesario definir los siguientes objetivos para superar las debilidades encontradas en la literatura.

1.9 Objetivos

1.9.1 Objetivo General

Modelar un metaheurístico que facilite la resolución del problema de programación de la preparación de pedidos, considerando los tiempos de entrega, K equipos heterogéneos y restricciones de la operación del almacén.

1.9.2 Objetivos específicos

1. Analizar el estado del arte de la programación de la preparación de pedidos en un almacén para identificar un problema relevante.

2. Definir los componentes del planteamiento del problema tales como la función objetivo y las restricciones que permitan el modelamiento de la problemática identificada.
3. Seleccionar el metaheurístico más apropiado que se ajuste a las características del problema identificado.
4. Desarrollar un escenario de aplicación que permita la implementación del metaheurístico seleccionado al problema definido.
5. Programar el metaheurístico seleccionado que permita solucionar el problema planteado.
6. Validar los resultados arrojados por el metaheurístico en el escenario construido.

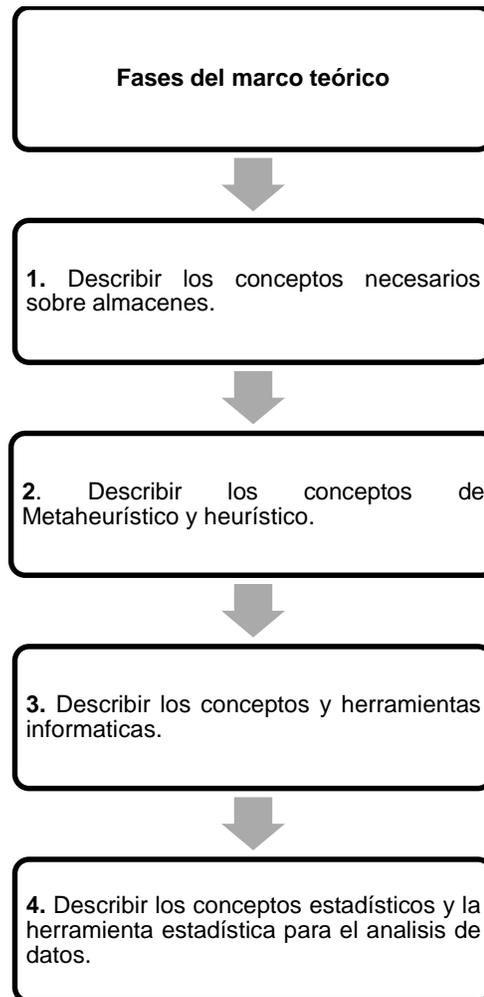
2. Marco teórico

En este capítulo, se muestran los principales conceptos y herramientas necesarias para desarrollar este trabajo de grado. Entre los temas que se abordan se encuentra: cadena de suministro, el Centro de Distribución (CEDI), la programación de operaciones, Heurísticos, Metaheurísticos, herramientas de programación (Lenguaje y Entorno de Programación, *Framework OpenTS*[®], Base de Datos), conceptos estadísticos y herramienta estadística (Pruebas de Hipótesis, Prueba *t* Pareada, herramienta estadística).

2.1 Flujo del marco teórico

Para la elaboración de este marco teórico se utilizaron tres fases: primero, se muestran los conceptos de cadena de suministro, centro de distribución, programación de operaciones, esto con el fin de describir los conceptos que se requiere en esta tesis, con respecto al tema de almacenes; segundo, se describen los conceptos de metaheurístico y heurístico para comprender cuál es más pertinente para el desarrollo de esta tesis; tercero, se describen los conceptos y herramientas informáticas que se requieren para poder desarrollar la tesis y por último, se describen los conceptos estadísticos y la herramienta que se necesita para realizar la evaluación de los resultados de la tesis. Estas fases, se muestran en la Figura 2-1.

Figura 2-1 Fases del marco teórico.



Fuente: Elaboración propia.

2.2 Cadena de suministro

Existen varias definiciones para el concepto de cadena de suministro. El cual, es el punto de partida para la contextualización del tema abordado en este trabajo de grado. Algunas de las definiciones que se encuentran en la literatura son:

“Red de instalaciones (almacenes, terminales, empresas, puertos, tiendas o hogares), vehículos (camiones, trenes, aviones o buques) y sistemas de información logísticos que

permiten la conexión con el proveedor de la empresa y los consumidores” (Frazelle, 2002).

“Es un conjunto de actividades funcionales (transporte, control de inventarios, etc.) que se repiten muchas veces a lo largo del canal de flujo, mediante las cuales la materia prima se convierte en productos terminados y se añade valor para el consumidor.” (Ballou, 2004).

“Una cadena de suministro consiste de todas las partes involucradas directa o indirectamente en satisfacer a un cliente. Esto implica incluir a transportistas, almacenes, intermediarios y a los clientes. Algunas etapas de la Cadena de Suministro son: Clientes, Minoristas, Distribuidores, Manufactureros, Proveedores de materia prima. El objetivo de la cadena de suministro es maximizar el valor global generado.” (Chopra & Meindl, 2012).

“Un conjunto de tres o más empresas conectadas o relacionadas con flujos de productos, servicios, finanzas e/o información desde el suministro hasta el cliente final” (Mentzer, 2004).

“Un conjunto de actividades funcionales que incluye redes de instalaciones, vehículos y sistemas de información logísticos que permiten conectar proveedores, fabricantes y distribuidores, con el fin que se transformen en productos terminados que intenten satisfacer las necesidades de los clientes” (Correa & Gómez, 2009).

2.3 CEDI (Centro de Distribución)

Al realizar una búsqueda en la literatura, se encuentra una amplia gama de definiciones para el concepto de centro de distribución (CEDI), algunas de estas son:

“un CEDI se puede definir como la parte de la cadena de suministro en la que se almacenan los productos provenientes de los puntos de producción hacia los puntos de consumo y donde se proporciona información a la empresa sobre el estado y disposición de los artículos almacenados” (De la Figuera, 2005).

“el CEDI es un enlace esencial dentro de la cadena de suministro, ya que es el intermediario entre la producción, compras, control de inventario, transporte y ventas, y se ejecuta sin problemas en colaboración con otros departamentos y con otros miembros de la cadena de suministro” (Van Den Berg, 2007).

“El CEDI es un proceso que permite regular los flujos de información y productos de los actores de la cadena de suministro a través de la planeación, implementación, control y mejoramiento de las operaciones de recepción, acomodo, almacenamiento, inventarios, preparación de pedidos y despacho.” (Correa, Rodríguez, & Gómez, 2014).

2.3.1 Elementos de un CEDI

Luego de entender lo que es un centro de distribución, es importante identificar cuáles son los elementos necesarios en este para su operación y funcionamiento. De acuerdo (Waters, 2003), estos elementos son definidos como: Muelle de llegada, es el lugar donde se entregan, comprueban y ordenan los productos que suministran los proveedores; área de almacenamiento, es el lugar donde los productos se almacenan e incrementan la cantidad del inventario; muelle de salida, es el lugar donde se consolidan las órdenes de los clientes para su respectivo envío; sistemas de almacenamiento y manejo de materiales, son los encargados del transporte de los productos en el CEDI; sistema de información, se encarga de registrar la ubicación de almacenamiento de los productos, la llegada de productos de los proveedores, el despacho de órdenes de los clientes, entre otras características propias del manejo de la información.

2.3.2 El CEDI como un tipo de almacén

En la literatura, se encuentran varios autores tales como (Peter Baker, 2004), quien indica que los centros de distribución (CEDI) se pueden considerar como un tipo de almacén que sirven de centro de consolidación, Centros de crossdocking y almacenamiento de productos terminados y como (Baker & Halim, 2007), que comentan que un almacén visto como CEDI se puede considerar como un punto centralizador en la

cadena de suministro que permite consolidar productos, agregar servicios de valor y despachar a los clientes de una forma eficiente.

2.4 Operaciones en un almacén

Existen cuatro operaciones básicas que se realizan en un almacén y que se consideran como los más importantes, las cuales son: recepción, almacenaje, preparación de pedidos y despacho, (Frazelle & Sojo, 2006). Estas son descritas a continuación: recepción, se centra en recibir los productos que llegan en camiones al CEDI, donde los operarios se encargan de revisarlos cuidadosamente verificando la cantidad y calidad de estos; almacenaje, se basa en transportar los productos de la zona de recepción a la zona de almacenamiento donde se procede con su almacenaje; preparación de pedidos, se basa en recolectar los productos ubicados en las estanterías o zonas de almacenamiento que han sido solicitados en las órdenes realizadas por los clientes. Se debe tener en cuenta las condiciones pactadas con el cliente respecto a referencias, cantidades y calidad (Henn, 2012); despacho, se basa en el envío de los pedidos a los clientes.

Entre estas operaciones, la preparación de pedidos es la más costosa en un almacén de acuerdo a lo indicado por (Rosenwein, 1996), (Ruben & Jacobs, 1999), (Van Den Berg, 1999). Esto también puede verse reflejado en lo dicho por (Coyle, Bardi, & Langley, 2003) quienes consideran que los costos operativos para la preparación de pedidos pueden llegar a ser tan altos como de un 65%, además de que indican que una consecuencia importante de esto es la cantidad de trabajo humano requerido.

Adicionalmente, a las operaciones antes mencionadas (R. De Koster, Le-Duc, & Roodbergen, 2007) comenta que existe la operación de cross-docking que se centra en recoger o trasladar los productos desde la plataforma de entrada (donde se reciben los productos) hasta la plataforma de salida (donde se despachan los productos al cliente), evitando almacenar y preparar la orden.

2.5 Recursos en un Almacén

En un almacén existen tres elementos esenciales para su buen funcionamiento de acuerdo a (Frazelle & Sojo, 2006). Estos elementos son: los equipos, la infraestructura del almacén y el recurso humano. A continuación, se describen cada uno de estos:

Equipos: Hacen referencia a equipos de procesamiento de datos, a equipo de almacenamiento (estanterías) y a equipos de manejo de materiales. De acuerdo a (Ballou, 2004), estos equipos pueden ser de uso manual, asistido con motor y mecanizado totalmente. Entre estos dispositivos se pueden encontrar para la extracción de productos los siguientes:

- Montacargas de contrapeso.
- Equipos pórticos (Straddle Trucks).
- Equipos pórticos extensibles (Straddle Reach Trucks).
- Carretillas tipo Walkie.
- Montacargas tipo Turret.
- Equipos de carga lateral.
- Vehículos híbridos de almacenamiento - extracción (S/R).
- Vehículos automatizados de almacenamiento - extracción (ASR).

Infraestructura: Hace referencia a la estructura física del almacén. La cual tiene que ser distribuida pensando siempre en los costos (Tompkins & Smith, 1998).

Recurso Humano: Son todas las personas que trabajan para que el almacén opere de manera adecuada. Hay que tener en cuenta que a mayor número de personas laborando mayor es el costo operacional, (Tompkins & Smith, 1998)

2.6 Programación de operaciones

De acuerdo a lo dicho por (Machuca, 1995), el objetivo de la programación de operaciones es determinar cuáles operaciones se deben realizar sobre cada uno de los

pedidos durante cada momento que pasa en el horizonte de la planificación. Todo esto, orientado a cumplir las fechas de entrega planificadas haciendo uso óptimo de los recursos e inventario disponible.

2.7 Datos de un problema de la programación

De acuerdo a lo dicho por (Pinedo, 2009) para un problema de programación se debe considerar el número de trabajos n y el número de máquinas m ya que siempre es un número finito con el que se cuenta. Adicionalmente, indica que para denotar un trabajo se puede usar la letra j y para designar una máquina se puede usar la i . También indica, que de los trabajos j se puede sacar los siguientes datos: tiempo de procesamiento, que es el tiempo que toma el trabajo j en ejecutarse en la máquina i ; fecha de lanzamiento, que es la fecha en la que se lanza el trabajo j . Es decir, es la fecha con la cual inicia el trabajo j su procesamiento; fecha de vencimiento, que es la fecha en la que se debe tener listo el trabajo j . Ya que esta, es la fecha que se pactó con el cliente. Si el trabajo finaliza después de esta fecha se incurrirá en una penalización y peso, que es el peso que se da a un trabajo j para dar prioridad a un trabajo sobre otro.

2.8 Representación de un problema de programación

De acuerdo a lo dicho por (Pinedo, 2009), un problema de programación se representa con una tripleta $\alpha | \beta | \gamma$. Donde α es el ambiente de la máquina, β son las características y restricciones de procesamiento y donde γ describe el objetivo a ser minimizado.

2.8.1 Ambientes de máquinas

Los ambientes de máquinas más conocidos de acuerdo a lo expresado por (Pinedo, 2009) son:

Máquina Individual: Es el ambiente donde solo se encuentra una máquina. Se considera también como un caso especial para los ambientes de máquinas más complicadas.

Máquinas en paralelo: Es una generalización del ambiente de una sola máquina. Dichas máquinas se encuentran paralelamente y pueden ser idénticas, lo que hace que un trabajo puede ser procesados en cualquiera de ellas o puede que no sean idénticas lo que hace que ciertos trabajos solo puedan ser procesado en algunas de ellas y en otras no.

Flow Shop: se le conoce al flujo de los trabajos que siguen una misma ruta de procesamiento a través de máquinas en serie. Es decir, al finalizar en una maquina el procesamiento de un trabajo este pasa a la cola la de maquina siguiente de su ruta.

Flexible Flow Shop: Es una generalización del Flow Shop y es donde se encuentra en cada etapa de la serie de máquinas un determinado número de máquinas en paralelo.

Job Shop: Es una generalización del Flow Shop donde los trabajos siguen diferentes rutas de procesamiento. En este ambiente se puede presentar o no, que un trabajo visite varias veces una máquina, lo que aumenta la complejidad y se conoce como recirculación.

2.8.2 Características y restricciones de procesamiento

(Pinedo, 2009) menciona que algunas características y restricciones que se pueden encontrar en los problemas de programación, unas de ellas son:

Restricción de precedencia: Solo podrá iniciarse un trabajo si previamente ya se completó un conjunto de trabajos.

Restricciones de elegibilidad de la máquina. Se presenta cuando se trabaja sobre un entorno de máquinas paralelas diferentes y se tienen trabajos que no pueden ser procesados en cualquiera de estas máquinas, ya que solo se puede hacer en un conjunto particular de ellas.

Interrupción de los trabajos: Se presenta cuando el procesamiento de un trabajo es detenido en una máquina por decisión del programador. Esta decisión puede tomarse cuando de pronto existe otro trabajo con mayor prioridad.

Procesamiento por lotes: Se da cuando una máquina puede procesar cierto número de trabajos a la misma vez y todo el lote se terminara de procesar sólo cuando termine el último trabajo de dicho lote, por ende el tiempo de finalización de todo el lote está dado por el trabajo con más largo tiempo de procesamiento.

2.8.3 Objetivos para el problema de programación

Algunos de los objetivos que se pueden plantear para resolver el problema de programación de acuerdo a lo indicado por (Pinedo, 2009) son:

Objetivo de rendimiento y de makespan: El rendimiento se mide por la velocidad de procesamiento, así pues para maximizar la tasa de rendimiento de una instalación se debe maximizar la tasa de rendimiento de los cuellos de botella formados en las diversas máquinas. Para esto, se debe equilibrar la carga sobre las diferentes máquinas y minimizar la suma de todos los tiempos de preparación. Por esto, el objetivo del makespan, que busca minimizar el tiempo finalización del último trabajo en un sistema, es muy similar al objetivo de rendimiento, ya que ambos buscan de cierta manera conseguir lo mismo.

Objetivo relacionado con la fecha de entrega: Se pueden distinguir varios objetivos a buscar. El primero, se centra en minimizar la máxima demora, la cual se puede hallar buscando la diferencia entre el tiempo de terminación de un trabajo con su fecha de entrega lo cual puede arrojar un resultado con valor positivo o no. El segundo, se centra en minimizar el número de trabajos que llegan tarde. El tercero, se centra en minimizar la tardanza total o a la tardanza promedio, la cual se calcula como el máximo entre la demora de un trabajo y cero, lo que significa que el resultado nunca tomara valores negativos. La demora y la tardanza se pueden considerar como funciones de penalidad. Pero adicionales a estas, también se puede encontrar otra que se llama la unidad de

penalidad que se calcula como uno si el tiempo de terminación de un trabajo es mayor que la fecha de vencimiento de este o cero en caso contrario.

2.9 Heurísticos

En la literatura se encuentran un gran número de definiciones del concepto de heurística, entre ellas se encuentran:

“Una heurística tiene como objetivo estudiar los métodos y reglas de descubrimiento o ayudar en la resolución de problemas, que es un proceso sistemáticamente tratado para alcanzar un objetivo preconcebido, pero no inmediatamente alcanzable”, (Zanakis & Evans, 1981).

“Una técnica heurística (o simplemente una heurística) es un método que busca buenas soluciones (es decir, soluciones cercanas al óptimo) a un costo computacional razonable sin poder garantizar optimalidad”, (Reeves, 1995).

Dadas las definiciones anteriores se concluye que una heurística es un método de búsqueda simple que puede aproximarse a una buena solución pero no garantiza la optimalidad.

2.10 Metaheurístico

En la literatura se encuentran varias definiciones para el término metaheurístico entre ellas se encuentran:

“Una estrategia maestra que guía y modifica otras heurísticas para producir soluciones más allá de aquéllas que son normalmente generadas en una solicitud por optimalidad local. Las heurísticas guiadas por tal metaestrategia pueden ser procedimientos de alto nivel o nada más que una descripción de movidas disponibles para transformar una solución en otra, junto con reglas de evaluación asociadas” (F. W. Glover & Laguna, 1997).

"Una estructura para producir heurísticas" (Lourenço, 2005).

Dicho lo anterior se puede distinguir como los Metaheurísticos son heurísticas de mayor nivel. Mostrando que puede desarrollar una búsqueda más compleja, que aunque no garantice la solución óptima si proporciona soluciones muy aproximadas. Este tipo de características los han convertido en un método muy atractivo en la literatura ya que permite solucionar problemas de tipo combinatorial o NP-hard.

2.11 Clasificación de las heurísticas y metaheurísticos

Dependiendo de la búsqueda, los heurísticos y metaheurísticos pueden considerarse con alcance local o global, siendo local cuando se parte de una solución inicial y se van realizando cambios que permiten encontrar nuevas soluciones con las cuales se hacen comparaciones con la mejor solución anterior, si la solución nueva resulta siendo mejor se almacena y si no se vuelve a la solución inicial; globales cuando buscan salir de los máximos locales y se tiene una exploración mayor en la zona a buscar.

El tipo de búsqueda que puede realizarse con ayuda de los metaheurísticos y heurísticos puede ser de manera Individual o poblacional. Para el tipo individual un ejemplo puede ser la Búsqueda Tabú cuando se realiza la búsqueda de solución en solución y para el poblacional un ejemplo puede ser un algoritmo genético que utiliza una búsqueda en paralelo de soluciones.

Los metaheurísticos son inspirados en situaciones basadas en la naturaleza, en el aprendizaje, en la evolución, en la gravedad, entre otras, (Zanakis & Evans, 1981).

2.12 Herramientas de programación

Para poder realizar el desarrollo de esta tesis, es necesario profundizar en los términos y herramientas que se mencionan en esta sección.

2.12.1 Lenguaje y entorno de programación

Los programas informáticos son desarrollos elaborados por programadores para resolver problemas de la vida cotidiana, (Morris, 1994). Para desarrollar un programa, se requiere seleccionar un lenguaje de programación, el cual consiste de líneas de código que contienen la lógica del programa que resolverá el problema identificado, y un entorno de programación con el cual se pueda ir mejorando y corrigiendo los errores que se presenten en las líneas de código.

Java[®] es un lenguaje de programación que fue comercializado por primera vez en 1995 por la empresa Sun Microsystems[®] y luego fue adquirido por la empresa Oracle[®]. Este lenguaje permite hacer desarrollos tecnológicos en múltiples plataformas (Linux[®], Windows[®], Pseries[®], Android[®], etc), lo cual hace que sobresalga sobre otros tipos de lenguajes de programación, además, de ser un lenguaje orientado a objetos, lo que permite reutilizar funcionalidades dentro del desarrollo. Por lo cual, se selecciona para desarrollar esta tesis, (Deitel, 2004), (Oracle Corporation, 2016a).

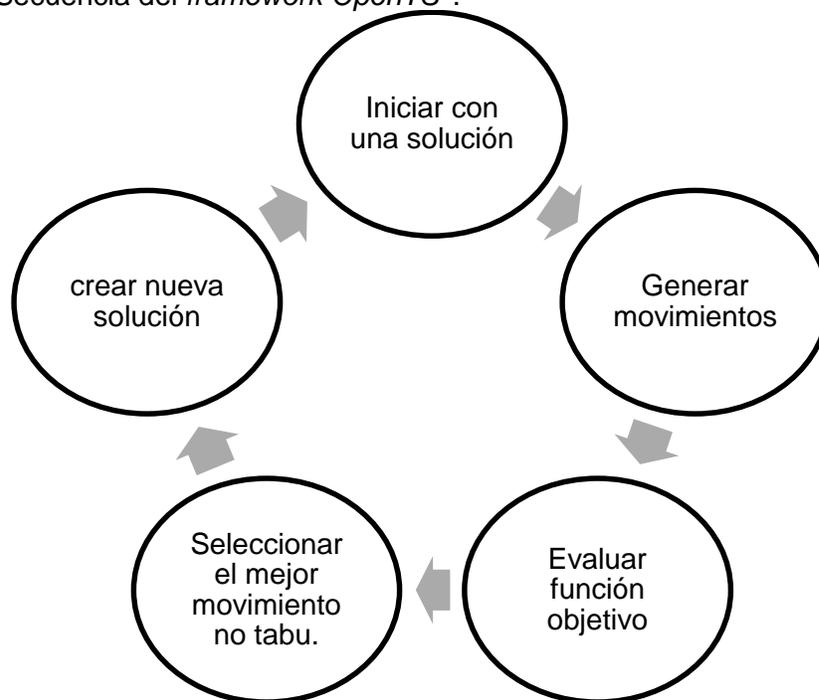
Existen diversos entornos de programación, tales como: Eclipse[®], Netbeans[®] y JCreator[®], los cuales, permiten crear, editar y ejecutar programas. Sin embargo, para el desarrollo de esta tesis, se opta por usar Netbeans[®], ya que tiene una interfaz gráfica fácil de usar por el programador, además de que permite identificar rápidamente en qué parte de las líneas de código de un programa se presenta un error, dando información detallada acerca de estos. (Oracle Corporation, 2016b), (Xinox Software, 2000), (Foundation Eclipse, 2016).

2.12.2 Framework OpenTS[®]

Para el desarrollo de esta tesis, se investiga el *framework OpenTS[®]*, el cual surge como parte del proyecto *Computational Infrastructure for Operations Research (COIN-OR)*, que consiste en desarrollar programas de código abierto para la investigación de operaciones. Este *framework*, permite trabajar con el metaheurístico Búsqueda Tabú para resolver problemas de diferentes tipos, tales como: enrutamiento de vehículos, el problema de asignación, el problema del agente viajero, entre otros (COIN-OR, 2016).

La secuencia que sigue la Búsqueda Tabú del *framework OpenTS*[®] es la siguiente: primero, se parte de una solución inicial o una solución actual que es encontrada como la mejor solución en una iteración pasada del algoritmo; segundo, se crea una lista de movimientos (vecindario); tercero, se evalúa el valor de la función objetivo con base a los elementos de la lista de movimientos; cuarto, se determina el mejor movimiento con el uso de una lista tabú; finalmente, con este movimiento, se crea la nueva solución actual. (COIN-OR, 2016). Esta secuencia, se muestra en la Figura 2-2

Figura 2-2 Secuencia del *framework OpenTS*[®].



Fuente: (COIN-OR, 2016).

2.12.3 Base de datos

Cuando se requiere almacenar información de un problema, por medio de un programa computacional, es necesario usar bases de datos. Las cuales están formadas por los siguientes elementos: primero, tablas que representan objetos del mundo real, como por ejemplo, puede existir una tabla llamada Estudiante, otra tabla llamada Materia; segundo, por atributos que representan las características de los objetos (tablas), para el ejemplo, se tiene que en la tabla Estudiante, sus atributos serían cedula, nombre, teléfono y para

la tabla Materia sus atributos serían código y nombre; tercero, registros que se conocen como tuplas de datos que se almacenan en la tabla, para el ejemplo un registro de la tabla Estudiante estaría conformada por el conjunto de datos cedula, nombre y teléfono de un estudiante y para la tabla Materia un registro estaría formado por un código y nombre de una materia; finalmente, las relaciones entre tablas, por ejemplo la tabla Estudiante tiene relación con la tabla Materia, ya que un estudiante en una universidad puede matricular varias materias. Estas relaciones, sirven para realizar consultas SQL con el fin de generar reportes de interés para la toma de decisiones en una organización determinada, (Kroenke, 2003).

Ahora, para poder crear, diseñar, almacenar y administrar una base de datos, se requiere el uso de un Sistema Gestor de Base de Datos (*SGBD*). En la actualidad, existen diversos *SGBD* tales como: DB2[®], Microsoft SQL server[®], MySQL[®], Microsoft Acces[®], PostgreSQL[®], Oracle Database[®], entre otros. Para este trabajo, se decide optar por el *SGBD* Oracle[®] ya que su instalación es rápida, tiene una interfaz simple para el programador, tiene buen soporte, es multiplataforma, se integra con Java[®], es utilizado comercialmente por grandes empresas a nivel mundial. (Oracle Corporation, 2016c), (Microsoft, 2016a), (IBM, 2016), (Oracle Corporation, 2016d), (Microsoft, 2016b), (PostgreSQL Global Development Group, 2016).

2.13 Conceptos estadísticos y herramienta estadística

Para poder analizar los datos que arrojará el desarrollo de esta tesis, se deberá utilizar un análisis estadístico, el cual permitirá probar si los resultados obtenidos en la tesis son mejores en comparación con de los resultados obtenidos con otro método. Debido a que se planea hacer una comparación entre dos métodos para una misma población de datos, es necesario profundizar sobre las pruebas de hipótesis y sobre el estadístico de *t* pareada. Además, de conocer la herramienta estadística que será útil para el desarrollo de esta tesis.

2.13.1 Prueba de hipótesis

“Una hipótesis estadística es un enunciado o afirmación ya sea acerca de los parámetros de una distribución de probabilidad o de los parámetros de un modelo. La hipótesis refleja alguna conjetura acerca de la situación del problema” (Montgomery, 2008). Existen dos tipos de hipótesis, la hipótesis nula y la hipótesis alternativa, las cuales son hipótesis opuestas de una población determinada, y se usan para probar por medio de un estadístico de prueba si se rechazan o no se rechazan los enunciados que se plantean.

Existen dos tipos de errores cuando se trabaja con hipótesis. El primero, es cuando la hipótesis nula se rechaza pero esta era verdadera, a esto se le conoce como error de tipo I y segundo, cuando la hipótesis nula no es rechazada pese a que es falsa, a lo esto se le conoce como error de tipo II (Montgomery, 2008).

2.13.2 Prueba T pareada

La prueba T pareada se utiliza para muestras dependientes que se desean comparar. Por ejemplo, se puede aplicar la prueba T pareada cuando se desea saber si una dieta ha dado resultado en un conjunto de sujetos. Para esto, se toma el peso del conjunto de sujetos antes y después de iniciar la dieta y se define tanto la hipótesis nula h_0 como la hipótesis alternativa h_a , las cuales se muestran en la ecuación (2-1) y en la ecuación (2-2) respectivamente, y donde μ_d es la media de las diferencias de las muestras (muestra uno: el antes y muestra dos: el después) y se aplica el estadístico de prueba T pareada, el cual se describe en la ecuación (2-3), donde, n es el tamaño de la muestra, \bar{d} es la media muestral de las diferencias que se calcula como se muestra en la ecuación (2-4) y S_d es la desviación estándar muestral de las diferencias que se calcula como se muestra en la ecuación (2-5). Ahora para saber si se rechaza la hipótesis nula h_0 se debe cumplir con la ecuación (2-6), donde α es el nivel de significancia (probabilidad de que ocurra el error tipo I) (Montgomery, 2008).

$$h_0 = \mu_d = 0 \quad (2-1)$$

$$h_1 = \mu_d \neq 0 \quad (2-2)$$

$$t_0 = \frac{\bar{d}}{S_d / \sqrt{n}} \quad (2-3)$$

$$\bar{d} = \frac{1}{n} \sum_{j=1}^n d_j \quad (2-4)$$

$$S_d = \left[\frac{\sum_{j=1}^n (d_j - \bar{d})^2}{n-1} \right]^{1/2} = \left[\frac{\sum_{j=1}^n d_j^2 - \frac{1}{n} (\sum_{j=1}^n d_j)^2}{n-1} \right]^{1/2} \quad (2-5)$$

$$|t_0| > t_{\alpha/2, n-1} \quad (2-6)$$

2.13.3 Herramienta estadística

Existen diversas herramientas estadísticas para el análisis de datos, entre ellas se encuentran: Minitab[®] desarrollada por la empresa Minitab Inc[®]. en 1972, Matlab[®] desarrollada por la empresa MathWorks[®] en el año de 1984, R[®] que surge como un lenguaje y entorno de programación y que se distribuye por la organización R Foundation[®], SAS[®] que es desarrollado por la empresas SAS Institute[®] en el año de 1971, SPSS[®] que fue desarrollado por la empresa IBM[®] en el año de 1969, entre otras. Para esta tesis, se decide optar por el uso de la herramienta estadística Minitab[®] (Minitab Inc, 2016) debido a: primero, (Montgomery, 2008) la usa como apoyo en gran parte de su libro para realizar análisis estadísticos; segundo, porque tiene buena documentación con ejemplos prácticos que están en internet; tercero, porque tiene una interfaz fácil de operar por el usuario y cuarto porque permite trabajar con pruebas t pareadas.

3. Planteamiento del problema de programación de preparación de pedidos y la selección del metaheurístico

En este capítulo se describe el proceso de preparación de pedidos en un almacén, a la vez de que se definen las variables de decisión, la función objetivo y las restricciones necesarias para resolver el problema de programación de la preparación de pedidos, considerando las fechas de entrega de las ordenes, k equipos heterogéneos y algunas restricciones de la operación de un almacén. Adicionalmente, se realiza una descripción de los metaheurísticos más usados en la literatura para así, poder determinar cuál es el más adecuado para resolver el problema tratado en esta tesis de maestría.

3.1 Descripción del proceso de preparación de pedidos en un almacén

Para poder comprender el problema tratado en esta tesis es necesario describir el proceso de recolección de mercancía que se sigue dentro de un almacén, donde este comienza con la llegada de una orden en la que se solicitan varios productos del almacén, y se define la fecha en la que se debe realizar la entrega de esta. Dichos productos, deben ser recogidos de sus posiciones de almacenamiento en el menor tiempo posible. Para ello, el operador debe visitar una secuencia de posiciones de almacenamiento, con el fin de recoger los productos que le solicitaron. Al finalizar, esta operación se obtiene el tiempo final de procesamiento de cada orden. Durante todo el proceso, el operador debe partir de un punto inicial (*depot*) y debe retornar a este cuando culmine la recolección de los productos de la orden, teniendo en cuenta que solo puede visitar cada posición de almacenamiento una vez.

3.2 Tiempo de procesamiento de la orden

Para resolver el problema de programación de la preparación de pedidos, considerando variables como: k equipos de manejo de materiales heterogéneos, n productos, m posiciones de almacenamiento, se debe saber previamente el tiempo de procesamiento de cada orden en el almacén; para con esto, poder determinar la secuencia en la que se deben procesar las órdenes, de manera tal que se pueda minimizar la tardanza total de estas.

Para el cálculo del tiempo de procesamiento de las órdenes considerando las variables antes descritas, se empleó la investigación realizada por (Gómez, 2015), la cual se complementó con el método propuesto en esta tesis, con el fin de encontrar la mejor secuencia para programar las órdenes, de tal manera que se minimice la tardanza total, obteniéndose con esto un aporte al conocimiento.

3.3 Planteamiento de las variables de decisión y la función objetivo

Es necesario, plantear cuales van a ser las variables de decisión y la función objetivo que se usará, para poder resolver el problema tratado en esta tesis. Para esto, se parte de un conjunto de órdenes disponibles o , las cuales deben ser secuenciadas de tal manera que se minimice la tardanza. Para ello, se parte de un tiempo inicial cero y un conjunto de órdenes disponibles en el almacén, considerando, que cada orden o , tiene una fecha de entrega Fe y un tiempo de procesamiento Tp .

Antes de formular las variables de decisión y la función objetivo, es importante aclarar que un atributo equivale a un código de una orden y una configuración es una solución formada por un conjunto de órdenes (atributos), las cuales forman la secuencia en la que se deben preparar las órdenes en el almacén. Dicho esto, para poder conseguir la secuencia (configuración) que minimice la tardanza de las órdenes, se debe tener en cuenta en la formulación de las variables de decisión y en la evaluación de la función

objetivo, el intercambio que se realizará entre atributos con posiciones consecutivas o atributos con posiciones no consecutivas en una configuración x . Estos intercambios, se realizarán con el fin de que se puedan hallar nuevas configuraciones. Dicho esto, se identifica que se deberán hacer dos planteamientos para resolver el problema tratado en esta tesis, el primero será usado cuando en una configuración se requiera minimizar la tardanza haciendo intercambios de posiciones consecutivas; y el segundo será usado cuando se requiera minimizar la tardanza haciendo intercambios de posiciones no consecutivas.

Se aclara que para realizar los intercambios tanto consecutivos como no consecutivos, es necesario formular el planeamiento para cada uno, de tal forma que se evidencie el funcionamiento de estos. Por ende, habrá segmentos de texto en la formulación que estarán repetidos, pero que son vitales para su entendimiento. A continuación, se plantean las variables de decisión y la función objetivo usadas para desarrollar esta tesis.

3.3.1 Variables de decisión para el problema tratado con intercambios consecutivos

Se identifican que para resolver el problema tratado en esta tesis haciendo uso de intercambios consecutivos, se requiere del uso de las siguientes variables de decisión:

n : Número de ordenes en el almacén.

Fe_o : Fecha de entrega en días de la orden o .

Tp_o : Tiempo de procesamiento en días de la orden o .

i : posición de una orden o en una configuración x .

PIO : posición de origen a intercambiar en la configuración x .

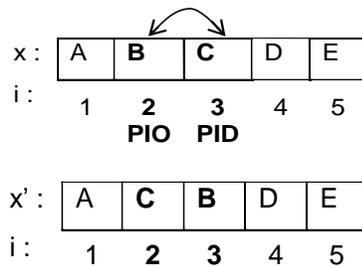
PID : posición de destino a intercambiar en la configuración x .

3.3.2 Función objetivo para el problema tratado con intercambios consecutivos

Para formular la función objetivo del problema tratado en esta tesis con intercambios consecutivos, se debe modificar la función objetivo de la tardanza expuesta por (Pinedo,

2009), de tal manera que se pueda calcular la tardanza teniendo en cuenta intercambios de atributos entre posiciones consecutivas en una configuración dada, por ejemplo, si se usa la configuración x mostrada en la Figura 3-1, se podría calcular la tardanza de una nueva configuración x' al intercambiar la orden B y C . También, se debe modificar la función objetivo formulada por (Pinedo, 2009) para que se pueda utilizar el tiempo de procesamiento de una orden como el tiempo inicial de procesamiento de la orden siguiente. Esto con el fin, de volver más real el tratamiento de las órdenes en un almacén, en el cual las órdenes se procesan una a una hasta completar la existencia de las órdenes que se requieren recolectar.

Figura 3-1 Ejemplo de una configuración donde se hace un intercambio consecutivo entre las órdenes de las posiciones PIO y PID . Las letras A, B, C, D y E corresponden a órdenes de un almacén.



Fuente: Elaboración propia.

Es importante resaltar, lo mencionado en el numeral 3.2, donde se indica que para minimizar la tardanza de las órdenes en un almacén, se requiere conocer el tiempo de procesamiento (T_p) de cada orden. En esta tesis, se utiliza el algoritmo propuesto por (Gómez, 2015) para calcular dicho T_p de la orden en el almacén.

Ahora, para formular la función objetivo usada en esta tesis, teniendo en cuenta intercambios consecutivos, ver ecuación

(3-1), se realizan los siguientes pasos: primero, se halla la tardanza de las ordenes que se encuentran en posiciones menores a la que tiene la posición PIO , en este caso para el ejemplo de la Figura 3-1 Ejemplo de una configuración donde se hace un intercambio consecutivo entre las órdenes de las posiciones PIO y PID , la orden que se encontraría

en una posición menor a la posición P_{IO} sería la orden A , por lo tanto la tardanza que se calcularía sería la de la orden A ; segundo, se suma la tardanza del primer paso con la tardanza de la orden que ha sido intercambiada de la posición P_{ID} a la posición P_{IB} , en el ejemplo de la Figura 3-1 Ejemplo de una configuración donde se hace un intercambio consecutivo entre las órdenes de las posiciones P_{IO} y P_{ID} , la tardanza que se sumaría sería la de la orden C ; tercero, a la tardanza que va acumulada del primero y el segundo paso se le suma la tardanza de la orden que fue intercambiada de la posición P_{IO} a la posición P_{ID} , en el ejemplo de la Figura 3-1 Ejemplo de una configuración donde se hace un intercambio consecutivo entre las órdenes de las posiciones P_{IO} y P_{ID} , se sumaría la tardanza de la orden B ; por último, se suma al acumulado de la tardanza del primero, segundo y tercer paso, la tardanza de las ordenes que se encuentran en posiciones mayores a las que tiene la posición P_{ID} , en el ejemplo de la Figura 3-1, se sumaría la tardanza de la orden D además de la tardanza de la orden E .

$$\begin{aligned}
Tardanza(x) = & \min(\sum_{i=1}^{P_{IO}-1} \max[0, (\sum_{j=1}^i Tp_j) - Fe_i] + \max\{0, [(\sum_{i=1}^{P_{IO}-1} Tp_i) + \\
& Tp_{P_{ID}}] - Fe_{P_{ID}}\} + \max\{0, [(\sum_{i=1}^{P_{IO}-1} Tp_i) + Tp_{P_{ID}} + Tp_{P_{IO}}] - Fe_{P_{IO}}\} + \\
& \sum_{i=P_{ID}+1}^n \max\{0, [(\sum_{i=1}^{P_{IO}-1} Tp_i) + Tp_{P_{ID}} + Tp_{P_{IO}} + Tp_i] - Fe_i\})
\end{aligned} \tag{3-1}$$

Sujeto a:

$$Fe > 0 \tag{3-2}$$

$$Tp > 0 \tag{3-3}$$

$$1 \leq P_{IO} \leq n - 1 \tag{3-4}$$

$$2 \leq P_{ID} \leq n \tag{3-5}$$

$$P_{ID} = P_{IO} + 1 \tag{3-6}$$

$$i, j = 1, \dots, n \tag{3-7}$$

$$\forall Fe, Tp \in \mathbb{R}^+ \tag{3-8}$$

$$\forall P_{IO}, P_{ID} \in \mathbb{N} \tag{3-9}$$

Con las ecuaciones (3-2) y (3-8), se asegura que la fecha de entrega que tiene cada orden sea mayor a cero días; con las ecuaciones (3-3) y (3-8), se asegura que el tiempo de procesamiento de cada orden sea mayor que cero días; con la ecuación (3-4) se asegura que la posición P_{IO} este entre uno y $n-1$, ya que para que se pueda intercambiar la posición P_{IO} y P_{ID} , P_{IO} no puede tomar el valor de n porque P_{ID} no podría tomar el

valor de $n+1$, ya que esta posición no existe en la configuración; la ecuación (3-5) se asegura que la posición PID este entre dos y n , ya que para que se pueda intercambiar la posición PIO y PID , PID no puede tomar el valor de 1 ya que si PID toma ese valor, PIO debería tener la posición de cero que no existe en la configuración, con la ecuación (3-6) se garantiza que PIO y PID siempre sean posiciones consecutivas, con la ecuación (3-7) se asegura que i y j tomen valores entre uno y el número total de órdenes que se van a secuenciar en el almacén, la ecuación (3-9) garantiza que PIO y PID sean números naturales.

3.3.3 Variables de decisión para el problema tratado con intercambios no consecutivos

Se identifican que para resolver el problema tratado en esta tesis haciendo uso de intercambios no consecutivos, se requiere del uso de una variable adicional a las antes mencionadas para los intercambios consecutivos, la cual es la variable y que se muestra a continuación:

n : Número de ordenes en el almacén.

Fe_o : Fecha de entrega en días de la orden o .

Tp_o : Tiempo de procesamiento en días de la orden o .

i : Posición de una orden o en una configuración x .

PIO : posición de origen a intercambiar en la configuración x .

PID : posición de destino a intercambiar en la configuración x .

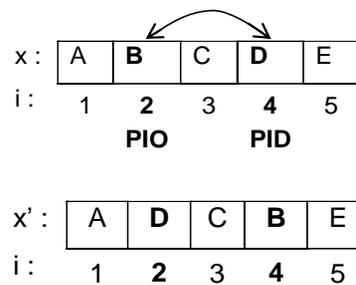
y : Diferencia entre la posición PID y la PIO .

3.3.4 Función objetivo para el problema tratado con intercambios no consecutivos

Para formular la función objetivo del problema tratado en esta tesis con intercambios no consecutivos, se realiza también una modificación sobre la función objetivo de la tardanza expuesta por (Pinedo, 2009), de tal manera que permita intercambios de

atributos que se encuentran en posiciones no consecutivos en una configuración dada, por ejemplo, si se usa la configuración x mostrada en la Figura 3-2, se podría calcular la tardanza de una nueva configuración x' al intercambiar la orden B y D . También, se debe modificar la función objetivo formulada por (Pinedo, 2009) para que se pueda utilizar el tiempo de procesamiento de una orden como el tiempo inicial de procesamiento de la orden siguiente. Esto último, también se mencionó en el numeral 3.3.2 debido a que se requiere comprender para utilizarlo en la función objetivo que usa intercambios no consecutivos.

Figura 3-2 Ejemplo de una configuración donde se hace un intercambio no consecutivo entre las órdenes de las posiciones PIO y PID . Las letras A , B , C , D y E corresponden a órdenes de un almacén.



Fuente: Elaboración propia.

Es importante también resaltar para los intercambios no consecutivos, lo mencionado en el numeral 3.2, donde se indica que para minimizar la tardanza de las órdenes en un almacén, se requiere conocer el tiempo de procesamiento (T_p) de cada orden. En esta tesis, se utiliza el algoritmo propuesto por (Gómez, 2015) para calcular dicho T_p de la orden en el almacén.

Ahora, para formular la función objetivo usada en esta tesis teniendo en cuenta intercambios no consecutivos, ver ecuación (3-10), se realizan los siguientes pasos: primero, se halla la tardanza de las ordenes que se encuentran en posiciones menores a la que tiene la posición PIO , en este caso para el ejemplo de la Figura 3-2, la orden que se encontraría en una posición menor a la posición PIO sería la orden A , por lo tanto la tardanza que se calcularía sería la de la orden A ; segundo, se suma la tardanza del primer paso con la tardanza de la orden que ha sido intercambiada de la posición PID a

la posición PIB , en el ejemplo de la Figura 3-2, la tardanza que se sumaría sería la de la orden D ; tercero, se suma la tardanza del primero y segundo paso con la tardanza de las ordenes que se encuentran entre las posiciones PIO y PID , en el ejemplo de la Figura 3-2, la tardanza que se sumaría sería la de la orden C ; cuarto, a la tardanza que va acumulada del primero, segundo y tercero paso se le suma la tardanza de la orden que fue intercambiada de la posición PIO a la posición PID , en el ejemplo de la Figura 3-2, se sumaría la tardanza de la orden B ; por último, se suma al acumulado de la tardanza del primero, segundo, tercero y cuarto paso, la tardanza de las ordenes que se encuentran en posiciones mayores a las que tiene la posición PID , en el ejemplo de la Figura 3-2, se sumaría la tardanza de la orden E .

$$\begin{aligned}
Tardanza(x) = & \min(\sum_{i=1}^{PIO-1} \max\{0, (\sum_{j=1}^i Tp_j) - Fe_i\} + \\
& \max\{0, [(\sum_{i=1}^{PIO-1} Tp_i) + Tp_{PID}] - Fe_{PID}\} + \sum_{i=PIO+1}^{PID-1} \max\{0, [(\sum_{i=1}^{PIO-1} Tp_i) + \\
& Tp_{PID} + (\sum_{j=i}^{PID-1} Tp_j)] - Fe_i\} + \max\{0, \sum_{i=PIO+1}^{PID-1} [(\sum_{i=1}^{PIO-1} Tp_i) + Tp_{PID} + \\
& (\sum_{j=i}^{PID-1} Tp_j)] + Tp_{PIO} - Fe_{PIO}\} + \sum_{i=PID+1}^n \max\{0, [(\sum_{i=1}^{PIO-1} Tp_i) + Tp_{PID} + \\
& (\sum_{j=i}^{PID-1} Tp_j) + Tp_{PIO} + Tp_i] - Fe_i\}) \quad (3-10)
\end{aligned}$$

Sujeto a:

$$Fe > 0 \quad (3-11)$$

$$Tp > 0 \quad (3-12)$$

$$1 \leq PIO \leq n - 1 \quad (3-13)$$

$$2 \leq PID \leq n \quad (3-14)$$

$$1 \leq y \leq n - 1 \quad (3-15)$$

$$PID = y + PIO \quad (3-16)$$

$$i, j = 1, \dots, n \quad (3-17)$$

$$\forall Fe, Tp \in \mathbb{R}^+ \quad (3-18)$$

$$\forall PIO, PID \in \mathbb{N} \quad (3-19)$$

Con la ecuación (3-11) y (3-18) se asegura que la fecha de entrega que tiene cada orden sea mayor a cero días, con la ecuación (3-12) y (3-18) se asegura que el tiempo de procesamiento de cada orden sea mayor que cero días, con la ecuación (3-13) se asegura la posición PIO este entre uno y $n-1$ órdenes, ya que para que se pueda

intercambiar la posición de PIO y PID , PIO no puede tomar el valor de n porque PID no podría tomar el valor de $n+1$, ya que esta posición no existe en la configuración, con la ecuación (3-14) se asegura que la posición PID este entre dos y n , con la ecuación (3-15) y (3-16) se garantiza que el valor de la diferencia entre la posición PIO y PID sea mínimamente uno o menor que $n-1$, esto con el fin de que se pueda hacer el intercambio entre posiciones no consecutivas, con la ecuación (3-17) se asegura que i y j tomen valores entre uno y el número total de órdenes que se van a secuenciar en el almacén, la ecuación (3-19) garantiza que PIO y PID sean números naturales.

3.4 Razones para seleccionar un metaheurístico

Para elaborar esta tesis se opta por usar un metaheurístico, debido a que el problema combinatorial que se desarrolla es del tipo *NP-Hard*, por lo cual se descarta el uso de métodos de optimización tradicionales, ya que para resolverlo se requeriría un alto uso computacional y un alto tiempo de ejecución (Rossi, van Beek, & Walsh, 2006), (Mendonca & Cowan, 2010). Adicionalmente, se descarta el uso de un heurístico dado que se requiere un método más robusto que de soluciones más cercanas a la óptima, usando características tales como la memoria (Wolpert & Macready, 1997). Por esto, en este trabajo se decide usar un metaheurístico que favorezca el tiempo computacional para resolver el problema y permita usar una estructura basada en la memoria, para encontrar una solución cercana a la óptima.

3.5 Teorema *NFL* (*No Free Lunch*)

El Teorema *NFL* (*No Free Lunch*) propuesto por (Wolpert & Macready, 1997), muestra que no todo metaheurístico se comporta igual para todos los problemas que existen y que para cada tipo de problema existe un mejor algoritmo el cual si usa estructuras de memoria será mejor que uno que no tiene. Por ende, se hace de vital importancia conocer el problema que se está tratando y adaptar el metaheurístico de acuerdo a las

características de este. Lo que ratifica que es mejor usar un metaheurístico que un heurístico para resolver el problema tratado en esta tesis.

Teniendo claro, la razón por la que se escoge un metaheurístico para el desarrollo de este problema, se realiza una evaluación diferentes algoritmos usados en la literatura tales como: la Búsqueda Tabú (Glover, 1989), los Algoritmos Genéticos (Holland, 1975), el Recocido Simulado (Kirkpatrick, Gelatt, & Vecchi, 1983), la Optimización de Enjambre de Partículas (Kennedy & Eberhart, 1995) y Colonia de Hormigas (Dorigo, 1992). Estos algoritmos también han sido mencionados en por (Escobar, Gallego, & Toro, 2015) y (Delgado, 2010).

3.6 Búsqueda Tabú

La Búsqueda Tabú fue inventada por Fred Glover en 1989 y es usada para resolver problemas complejos de diferentes áreas. La Búsqueda Tabú está inspirada en algunos conceptos de la inteligencia artificial, tales como la memoria adaptativa (memoria a corto y largo plazo) y la exploración sensible, que permite extraer información de la selección de configuraciones (soluciones) malas y buenas para guiar la búsqueda. (Escobar et al., 2015), (Glover, 1989).

La Búsqueda Tabú es de tipo local y se caracteriza por partir de una configuración x , definir una vecindad de x (conjunto de posibles soluciones del problema), e ir seleccionando la mejor configuración vecina o a la que tenga menor impacto en la función objetivo, hecho que permite salir de óptimos locales mientras se realiza la búsqueda. Adicionalmente, emplea el uso de memoria por medio de la lista tabú, que es la que permite almacenar esos atributos (elementos que forman una configuración) que ya han sido explorados y han resultado dar los mejores resultados de la función objetivo, para así no volver a visitarlos durante un número de iteraciones. (Escobar et al., 2015), (Glover, 1989).

La memoria de corto plazo es conocida por tomar los hechos recientes y evitar volver a seleccionar configuraciones ya visitadas y que tienen atributos que se encuentran como tabú. Para almacenar dichos atributos tabú, se utiliza la *lista tabú*, la cual puede

implementarse conservando como prohibidos los atributos tabú un número de iteraciones w , para que no sean tenidos en cuenta en la búsqueda, a este número de iteraciones se le conoce como *Tabú tenure*. Ahora bien, si una configuración usa un atributo que está en la *lista tabú*, este puede salir por medio del *criterio de aspiración* el cual elimina la prohibición siempre y cuando se encuentre una mejor solución que la que se tenga como actual. (Escobar et al., 2015), (Glover, 1989).

La memoria a largo plazo se centra en la estrategia de frecuencia que trata de almacenar la cantidad de veces que un atributo ha sido seleccionado para formar una configuración durante el proceso de la Búsqueda Tabú con memoria a corto plazo para así poder incentivar alguna configuración o cuando se usa la estrategia de intensificación que busca centrar la búsqueda en zonas que han mostrado al ejecutar la Búsqueda Tabú con memoria de corto plazo que son prometedoras para encontrar buenas soluciones o cuando se usa la estrategia de intensificación que consiste en explorar zonas que no han sido visitadas al ejecutar la Búsqueda Tabú con memoria de corto plazo y para ello se hace uso de la penalización de parámetros frecuentemente visitados hasta que se encuentre una configuración que mejore a la que se encontró en la Búsqueda Tabú básica con memoria de corto plazo como la mejor. (Escobar et al., 2015), (Glover, 1989).

El metaheurístico de Búsqueda Tabú usa un criterio de parada que se refiere a la condición que debe cumplirse para que se deje de realizar la Búsqueda Tabú. Se distinguen dos criterios de parada, el primero es cuando se cumple un número l máximo de iteraciones y el segundo es cuando se cumple un número máximo de diversificaciones. (Escobar et al., 2015), (Glover, 1989).

3.7 Algoritmos Genéticos

Los algoritmos genéticos fueron creados por John Henry Holland en 1975 basándose en la teoría evolutiva propuesta por Darwin en 1859. Por ende, lo que hacen es simular un proceso de selección natural para así conseguir la solución a un problema de optimización. (Holland, 1975), (Escobar et al., 2015).

Los algoritmos genéticos usan una búsqueda de tipo poblacional y se caracterizan por convertir el universo de soluciones en cadenas de tipo binarias las cuales se conocen

como cromosomas. Adicionalmente, generan la población inicial de manera aleatoria y se van haciendo las evaluaciones de los cromosomas de acuerdo a la función de ajuste que se defina, se selecciona los mejores individuos para que perdure su herencia por medio del cruce, mientras que el resto de individuos mueren o son mutados de acuerdo a los parámetros que se definan en el algoritmo. (Holland, 1975), (Escobar et al., 2015).

Para detener la búsqueda se usan tres criterios. El primero, consiste en detener la búsqueda cuando se ha generado un número y de generaciones. El segundo, es cuando la mejor configuración encontrada en el proceso (cromosoma) cumple con una calidad mínima dada. El tercero, es cuando las configuraciones que se tienen son parecidas y por ende no se puede seguir evolucionando. (Holland, 1975), (Escobar et al., 2015).

3.8 Recocido Simulado

El recocido simulado fue propuesto por Kirkpatrick, Gelatt y Vecchi en 1983 y está inspirado en un proceso físico de calentamiento de un sólido, seguido por el enfriamiento de este hasta que se obtiene una estructura deseada. Este metaheurístico es usado en problemas de tipo combinatorial. (Kirkpatrick et al., 1983), (Escobar et al., 2015).

El metaheurístico de recocido simulado usa un tipo de búsqueda local, y se caracteriza por partir de una temperatura muy alta que es lo que permite aceptar o no configuraciones (soluciones) y a medida que se va iterando en el algoritmo se va disminuyendo la temperatura para llegar a las mejores configuraciones. Para esto, se parte de una configuración inicial y se evalúa la función objetivo, si la configuración es mejor que la que se tiene como actual se actualiza, en caso contrario de acuerdo a la función de aceptación se selecciona una nueva configuración. Luego se reduce la temperatura de acuerdo a una función inicialmente dada para el algoritmo. (Kirkpatrick et al., 1983), (Escobar et al., 2015).

Para detener la búsqueda se usan dos criterios, el primero es detener la búsqueda cuando la temperatura llegue a cero, el segundo es cuando se cumpla un número máximo de iteraciones. (Kirkpatrick et al., 1983), (Escobar et al., 2015).

3.9 Optimización de enjambre de partículas (*PSO*)

El metaheurístico de optimización de enjambre de partículas (*PSO*) fue inventado por James Kennedy y Russell Eberhart en 1995 y se inspira en la formación que sigue un grupo de aves, este tipo de metaheurístico es usado para resolver problemas no lineales. (Kennedy & Eberhart, 1995), (Escobar et al., 2015).

Este metaheurístico es del tipo poblacional y se caracteriza por definir el tamaño de población, las constantes de aceleración (c_1 y c_2) que son los valores que van guiando las partículas hacia la mejor posición local o global, la posición, velocidad y pesos de las partículas, para luego calcular la función de adaptación de estas y así validar cual es la mejor ubicación que deben seguir. Para esto se usan dos valores como memoria, la mejor ubicación local que representa la memoria de cada individuo y la mejor ubicación global que representa la memoria del grupo, el metaheurístico seguirá con el cálculo de las nuevas velocidades y posiciones de las partículas y actualizará los pesos de las partículas hasta que se cumpla el criterio de parada. La idea en este algoritmo es que las partículas giran en torno a la que mejor ubicación tiene que sería el líder de la manada, este líder puede ser remplazado siempre y cuando una partícula mejore la ubicación que tiene el líder con respecto a la función de adaptación. (Kennedy & Eberhart, 1995), (Escobar et al., 2015).

Para detener la búsqueda se pueden usar dos criterios, el primero es cuando se supere un número máximo de iteraciones y el segundo es cuando tras un número de iteraciones no se encuentra otra mejor solución. (Kennedy & Eberhart, 1995), (Escobar et al., 2015).

3.10 Colonia Hormigas

El metaheurístico conocido como Colonia de Hormigas fue creado por Marco Dorigo en 1992 y fue inspirado en el estudio de la naturaleza, en donde se observó cómo las hormigas dejaban rastros de feromonas para guiar a la colonia en la búsqueda de

alimento. Este tipo de metaheurístico es usado para trabajar con problemas de tipo combinatorial. (Dorigo, 1992), (Escobar et al., 2015).

La Colonia de hormigas es un algoritmo de tipo poblacional, donde una configuración (solución) es conocida como una hormiga y la colonia de hormigas es entonces el grupo de j configuraciones. Este metaheurístico usa estructura de memoria por medio del rastro de feromonas que deja la hormiga, donde la feromona indica el grado de aceptación que tienen las hormigas para cada configuración. El metaheurístico cuenta con tres mecanismos para encontrar la mejor solución, el primero es por medio de la explotación que consiste en seguir la configuración con mayor rastro de feromonas, el segundo es el mecanismo de exploración que se basa en que no todas las hormigas sigan el mayor rastro de feromonas lo que permite conocer nuevas configuraciones (soluciones) y el tercero es el mecanismo de evaporación de feromona que consiste en equilibrar la exploración y explotación para permitir al metaheurístico converger de manera oportuna y salir de óptimos locales. (Dorigo, 1992), (Escobar et al., 2015).

Para detener la búsqueda se usan se usan los siguientes dos criterios: cuando se cumple un número máximo de iteraciones o cuando todas las hormigas usen la misma configuración. (Dorigo, 1992), (Escobar et al., 2015).

3.11 Metaheurístico seleccionado

Para tomar la decisión de cuál metaheurístico usar en esta tesis, se hace un comparativo donde se exponen las características de los principales metaheurísticos utilizados en la literatura y que se muestran en la Tabla 3-1.

Tabla 3-1 Comparativo de metaheurísticos más utilizados en la literatura.

Metaheurístico	Búsqueda	Inspirado	Usa memoria	Propuesto
Búsqueda Tabú	Local	Inteligencia Artificial	Si (Lista Tabú)	(Glover, 1989)
Algoritmos Genéticos	Poblacional	Genética	No	(Holland, 1975)
Recocido Simulado	Local	Termodinámica	No	(Kirkpatrick et al., 1983)

Metaheurístico	Búsqueda	Inspirado	Usa memoria	Propuesto
Optimización de enjambre de partículas (PSO)	Poblacional	La inteligencia colectiva	Si (Mejor Posición)	(Kennedy & Eberhart, 1995)
Colonia Hormigas	Poblacional	Naturaleza	Si (Feromonas)	(Dorigo, 1992)

Fuente: Elaboración propia

A partir del análisis hecho en la Tabla 3-1, se encuentra que los algoritmos genéticos como el recocido simulado no usan memoria y por el teorema *NFL* propuesto por (Wolpert & Macready, 1997) se descartan, luego se observa que por las características del problema el que mejor se adapta es el metaheurístico de Búsqueda Tabú, ya que usa un tipo memoria más elaborado que el de colonia de hormigas y el de optimización de enjambre de partículas (*PSO*), pues posee memoria a corto y largo plazo, penaliza soluciones las cuales son enviadas a una lista tabú lo que permite acercarse a mejores soluciones de acuerdo a lo dicho por (Wolpert & Macready, 1997), salir de óptimos locales, también porque es usado para resolver problemas de tipo agente viajero *TSP*, lo cual se asemeja mucho al problema de secuenciación que se aborda en esta tesis, ya que se requiere programar las ordenes de tal manera que no se repitan y que se encuentre la mejor secuencia en que se deben programar las órdenes para ser despachadas al cliente.

4. Programación del metaheurístico

En este capítulo, se describe el proceso de preparación de pedidos que se sigue en un almacén, junto con las características del almacén que se selecciona para desarrollar la tesis. También, se crean unos escenarios de aplicación, los cuales son necesarios para realizar el desarrollo computacional que soluciona el problema tratado en esta tesis y que se tratan en este capítulo. A la vez, que se muestra la estructura básica del metaheurístico de Búsqueda Tabú seleccionado y las variaciones realizadas para adaptarlo al problema tratado en la tesis.

4.1 Proceso de preparación de pedidos en un almacén

En los almacenes se utiliza la operación de preparación de pedidos para recoger los productos que son solicitados por los clientes a través de las órdenes que llegan al almacén. Así pues, para realizar este proceso, los operadores del almacén, debe partir de un punto inicial conocido como *Depot* junto con la ruta que deben seguir para recolectar los productos solicitados en las órdenes. Para esto, los operadores utilizan los equipos de manejo de materiales disponibles en el almacén y visitan cada una de las posiciones de almacenamiento indicadas para recolectar los productos de las órdenes. Luego, de que los productos han sido recolectados de sus posiciones de almacenamiento, los operadores se dirigen con los productos al punto inicial (*Depot*) para descargarlos y disponerlos para el envío de la orden al cliente.

Para poder realizar el proceso de preparación de pedidos, se requiere que existan órdenes en el almacén. A demás, se requiere conocer los siguientes atributos de la orden: código, fecha de entrega, productos, cantidad del producto, esto con el fin de manejar el inventario en el almacén.

Debido a que en el almacén se procesan múltiples órdenes, se vuelve importante para las organizaciones, velar por el cumplimiento de las fechas de entrega de las órdenes a los clientes. Por esto, es necesario que en los almacenes se utilice un modelo que sirva para programar la operación de preparación de pedidos, de tal manera, que se pueda minimizar la tardanza de las órdenes en el almacén y teniendo en cuenta variables tales como: k equipos de manejo de materiales heterogéneos, n productos, m posiciones de almacenamiento y la actualización del nivel de inventario.

4.2 Descripción del Almacén seleccionado

Para este trabajo, se usa el almacén empleado por (Gómez, 2015) ver Figura 4-1, un *layout* multipasillo en forma de bloque con cuatro pasillos, $m=192$ posiciones de almacenamiento y un área de $720 m^2$, la distancia entre pasillos es de cinco metros, se utilizan tres equipos de manejo de materiales (*EM*) heterogéneos (diferente capacidad, velocidad y peso), 60 productos, se usa la misma matriz de distancias entre posiciones de almacenamiento, la misma distribución de productos en los estantes y la ubicación del *depot* al lado izquierdo. Esto se hace con el fin de poder integrar el desarrollo hecho por (Gómez, 2015) con el desarrollo hecho en esta tesis. Para así, velar por la disminución de la tardanza de las órdenes de los clientes.

Figura 4-1 Almacén de tipo *layout* multipasillo, de dos pisos, con códigos de las posiciones de almacenamiento y la ubicación del *Depot*.

#																
PISO	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
	12	24	36	48	60	72	84	96	108	120	132	144	156	168	180	192
	11	23	35	47	59	71	83	95	107	119	131	143	155	167	179	191
	10	22	34	46	58	70	82	94	106	118	130	142	154	166	178	190
	9	21	33	45	57	69	81	93	105	117	129	141	153	165	177	189
	8	20	32	44	56	68	80	92	104	116	128	140	152	164	176	188
	7	19	31	43	55	67	79	91	103	115	127	139	151	163	175	187
	6	18	30	42	54	66	78	90	102	114	126	138	150	162	174	186
	5	17	29	41	53	65	77	89	101	113	125	137	149	161	173	185
	4	16	28	40	52	64	76	88	100	112	124	136	148	160	172	184
	3	15	27	39	51	63	75	87	99	111	123	135	147	159	171	183
	2	14	26	38	50	62	74	86	98	110	122	134	146	158	170	182
	1	13	25	37	49	61	73	85	97	109	121	133	145	157	169	181

Depot

Fuente: (Gómez, 2015).

4.3 Escenarios creados para el proceso de preparación de pedidos

Para esta tesis, se diseñan treinta escenarios mostrados en el anexo A, donde cada uno está formado por un conjunto de órdenes que llegan al almacén para su procesamiento, dichas ordenes tienen los siguientes atributos: código, que hace referencia a la identificación de cada orden; fecha de entrega, que hace referencia a la fecha en la cual el cliente requiere que se le entregue los productos solicitados en la orden; producto, que hace referencia al listado de productos que solicita el cliente a través de la orden, teniendo en cuenta, el listado de productos que ofrece el almacén ver Tabla B-1 del anexo B y cantidad productos, que hace referencia al número de unidades del producto que solicita el cliente en la orden. Ver Tabla 4-1.

Tabla 4-1 Atributos de las órdenes con valores de ejemplo.

Orden			
Código (O)	Fecha Entrega (Fe)	Producto (P)	Cantidad productos (Cp)
O1	5 días	p1	100
		p2	80
		p3	162
		p4	75
O2	2 días	p1	30
		p4	35
		p7	120
O _n	Fe _n	P _n	Cp _n

Fuente: Elaboración propia.

Las características de los equipos de manejo de materiales utilizados en el almacén y en cada uno de los escenarios construidos, se muestra en la Tabla 4-2. En donde el campo: Equipo Manejo Materiales, indica el código del equipo de manejo de materiales; Capacidad Peso, indica el peso máximo que puede transportar este en *Kg* y Velocidad, indica la velocidad promedio dada en *Km* por hora, a la que viaja el equipo de manejo de materiales.

Tabla 4-2 Características de los equipos de manejo de materiales.

Equipo Manejo Materiales	Capacidad Peso (Kg)	Velocidad (Km/h)
1	1600	2.7
2	1000	3.6
3	1550	3

Fuente: Elaboración propia.

Ya con el diseño de los treinta escenarios, se procede a desarrollar el metaheurístico de Búsqueda Tabú que se requiere para esta tesis.

4.4 Estructura básica de la Búsqueda Tabú

Para poder entender la lógica con la que opera el metaheurístico de Búsqueda Tabú, se requiere revisar la estructura básica de este. En esta sección, se explica el pseudocódigo que utiliza este metaheurístico ver Figura 4-2. Donde, se siguen los siguientes pasos: primero, se definen las variables que se utilizarán para denotar: la solución actual (x), la nueva solución (y), la lista tabú (T), el vecindario (V) y el vecindario reducido (V^*); segundo, se genera una solución inicial; tercero, se genera un vecindario V y V^* ; cuarto, se obtiene la nueva solución y , realizando movimientos en los atributos de la solución x con base al V^* ; quinto, se evalúan las soluciones x y y con respecto a la función objetivo definida y se escoge la mejor solución; sexto, se marcan los atributos tabú, séptimo, se eliminan elementos antiguos de la lista tabú y octavo, se verifica si el criterio de parada se ha cumplido para terminar la ejecución del metaheurístico.

Figura 4-2 Metaheurístico básico Búsqueda Tabú.

```
{x: TipoSolucion} = TS(ST,Sv,Sv*:integer; f: TipoFunciónObjetivo)
Var

x,y: TipoSolucion;
T: array [1...ST] of TipoSolucion; // Memoria Tabú;
V: array [1...Sv] of TipoSolucion; // Estructura de vecindad;
V*: array [1...Sv*] of TipoSolucion; // Estructura de vecindad reducida;

Begin
{x} := GenerarSolucionInicial();
While not terminación do
    V := GenerarVecindad(x);
    V* := GenerarVecindadReducida(V);
    {y} := SeleccionarMejor (V*, f);
    If f(y) ≥ f(x) then
        x:= y; // se actualiza la solución actual.
    End if
    {T}:= GrabarMovimientoMemoria(x,T); // Marcar la solución como tabú-activo
    {T}:= ActualizarMemoria(T); // Borrar elementos tabúes antiguos si procede.
End While
End
```

Fuente: (Muñoz, 2007).

4.5 Estructura de Búsqueda Tabú adaptada al problema tratado en esta tesis

En esta sección, se adapta la estructura del metaheurístico básico de Búsqueda Tabú ver Figura 4-2 al problema tratado en esta tesis. De tal manera, que se pueda encontrar la secuencia en que deben procesar las órdenes que llegan al almacén, para que se minimice la tardanza de estas. Así pues, la función objetivo equivaldrá a la ecuación (3-1) cuando se estén realizando movimientos consecutivos o a la ecuación (3-10) cuando se estén realizando movimientos no consecutivos, ver Figura 4-3.

Figura 4-3 Estructura de Búsqueda Tabú adaptada al problema tratado en esta tesis.

```

{x: TipoSolucion} = TS (ST,Sv,Sv*:integer; f: CalcularTardanza)
Var
x,y: TipoSolucion;
T: array [1...ST] of TipoSolucion; // Memoria Tabú;
V: array [1...Sv] of TipoSolucion; // Estructura de vecindad;
V*: array [1...Sv*] of TipoSolucion; // Estructura de vecindad reducida;
Ta, Ta': Integer;

Begin
{x} := GenerarSolucionInicial();

While not terminación do
V := GenerarVecindad(x);
V* := GenerarVecindadReducida(V);
{y} := SeleccionarMejor (V*, f);
Ta:= f(x); // Se calcula la tardanza de la solución x.
Ta':= f(y); // Se calcula la tardanza de la solución y.
If Ta ≥ Ta' then
    x:= y; // se actualiza la solución actual.
    Ta=Ta'; //se almacena la mejor tardanza;
End if
{T}:= GrabarMovimientoMemoria(x,T); // Marcar la solución como tabú-activo
{T}:= ActualizarMemoria(T); // Borrar elementos tabúes antiguos si procede.
End While
End

```

Fuente: Elaboración propia basada en (Muñoz, 2007).

4.6 Herramientas empleadas para el desarrollo del metaheurístico de Búsqueda Tabú

Para desarrollar el trabajo, se usó el *Framework OpenTS*[®], el cual se describió en el capítulo 2. Este *framework*, es utilizado con el fin modificarlo y adaptarlo para resolver el

problema identificado en esta tesis. Para la programación del metaheurístico de Búsqueda Tabú, se empleó el lenguaje Java[®] junto con el *SGBD Oracle Database 11g Express Edition Release 11.2.0.2.0[®]*, el cual se utiliza para almacenar la información de la tesis en una base de datos, que contiene la tabla órdenes, productos, estantes y montacargas.

4.7 Selección de la configuración inicial

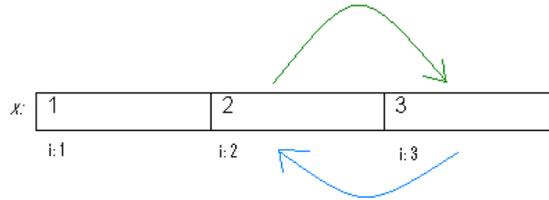
Para este trabajo, se selecciona como configuración inicial el código del listado de órdenes, las cuales son ordenadas inicialmente por la regla de prioridad *Earliest Due Date (EDD)* (Achedad & Giménez, 2011), la cual consiste en ordenar de acuerdo a la fecha de entrega el listado de las ordenes. Esto se hace con el fin de iniciar la Búsqueda Tabú con una buena solución y no con una configuración inicial aleatoria.

4.8 Conformación del vecindario

El vecindario que se crea para abordar esta tesis se forma de acuerdo al rango de los posibles movimientos que puede hacer un atributo en una configuración (solución), donde el máximo rango de movimientos permitidos se define por un número u de posiciones i . Esto quiere decir, que se pueden realizar movimientos de atributos un número f de posiciones atrás de donde se encuentra el atributo o un número f de posiciones después de este, teniendo en cuenta que f es un valor menor o igual al de u . Lo cual, permite crear vecindarios variables de acuerdo a la posición en que se encuentre el atributo a intercambiar o mover. Para el desarrollo de esta tesis, se usa u igual a 5.

En la Figura 4-4, se ilustra un ejemplo con un posible intercambio que puede realizarse para una configuración x , con $u=2$. Además, se muestran los atributos vecinos x_i con los cuales se pueden realizar intercambios. En la Tabla 4-3, se puede visualizar todos los posibles movimientos o intercambios para el ejemplo mostrado en la Figura 4-4, el cual ayuda a comprender cómo se forman los vecindarios en el metaheurístico desarrollado.

Figura 4-4 Intercambio posible entre el atributo x2 con x3 en una configuración x (solución) de ejemplo.



Fuente: Elaboración propia.

Tabla 4-3 Intercambios posibles entre atributos que forman el vecindario del ejemplo de la Figura 4-4. Se resalta en color azul y verde el intercambio realizado.

# Intercambios \ Posición i	Posición i		
	1	2	3
1	1	3	2
2	2	1	3
3	2	3	1
4	3	1	2
5	3	2	1

Fuente: Elaboración propia

4.9 Evaluación Función Objetivo

Para calcular la función objetivo que se basa en minimizar la tardanza total de las ordenes que llegan a un almacén y que se describió en la ecuación (3-1) y (3-10), se debió adaptar el metaheurístico de Búsqueda Tabú, el cual requiere como dato de entrada el listado de órdenes, la fecha de entrega que está en cada orden que realiza el cliente y el tiempo de procesamiento. El cual, es calculado por medio del trabajo desarrollado por (Gómez, 2015), con el fin de tener en cuenta en dicho calculo restricciones tales como EM equipos de manejo de materiales heterogéneos, n productos, m posiciones de almacenamiento. Para este último dato de entrada, se debió realizar un cálculo adicional, el cual permitiera acumular el tiempo de procesamiento de

una orden con el de las órdenes anteriores para así adaptar a la vida real dicho tiempo de procesamiento suministrado por (Gómez, 2015).

Para evaluar la función objetivo dentro del metaheurístico desarrollado se usaron los dos criterios mencionados en el capítulo 3 y se seleccionaron las variables i : para definir la posición de la orden dentro de la configuración x , $Tp(i)$ que es el tiempo de procesamiento de la orden que está en la posición i , $Fe(i)$ que es la fecha de entrega de la orden i , $SumaTp$ que empieza en cero y acumula los tiempos de procesamiento de las ordenes, $TardanzaActual$ que empieza que empieza en cero y almacena la tardanza de las ordenes de la configuración x , $MejorTardanza$ que empieza con la tardanza de la configuración inicial y será la que almacenara la mejor tardanza al finalizar las iteraciones del metaheurístico.

Para el primer criterio, se recorre toda la configuración y se evalúa si la posición de la orden de origen y la orden destino mostrada en la Figura 3-1 cumple alguno de los condicionales ver Figura 4-5, para así poder calcular la tardanza teniendo en cuenta el intercambio de posiciones consecutivas.

Figura 4-5 Pseudocódigo de los intercambios entre atributos consecutivos.

<p><i>Si</i> ($i < PIO$ ó $i > PID$)</p> <p>$SumaTp = SumaTp + Tp(i)$</p> <p>$TardanzaActual = TardanzaActual + \max(0, SumaTp - Fe(i))$</p> <p><i>Si</i> ($i = PIO$)</p> <p>$SumaTp = SumaTp + Tp(PID)$</p> <p>$TardanzaActual = TardanzaActual + \max(0, SumaTp - Fe(PID))$</p> <p><i>Si</i> ($i = PID$)</p> <p>$SumaTp = SumaTp + Tp(PIO)$</p> <p>$TardanzaActual = TardanzaActual + \max(0, SumaTp - Fe(PIO))$</p>
--

Fuente: Elaboración propia.

Para el criterio dos, se recorre toda la configuración y se evalúa si la posición de la orden de origen y de la orden de destino mostrada en la Figura 3-2 cumple alguno de los condicionales ver Figura 4-6, para así poder calcular la tardanza teniendo en cuenta el intercambio de posiciones no consecutivas.

Figura 4-6 Pseudocódigo de los intercambios entre atributos no consecutivos.

Si ($i < PIO$ ó $i > PID$)

$SumaTp = SumaTp + Tp(i)$

$TardanzaActual = TardanzaActual + \max(0, SumaTp - Fe(i))$

Si ($i < PID$ y $i > PIO$)

$SumaTp = SumaTp + Tp(i)$

$TardanzaActual = TardanzaActual + \max(0, SumaTp - Fe(i))$

Si ($i = PIO$)

$SumaTp = SumaTp + Tp(PID)$

$TardanzaActual = TardanzaActual + \max(0, SumaTp - Fe(PID))$

Si ($i = PID$)

$SumaTp = SumaTp + Tp(PIO)$

$TardanzaActual = TardanzaActual + \max(0, SumaTp - Fe(PIO))$

Fuente: Elaboración propia.

Luego de escoger el planteamiento adecuado para cada uno de los intercambios, se calcula si la configuración evaluada disminuye la tardanza de las órdenes. Para ello, se compara la mejor tardanza (MejorTardanza) que se ha obtenido hasta el momento con la Búsqueda Tabú con la tardanza de la configuración actual (TardanzaActual), si la configuración actual (TardanzaActual) resulta tener una menor tardanza que la mejor tardanza lo que se realiza es escoger dicha configuración como la que tiene mejor tardanza, esto se puede ver en la Figura 4-7:

Figura 4-7 Pseudocódigo con el cálculo de la mejor tardanza.

```
Si (MejorTardanza ≤ TardanzaActual)
  Mostrar MejorTardanza

Si no
  MejorTardanza = TardanzaActual
  Mostrar MejorTardanza
```

Fuente: Elaboración propia.

4.10 Memoria de corto plazo (Lista Tabú)

Para evitar caer en óptimos locales y poder reactivar la Búsqueda Tabú, se construyó una lista tabú de siete posiciones. Dicha lista, almacena la posición de la orden que se intercambié en la configuración y que al evaluar la función objetivo logro obtener una menor tardanza que la que se conocía hasta ese momento. De esta manera, esa posición no se podrá intercambiar nuevamente hasta que trascurren siete iteraciones, instante en el cual la posición sale de la lista tabú.

Estas posiciones, son almacenadas en la lista tabú, con el fin de explorar nuevas configuraciones dejando fija la posición que obtuvo mejores resultados, y así facilitar que durante las siete iteraciones siguientes se pueda encontrar configuraciones que sean mejores que la actual.

4.11 Criterio de parada

Para el metaheurístico desarrollado, se usó como criterio de parada el número máximo de iteraciones en las cuales se busca la mejor secuencia de órdenes que minimicen la tardanza. Para este trabajo, se usó como número máximo de cien iteraciones.

Con la definición del criterio de parada del metaheurístico empleado en esta tesis, se concluye el capítulo 4. Logrando describir: el proceso de preparación de pedidos y las características del almacén seleccionado para esta tesis. Adicionalmente, se crean

treinta escenarios de aplicación y se describe la estructura básica del metaheurístico de Búsqueda Tabú y las variaciones que se realizan para adaptarlo al problema tratado.

5. Análisis de Resultados

En este capítulo, se comparan los resultados obtenidos con el metaheurístico de Búsqueda Tabú y con la regla de priorización *Earliest Due Date (EDD)*, en cada uno de los escenarios planteados en el capítulo 4. Dicha comparación, se realiza por medio de un diseño de experimentos.

5.1 Regla de despacho seleccionada para comparar los resultados

Se revisan varias reglas de prioridad, entre las cuales se encuentran la regla *FIFO (First In First Out)*, que consiste en priorizar la llegada de las ordenes sin importar el tiempo de entrega de estas. Por lo cual, se descarta para este análisis; se revisa también la regla de priorización *LIFO (Last In First Out)*, la cual prioriza la última orden en ingresar atendiéndola de primera en el almacén. Sin embargo, también se descarta esta regla ya que no tiene en cuenta la fecha de entrega de los pedidos de los clientes; se revisa también la regla de prioridad *SPT (Shortest Process Time)* que consiste en ordenar las ordenes de acuerdo al tiempo de procesamiento de estas en el almacén, concentrándose en atender las ordenes que menor tiempo de procesamiento toman, por lo cual si una orden de un cliente tiene productos cuyo tiempo de procesamiento es alto, pese a que su fecha de vencimiento esta próxima, no se priorizara y se incumplirá en la entrega de la orden al cliente. Por último, se revisa la regla *Earliest Due Date*, que se centra en atender las órdenes de acuerdo a su fecha de entrega, lo que permite priorizar las órdenes cuya fecha de entrega está más cercana. Lo que genera que se concentren los esfuerzos del almacén en el cumplimiento de las fechas de entrega de los clientes, lo cual coincide con los propósitos del metaheurístico desarrollado. Por lo anterior, se decide utilizar la regla de priorización *Earliest Due Date* para realizar el análisis comparativo.

5.2 Prueba de hipótesis seleccionada

Para realizar la validación de los resultados en esta tesis, se decidió emplear una prueba *T* pareada, debido a que se hará una comparación entre dos métodos diferentes (regla de priorización *Earliest Due Date (EDD)* y Metaheurístico de Búsqueda Tabú) usando el mismo conjunto de datos en cada prueba experimental, lo que hace que las muestras sean dependientes entre sí. Esto, se hace con el fin de determinar cuál de los dos métodos es más conveniente para abordar el tema tratado en esta tesis.

5.3 Obtención de datos iniciales

Con el fin de obtener los datos iniciales que nos permitan estimar el tamaño muestral necesario para darle confiabilidad al estudio. Se define como la muestra uno, el valor total de las tardanzas obtenidas por medio del metaheurístico de Búsqueda Tabú y como la muestra dos, el valor total de las tardanzas obtenidas por medio de la regla de priorización *Earliest Due Date (EDD)*, para obtener estos datos de cada una de estas muestras, en cada iteración se variaron los parámetros: fecha de entrega, tiempo de procesamiento, cantidad de órdenes a secuenciar en el almacén, número de productos por orden, cantidad de productos por orden, número de equipos de manejo de materiales a emplear.

Para el cálculo de la tardanza, tanto para el método que usa la regla *Earliest Due Date (EDD)*, como para el que usa el metaheurístico de Búsqueda Tabú, se usa la misma fecha de entrega, el mismo tiempo de procesamiento, la misma cantidad de órdenes a secuenciar en el almacén, el mismo número de productos por orden, la misma cantidad de productos por orden, el mismo número de equipos de manejo del almacén con base en los escenarios planteados en el capítulo 4. Ver resultados obtenidos en la Tabla 5-1.

Tabla 5-1 Resultados de las tardanzas obtenidos por el metaheurístico de Búsqueda Tabú desarrollado en esta tesis (uno) y por la regla *Earliest Due Date (EDD)* (dos)

Dato Número	Búsqueda Tabú (uno): Tardanza en días	<i>Earliest Due Date</i> (dos): Tardanza en días
1	3,13	3,59
2	1,64	2,70

Dato Número	Búsqueda Tabú (uno): Tardanza en días	<i>Earliest Due Date</i> (dos): Tardanza en días
3	4,46	5,36
4	4,23	5,08
5	1,81	1,97
6	2,60	2,90
7	2,14	2,82
8	3,33	3,68
9	2,96	3,87
10	4,95	4,98
11	3,01	3,57
12	3,46	4,97
13	1,60	1,80
14	3,74	5,76
15	4,87	7,07
16	1,72	2,23
17	2,14	2,64
18	2,65	3,06
19	3,96	4,58
20	4,59	5,44
21	4,29	5,03
22	4,72	5,86
23	4,15	5,81
24	2,16	3,00
25	2,49	2,78
26	4,89	6,04
27	3,31	4,02
28	2,00	2,04
29	1,48	2,05
30	2,66	2,71

Fuente: Elaboración propia.

Debido a la prueba de hipótesis seleccionada, se requiere la estimación de las diferencias de las muestras uno y dos, las cuales se muestran en la última columna de la Tabla 5-2.

Tabla 5-2 Diferencia entre la muestra uno y la muestra dos

Dato Número	Diferencia (Uno – dos)
1	-0,46
2	-1,06
3	-0,90
4	-0,85
5	-0,16
6	-0,30
7	-0,68
8	-0,35
9	-0,91
10	-0,03
11	-0,56
12	-1,51
13	-0,20
14	-2,02

Dato Número	Diferencia (Uno – dos)
15	-2,20
16	-0,51
17	-0,50
18	-0,41
19	-0,62
20	-0,85
21	-0,74
22	-1,14
23	-1,66
24	-0,84
25	-0,29
26	-1,15
27	-0,71
28	-0,04
29	-0,57
30	-0,05

Fuente: elaboración propia.

5.4 Calculo del tamaño de la muestra

Para la estimación del tamaño de la muestra, se definen los parámetros necesarios, tales como: nivel de significancia (0,05) y valor de la potencia (0,90), usualmente, utilizados para la experimentación en ingeniería, (Montgomery, 2008); el valor de la desviación estándar, las medias de las muestras y su diferencia se calculan a partir de los datos obtenidos en la Tabla 5-1 de la diferencia de las medias entre las dos muestras usando la herramienta Minitab®. Los cuales se muestran en la Tabla 5-3.

Tabla 5-3 Parámetros de entrada ingresados en la herramienta Minitab® para el cálculo del tamaño de la muestra poblacional

Parámetros de entrada	Valor
Nivel de significancia	0,05
Valor potencia	0,90
Desviación Estándar de diferencias pareadas	0,50
Media muestra uno	2,00
Media muestra dos	2,40
Diferencia (media población dos menos media población uno)	0,40

Fuente: Elaboración propio.

El valor obtenido para el tamaño de la muestra necesaria para identificar una diferencia entre las medias de las muestras a evaluar con la prueba T pareada, es de diecinueve

datos, para ambas muestras, lo cual se observa en la Tabla 5-4. Dado que el tamaño de muestra necesario es menor al tamaño de la muestra inicial de treinta datos, se decide continuar trabajando con los datos de la Tabla 5-1, con el fin de hacer los análisis estadísticos necesarios.

Tabla 5-4 Resultados de Minitab® para el tamaño de la muestra poblacional que se debe emplear para el diseño experimental

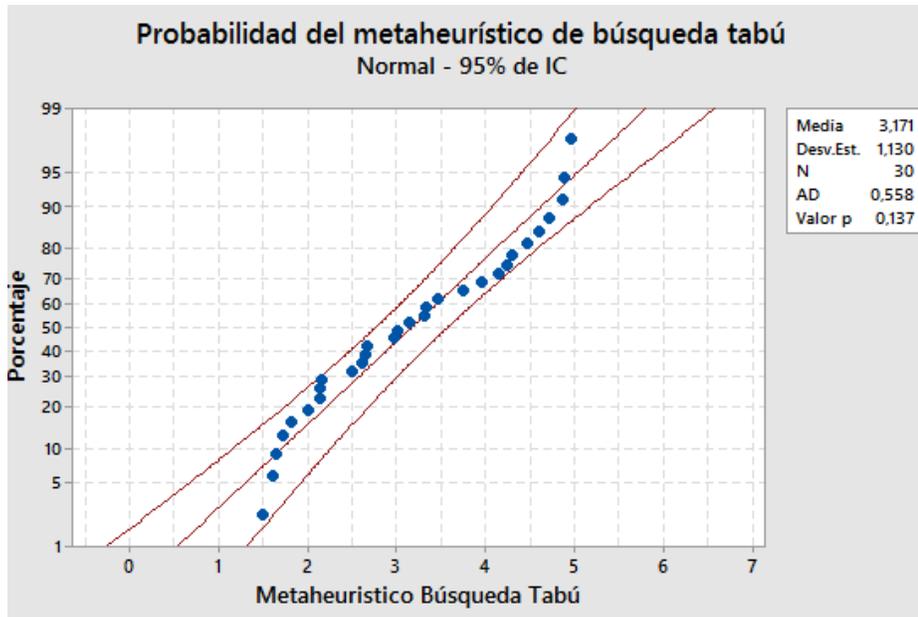
Diferencia	Tamaño de la muestra	Potencia Objetivo	Potencia Real
0,40	19	0,90	0,91

Fuente: Resultados arrojados por Minitab®.

5.5 Prueba de normalidad de las muestras

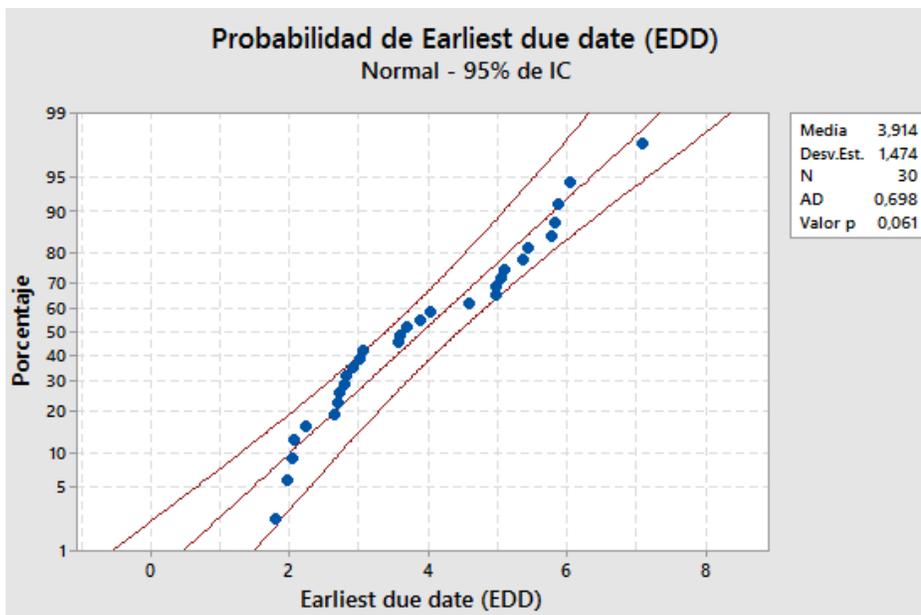
Para realizar el análisis de las pruebas de hipótesis T pareado se verifica la normalidad de los datos de las muestras usando el estadístico Anderson-Darling. Los resultados que se obtienen con la herramienta Minitab®, muestran que tanto para la muestra de datos obtenidos por el metaheurístico de Búsqueda Tabú desarrollado en esta tesis como para los obtenidos por medio de la regla de priorización *Earliest Due Date (EDD)*, se sigue una distribución normal, con un nivel de confianza del 95%, ya que el valor p de ambas muestras es mayor que 0,05. Ver Figura 5-1 para la muestra del metaheurístico y ver Figura 5-2 para la muestra de la regla de prioridad *Earliest Due Date*.

Figura 5-1 Prueba de normalidad para la muestra uno (Metaheurístico de Búsqueda Tabú), donde el valor p equivale a 0,14



Fuente: Resultados arrojados por Minitab®.

Figura 5-2 Prueba de normalidad para la muestra dos (*Earliest Due Date (EDD)*), donde el valor p equivale a 0,06



Fuente: Resultados arrojados por Minitab®.

5.6 Resultados de la prueba T pareada

Luego de verificar la normalidad de las muestras, se realiza la prueba T pareada en la herramienta Minitab®, con el fin de verificar que se obtienen valores menores para la tardanza usando el metaheurístico de Búsqueda Tabú desarrollado en esta tesis, en comparación con la regla de priorización *Earliest Due Date (EDD)*. Para esta prueba, se elige un nivel de confianza del 95%, se define a μ_1 como la media de los datos de la muestra del metaheurístico de Búsqueda Tabú (muestra uno), se define a μ_2 como la media de los datos de la muestra de la regla *Earliest Due Date (EDD)* (muestra dos), se define a μ_d como la diferencia de las medias de las muestras pareadas, se define la hipótesis nula que se muestra en la ecuación (5-1) y la hipótesis alternativa que se muestra en la ecuación (5-2).

$$H_0: \mu_1 - \mu_2 = 0 \text{ ó lo que equivale a } H_0: \mu_1 = \mu_2 \text{ o } H_0: \mu_d = 0 \quad (5-1)$$

H_0 : Es la media de la muestra del metaheurístico de Búsqueda Tabú igual a la media de la muestra de la regla *Earliest Due Date (EDD)*?

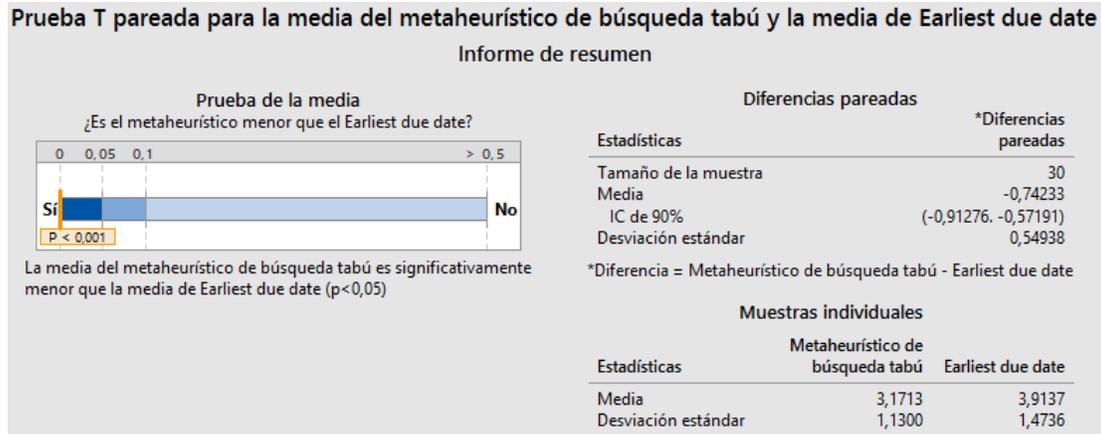
$$H_1: \mu_1 < \mu_2 \quad (5-2)$$

H_1 : Es la media de la muestra del metaheurístico de Búsqueda Tabú menor que la media de la muestra de la regla *Earliest Due Date (EDD)*?

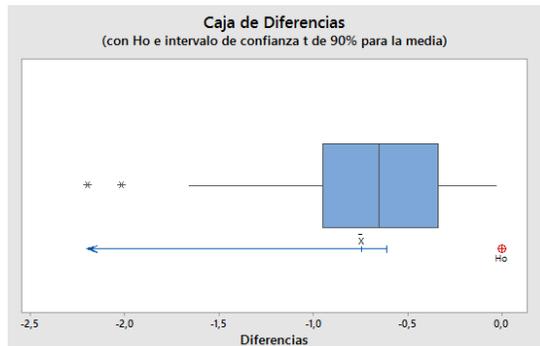
Los resultados obtenidos con la prueba T pareada en la herramienta Minitab®, Se muestran en la Figura 5-3, en donde en: (a) se confirma con un nivel de confianza del 95% que se rechaza la H_0 , ya que el valor p (0,001) es inferior a 0,05. Por lo cual, se acepta la H_1 que indica que la media de la muestra del metaheurístico de Búsqueda Tabú es menor que la media de la muestra de la regla *Earliest Due Date (EDD)*; en (b),(c) y (d) se muestra respectivamente el diagrama de cajas y bigotes de las diferencias de las medias, el diagrama de valores individuales de las diferencias de las medias y el histograma de la diferencia de las medias, los cuales muestran que la media de las diferencias entre μ_1 y μ_2 se aleja de cero, indicando que la media de las muestras son diferentes, además, la media de los datos se ubica en valores negativos, evidenciando que $\mu_1 < \mu_2$, por lo cual, se puede afirmar que se obtiene una menor tardanza usando el

modelo desarrollado en esta tesis en comparación a la regla de *Earliest Due Date (EDD)*. Ahora para determinar en qué porcentaje es mejor, se utilizan los datos de la Tabla 5-1 y Tabla 5-2 con los cuales se realiza una regla de tres con cada registro, luego se promedian estos resultados y se obtiene que con el metaheurístico desarrollado en esta tesis se obtiene una mejora promedio del 18% y un valor máximo de mejora del 39%.

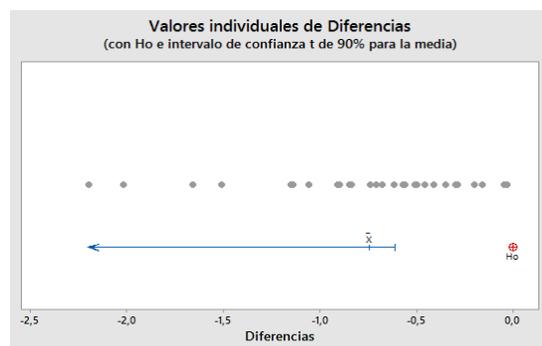
Figura 5-3 (a) Resultado arrojado por la herramienta Minitab® al aplicar la prueba *T* pareada, (b) Caja de diferencias, (c) Valores individuales de diferencias y (d) Histograma de diferencias.



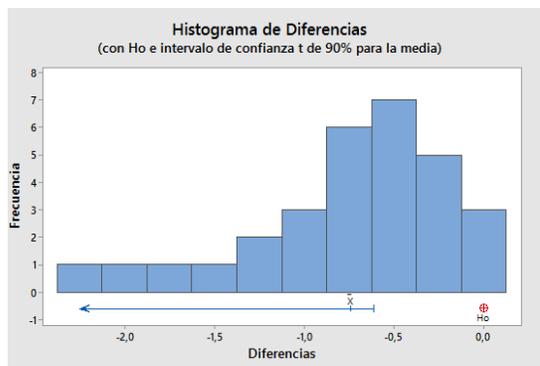
(a)



(b)



(c)



(d)

Fuente: Resultados arrojados por Minitab®.

6. Conclusiones y recomendaciones

6.1 Conclusiones

Se analizó el estado del arte de la programación de la preparación de pedidos en un almacén, se identificó una oportunidad investigativa, ya que actualmente no se tiene un modelo que permita determinar la secuencia en que se deben procesar las órdenes en la operación de preparación de pedidos, de tal manera que se minimice la tardanza teniendo en cuenta restricciones tales como: k equipos de manejo de materiales heterogéneos, n productos, m posiciones de almacenamiento y actualización del nivel de inventario.

Se planteó la función objetivo y las restricciones propias del problema de la programación de la preparación de pedidos en un almacén, considerando las variables definidas para este problema, que son fundamentales para el desarrollo del metaheurístico. Esta función, es novedosa ya que permite el cálculo de la tardanza según el tipo de intercambio.

Se realizó un análisis comparativo de los metaheurísticos más usados en la literatura, tales como: Búsqueda Tabú, Algoritmos Genéticos, Recocido Simulado, Optimización de enjambre de partículas (PSO) y Colonia Hormigas. A partir de esto, se tomó la decisión de usar el algoritmo de Búsqueda Tabú, ya que posee unas características que se acoplan muy bien al problema tratado, tales como: uso de memoria, penalización de atributos que permiten escapar de óptimos locales y sirve para resolver problemas como el del agente viajero.

Se usó una configuración de almacén existente, con la cual se crearon treinta escenarios de aplicación. Esto con el fin, de programar el metaheurístico de Búsqueda Tabú y realizar pruebas estadísticas.

Se programó en el lenguaje Java®, el metaheurístico de Búsqueda Tabú que permitiera encontrar la secuencia de órdenes de menor tardanza en el almacén, teniendo en cuenta restricciones como: k equipos de manejo de materiales heterogéneos, n productos, m posiciones de almacenamiento y actualización del nivel de inventario. Siendo esta la primera vez, que se aborda este tipo de modelo para la secuenciación de órdenes en un almacén.

Se compararon los resultados obtenidos con el metaheurístico de Búsqueda Tabú y con la regla *Earliest Due Date (EDD)*, a través de una prueba de hipótesis, la cual mostró con un nivel de confianza del 95%, que al usar el metaheurístico de Búsqueda Tabú propuesto, se consiguen menores tardanzas en comparación con las obtenidas por la regla *Earliest Due Date (EDD)*. Lo que nos confirma la bondad del método propuesto.

6.2 Recomendaciones y trabajos futuros

De esta tesis se desprenden los siguientes temas que podrían ser abordados en futuras investigaciones.

- Para la realizar esta tesis se usó un metaheurístico de Búsqueda Tabú con un tamaño de lista tabú de siete, y un criterio de parada de cien iteraciones. Por lo cual, se recomienda como trabajo futuro, evaluar el impacto que tiene el cambio de estos parámetros en los resultados de esta tesis.
- Investigar cómo se podría abordar el problema tratado en esta tesis con el uso de: matheurísticas, las cuales combinan técnicas de programación matemática con técnicas metaheurísticas; y algoritmos meméticos, que combinan diferentes características y estrategias de varios metaheurísticos. Para así, comparar si se obtienen mejores resultados que con el metaheurístico desarrollado en esta.
- Se recomienda trabajar en la mejora de los tiempos de ejecución del algoritmo desarrollado en la tesis por medio de la codificación.

- Se sugiere complementar el estudio realizando en esta tesis por medio de un análisis de varianza que permita comparar el método desarrollado en esta tesis con otras reglas de prioridad diferentes al *Earliest Due Date (EDD)*, con el fin de validar la pertinencia del modelo desarrollado.

A. Escenarios de aplicación

Se adjunta archivo de Excel con los treinta escenarios usados en el desarrollo de esta tesis.



Anexo_B_Escenarios
_de_aplicacion.xlsx

B. Listado de los productos que se usan en el almacén seleccionado

En este anexo, se muestra el listado de productos que se manejan en el almacén que se seleccionó en el capítulo 4.

Tabla B-1 Listado de productos en el almacén

Producto	
Código	Nombre
p1	ESPUMA MACH3 IRRITACION DEFENSE
p2	VEL ROSITA JABON 3X250 GR
p3	PURO JABON AZUL 3X235 GR
p4	LAVOMATIC POLVO FLORAL 2500 GR
p5	SALSA DE TOMATE FRUCO DOY PACK
p6	FRUCO MAY LIGHT DOYP 400 GR
p7	SPAGHETTI DORIA 1000 GR
p8	TORNILLO VERDURAS DORIA 500 GR
p9	SPAGHETTI HUEVO DORIA 1000 GR
p10	MACARRON CORTO DORIA 250 GR
p11	DORITOS MEGAQUESO X10 UNIDADES
p12	FRITO LAY LONCHERA
p13	RAMA MARGARINA CON SAL 250 GR
p14	RAMA MARGARINA CULINARIA 250 GR
p15	RAMA MARGARINA LINE 250 GR
p16	MAIZENA YUCARINA 300 GR
p17	MAIZENA FECULA MAIZ 380 GR
p18	MAIZENA NATILLA 340 GR
p19	MAIZENA BUNUELO 350 GR
p20	TOSTI EMPANADA
p21	PALOMITAS CARAMELO YUPI
p22	MANI CON UVAS PASAS
p23	MANI CON SAL LA ESPECIAL TARRO
p24	RAPIPAPA TRADICIONAL 1500 GR
p25	CHOCOLISTO TARRO 1000 GR
p26	CEREAL ZUCARITAS KELLOGGS 300GR
p27	CHOCOLATE CORONA 500GR
p28	MEZCLA DE NUECES FINAS DEL ALBA
p29	SALCHICHA VIENA ZENU ESPECIAL
p30	SALCHICHA VIENA ZENU POLLO
p31	GALLETA SALTIN TACO
p32	AREQUIPE VASO ALPINA
p33	CAPPUCINO CLASSIC COLCAFE 270 GR

Producto	
Código	Nombre
p34	COLCAFE PREMIUM 200 GR
p35	MILO LATA 400 GR
p36	CHOCOLYNE CON SPLENDA 120 GR
p37	LENTEJA ÉXITO
p38	SUSTAGEN VAINILLA 400 GR
p39	CHOCOLISTO DE FRESA 200 GR
p40	DOG CHOW ADULTOS RAZAS PEQUENAS
p41	DEO PERF TONE BARRA LADY SPEED STICK 45 GR
p42	DEO SPEED STICK STAINGUARD SPEED STICK 50 GR
p43	TALCO DESODORANTE LADY EXITO 150 GR
p44	TALCO DESODORANTE LADY EXITO 85 GR
p45	DESODORANTE CLINICAL SECRET SPORT 45 GR
p46	DESODORANTE GILLETTE COOL GEL 113 GRÂ
p47	TALCO MEDICINAL MEXSANA 150 GR
p48	FAB POLVO FLORAL 1800 GR
p49	DETERGENTE RINDEX FLORES ESPECIAL 2350 GR
p50	JABON BARRA EXITO X 3UNDS
p51	DETERGENTE PREMIUM ÉXITO
p52	CHOCOLYNE CON SPLENDA TARRO
p53	CHOCOLATE DIETETICO
p54	MALTEADA CHOCOLATE CHOCOLYNE
p55	LENTEJAS CON SALCHICHAS ÉXITO
p56	FRIJOL BOLA ROJA EXITO 1000 GR
p57	DESODORANTE CLINICAL SECRET SENSITIVE SKIN 45 GR
p58	TALCO LADY MEXSANA
p59	MEXSANA AVENA MEXSANA
p60	FAB POLVO SUAVIZ SOFLAN 2500 GR

Fuente: (Gómez, 2015).

Bibliografía

- Achedad, P. C., & Giménez, L. O. (2011). *Ingeniería de organización: Modelos y aplicaciones*. Ediciones Díaz de Santos.
- Alonso-Ayuso, A., Tirado, G., & Udías, Á. (2013). On a selection and scheduling problem in automatic storage and retrieval warehouses. *International Journal of Production Research*, 51(17), 5337-5353. <https://doi.org/10.1080/00207543.2013.813984>
- Azadnia, A. H., Taheri, S., Ghadimi, P., Mat Saman, M. Z., & Wong, K. Y. (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, e246578. <https://doi.org/10.1155/2013/246578>
- Baker, P. (2004). Aligning Distribution Center Operations to Supply Chain Strategy. *The International Journal of Logistics Management*, 15(1), 111-123. <https://doi.org/10.1108/09574090410700266>
- Baker, P., & Halim, Z. (2007). An exploration of warehouse automation implementations: Cost, service and flexibility issues. *Supply Chain Management*, 12(2), 129-138. <https://doi.org/10.1108/13598540710737316>
- Ballou, R. H. (2004). *Business Logistics/supply Chain Management: Planning, Organizing, and Controlling the Supply Chain*. Pearson Prentice Hall.
- Bozer, Y. A., & Kile, J. W. (2008). Order batching in walk-and-pick order picking systems. *International Journal of Production Research*, 46(7), 1887-1909. <https://doi.org/10.1080/00207540600920850>

- Chopra, S., & Meindl, P. (2012). *Supply Chain Management: Strategy, Planning, and Operation* (5.^a ed.). Boston: Pearson.
- Choy, K. I., Sheng, N., Lam, H. y., Lai, I. K. W., Chow, K. h., & Ho, G. t. s. (2014). Assess the effects of different operations policies on warehousing reliability. *International Journal of Production Research*, 52(3), 662-678.
<https://doi.org/10.1080/00207543.2013.827807>
- COIN-OR. (2016). Computational Infrastructure for Operations Research Home Page. Recuperado 24 de octubre de 2016, a partir de <http://www.coin-or.org/>
- Correa, A. A., Rodríguez, E., & Gómez, R. A. (2014). Modelamiento del ruteo del acomodo de tiempo mínimo en centros de distribución (CEDI) usando Búsqueda Tabú. *Revista Soluciones de Postgrado*, 6(12), 15-28.
- Correa, A., & Gómez, R. A. (2009). Tecnologías de la información en la cadena de suministro. *DYNA*, 76(157), 37-48.
- Coyle, J. J., Bardi, E. J., & Langley, C. J. (2003). *The Management of Business Logistics: A Supply Chain Perspective*. South-Western/Thomson Learning.
- De Koster, M. B. M., Van der Poort, E. s., & Wolters, M. (1999). Efficient orderbatching methods in warehouses. *International Journal of Production Research*, 37(7), 1479-1504. <https://doi.org/10.1080/002075499191094>
- De Koster, R., Le-Duc, T., & Roodbergen, K. J. (2007). Design and control of warehouse order picking: A literature review. *European Journal of Operational Research*, 182(2), 481-501. <https://doi.org/10.1016/j.ejor.2006.07.009>
- De la Figuera, D. S. (2005). *La logística empresarial en el nuevo milenio*. Gestión 2000.
- Deitel, P. J. (2004). *Cómo programar en Java*. Pearson Educación.

- Delgado, E. J. M. (2010). *Supernova un algoritmo novedoso de optimización global*. Universidad Nacional de Colombia - Sede Medellín, Colombia. Recuperado a partir de <http://www.bdigital.unal.edu.co/2035/>
- Dorigo, M. (1992). *Optimization, learning and natural algorithms* (Ph. D. Thesis). Politecnico di Milano, Italy. Recuperado a partir de <http://ci.nii.ac.jp/naid/10016599043/>
- Escobar, A. E., Gallego, R. A., & Toro, E. M. (2015). *Técnicas Heurísticas y Metaheurísticas*. Universidad Tecnológica de Pereira.
- Foundation Eclipse. (2016). Eclipse Neon. Recuperado 24 de octubre de 2016, a partir de <https://eclipse.org/>
- Frazelle, E. (2002). *Supply Chain Strategy: The Logistics of Supply Chain Management*. McGraw-Hill.
- Frazelle, E., & Sojo, R. (2006). *Logística de Almacenamiento y Manejo de Materiales de Clase Mundial*. Grupo Editorial Norma.
- Glover, F. (1989). Tabu Search—Part I. *ORSA Journal on Computing*, 1(3), 190-206. <https://doi.org/10.1287/ijoc.1.3.190>
- Glover, F. W., & Laguna, M. (1997). *Tabu Search* (1.^a ed.). Springer US.
- Gómez, R. A. (2015). *Problemas de conformación de lotes con ruteo en el acomodo y la preparación de pedidos considerando K equipos heterogéneos* (Ph. D. Thesis). Universidad Nacional de Colombia - Sede Medellín, Colombia. Recuperado a partir de <http://www.bdigital.unal.edu.co/50619/>
- Henn, S. (2012). Algorithms for on-line order batching in an order picking warehouse. *Computers & Operations Research*, 39(11), 2549-2563. <https://doi.org/10.1016/j.cor.2011.12.019>

- 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(1), 86-114. <https://doi.org/10.1007/s10696-012-9164-1>
- Henn, S., & Schmid, V. (2013). Metaheuristics for order batching and sequencing in manual order picking systems. *Computers & Industrial Engineering*, 66(2), 338-351. <https://doi.org/10.1016/j.cie.2013.07.003>
- Ho, Y.-C., Su, T.-S., & Shi, Z.-B. (2008). Order-batching methods for an order-picking warehouse with two cross aisles. *Computers & Industrial Engineering*, 55(2), 321-347. <https://doi.org/10.1016/j.cie.2007.12.018>
- Ho, Y.-C., & Tseng, Y.-Y. (2006). A study on order-batching methods of order-picking in a distribution centre with two cross-aisles. *International Journal of Production Research*, 44(17), 3391-3417. <https://doi.org/10.1080/00207540600558015>
- Holland, J. H. (1975). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press.
- Hsieh, L. F., & Fan, C. Y. (2010). A new SOM batching heuristic for order picking systems. En *2010 IEEE 17Th International Conference on Industrial Engineering and Engineering Management (IE EM)* (pp. 1478-1481). <https://doi.org/10.1109/ICIEEM.2010.5646042>
- Hsieh, L. F., & Huang, Y. C. (2011). New batch construction heuristics to optimise the performance of order picking systems. *International Journal of Production Economics*, 131(2), 618-630. <https://doi.org/10.1016/j.ijpe.2011.02.006>
- IBM. (2016). IBM DB2 – Database software – IBM Analytics. Recuperado 24 de octubre de 2016, a partir de <http://www.ibm.com/analytics/us/en/technology/db2/>

- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. En , *IEEE International Conference on Neural Networks, 1995. Proceedings* (Vol. 4, pp. 1942-1948 vol.4).
<https://doi.org/10.1109/ICNN.1995.488968>
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598), 671-680.
- Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering – A systematic literature review. *Information and Software Technology*, 51(1), 7-15.
<https://doi.org/10.1016/j.infsof.2008.09.009>
- Kroenke, D. M. (2003). *Procesamiento de bases de datos: fundamentos, diseño e implementación*. Pearson Educación.
- Lourenço, H. R. (2005). Logistics management: An opportunity for metaheuristics. *Operations Research/ Computer Science Interfaces Series*, 30, 329-356.
- Machuca, J. A. D. (1995). *Dirección de operaciones: aspectos estratégicos en la producción y los servicios*. McGraw-Hill.
- Mendonca, M., & Cowan, D. (2010). Decision-making coordination and efficient reasoning techniques for feature-based configuration. *Science of Computer Programming*, 75(5), 311-332. <https://doi.org/10.1016/j.scico.2009.12.004>
- Mentzer, J. (2004). *Fundamentals of Supply Chain Management: Twelve Drivers of Competitive Advantage*. United States: SAGE Publications, Inc.
- Microsoft. (2016b). Software y aplicaciones de base de datos | Microsoft Access.
Recuperado 24 de octubre de 2016, a partir de <https://products.office.com/ES-CO/access>
- Microsoft. (2016a). SQL Server 2016 | Microsoft. Recuperado 24 de octubre de 2016, a partir de <https://www.microsoft.com/en-us/sql-server/sql-server-2016>

Minitab Inc. (2016). Minitab. Recuperado 24 de octubre de 2016, a partir de

<https://www.minitab.com/es-mx/>

Montgomery, D. C. (2008). *Design and Analysis of Experiments*. John Wiley & Sons.

Morris, M. (1994). *Arquitectura de computadoras* (3.^a ed.). Pearson Educacion.

Muñoz, A. D. (2007). *Metaheurísticas*. Madrid: Librería-Editorial Dykinson.

Oracle Corporation. (2016a). Java. Recuperado 24 de octubre de 2016, a partir de

<https://www.java.com/es>

Oracle Corporation. (2016d). MySQL. Recuperado 24 de octubre de 2016, a partir de

<https://www.mysql.com/>

Oracle Corporation. (2016c). Oracle | Integrated Cloud Applications and Platform

Services. Recuperado 24 de octubre de 2016, a partir de

<https://www.oracle.com/index.html>

Oracle Corporation. (2016b). Welcome to NetBeans. Recuperado 24 de octubre de 2016,

a partir de <https://netbeans.org/>

Pinedo, M. L. (2009). Manufacturing Models. En *Planning and Scheduling in*

Manufacturing and Services (pp. 19-35). Springer New York. Recuperado a partir

de http://link.springer.com/chapter/10.1007/978-1-4419-0910-7_2

Piroird, F., & Dale, B. G. (1998). The importance of lead time control in the order fulfilment process. *Production Planning & Control*, 9(7), 640-649.

<https://doi.org/10.1080/095372898233632>

PostgreSQL Global Development Group. (2016). PostgreSQL: The world's most

advanced open source database. Recuperado 24 de octubre de 2016, a partir de

<https://www.postgresql.org/>

- Reeves, C. R. (1995). *Modern Heuristic Techniques for Combinatorial Problems*. McGraw-Hill.
- Rosenwein, M. B. (1996). A comparison of heuristics for the problem of batching orders for warehouse selection. *International Journal of Production Research*, 34(3), 657-664. <https://doi.org/10.1080/00207549608904926>
- Rossi, F., van Beek, P., & Walsh, T. (2006). *Handbook of Constraint Programming* (1.^a ed.). Elsevier Science.
- Ruben, R. A., & Jacobs, F. R. (1999). Batch Construction Heuristics and Storage Assignment Strategies for Walk/Ride and Pick Systems. *Management Science*, 45(4), 575-596. <https://doi.org/10.1287/mnsc.45.4.575>
- Rubrico, J. I. U., Higashi, T., Tamura, H., & Ota, J. (2011). Online rescheduling of multiple picking agents for warehouse management. *Robotics and Computer-Integrated Manufacturing*, 27(1), 62-71. <https://doi.org/10.1016/j.rcim.2010.06.011>
- Rubrico, J. I. U., Ota, J., Higashi, T., & Tamura, H. (2006). Scheduling multiple agents for picking products in a warehouse. En *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. (pp. 1438-1443). <https://doi.org/10.1109/ROBOT.2006.1641911>
- Tompkins, J. A., & Smith, J. D. (1998). *The Warehouse Management Handbook* (2.^a ed.). Tompkins Pr.
- Van Den Berg, J. P. (1999). A literature survey on planning and control of warehousing systems. *IIE Transactions (Institute of Industrial Engineers)*, 31(8), 751-762.
- Van Den Berg, J. P. (2007). *Integral Warehouse Management: The Next Generation in Transparency, Collaboration and Warehouse Management Systems*. Management Outlook.

- Velásquez, J. D. (2015a). Una Guía Corta para Escribir Revisiones Sistemáticas de Literatura Parte 3. *DYNA*, 82(189), 9-12.
<https://doi.org/10.15446/dyna.v82n189.48931>
- Velásquez, J. D. (2015b). Una Guía Corta para Escribir Revisiones Sistemáticas de Literatura Parte 4. *DYNA*, 82(190), 9-12.
<https://doi.org/10.15446/dyna.v82n190.49511>
- Waters, C. D. J. (2003). *Global Logistics and Distribution Planning: Strategies for Management* (4.^a ed.). United kingdom: Kogan Page.
- Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1), 67-82.
<https://doi.org/10.1109/4235.585893>
- Xinox Software. (2000, 2015). JCreator — Java IDE. Recuperado 24 de octubre de 2016, a partir de <http://www.jcreator.com/>
- Zanakis, S. H., & Evans, J. R. (1981). Heuristic «Optimization»: Why, When, and How to Use It. *Interfaces*, 11(5), 84-91. <https://doi.org/10.1287/inte.11.5.84>