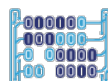




Universidad del  
**Rosario**

Escuela de Ingeniería,  
Ciencia y Tecnología



**MACC**  
Matemáticas Aplicadas y  
Ciencias de la Computación

TRABAJO FIN DE GRADO

# Algoritmos Heurísticos para la Optimización del Problema de Asignación de Contenedores en un Puerto Marítimo

Realizado por  
**Santiago Hoyos Ortiz**

Para la obtención del título de  
Profesional en Matemáticas Aplicadas y Ciencias de la Computación

Dirigido por  
Juan Camilo Yepes Borrero

Codirigido por  
Yiby Karolina Morales Pinto

En la escuela de  
Ingeniería, Ciencia y Tecnología

# Agradecimientos

---

En primer lugar quisiera expresar mi más profundo agradecimiento a mi madre, por ser mi apoyo incondicional durante toda mi vida y educación.

Asimismo, agradezco de corazón a mis hermanos y amigos, por motivarme y darme aliento en los momentos más difíciles de este proceso, y por compartir conmigo tantos momentos de alegría que me dieron energía para continuar.

Mi reconocimiento sincero a todos mis profesores a lo largo de la carrera, por transmitirme sus valiosos conocimientos que hoy plasmo en este trabajo. En especial, mi profundo agradecimiento a Juan Camilo Yepes que, más que un profesor, ha sido mi mentor y amigo; que siempre creyó en mí y me inspiró con su ejemplo a esforzarme por alcanzar mis metas.

Finalmente, mi gratitud a la Universidad por darme la oportunidad de forjarme como profesional, y a todos quienes fueron parte de este camino. Se cierra una etapa muy importante, y el aprendizaje obtendrá sus frutos en el futuro que comienza.

# Glosario

---

- **Heurística:** Método o técnicas que encuentran soluciones óptimas a problemas computacionales complejos en tiempos de ejecución prácticos, mediante la aplicación de reglas empíricas y estrategias que reducen el espacio de soluciones a explorar.
- **Metaheurística:** Heurística de mayor nivel, usualmente de naturaleza iterativa, que coordina y guía de forma inteligente la interacción entre diversos procedimientos heurísticos más simples, con el objetivo de encontrar soluciones de muy alta calidad para problemas complejos de optimización combinatoria.
- **GRASP:** Sigla en inglés de Procedimiento de Búsqueda Adaptativa Aleatoria (Greedy Randomized Adaptive Search Procedure). Popular metaheurística que funciona integrando de forma iterativa e incremental una fase de construcción de soluciones iniciales mediante un proceso goloso aleatorizado, y una fase de mejoramiento iterativo de estas soluciones mediante una búsqueda local.
- **RPD:** Sigla de Desviación Porcentual Relativa (Relative Percentage Deviation). Métrica comúnmente utilizada para medir el desempeño de un algoritmo heurístico, calculada como la desviación porcentual de la solución obtenida por dicho algoritmo con respecto a la mejor solución conocida para una instancia de prueba.
- **ANOVA:** Sigla en inglés de Análisis de Varianza (Analysis of Variance). Técnica estadística inferencial para determinar si existen diferencias significativas entre las medias de dos o más grupos de una variable cuantitativa. El ANOVA compara las varianzas internas y entre grupos para hacer esta detección.
- **Prueba de Tukey:** Método estadístico basado en el ANOVA que se utiliza en las comparaciones múltiples por pares de medias, cuando se desea encontrar qué

grupos específicos dentro de un análisis de varianza poseen medias significativamente diferentes entre sí. Lleva el nombre de su creador, el estadístico John Tukey.

- **Aleatorización:** Técnica heurística que incorpora elementos aleatorios controlados dentro de un algoritmo con el fin de aumentar la diversificación durante la búsqueda y exploración de un espacio de soluciones, evitando quedar atrapado prematuramente en óptimos locales como ocurriría con un enfoque puramente voraz.
- **Búsqueda local:** Procedimiento iterativo implementado típicamente después de una fase constructiva en metaheurísticas, que busca mejorar de forma incremental una solución inicial mediante la exploración intensiva de su entorno o vecindario cercano, a través de movimientos, intercambios o pequeñas perturbaciones.

# Resumen

---

La optimización de la asignación de contenedores en una terminal portuaria implica desarrollar estrategias eficientes para organizar y mover los contenedores de manera eficaz. Esto incluye considerar factores como la disponibilidad de espacio, la duración del traslado y la minimización del tiempo, teniendo en cuenta si los contenedores se destinan al transporte marítimo o terrestre.

Mediante la implementación de técnicas de optimización, es posible mejorar la eficiencia en la asignación de contenedores. Esto no solo agiliza los procesos en el patio de contenedores, sino que también contribuye a maximizar la productividad, sostenibilidad y los beneficios económicos del puerto.

El enfoque principal de este trabajo de fin de grado es generar una propuesta de mejora para resolver el problema de asignación de contenedores, que minimice el tiempo total requerido para la asignación de cada contenedor en los bloques. Para resolver el problema, se emplean conceptos y estrategias algorítmicas y estadísticas, específicamente métodos de optimización metaheurísticos como el algoritmo GRASP.

Por último, por medio de diferentes experimentos haciendo uso de las instancias generadas previamente en la literatura, se analizarán los resultados del GRASP planteado con otro algoritmo de proximidad existente. De esta forma, se demostrará qué método genera la mejor solución para cada caso.

**Palabras clave:** Optimización, algoritmo GRASP, modelos metaheurísticos

# Abstract

---

Optimizing container allocation in a port terminal involves developing efficient strategies to organize and move containers effectively. This includes considering factors such as space availability, duration of the transfer and minimization of time, taking into account whether the containers are destined for sea or land transport.

By implementing optimization techniques, it is possible to improve the efficiency of container allocation. This not only streamlines processes at the container yard, but also helps maximize the port's productivity, sustainability and economic benefits.

The main focus of this final degree project is to generate an improvement proposal for the container assignment problem, which minimizes the total time required for the assignment of each container in the blocks. To solve the problem, algorithmic and statistical concepts and strategies are used, specifically metaheuristic optimization methods such as the GRASP algorithm.

Finally, through different experiments using the instances previously generated in the literature, the results of the GRASP with another existing proximity algorithm will be analyzed. In this way, it will be demonstrated which method generates the best solution for each case.

**Keywords:** Optimization, proximity algorithm, metaheuristic models

# Índice general

---

<b>1</b>	<b>Introducción</b>	<b>12</b>
1.1.	Objetivos	13
1.1.1.	Objetivo Principal	13
1.1.2.	Objetivos Específicos	13
1.2.	Motivación	13
1.3.	Metodología	15
1.3.1.	Esquema de Metodología	18
1.4.	Estructura del documento	19
<b>2</b>	<b>Terminales Portuarias</b>	<b>21</b>
2.1.	Contexto histórico, económico y ambiental del transporte marítimo en Colombia	21
2.2.	Funcionamiento de una terminal portuaria	25
2.2.1.	Estructura y elementos de un puerto	25
2.2.2.	Configuración Europea y Asiática	29
2.2.3.	Problemas de optimización y logística en una terminal portuaria	33
<b>3</b>	<b>Revisión Bibliográfica</b>	<b>35</b>
<b>4</b>	<b>Problema de Asignación de Contenedores</b>	<b>39</b>
4.1.	Tipos de contenedores	39
4.2.	Descripción Formal del Problema	40
4.3.	Modelación Matemática del Problema	42
4.3.1.	Modelo	43

<b>5 Experimentos Computacionales</b>	<b>46</b>
5.1. Algoritmo de Proximidad	46
5.2. Algoritmo GRASP	49
5.2.1. Aleatorizacion	49
5.2.2. Búsqueda Local	52
<b>6 Resultados</b>	<b>55</b>
6.1. ANOVA y Prueba Tukey	60
<b>7 Conclusiones y futuras líneas de investigación</b>	<b>63</b>
7.1. Futuras Líneas de Investigación	64
<b>Bibliografía</b>	<b>66</b>

# Índice de figuras

---

1.1. Esquema de metodología. . . . .	18
2.1. Quayside Crane. [16] . . . . .	26
2.2. Estructura de una terminal portuaria. [14] . . . . .	27
2.3. Estructura de una terminal portuaria. [56] . . . . .	27
2.4. Zona de almacenamiento y bloques. [33] . . . . .	28
2.5. Rubber tyred gantry cranes (RTGCs). [64] . . . . .	29
2.6. Configuración europea en el Puerto de Rotterdam. [37] . . . . .	30
2.7. Configuración asiática en el Puerto de Cartagena. [19] . . . . .	31
2.8. Configuración asiática (a) y configuración europea (b). [11] . . . . .	32
6.1. Comparación gráfica en unidades de tiempo entre cada algoritmo. . . . .	58

# Índice de tablas

---

6.1. Tabla de comparación de los 4 algoritmos para tiempos de asignación (ut), tiempos de computo (ms) y RPD. . . . .	56
6.2. Resultados del ANOVA . . . . .	60
6.3. Resultados de la Prueba de Tukey . . . . .	61

# Índice de Algoritmos

---

1.	Algoritmo de proximidad de Carlos Romero . . . . .	48
2.	Fase Constructiva - Aleatorización . . . . .	51
3.	Búsqueda Local General . . . . .	52
4.	GRASP . . . . .	54

# 1. Introducción

---

Desde el siglo XX, se viene observando un creciente interés y necesidad de abordar los desafíos ambientales derivados de la actividad industrial y social [42]. En este contexto, los buques de carga y los puertos marítimos han desempeñado un papel crucial en los aspectos ambientales y económicos. El transporte marítimo se ha consolidado como el medio más eficiente para el traslado de mercancías, gracias a su bajo costo, capacidad para transportar grandes volúmenes a largas distancias y su rentabilidad [21].

Sin embargo, a pesar de las ventajas en términos de transporte, esta industria conlleva un significativo impacto ambiental debido al tamaño de los buques y a la eficiencia de los puertos. Se ha reconocido que una mayor capacidad de carga de los buques se traduce en un menor consumo de combustible por tonelada transportada, lo que reduce el impacto ambiental [65]. Además, los tiempos de espera de los buques en las terminales portuarias generan un consumo continuo de combustible, ya que los motores nunca se detienen durante los procesos de carga y descarga [39].

En términos económicos, los puertos desempeñan un papel clave en el desarrollo del comercio y la industria de las zonas en las que se encuentran, generando empleo e inversiones significativas [6]. Por tanto, la eficiencia logística en el funcionamiento interno de cada terminal portuaria resulta esencial. Algunos de los procesos logísticos relevantes incluyen la planificación del atraque, la asignación y localización de grúas y contenedores [56].

En este contexto, el presente trabajo de investigación se centra en la optimización de la asignación de contenedores en una terminal portuaria, por lo que se presentará la estructura general de un puerto. La gestión óptima de la asignación de los contenedores es crucial para las compañías del ámbito logístico y portuario. Al disminuir los gastos operativos y los tiempos de espera de los barcos, se alcanza una mejora notable en la eficacia y rentabilidad de las operaciones.

## 1.1. Objetivos

### 1.1.1. Objetivo Principal

Resolver el problema de asignación de contenedores en terminales portuarias, buscando minimizar los tiempos de operación.

### 1.1.2. Objetivos Específicos

- Desarrollar un algoritmo GRASP que permita escoger eficientemente el sector más idóneo entre los bloques más próximos a un contenedor extraído de un buque.
- Analizar los resultados generados por el algoritmo GRASP planteado y contrastarlos con los reportados previamente en la literatura.
- Identificar posibles mejoras en problemas de optimización combinatoria aplicadas a las terminales portuarias.

## 1.2. Motivación

Las actividades en los puertos de carga se han caracterizado por dirigirse bajo sistemas de organización y logística como la asignación de contenedores; los cuales se ubican en los diferentes bloques de la terminal buscando una planificación efectiva para los futuros traslados y almacenamiento [2]. Ahora bien, dicha forma de organización requiere de un sistema de mejora, el cual permita impactar positivamente en aspectos económicos, sociales y ambientales.

Implementar sistemas que permitan optimizar organización y tiempos de asignación en las terminales portuarias abre la posibilidad de generar un crecimiento económico regional dado que promueve un tránsito marítimo ágil, generando un desarrollo portuario; el cual permite una mejora en la tasa de empleabilidad de la zona [6].

Además, esta optimización no solo beneficia la eficiencia en el mar, sino que también se agiliza la parte terrestre en el puerto, lo que impulsa aún más el flujo de mercancías y reduce los costos de almacenamiento de los contenedores [41].

Dentro de las diversas estrategias que se han aplicado para la mejora de los puertos, se encuentra el Sistema de comunidad portuaria (PCS), el cual es utilizado en diferentes puertos marítimos alrededor del mundo [36]. Este permite optimizar y automatizar los procesos logísticos a través de un único envío de datos que conecta las cadenas logísticas y de transporte. Así pues, su implementación podría generar importantes beneficios en terminales importantes de Colombia; por ejemplo, el puerto de Buenaventura que mueve aproximadamente 14.356 millones de dólares por año [45].

Históricamente, las empresas han optado por el desarrollo u obtención de nuevas estrategias que les permitan aumentar su competitividad y mejorar el consumo de recursos [56]. Las terminales portuarias no son ajenas a dicha corriente; en consecuencia, la asignación de contenedores en el bloque ideal resulta en un interesante tema de investigación.

Por otra parte, agilizar las estrategias que se utilizan actualmente para la asignación de contenedores en los puertos, contribuye positivamente al medio ambiente. Según la Organización Marítima Internacional, los buques de carga son una de las principales fuentes de contaminación en el planeta, emitiendo alrededor de 1 000 millones de toneladas de dióxido de carbono (CO<sub>2</sub>). Esta cifra supera las emisiones anuales de dióxido de carbono de la industria de la aviación, que alcanzan los 781 millones de toneladas de CO<sub>2</sub> [57]. Y si bien los buques disminuyen la velocidad para mitigar la quema de combustible, siguen teniendo un impacto ecológico al no apagar sus motores en su llegada a la terminal portuaria, lo que contribuye significativamente al cambio climático y a la acidificación del océano [46]. Por tanto, aminorar el tiempo de espera que los buques requieren para la descarga y organización de contenedores es fundamental para la generación de un sistema portuario más sostenible y responsable con el medio ambiente.

Igualmente, la adopción de tecnología en la optimización de la asignación de

contenedores en los bloques de una terminal portuaria es crucial para mejorar la logística en este proceso. La implementación de sistemas de gestión avanzados, análisis de datos y algoritmos de optimización permite una asignación más eficiente de los contenedores [10]. Con sistemas de información en tiempo real y tecnologías interconectadas, se logra una mejor planificación de la carga y descarga, una asignación óptima de recursos y una reducción de tiempos de espera [53].

En resumen, la incorporación de tecnología en la asignación de contenedores permite una logística más precisa, eficiente y colaborativa, mejorando la operación y competitividad de la terminal portuaria, así como su impacto ambiental. De igual manera, se espera aportar al diseño de metodologías para la asignación de contenedores en trabajos de investigación futuros.

### **1.3. Metodología**

Contar con una buena logística de asignación de contenedores es un proceso crítico en las terminales portuarias. En este caso, un algoritmo eficiente puede ayudar a reducir el tiempo de carga y descarga de contenedores, lo que puede conducir a ahorros de costos y un mejor servicio al cliente.

El presente trabajo plantea desarrollar un nuevo algoritmo heurístico para resolver el problema de asignación de contenedores en terminales portuarias, buscando optimizar el proceso de asignación.

Se propone implementar una metodología GRASP (Greedy Randomized Adaptive Search Procedure) que permita explorar un espacio de soluciones más amplio en comparación con el algoritmo de proximidad propuesto previamente por [56] para este mismo problema.

El algoritmo GRASP fue propuesto inicialmente en [24] y ha sido ampliamente utilizado en diferentes problemas de optimización combinatoria. Este combina una fase constructiva greedy aleatorizada para generar diversas soluciones iniciales de alta

calidad, y una fase de búsqueda local para mejorar iterativamente cada solución inicial hasta alcanzar óptimos locales.

De esta manera, se espera que este nuevo enfoque mejore el desempeño en la asignación de contenedores en comparación con el algoritmo de proximidad; donde los resultados obtenidos se compararán con base en métricas de desempeño como tiempo de ejecución y costo de la solución.

Debido a esto, se propone una nueva regla para determinar a qué bloque iría cada contenedor; se probará en un conjunto de datos de contenedores en instancias aleatorias que se trabajan previamente en la literatura, con el fin de observar si se pueden obtener mejoras respecto al algoritmo de proximidad original.

Una vez planteada la nueva regla, se programará una nueva función que retorne el costo total en unidades de tiempo en el proceso de asignación de contenedores. Se espera que haciendo uso de esta función se obtenga el costo del algoritmo con la nueva regla propuesta.

Por otro lado, contar con este costo, permitirá desarrollar algoritmos de búsqueda local, los cuales son un tipo de algoritmo de optimización que comienza con una solución inicial y luego itera sobre ella, haciendo intercambios o pequeñas mejoras hasta encontrar una solución localmente óptima.

Para verificar la efectividad de la aleatorización y de la búsqueda local, se propone realizar una comparación entre diferentes métodos y combinaciones. En primer lugar, se evaluará el algoritmo greedy de proximidad simple, que servirá como línea base. Posteriormente, se analizará el efecto de añadir búsqueda local al algoritmo greedy de proximidad. Esto permitirá observar si la búsqueda local por sí sola es capaz de mejorar el desempeño con respecto al método voraz puro. Luego, se probará una variante donde se incorpora aleatorización al algoritmo greedy, sin aplicar búsqueda local. De esta forma se podrá aislar el impacto de introducir aleatorización en la construcción de la solución inicial.

Finalmente, se evaluará el algoritmo GRASP completo, que integra tanto la

aleatorización en la fase greedy, como la búsqueda local posterior. Se espera que esta combinación presente los mejores resultados.

Los cuatro métodos se compararán computacionalmente en un conjunto de instancias generadas aleatoriamente por [56], analizando métricas de rendimiento como costo de la solución y tiempo de ejecución. Adicionalmente, mediante un análisis de varianza (ANOVA) se evaluará si el algoritmo propuesto logra mejoras significativas en el desempeño respecto a los existentes. Esto permitirá extraer conclusiones sólidas sobre los aportes individuales y conjuntos de la aleatorización y la búsqueda local en el algoritmo GRASP propuesto.

### 1.3.1. Esquema de Metodología

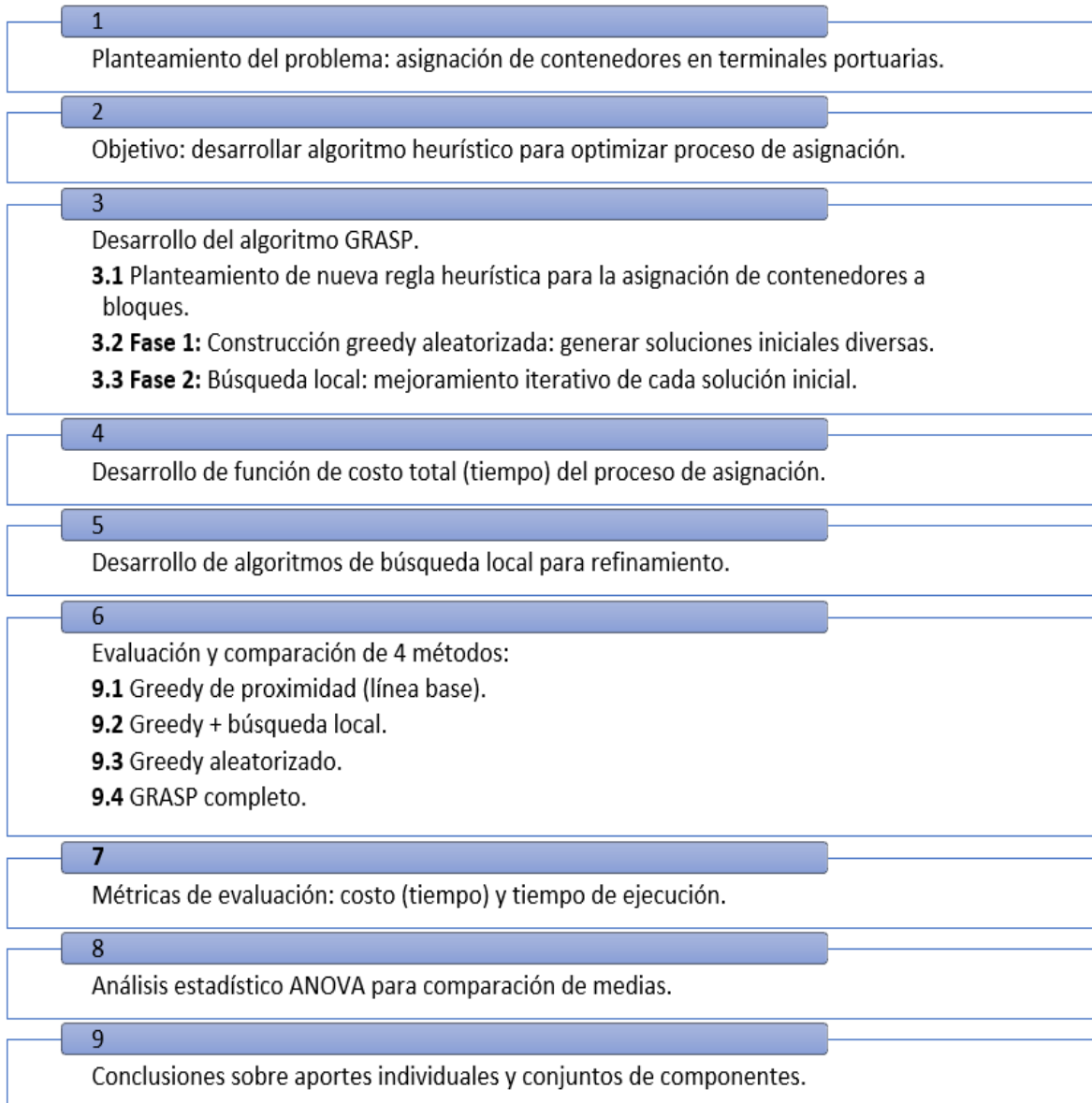


Figura 1.1: Esquema de metodología.

## 1.4. Estructura del documento

Este Trabajo de Grado se organiza de la siguiente manera para facilitar su comprensión y navegación:

- En este primer capítulo, se presenta una visión general de la investigación, incluyendo el resumen de la tesis, los objetivos generales y específicos, la motivación que impulsó este estudio, la metodología utilizada para llevar a cabo la investigación y una descripción de la estructura del documento.
- El segundo capítulo se centra en proporcionar una comprensión profunda del contexto relacionado con las terminales portuarias en Colombia, estudiando ámbitos como: la evolución histórica del transporte marítimo en el país, contextualizando su desarrollo a lo largo del tiempo. Se explora las implicaciones medioambientales del transporte marítimo, destacando las cuestiones críticas relacionadas con la sostenibilidad y el medio ambiente. Además, se analiza la importancia económica del transporte marítimo en la economía de Colombia, destacando su contribución al comercio y desarrollo económico del país. Finalmente, se describe los elementos fundamentales de una terminal portuaria y su funcionamiento general
- El tercer capítulo presenta una revisión exhaustiva de la literatura relacionada con el tema de la investigación. Se examinan y analizan los trabajos previos y las teorías relevantes que respaldan el desarrollo de la tesis.
- En el cuarto capítulo se detalla el problema central abordado en este trabajo: el Problema de Asignación de Contenedores. Se presenta la formulación del problema y se establecen las bases para su resolución. También se describen las implementaciones algorítmicas específicas utilizadas para abordar el Problema de Asignación de Contenedores. Se presentan los algoritmos y las estrategias utilizadas en este estudio.
- En el sexto capítulo se realiza un análisis exhaustivo de los algoritmos

implementados, se presentan los resultados obtenidos y se discuten las mejoras realizadas para abordar el problema de manera más efectiva.

- Finalmente, en el último capítulo, se resumen las conclusiones clave de la investigación y se proponen posibles direcciones para futuros estudios relacionados con el tema.

## 2. Terminales Portuarias

---

El transporte marítimo es el medio de transporte con fines comerciales más antiguo y destacable del mundo [18]. Y ha jugado un papel fundamental en el desarrollo de la civilización y en el crecimiento económico mundial.

En tal sentido, los primeros barcos se utilizaron para la pesca y transporte de personas y mercancías. Así, con el desarrollo de la navegación, los barcos se hicieron más grandes, contando con mayores capacidades para navegar en alta mar; lo que posteriormente permitió el comercio entre diferentes países y condujo a un mayor intercambio de ideas y tecnologías [35].

Actualmente se ha podido evidenciar el crecimiento económico a nivel internacional y el desarrollo de industrias emergentes debido a la evolución de este estilo de comercio. Sin embargo, cabe hacer énfasis en que los buques tienen una gran responsabilidad respecto a las emisiones globales de gases de efecto invernadero; por lo que es indispensable seguir trabajando en la innovación y optimización de la logística de la navegación marítima [47].

### 2.1. Contexto histórico, económico y ambiental del transporte marítimo en Colombia

La historia y evolución del transporte marítimo en Colombia han jugado un papel clave para el desarrollo del país. Anteriormente, las rutas fluviales y marítimas del territorio formaban un sistema estratégico para el comercio y transporte desde antes de la colonización por parte de los europeos [38]. Siguiendo en este razonamiento, las comunidades precolombinas ya aprovechaban esta ventaja natural para el comercio y contacto con otros pueblos. De ahí que, el país siempre ha tenido la oportunidad de capitalizar sus dos costas y cuenca hidrográfica para establecer conexiones comerciales

con el resto del mundo.

Posteriormente, con el advenimiento de la época colonial el transporte marítimo en Colombia sufrió una serie de eventos evolutivos hasta su conformación y funcionamiento actual. En un inicio, el río Magdalena fue la ruta de comunicación con mayor importancia puesto que era una guía para llegar al Perú y, además, servía para el tráfico de alimentos para las tropas [59]. Más adelante, durante el desarrollo del siglo XIX y con la invención de los barcos a vapor durante la revolución industrial, Colombia comienza su incursión de nuevas tecnologías y se da el comienzo de la navegación con barcos de vapor gracias al comerciante alemán Juan Bernardo Elbers; quien logró jurisdicción para la importación de los primeros tres barcos de este tipo desde los Estados Unidos de América: Fidelidad, Gran Bolívar y el Gran Santander. No obstante, estas primeras embarcaciones resultaron poco eficientes frente a las características geográficas del río, así como a las constantes variaciones en su profundidad [8].

Por otro lado, los españoles establecieron diferentes puertos a lo largo del Caribe para facilitar el comercio con Europa; en primer lugar, se fundó el puerto de Santa Marta en 1525; siendo un lugar con aspecto de villa pequeña, donde se destacaban el contrabando, sus ideas conservadoras tradicionalistas y las desembarcaciones de piratas franceses, ingleses y holandeses.

Más adelante, el puerto de Cartagena de Indias, fundado en 1533, fue el principal puerto del en ese entonces llamado Nuevo Reino de Granada, convirtiéndose en un importante centro comercial y militar. Este, cumplió un papel fundamental en cuanto a la comercialización negrera, el “saqueo” y venta de mercancías, recursos minerales y naturales de la Nueva Granada hacia la corona española [7].

Para el año de 1719, la Corona Española eligió a Buenaventura como uno de los principales puertos, allí fue donde se realizaron las primeras embarcaciones del General Sucre y del Libertador Simón Bolívar, para la expansión del movimiento libertador en Ecuador y Perú [9].

Durante aquel periodo, los puertos colombianos no solo servían como enclaves

estratégicos para el envío de productos hacia Europa, sino que también eran testigos de la convergencia de culturas y mercancías.

Siglos después, el mundo fue testigo de un hito trascendental en la historia del mundo: la creación del canal de Panamá. Construido entre 1904 y 1914, el canal se volvería pieza clave para la evolución del transporte marítimo internacional; puesto que recortó drásticamente la distancia a recorrer entre el océano Pacífico y Atlántico. Dicho suceso destacaría el desarrollo del comercio colombiano en la tercera década del siglo XX, impulsando el avance de terminales portuarias clave. En tal sentido, en 1936 se establece el puerto marítimo y fluvial de Barranquilla, ubicado a 22 km de la desembocadura del río Magdalena en el mar Caribe; convirtiéndose en uno de los lugares más importantes para impulsar el comercio nacional y extranjero.

Ahora bien, en la actualidad Colombia cuenta con diez terminales portuarias a lo largo de sus dos costas: ocho en el Caribe y dos en el Pacífico. De acuerdo a La Comisión Económica para América Latina (CEPAL), en el 2019 los puertos del país se ubicaron en el cuarto lugar en cuanto volumen de mercancías a nivel de Latinoamérica [49], lo cual resalta el papel fundamental de el transporte portuario Colombiano en la región. De igual modo, el instituto mencionado asegura que ha habido un crecimiento del 36 % en cuanto a la capacidad portuaria. Además, en investigaciones recientes realizadas por la ANDI, se ha señalado que entre 2010 y 2020, las inversiones en la zona pública de los puertos ascendieron a más de 2.600 millones de dólares [52], notando como sigue en crecimiento dicho fenómeno.

Con respecto a los impactos ambientales del transporte marítimo en Colombia, se resaltan aspectos negativos como la contaminación por derrames de petróleo e hidrocarburos, la emisión de gases de efecto invernadero que contribuyen al calentamiento global, y la contaminación acústica submarina que afecta la vida marina. Grandes desastres como los derrames de petróleo en El Lizama y Tumaco han tenido consecuencias devastadoras para los ecosistemas y la biodiversidad, con impactos que pueden durar años [68]. Las emisiones de dióxido de carbono y otros gases de buques que usan combustibles fósiles también agravan la crisis climática [50]. El ruido

submarino de motores y hélices interfiere en la comunicación y comportamiento de animales marinos [20], potencialmente alterando sus hábitats. Mitigar estos impactos requiere transición energética en el transporte marítimo, mayor educación ambiental y colaboración público-privada para innovaciones como la optimización de tiempos de espera de buques que reduzcan sus emisiones contaminantes [32]. Un transporte marítimo sostenible en Colombia es crucial para proteger sus frágiles ecosistemas costeros y marinos [17].

En cuanto al impacto económico, el transporte marítimo ha sido un motor clave del comercio internacional y la competitividad de Colombia, representando más del 90 % del intercambio de carga exterior del país [15]. La modernización portuaria desde los años 90 mediante concesiones a operadores privados logró atraer cuantiosas inversiones que permitieron una gran expansión de capacidad e infraestructura [27]. Esto produjo un aumento del 100 % en el comercio de los principales puertos en esa década y una mejora sustancial en tiempos de atraque [66]. Entre 2010-2016 se invirtieron US\$2.370 millones en puertos colombianos [54], incrementando la carga movilizada de 188 a 200 millones de toneladas [25]. Los puertos de Buenaventura y Cartagena lideran el sector, moviendo en conjunto sobre el 50 % de la carga exterior [45]. Retos a futuro incluyen seguir atrayendo inversión privada para financiar ambiciosos planes de expansión y modernización con tecnologías como automatización e inteligencia artificial. Contar con puertos eficientes y competitivos a nivel global es clave para el desarrollo económico de Colombia.

Finalmente, como se puede notar a lo largo de la historia del transporte marítimo y fluvial de Colombia, estos nodos portuarios no solo son pilares para las exportaciones e importaciones del país, sino que también han experimentado cuantiosas inversiones en infraestructuras para conservar su competitividad a nivel global, sirviendo como impulsores esenciales de la economía colombiana.

## 2.2. Funcionamiento de una terminal portuaria

Las terminales portuarias constituyen una pieza clave e indispensable en las cadenas logísticas globales del transporte marítimo de contenedores [55]. Son instalaciones especializadas que permiten la transferencia de mercancías entre buques de gran calado y otros modos de transporte terrestre [51]. Todo contenedor que se moviliza por vía marítima pasa por al menos una terminal durante su trayecto. La operación de estos nodos portuarios involucra una compleja coreografía de grúas, vehículos de patio, camiones y trenes. Optimizar los flujos y procesos en una terminal es fundamental, pero muy desafiante [62]. Requiere una comprensión profunda de los distintos subsistemas y equipos que la componen. Las tecnologías de automatización y los sistemas de información brindan oportunidades para mejorar la eficiencia de las terminales acorde continúan creciendo los volúmenes de carga [62].

### 2.2.1. Estructura y elementos de un puerto

La estructura de una terminal portuaria está diseñada para permitir un flujo eficiente de contenedores entre el transporte marítimo y terrestre. Se pueden identificar tres zonas principales interconectadas:

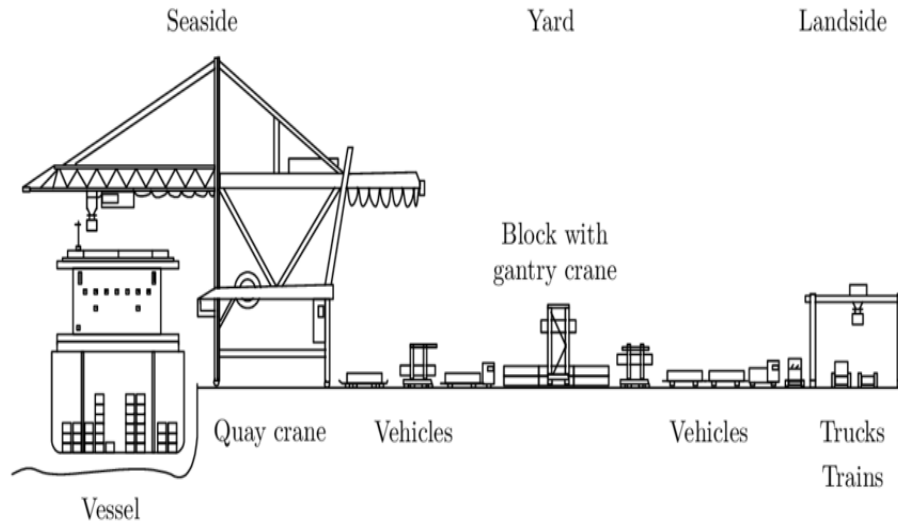
La zona de mar, también llamada lado marítimo o waterside, comprende toda la infraestructura y actividades relacionadas con la interfaz buque-puerto. Lo más destacado es la línea de atraque, que consta de varios sitios donde los buques portacontenedores pueden amarrar para proceder a la descarga y carga de contenedores [63]. Cada sitio de atraque está equipado con grúas de muelle o quayside cranes, que son las encargadas de izar los contenedores desde la cubierta del barco y depositarlos en el patio o en vehículos de transporte [1]. En la Figura 2.1 puede verse como este tipo de grúas están situadas a lo largo del muelle, donde además son movidas por medio de rieles. Otros equipos típicos en la zona de mar son las grúas móviles sobre neumáticos, usadas para apilar contenedores en la zona de respaldo del muelle mientras esperan ser

procesados. Allí también puede haber una zona de inspección aduanera de los contenedores que llegan [67].

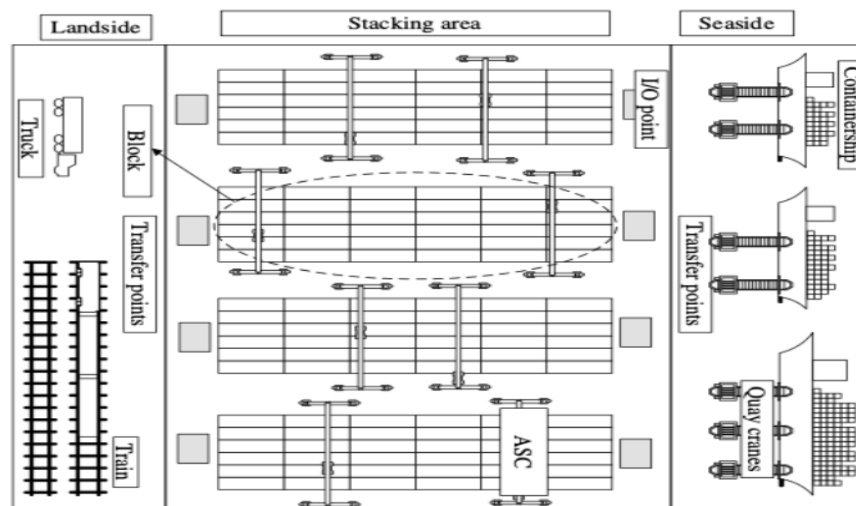


**Figura 2.1:** Quayside Crane. [16]

La zona terrestre o landside corresponde a la interfaz entre la terminal portuaria y los medios de transporte por carretera o ferroviarios. Consta de accesos viales para los camiones que van a recoger o dejar contenedores, zonas de espera, áreas para inspeccionar y sellar contenedores antes de salir, y portones de entrada/salida para controlar el flujo [67]. Si la terminal también ofrece servicio ferroviario, debe contar con vías férreas y una zona intermodal para transferencia de contenedores entre trenes y camiones. Idealmente, las vías férreas ingresan directamente al patio de contenedores para minimizar recorridos. En esta zona suelen ubicarse también el edificio administrativo, la aduana de exportaciones e importaciones, zonas de servicios para conductores y estacionamientos para camiones [63]. En las Figuras 2.2 y 2.3 se muestra la distribución general de los puertos:



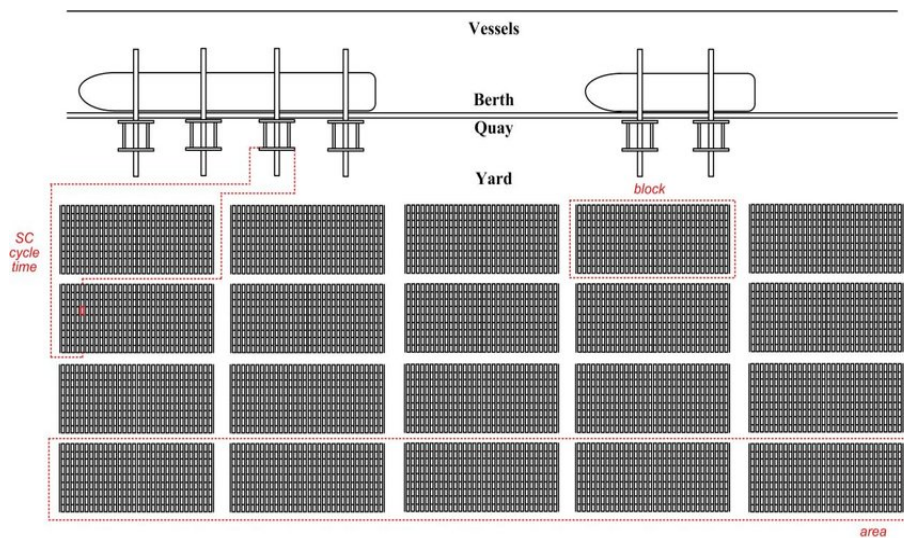
**Figura 2.2:** Estructura de una terminal portuaria. [14]



**Figura 2.3:** Estructura de una terminal portuaria. [56]

La zona de almacenamiento o storage yard representa el corazón de una terminal portuaria. Como se ve en la Figura 2.4, aquí es donde los contenedores son temporalmente apilados y almacenados mientras esperan ser recogidos o embarcados [63]. El componente principal es el patio de contenedores, que a su vez se divide en varias secciones; de acuerdo a [48], típicamente, estos bloques se componen de 40 compartimentos, dispuestos en 7 filas y con una variación en la altura que depende de la capacidad de elevación de la grúa que realice la carga. Asimismo, en la Figura 2.5 puede

verse las características de las grúas pórtico o rubber tyred gantry cranes (RTGCs), que son las encargadas de formar las pilas dentro de los bloques de almacenamiento [26]. El patio también debe tener espacios para el tránsito y maniobras de los vehículos que transportan contenedores como camiones, trailers y straddle carriers. Las vías interiores y el layout de los bloques determinan los flujos de circulación dentro del patio [67]. Además, en esta zona se ubican las subestaciones eléctricas que proveen energía a las grúas y los talleres de reparación y mantenimiento de equipos. También pueden construirse vías férreas internas y zonas de inspección de contenedores.



**Figura 2.4:** Zona de almacenamiento y bloques. [33]



**Figura 2.5:** Rubber tyred gantry cranes (RTGCs). [64]

En conclusión, el patio de contenedores es un elemento central y crítico que condiciona la formulación y resolución del problema de asignación óptima de contenedores entrantes en una terminal portuaria. Son precisamente los bloques del patio los que representan las ubicaciones disponibles a las cuales el algoritmo debe designar cada contenedor entrante; también debe considerar constantemente la capacidad remanente de cada bloque antes de asignar un nuevo contenedor entrante. De esta manera, las limitaciones de recursos dentro del patio de contenedores, tanto físicas como operacionales, resultan determinantes para representar adecuadamente el problema y poder diseñar un algoritmo de asignación óptimo. El patio es, por lo tanto, el corazón de la terminal y un elemento central en la formulación del problema.

### **2.2.2. Configuración Europea y Asiática**

Existen dos configuraciones predominantes en el diseño de terminales portuarias de contenedores: la europea y la asiática. La diferencia principal radica en la orientación de los bloques de almacenamiento en relación al atraque de los buques.

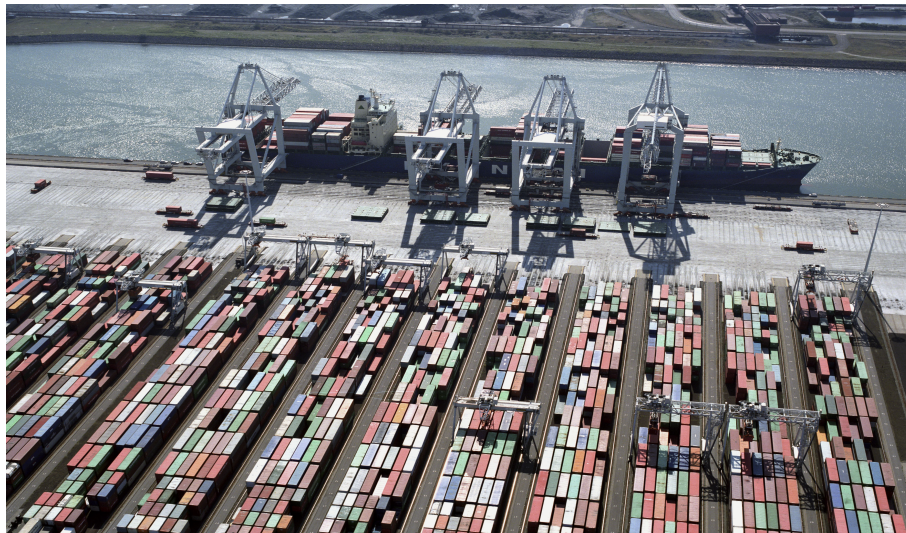
La configuración europea se caracteriza por tener los bloques de almacenamiento perpendiculares a la línea de atraque. Los puntos de entrada y salida de contenedores se

ubican al final de cada bloque, tanto en el lado mar como en el lado tierra [11].

Esta disposición presenta varias ventajas. Por un lado, al tener los bloques perpendiculares al muelle se aprovecha mejor el espacio, permitiendo un patio de contenedores más largo [31]. Además, los camiones y grúas tienen acceso directo a los contenedores sin tener que rodear los bloques, lo que mejora los tiempos de carga y descarga [30].

Otra ventaja es que al concentrar los puntos de entrada/salida al final de los bloques se reducen los conflictos entre grúas y vehículos de transporte terrestre dentro del patio [67]. También permite una mayor flexibilidad en la asignación de zonas a importación/exportación dependiendo de la demanda [40].

La configuración europea es cada vez más popular y ha sido adoptada en terminales de puertos importantes como Rotterdam, Amberes, Algeciras y Valencia; como se demuestra en la Figura 2.6:



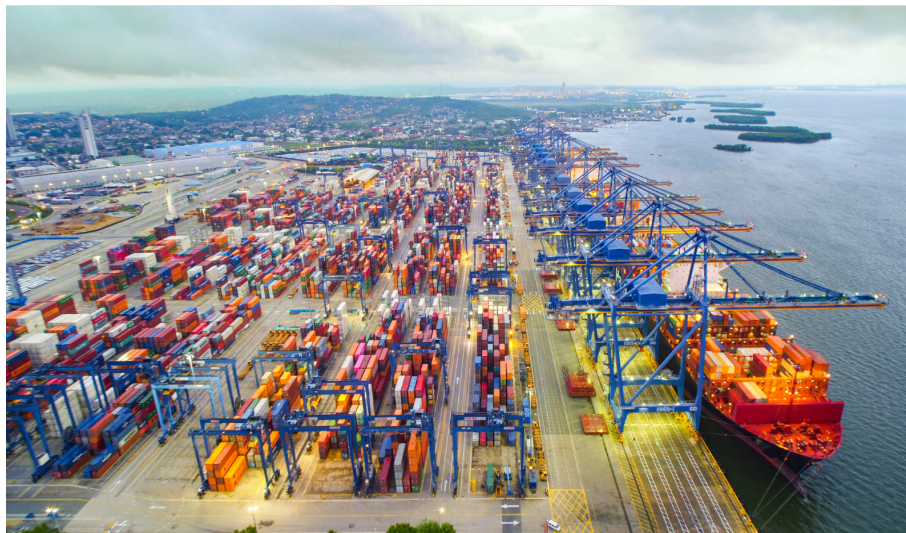
**Figura 2.6:** Configuración europea en el Puerto de Rotterdam. [37]

Por otro lado, la configuración asiática se distingue porque los bloques de contenedores son paralelos al muelle. Además, cada bloque tiene una fila dedicada al tránsito de camiones y otros vehículos de transporte terrestre [11].

La ventaja principal es que al tener una zona exclusiva para el tráfico vehicular se evitan interferencias entre las grúas pórtico y los camiones que entran y salen del patio. Esto mejora la productividad de las grúas al no tener que detenerse para dejar pasar vehículos [40].

Sin embargo, la configuración asiática tiene algunas desventajas. Al poner los bloques paralelos al muelle se desperdicia espacio, limitando el largo usable del patio. Además, los camiones deben rodear los bloques para acceder a los puntos de carga/descarga, incrementando distancias recorridas [31].

Finalmente, concentrar el tráfico en una sola vía perimetral puede generar congestión y cuellos de botella, especialmente en terminales grandes [67]. A pesar de esto, la configuración asiática sigue siendo popular en Asia, especialmente en puertos con alta densidad de tráfico como Shanghai, Singapur y Hong Kong. De acuerdo con la Figura 2.7, puede verse que en Colombia está presente en terminales importantes como la de Cartagena.

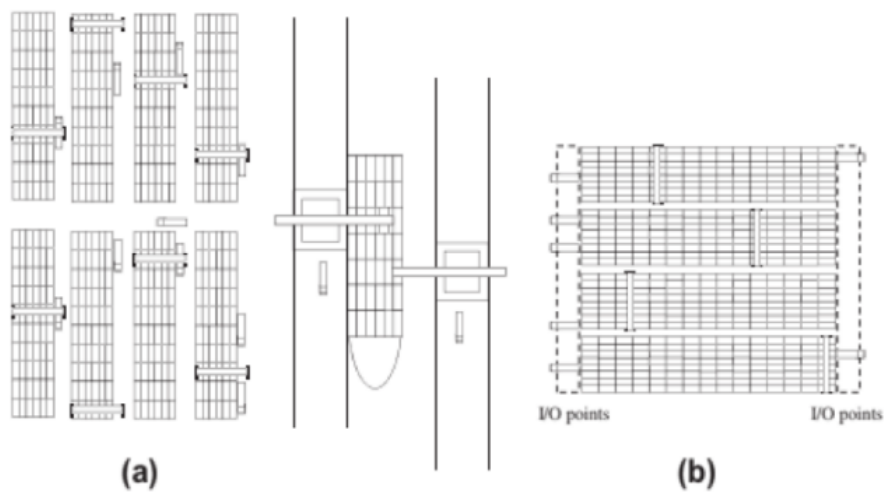


**Figura 2.7:** Configuración asiática en el Puerto de Cartagena. [19]

En resumen, la configuración europea es más eficiente en el uso del espacio al tener los bloques perpendiculares al muelle. Aprovecha mejor el largo del patio y reduce recorridos

de vehículos. Pero la asiática minimiza interferencias entre grúas y camiones al segregar el tráfico. En la Figura 2.8, se compara y evidencia la diferencia entre ambas configuraciones.

No hay una mejor universalmente. La elección depende de factores como espacio disponible, volumen de carga, mezcla de tipos de contenedores, etc. Y aunque en Colombia es común encontrar la configuración asiática en terminales portuarias, este trabajo se enfocará en el diseño europeo debido a que es la tendencia global actual por permitir mayor flexibilidad y productividad en patios de gran tamaño y alto volumen de carga [31]. La configuración europea con varios bloques paralelos interconectados facilita los flujos de contenedores entre el lado marítimo y terrestre. Además, esta investigación busca aportar a estudios previos que también han utilizado el modelo europeo para el problema de asignación de contenedores como el trabajo de [56]. Finalmente, el algoritmo y análisis propuesto podría ser fácilmente extrapolado a una configuración de patio asiático, dado que el problema de optimización es análogo en ambos casos. Por estas razones, se trabajará bajo el supuesto de una terminal con distribución europea de bloques a pesar de no ser el más común en Colombia.



**Figura 2.8:** Configuración asiática (a) y configuración europea (b). [11]

### 2.2.3. Problemas de optimización y logística en una terminal portuaria

Las terminales portuarias enfrentan múltiples desafíos de optimización y logística debido a la complejidad de sus operaciones. Un puerto eficiente requiere coordinar de manera óptima una amplia gama de recursos, incluyendo patio de contenedores, grúas, vehículos de patio, transporte terrestre y por supuesto, la gestión de los propios contenedores.

Uno de los principales retos es la asignación de grúas de muelle a los buques que arriban para minimizar tiempos de espera y completar las operaciones de carga/descarga en el menor tiempo posible. Esto implica designar grúas disponibles a buques entrantes considerando restricciones de recursos y concatenación de tareas [5].

Otro desafío clave es calendarizar de manera óptima las tareas de las grúas de patio que apilan y retiran contenedores en los bloques de almacenamiento. El objetivo es maximizar la productividad de las grúas reduciendo movimientos improductivos y tiempos de espera.

También es crucial planificar la asignación y rutas de los vehículos de patio (camiones, straddle carriers, AGVs) que transportan contenedores entre las zonas de almacenamiento y los puntos de transferencia con modos terrestres. La meta es minimizar distancias recorridas y evitar congestión vehicular dentro del patio [58].

Asimismo, es importante optimizar la ubicación de contenedores en los bloques de almacenamiento considerando factores como tamaño, peso, puerto de destino y fecha de zarpe del buque. Una adecuada asignación de posiciones reduce movimientos posteriores dentro del patio [22].

La gestión de llegadas y despachos de camiones en la zona terrestre también requiere optimización para evitar congestión en los accesos [13]. Igualmente crucial es coordinar las operaciones terrestres y marítimas, sincronizando el flujo vehicular con la disponibilidad de grúas de muelle [53].

Como se puede apreciar, en una terminal portuaria existen múltiples procesos por

optimizar. En este trabajo de grado se trabajará en uno de los problemas fundamentales: la asignación óptima de contenedores a bloques de almacenamiento en el patio; puesto que el objetivo es lograr mejores resultados a los obtenidos en la investigación de [56]

### 3. Revisión Bibliográfica

---

El problema de asignar de forma óptima contenedores entrantes a diferentes bloques dentro de un patio portuario es un reto importante en logística y gestión portuaria. Se trata de un problema NP-complejo de empaquetamiento multidimensional, como afirma [60]. La complejidad radica en que existen múltiples contenedores de distintos tamaños que deben encajarse de forma compacta en bloques de tamaño fijo. La meta es minimizar la distancia total de transporte al asignar cada contenedor entrante al bloque más cercano posible con capacidad disponible; lograr la asignación óptima permite reducir tiempos y costos de movimiento dentro del patio portuario.

Una de las primeras heurísticas propuestas para atacar este problema fue el algoritmo de clasificación en bloques (block stacking algorithm) presentado en [28]. Este método greedy funciona asignando secuencialmente cada contenedor entrante al bloque con la menor altura actual y que tenga espacio disponible. De esta manera, se maximiza el aprovechamiento del espacio en vertical dentro de cada bloque. Los autores reportaron que este enfoque simple permite encontrar buenas soluciones en tiempos de cómputo cortos, aunque no garantiza la optimalidad.

Otros investigadores han propuesto mejoras y alternativas al algoritmo greedy inicial expuesto en [28]. Por ejemplo, en [43] y [4] se diseña una búsqueda Tabú para explorar de forma más amplia el espacio de posibles asignaciones. En cada iteración se realizan movimientos hacia vecinos próximos, intentando evitar quedarse atorado en óptimos locales. Los resultados obtenidos mejoraron la distancia total de transporte en varios casos de prueba.

Asimismo, en [44] se combina el algoritmo greedy con programación matemática. La idea es usar el greedy para una asignación inicial rápida, y luego aplicar programación entera para refinar y encontrar la distribución óptima de contenedores. Esta hibridación logró tiempos de cómputo aceptables al tiempo que se incrementó la calidad de las

soluciones.

El trabajo de [12] presenta una nueva y compacta formulación matemática binaria para el problema de reubicación de bloques en terminales portuarios, incorporando restricciones realistas como capacidad de carga. Desarrollan un algoritmo de ramificación y corte que permite resolver casos de tamaño mediano de forma óptima en tiempos razonables. Los autores discuten como analizar las soluciones óptimas puede guiar el diseño de heurísticas rápidas, por ejemplo derivando reglas de prioridad para algoritmos greedy constructivos. Así, la programación matemática aporta entendimiento valioso sobre la estructura óptima del problema que luego se explota para mejorar métodos aproximados

Por otro lado, también se ha abarcado el problema por medio del algoritmo GRASP, que es una metaheurística que combina un proceso greedy randomized con búsqueda local reactiva. En [3] se evalúa su aproximación en instancias de benchmarking para el problema de carga de contenedores tridimensional con restricciones realistas de soporte de carga. Los resultados reportados muestran que su método GRASP reactivo logra mejores soluciones que otras heurísticas existentes para el problema de empaquetamiento considerado. Resaltan que al integrar reactividad durante la construcción de la solución, se promueve la factibilidad y se mejora el rendimiento.

Por la misma época, [34] propusieron una taxonomía que clasifica las contribuciones previas para el problema de carga, descarga y pre-marshalling de contenedores apilados en áreas de almacenamiento; y analizan diferencias en supuestos y enfoques de modelado. Posteriormente, proponen una metaheurística basada en algoritmo de colonia de hormigas y búsqueda de entorno variable para atacar el problema. Mediante experimentos computacionales demuestran que su propuesta encuentra mejores soluciones que simples heurísticas constructivas en un conjunto de instancias benchmark. Logran un método competitivo en términos de tiempo de cómputo y calidad de solución para el complejo problema estudiado.

Más recientemente, con el auge del aprendizaje automático, en [61] describen un sistema de recomendación para asignación de slots de contenedores usando modelos de

aprendizaje automático supervisado como regresión y árboles de decisión. Reportan que su enfoque permite recomendaciones precisas y rápidas que mejoran la utilización del espacio disponible en el patio de contenedores. Asimismo, [29] presentan un enfoque de optimización de apilamiento de contenedores basado en aprendizaje profundo por refuerzo (Deep Reinforcement Learning). Entrenan un agente inteligente mediante deep RL que aprende una política efectiva a través de interacción en un ambiente simulado de patio de contenedores. Comparado con métodos analíticos y heurísticos, su propuesta basada en deep RL logra mejores soluciones en menor tiempo de computación en casos de estudio realistas.

Dentro de los estudios más actuales, el trabajo de [69] propone un novedoso algoritmo de búsqueda en vecindarios grandes (Large Neighbourhood Search) que determina empaquetamientos efectivos de un conjunto de ítems en contenedores, con el objetivo de maximizar el valor total cargado; maneja restricciones realistas como orientación, apilamiento, separación, límites de peso, prioridades y estabilidad; implementan el algoritmo en una planilla Excel que almacena datos, resuelve de forma eficiente y presenta gráficamente las soluciones; experimentos en instancias benchmark demuestran la alta calidad de las soluciones logradas, superando a otros métodos como la búsqueda tabú; los autores concluyen que su enfoque LNS puede resolver variaciones del problema de carga de contenedores con distintos niveles de heterogeneidad en ítems y contenedores, así como manejar diversas restricciones prácticas, resultado en una herramienta útil para aplicaciones logísticas reales.

El presente trabajo busca desarrollar un algoritmo heurístico híbrido que combine la velocidad de los métodos greedy con la exploración amplia del espacio de búsqueda de metaheurísticas como GRASP. En ese sentido, la propuesta tiene similitudes con el enfoque presentado en [56], que también plantea una hibridación de greedy y programación matemática para mejorar tiempos de ejecución al tiempo que se incrementa la calidad de las soluciones.

Sin embargo, esta investigación se diferencia en su uso de GRASP en lugar de programación entera para el refinamiento posterior a la fase greedy inicial. Asimismo, el

diseño reactivo de la construcción greedy permite promover factibilidad durante la generación de soluciones, tomando ideas de [56]. Por otro lado, a diferencia de la mayoría de trabajos revisados que se enfocan en maximizar aprovechamiento de espacio, este estudio considera directamente la minimización de la distancia total de transporte en la asignación de contenedores a bloques.

De esta manera, esta propuesta complementa y extiende ideas presentadas previamente en la literatura, resultando potencialmente en un algoritmo competitivo tanto en velocidad como calidad de las soluciones para el problema específico abordado.

En conclusión, las heurísticas clásicas como la de Kim y Park siguen siendo valiosas por su velocidad y facilidad de implementación. Pero técnicas más nuevas como la optimización matemática y el aprendizaje por refuerzo han demostrado poder mejorar la calidad de las soluciones al problema de asignación de contenedores en patios portuarios. Queda un amplio campo de investigación en como hibridar lo mejor de ambos paradigmas.

# 4. Problema de Asignación de Contenedores

---

En este capítulo se abordará en detalle el problema de asignación de contenedores en el patio de una terminal portuaria, para lo cual, primeramente se referenciará un modelo matemático que permita encontrar soluciones óptimas y sirva como base de comparación. No obstante, antes de adentrarnos en la descripción del problema, es necesario explicar la definición y los diferentes tipos de contenedores con los que se trabajará.

## 4.1. Tipos de contenedores

Los contenedores son cajas metálicas de tamaño estandarizado que se utilizan para el transporte de carga. Permiten movilizar mercancías de forma segura, protegiéndolas y facilitando su manipulación. En consecuencia, tienen una importancia primordial en el comercio marítimo global [35].

Existen tres tipos principales de contenedores en una terminal portuaria:

- Contenedores de importación (import): Son contenedores cargados en un buque portacontenedores en el puerto de origen, que llegan por vía marítima a la terminal portuaria. Una vez descargados del barco mediante grúas de muelle, deben ser temporalmente almacenados en el patio a la espera de ser recogidos por camiones o trenes para su distribución terrestre hacia destinos finales.
- Contenedores de exportación (export): Son contenedores que arriban a la terminal portuaria por transporte terrestre, ya sea camiones o trenes. Vienen de distintos orígenes transportando mercancías de exportación. Son apilados en el patio de contenedores mientras aguardan para ser embarcados en los buques

portacontenedores con destino al extranjero.

- Contenedores de transbordo (Transshipment): También llegan por vía marítima en barcos portacontenedores, pero a diferencia de los contenedores de importación, estos no abandonan la terminal por transporte terrestre. Sólo son almacenados temporalmente en el patio para luego ser estibados en otro buque distinto con destino a otro puerto. Son contenedores de paso entre dos rutas marítimas diferentes.

Cada tipo de contenedor presenta características y restricciones particulares que deben considerarse al asignarlos a los bloques de almacenamiento en el patio de la terminal.

## 4.2. Descripción Formal del Problema

El problema que se desea resolver consiste en la asignación óptima de contenedores entrantes en los distintos bloques que conforman el patio de contenedores de una terminal portuaria. De esta forma, se busca complementar el modelo expuesto por [56] donde esta terminal cuenta con una configuración típica que incluye una zona marítima para la llegada y salida de buques, una zona terrestre para la entrada y salida de camiones, y el patio de contenedores propiamente dicho.

El patio está dividido en múltiples bloques paralelos de forma rectangular, separados por corredores de circulación interna. Existen  $B$  bloques disponibles indexados como  $j = 1, 2, \dots, B$ , y cada uno posee dos puntos extremos de entrada/salida (I/O points), referidos como lado marítimo y lado terrestre respectivamente. Estos servirán para ingresar o retirar contenedores del bloque según su tipo y próximo destino.

El proceso de asignación involucra un conjunto total de  $N$  contenedores entrantes, indexados como  $i = 1, 2, \dots, N$ . Cada contenedor  $i$  tiene asociado un tiempo de llegada  $r_i$  que indica cuando estará disponible para ser asignado, con los  $r_i$  ordenados de forma

creciente. Además, cada  $i$  tiene un tiempo futuro de salida  $d_i$  que determinará cuándo dejará de ocupar capacidad en su bloque asignado.

Otros dos parámetros importantes asociados a cada contenedor  $i$  son su tipo de entrada  $C_i$ , que indica si ingresa por vía marítima ( $C_i = 1$ ) o terrestre ( $C_i = 2$ ); y su próximo destino  $D_i$ , que señala si después dejará la terminal por mar ( $D_i = 1$ ) o tierra ( $D_i = 2$ ). Con base en  $C_i$  y  $D_i$  se diferencia si un contenedor es de importación, exportación o transbordo. Conocer el tipo de contenedor es crucial porque determina en qué I/O point debe asignarse dentro del bloque  $j$ . Si  $C_i = 1$ , se asigna en el lado marítimo teniendo en cuenta la variable  $LM_{jt}$ , que indica la capacidad libre del bloque  $j$  en el lado de mar en un instante  $t$ . Si  $C_i = 2$ , se asigna en el lado terrestre teniendo en cuenta la variable  $LT_{jt}$ , que indica la capacidad libre del bloque  $j$  en el lado de tierra en un instante  $t$ . Esta restricción debe cumplirse estrictamente en cualquier solución factible.

Se conoce también el tiempo  $T_{ij}$  que tomará mover y asignar cada  $i$  desde su punto de entrada a cada bloque  $j$  específico. Estos tiempos guiarán la decisión de asignación buscando minimizar el tiempo total.

El algoritmo propuesto debe entonces asignar secuencialmente cada nuevo contenedor  $i$  al bloque  $j$  más adecuado, en cuanto  $r_i$  indique disponibilidad. Debe elegir un  $j$  que tenga capacidad disponible en el lado correcto, minimizando  $T_{ij}$ , y actualizando las capacidades dinámicas tras cada asignación, dado que los contenedores tienen distintos instantes de salida.

El proceso de asignación debe realizarse de forma secuencial, a medida que los contenedores van arribando a la terminal en sus respectivos horarios  $r_i$ . No todos llegan al mismo tiempo, sino que van ingresando en distintos momentos, con tiempos  $r_i$  que han sido previamente coordinados y registrados.

Una vez que se asigna un contenedor  $i$  a un bloque  $j$ , esa asignación es definitiva. Los contenedores no pueden ser posteriormente reubicados o movidos a otro sector. Por lo tanto, el modelo debe realizar cada asignación de la forma más eficiente posible.

El objetivo final es realizar la asignación completa de los  $N$  contenedores entrantes en el menor tiempo total posible, minimizando el último tiempo  $t$  registrado al finalizar la última asignación. Esto permitirá mejorar el flujo de contenedores en la terminal portuaria y la eficiencia de sus operaciones.

### 4.3. Modelación Matemática del Problema

Para representar adecuadamente el problema de asignación de contenedores y poder resolverlo mediante técnicas de optimización matemática, es necesario formular un modelo matemático que capture las principales características y restricciones involucradas.

En esta sección se complementa el modelo simplificado desarrollado por [56] para el problema de asignación contenedores a bloques dentro de una terminal portuaria. En este trabajo, la principal diferencia es la incorporación del subíndice  $t$  para indexar las variables por tiempo.

De forma similar al modelo original, se considera únicamente la llegada de contenedores nuevos al puerto, sin tomar en cuenta la salida o retiro de contenedores. Es decir, el enfoque está puesto en asignar cada contenedor entrante al puerto a un bloque, minimizando el tiempo total de asignación.

El añadir el subíndice  $t$  permite modelar explícitamente el instante de tiempo en que cada contenedor está disponible para ser asignado, representado por el parámetro  $r_i$ . Esto posibilita incluir las restricciones 10 a 12, que impiden asignar un contenedor antes de que éste haya llegado al puerto.

De esta manera, el modelo extendido captura de mejor forma la dinámica del problema real, al considerar la disponibilidad de los contenedores en el tiempo. Las demás características y supuestos se mantienen igual que en el modelo original, pero la indexación por tiempo y las nuevas restricciones permiten representar con mayor precisión el proceso de asignación de contenedores a medida que van llegando al puerto.

### 4.3.1. Modelo

Sea  $N = \{1, 2, \dots, N\}$  el conjunto de contenedores a ser asignados, donde los subíndices de 1 a  $N_1$  representan contenedores de importación, de  $N_1 + 1$  a  $N_2$  los de exportación, y de  $N_2 + 1$  a  $N$  los de transbordo.

Sea  $B = \{1, 2, \dots, B\}$  el conjunto de bloques disponibles en la terminal.

Sea  $T = \{1, 2, \dots, T\}$  el horizonte de tiempo del problema.

Se define el parámetro  $T_{ij}$  como el tiempo requerido para asignar el contenedor  $i$  al bloque  $j$ .

Sea  $r_i$  el instante de tiempo en que el contenedor  $i$  está disponible para ser asignado.

Las restricciones 2 a 4 garantizan que cada contenedor se asigne a exactamente un bloque.

Las restricciones 5 y 6 restringen que se supere la capacidad de cada bloque en los lados tierra y mar respectivamente.

Las restricciones 7, 8 y 9 corresponden a variables de decisión binarias indexadas por el tiempo:

- $Y_{ijt} = 1$  si el contenedor  $i$  del tipo import es asignado al bloque  $j$  en el instante  $t$ , 0 en caso contrario.
- $X_{ijt} = 1$  si el contenedor  $i$  del tipo export es asignado al bloque  $j$  en el instante  $t$ , 0 en caso contrario.
- $S_{ijt} = 1$  si el contenedor  $i$  del tipo transbordo es asignado al bloque  $j$  en el instante  $t$ , 0 en caso contrario.

Las restricciones 10 a 12 modelan que los contenedores no están disponibles en el instante inicial  $t = 0$ .

La función objetivo busca minimizar el tiempo total de asignación:

$$\text{Min} \sum_{i=1}^{N_1} \sum_{j=1}^B \sum_{t=1}^T T_{ij} Y_{ijt} + \sum_{i=N_1+1}^{N_2} \sum_{j=1}^B \sum_{t=1}^T T_{ij} X_{ijt} + \sum_{i=N_2+1}^N \sum_{j=1}^B \sum_{t=1}^T T_{ij} S_{ijt} \quad (4.1)$$

**sujeto a:**

$$\sum_{j=1}^B \sum_{t=1}^T Y_{ijt} = 1, \quad \forall i \in [1, \dots, N_1] \quad (2)$$

$$\sum_{j=1}^B \sum_{t=1}^T X_{ijt} = 1, \quad \forall i \in [N_1 + 1, \dots, N_2] \quad (3)$$

$$\sum_{j=1}^B \sum_{t=1}^T S_{ijt} = 1, \quad \forall i \in [N_2 + 1, \dots, N] \quad (4)$$

$$\sum_{i=1}^{N_1} \sum_{t=1}^T Y_{ijt} \leq LT_{j0}, \quad \forall j \in B \quad (5)$$

$$\sum_{i=N_1+1}^{N_2} \sum_{t=1}^T X_{ijt} + \sum_{i=N_2+1}^N \sum_{t=1}^T S_{ijt} \leq LM_{j0}, \quad \forall j \in B \quad (6)$$

$$Y_{ijt} \in [0, 1], \quad \forall i \in [0, \dots, N_1], \forall j \in B, \forall t \in T \quad (7)$$

$$X_{ijt} \in [0, 1], \quad \forall i \in [N_1 + 1, \dots, N_2], \forall j \in B, \forall t \in T \quad (8)$$

$$S_{ijt} \in [0, 1], \quad \forall i \in [N_2 + 1, \dots, N], \forall j \in B, \forall t \in T \quad (9)$$

$$Y_{ijt} = 0, \quad \forall i \in [0, \dots, N_1], \forall j \in B, \forall t \in T < r_i \quad (10)$$

$$X_{ijt} = 0, \quad \forall i \in [N_1 + 1, \dots, N_2], \forall j \in B, \forall t \in T < r_i \quad (11)$$

$$S_{ijt} = 0, \quad \forall i \in [N_2 + 1, \dots, N], \forall j \in B, \forall t \in T < r_i \quad (12)$$

Resolviendo este modelo se obtiene la asignación de contenedores a bloques que minimice el tiempo total requerido, cumpliendo con todas las restricciones impuestas.

De esta forma, el problema real se representa matemáticamente para poder aplicar algoritmos de optimización que encuentren la mejor solución posible.

# 5. Experimentos Computacionales

---

Para evaluar el desempeño de los algoritmos propuestos por Romero, se utilizaron las mismas 25 instancias aleatorias generadas para su investigación. Las instancias contenían entre 500 y 2500 contenedores para asignar. En todos los casos se utilizaron 5 bloques disponibles en el patio de la terminal. Además, también se varían parámetros como las capacidades iniciales de cada bloque, los tiempos de llegada y salida de los contenedores, entre otros.

En cuanto a los experimentos computacionales de este trabajo se llevaron a cabo en un computador personal con procesador AMD Ryzen 7 4700U de ocho núcleos, velocidad base de 2.0 GHz, 8 GB de memoria RAM DDR4 y unidad de estado sólido SSD de 512 GB. El código fue programado en lenguaje Python versión 3.9.4.

Cada una de las instancias fue resuelta utilizando los algoritmos desarrollados, registrando métricas de desempeño como el tiempo de CPU requerido y la calidad de la solución obtenida en términos del tiempo total de asignación. Los resultados permitieron comparar los distintos enfoques algorítmicos y extraer conclusiones respecto a su eficiencia. El siguiente capítulo analiza en detalle los hallazgos de estos experimentos computacionales.

## 5.1. Algoritmo de Proximidad

En esta sección se expone el algoritmo de proximidad propuesto en [56]. Este comienza inicializando las variables  $ts$  que representa el instante de tiempo actual; también inicializa la variable solución, que representa una lista de listas, donde cada sublista está formada por un contenedor, un bloque asignado y un tiempo en el que finaliza su asignación.

Posteriormente, se ordena la lista de contenedores según su tiempo de llegada  $r_i$  de menor a mayor. Luego inicia un bucle for que recorre cada contenedor  $i$  hasta  $N$ . Dentro del bucle se inicializa una variable “asignado” en 0 indicando que aún no se ha asignado ese contenedor.

Mediante un bucle while se asegura que no se comience la asignación hasta que el tiempo actual  $ts$  sea mayor o igual al tiempo de llegada  $r_i$  del contenedor  $i$ . Durante este bucle se comprueban salidas previstas en los bloques entre  $ts$  y  $ts + 1$  para actualizar capacidades.

Luego comienza un nuevo bucle while mientras no se haya asignado el contenedor ( $asignado == 0$ ). Dentro de este bucle se comprueba si el destino  $D_i$  es mar ( $D_i == 1$ ), en cuyo caso se selecciona el bloque  $j$  con menor tiempo  $T_{ij}$ . Si hay capacidad libre  $LM_{jt}$  en el lado mar se asigna el contenedor, reduciendo  $LM_{jt}$  en 1. Si no, se elige aleatoriamente otro bloque con capacidad.

Análogamente, si  $D_i$  es tierra (else) se selecciona el bloque  $j$  con  $T_{ij}$  mínimo. Si hay capacidad  $LT_{jt}$  libre en tierra se asigna reduciendo  $LT_{jt}$ . Si no, se elige aleatoriamente.

Cuando se asigna, se incrementa  $ts$  en  $T_{ij}$ , se comprueban salidas en  $[ts, ts + T_{ij}]$  y se actualizan capacidades. Se añade la asignación a la solución y se sale del bucle interno al estar asignado.

El proceso se repite para cada contenedor  $i$  hasta finalizar la lista ordenada. De esta forma se va construyendo la solución minimizando el tiempo total, respetando capacidades y seleccionando de forma heurística el bloque más cercano a cada contenedor o uno aleatorio si no hay capacidad en el más cercano. En el Algoritmo 1 se representan cada uno de los pasos mencionados previamente.

El algoritmo es bastante simple pero efectivo, ya que al ordenar por llegada e ir asignando secuencialmente se construye una solución factible y completa. La heurística de proximidad busca reducir los tiempos individuales para minimizar el tiempo total. También, permite resolver instancias de tamaño realista de forma rápida.

---

**Algoritmo 1:** Algoritmo de proximidad de Carlos Romero

---

```
ts = 0 ;
solucion =  $\emptyset$  ;
Ordenar de menor a mayor la lista de contenedores según su  $r_i$  ;
for cada contenedor  $i$  hasta  $N$  do
    asignado = 0 ;
    while  $ts < r_i$  do
        Comprobamos salidas previstas entre  $ts$  y  $ts + 1$  ;
         $ts = ts + 1$  ;
    end
    while asignado == 0 do
        if  $D_i == 1$  then
            Seleccionar mínimo  $T_{ij}$  del contenedor  $i \forall j \in B$  ;
            Bloque  $j =$  posición del mínimo  $T_{ij}$  ;
            if  $LM_{jts} > 0$  then
                 $LM_{jts} = LM_{jts} - 1$  ;
                asignado = 1 ;
            else
                Elegir un bloque  $B$  aleatorio de los restantes que tenga espacio ;
                asignado = 1 ;
            end
        else
            Seleccionar mínimo  $T_{ij}$  del contenedor  $i \forall j \in B$  ;
            Bloque  $j =$  posición del mínimo  $T_{ij}$  ;
            if  $LT_{jts} > 0$  then
                 $LT_{jts} = LT_{jts} - 1$  ;
                asignado = 1 ;
            else
                Elegir un bloque  $B$  aleatorio de los restantes que tenga espacio ;
                asignado = 1 ;
            end
        end
        if asignado == 1 then
            Comprobamos salidas previstas en  $\forall j \in B$  entre  $ts$  y  $ts + T_{ij}$  ;
            Actualizamos el tiempo  $\rightarrow ts = ts + T_{ij}$  ;
            Actualizamos  $LT_{jts}$  y  $LM_{jts}$  ;
            Añadimos a solucion el contenedor  $i$ , el bloque asignado  $j$  y  $ts$  ;
        end
    end
end
return solucion
```

---

## 5.2. Algoritmo GRASP

En esta sección se explica la propuesta de mejora al algoritmo anterior mediante la metodología GRASP (Greedy Randomized Adaptive Search Procedure), que es una metodología metaheurística desarrollada originalmente por [23] para resolver problemas de optimización combinatoria.

Consiste en la integración de dos técnicas: una fase constructiva greedy aleatorizada, donde se genera una solución inicial de forma iterativa agregando elementos de manera casi voraz pero con cierta aleatorización; y una fase de búsqueda local, donde se explora el entorno de la solución inicial buscando mejorarla mediante intercambios.

La aleatorización en la fase constructiva permite una mayor exploración del espacio de soluciones en comparación con métodos puramente greedy, evitando quedar atrapado en óptimos locales y obteniendo soluciones iniciales de buena calidad. La búsqueda local posterior explota estas soluciones iniciales para mejorarlas aún más.

Una característica distintiva del GRASP es su naturaleza multiarranque: se ejecutan múltiples iteraciones con diferentes semillas aleatorias, obteniendo una solución en cada iteración. Al final se elige la mejor solución encontrada en todas las iteraciones.

Esta combinación innovadora de greedy aleatorizado y búsqueda local en un esquema iterativo e incremental convirtió a GRASP en una metodología ampliamente adoptada para resolver complicados problemas de optimización combinatoria en diversas áreas.

### 5.2.1. Aleatorización

La fase constructiva del GRASP aplicado al problema de asignación de contenedores consiste en asignar cada contenedor  $i$  al bloque  $j$  de manera aleatoria, en lugar de siempre escoger el bloque con el tiempo mínimo  $T_{ij}$ .

Debido a esto, se propone la siguiente regla para determinar a qué bloque iría cada

contenedor:

$$T = \text{mín}(dist) + (\alpha * (\text{máx}(dist) - \text{mín}(dist)))$$

donde cada elemento corresponde a lo siguiente:

- T: métrica calculada que combina la distancia más corta con una ponderación de la variabilidad en las distancias. Así, puede utilizarse en diversas aplicaciones para evaluar y comparar conjuntos de datos de distancias, lo que puede ser útil para la toma de decisiones y análisis en contextos.
- $\alpha$ : factor de ponderación que multiplica la diferencia entre el valor máximo y el valor mínimo de las distancias. Es un parámetro entre 0 y 1 que controla el nivel de aleatorización. Un  $\alpha$  cercano a 0 implica una solución casi greedy, mientras que con  $\alpha$  cercano a 1 la solución es muy aleatoria.
- $\text{máx}(dist)$ : valor de la distancia del contenedor al bloque más lejano.
- $\text{min}$ : valor de la distancia del contenedor al bloque más próximo.

El Algoritmo [2](#) muestra el procedimiento general del método:

---

**Algoritmo 2:** Fase Constructiva - Aleatorización

---

```
ts = 0 ;
solucion =  $\emptyset$  ;
start_time = time() ;
Ordenar de menor a mayor la lista de contenedores según su  $r_i$  ;
while start_time < 60s do
  for cada contenedor i hasta N do
    asignado = 0 ;
     $T = \min(\text{dist}) + (\alpha * (\max(\text{dist}) - \min(\text{dist})))$ ;
    distT = [];
    Guardar en distT las distancias menores que el parámetro T;
    while ts <  $r_i$  do
      Comprobamos salidas previstas entre ts y ts + 1 ;
      ts = ts + 1 ;
    end
    while asignado == 0 do
      if  $D_i == 1$  then
        Seleccionar aleatoriamente  $T_{ij}$  en distT del contenedor  $i \forall j \in B$  ;
        Bloque  $j =$  posición del mínimo  $T_{ij}$  ;
        if  $LM_{jts} > 0$  then
           $LM_{jts} = LM_{jts} - 1$  ;
          asignado = 1 ;
        else
          Elegir un bloque B aleatorio de los restantes que tenga espacio ;
          asignado = 1 ;
        end
      else
        Seleccionar aleatoriamente  $T_{ij}$  en distT del contenedor  $i \forall j \in B$  ;
        Bloque  $j =$  posición del mínimo  $T_{ij}$  ;
        if  $LT_{jts} > 0$  then
           $LT_{jts} = LT_{jts} - 1$  ;
          asignado = 1 ;
        else
          Elegir un bloque B aleatorio de los restantes que tenga espacio ;
          asignado = 1 ;
        end
      end
    end
    if asignado == 1 then
      Comprobamos salidas previstas en  $\forall j \in B$  entre ts y ts +  $T_{ij}$  ;
      Actualizamos el tiempo  $\rightarrow ts = ts + T_{ij}$  ;
      Actualizamos  $LT_{jts}$  y  $LM_{jts}$  ;
      Añadimos a solucion el contenedor i, el bloque asignado j y ts ;
    end
  end
end
return solucion
```

---

De esta forma se genera una solución inicial factible pero introduciendo aleatorización controlada para explorar un espacio de soluciones más amplio y evitar quedar atrapado en óptimos locales, en comparación con una heurística puramente greedy.

### 5.2.2. Búsqueda Local

La búsqueda local es la segunda fase iterativa en la metodología GRASP, después de la construcción greedy aleatorizada de la solución inicial. El objetivo es explorar el entorno de la solución inicial tratando de mejorarla mediante pequeños cambios o movimientos; en el Algoritmo 3 se presenta su estructura general con el fin de comprender mejor su concepto y objetivo.

Para guiar la búsqueda local se define una función de costo que calcula el valor de la solución en términos de la métrica objetivo, en este caso el tiempo total de asignación en unidades de tiempo. De esta manera, se puede evaluar si los movimientos propuestos por la búsqueda local reducen el costo y por tanto mejoran la solución inicial o no.

---

**Algoritmo 3: Búsqueda Local General**

---

```
solucion_actual = copiar(solucion);  
mejores_soluciones = [];  
while tiempo < 60s do  
  Se modifica la solución de acuerdo a diferentes criterios;  
  Calcular nuevo costo de solucion_actual  
  if costo_solucion_actual < costo_solucion then  
    | mejores_soluciones.append(solucion_actual);  
    | solucion_actual = copiar(solucion);  
  end  
  else  
    | solucion_actual = copiar(solucion);  
  end  
end  
return mejores_soluciones
```

---

En este trabajo se proponen cuatro variantes de búsqueda local:

- En la primera se escogen aleatoriamente dos contenedores y se intercambian sus posiciones en la secuencia de asignación. Se calcula el nuevo costo y se compara con el de la solución original.
- En la segunda se escoge un contenedor al azar y se desplaza posición a posición en la lista, calculando en cada movimiento el nuevo costo y comparándolo con el original.
- La tercera variante escoge aleatoriamente tres contenedores, se intercambian con otros tres contenedores también escogidos al azar, y se evaluó el costo de esta nueva solución en comparación con la original.
- Por último, en la cuarta búsqueda local se selecciona un contenedor al azar y se mueve al final de la lista, recalculando el costo y aceptando el movimiento solo si representa una reducción del mismo.

En todas las variantes, si el movimiento no reduce el costo se regresa la solución original y se itera nuevamente con otros contenedores seleccionados al azar. El proceso de búsqueda local se repite durante 60 segundos hasta que no se encuentren mejoras adicionales en una iteración completa.

En resumen, el algoritmo GRASP desarrollado consta de dos fases iterativas principales, como se ve en el Algoritmo 4. La primera es una construcción greedy aleatorizada, donde se genera una solución inicial asignando contenedores usando una regla para controlar el nivel de aleatorización. La segunda fase es una búsqueda local alrededor de esta solución inicial, donde se realizan movimientos para generar soluciones vecinas y se calcula el costo de cada una, aceptando el movimiento sólo si disminuye el costo total. Se implementan cuatro variantes de movimientos: intercambio de contenedores, desplazamiento individual, intercambio múltiple e inserción al final. El proceso se repite durante 60 segundos buscando mejorar incrementalmente la solución inicial.

De esta forma, al integrar construcción greedy aleatoria y búsqueda local intensiva,

GRASP permite explorar ampliamente el espacio de soluciones al tiempo que explota las mejores encontradas, mejorando la calidad respecto a heurísticas puramente voraces.

---

**Algoritmo 4: GRASP**

---

*mejores\_soluciones* = [];

solución inicial ← Fase constructiva con aleatorización;

Local search(solución inicial);

**return** *mejores\_soluciones*

---

## 6. Resultados

---

Para evaluar y comparar el desempeño de los 4 algoritmos implementados en las 25 instancias generadas aleatoriamente para el problema de asignación de contenedores, se utilizó como métrica de evaluación el RPD (Relative Percentage Deviation). Este mide la desviación porcentual de la solución  $S$  de un algoritmo respecto a la mejor solución conocida  $S^*$  para esa instancia.

El RPD se calcula como:

$$RPD = \frac{S - S^*}{S^*} * 100$$

Valores de RPD cercanos a 0 indican que el algoritmo logra encontrar soluciones próximas o iguales a la mejor solución conocida. Mientras que RPD altos evidencian resultados de mala calidad alejados de la solución.

Analizando la Tabla 6.1, se puede notar que el algoritmo original de Proximidad obtiene RPD altos en la mayoría de instancias, con un promedio de 6.09% y un máximo de 21% en la instancia Ins\_B5\_NC500\_3. Esto indica que el enfoque puramente voraz de asignar cada contenedor siempre al bloque más cercano no es efectivo para minimizar el tiempo total de la solución.

Instancia	Proximidad			Local Search			Aleatorización			Grasp		
	Ut	tc (ms)	RPD	Ut	tc (seg)	RPD	Ut	tc (seg)	RPD	Ut	tc (seg)	RPD
Ins_B5_NC500.1	1560	0.0269	14.875	1508	240	11.046	1389	60	2.283	1358	300	0
Ins_B5_NC500.2	1438	0.0155	7.394	1391	240	3.883	1355	60	1.195	1339	300	0
Ins_B5_NC500.3	1364	0.0229	21.137	1334	240	18.472	1143	60	1.510	1126	300	0
Ins_B5_NC500.4	1798	0.0169	10.239	1734	240	6.315	1670	60	2.391	1631	300	0
Ins_B5_NC500.5	1840	0.0185	13.022	1788	240	9.828	1668	60	2.457	1628	300	0
Ins_B5_NC1000.1	2623	0.0551	7.061	2605	240	6.327	2467	60	0.694	2450	300	0
Ins_B5_NC1000.2	2634	0.0569	4.358	2605	240	3.209	2538	60	0.555	2524	300	0
Ins_B5_NC1000.3	2439	0.0525	5.493	2426	240	4.931	2322	60	0.433	2312	300	0
Ins_B5_NC1000.4	2547	0.0509	3.410	2512	240	1.989	2479	60	0.650	2463	300	0
Ins_B5_NC1000.5	2594	0.0500	4.302	2571	240	3.378	2500	60	0.523	2487	300	0
Ins_B5_NC1500.1	4051	0.0965	6.437	4021	240	5.649	3821	60	0.394	3806	300	0
Ins_B5_NC1500.2	4121	0.1741	1.904	4105	240	1.508	4064	60	0.495	4044	300	0
Ins_B5_NC1500.3	3982	0.1727	3.644	3962	240	3.123	3856	60	0.364	3842	300	0
Ins_B5_NC1500.4	3970	0.1852	3.197	3948	240	2.625	3867	60	0.520	3847	300	0
Ins_B5_NC1500.5	3758	0.0975	3.072	3741	240	2.606	3663	60	0.466	3646	300	0
Ins_B5_NC2000.1	5585	0.1674	3.273	5569	240	2.977	5427	60	0.351	5408	300	0
Ins_B5_NC2000.2	5936	0.1921	4.470	5890	240	3.661	5736	60	0.950	5682	300	0
Ins_B5_NC2000.3	5558	0.3921	3.040	5531	240	2.540	5415	60	0.389	5394	300	0
Ins_B5_NC2000.4	5752	0.3018	3.658	5737	240	3.388	5561	60	0.216	5549	300	0
Ins_B5_NC2000.5	5938	0.3527	3.630	5910	240	3.141	5743	60	0.227	5730	300	0
Ins_B5_NC2500.1	7381	0.5171	8.289	7225	240	6.001	6831	60	0.220	6816	300	0
Ins_B5_NC2500.2	7562	0.2655	2.508	7552	240	2.372	7411	60	0.461	7377	300	0
Ins_B5_NC2500.3	7605	0.2927	4.164	7435	240	1.835	7382	60	1.109	7301	300	0
Ins_B5_NC2500.4	7817	0.3267	8.179	7643	240	5.771	7269	60	0.595	7226	300	0
Ins_B5_NC2500.5	7813	0.2798	1.732	7786	240	1.380	7709	60	0.378	7680	300	0
<b>Promedio</b>	4306,64	0,1672	6,0996	4261,16	240	4,7183	4131,44	60	0,7930	4106,64	300	0

**Tabla 6.1:** Tabla de comparación de los 4 algoritmos para tiempos de asignación (ut), tiempos de computo (ms) y RPD.

Al complementar Proximidad con Búsqueda Local, se logra una reducción importante del RPD, con un promedio de 4.7% en las 25 instancias. Sin embargo, en algunos casos el RPD aún supera el 10%, evidenciando que la búsqueda local no es suficiente para encontrar buenas soluciones debido a las limitaciones del enfoque greedy inicial.

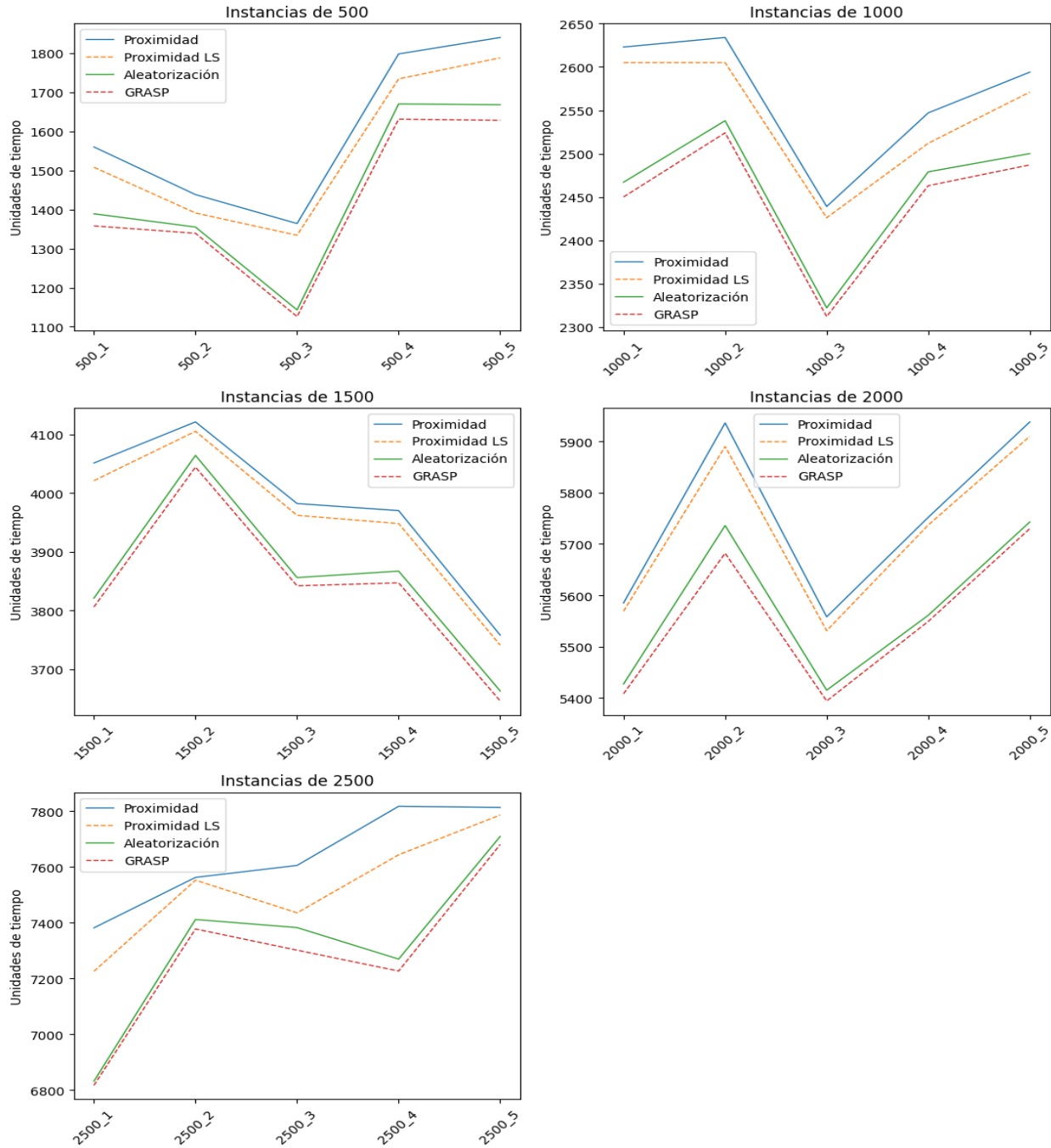
El algoritmo con Aleatorización Controlada presenta una mejora sustancial, con RPD promedio de solo 0.79% en las 25 instancias. El máximo RPD es 2.45% en la instancia Ins\_B5\_NC500.5. Esto demuestra que al incorporar aleatorización en la construcción de la solución inicial se logra explorar un espacio de soluciones mucho más amplio, encontrando asignaciones próximas al óptimo incluso partiendo de una solución

aleatoria.

Por último, el algoritmo GRASP basado en Aleatorización y Búsqueda Local obtiene un RPD de 0% en las 25 instancias, igualando la solución  $S^*$  encontrada por cualquier algoritmo en todos los casos. La búsqueda local intensifica la exploración del entorno de las diversas soluciones aleatorias iniciales, permitiendo alcanzar mejores soluciones.

Cabe destacar que para una comparación justa entre los algoritmos, cada uno se ejecutó 10 veces mostrando el peor caso de Proximidad y el mejor de GRASP. Aún bajo estas condiciones adversas para Proximidad, GRASP arrojó la mejor solución en todos los casos, validando la robustez del método. Adicionalmente, se modificaron las capacidades iniciales de los bloques en cada instancia, reduciéndolas para generar casos más exigentes. De lo contrario, con capacidades muy amplias, el ejercicio podría volverse trivial ya que Proximidad siempre tendría dónde asignar contenedores al bloque más cercano. Al limitar la capacidad se generan casos más realistas y complejos, donde la aleatorización de GRASP demuestra su valor al distribuir mejor los contenedores entre bloques con capacidad restringida. Incluso en estas instancias modificadas, GRASP fue capaz de dar con el mejor resultado, resaltando la eficiencia de la aleatorización y búsqueda local ante restricciones de capacidad.

En la Figura 6.1 se muestra una colección de las gráficas comparativas entre los algoritmos y por grupos de instancias:



**Figura 6.1:** Comparación gráfica en unidades de tiempo entre cada algoritmo.

En cuanto a tiempos computacionales, la Tabla 6.1 reporta en las columnas  $t_c$  el tiempo requerido o establecido para la ejecución de los algoritmos en cada una de las 25 instancias generadas. Para Local Search, Aleatorización y GRASP, se asignaron límites de tiempo amplios con el fin de permitir la mayor exploración y convergencia posible

dentro del proceso experimental.

Específicamente, para la búsqueda local se estableció un tiempo máximo de 240 segundos (4 minutos) por instancia. Este prolongado tiempo se fijó con el objetivo de que el algoritmo pudiera realizar una exploración completa y exhaustiva del entorno cercano a la solución inicial obtenida con el método de proximidad greedy. Al poder iterar ampliamente los movimientos e intercambios durante 4 minutos, se maximizan las oportunidades de encontrar mejoras locales significativas.

En cuanto al algoritmo con aleatorización controlada, se ejecutó con un límite de 60 segundos por instancia. Este corto tiempo fue suficiente debido a la rápida construcción de múltiples soluciones iniciales factibles al incorporar el elemento aleatorio. En menos de 1 minuto se pueden generar y evaluar diversas alternativas iniciales de alta calidad antes de alcanzar convergencia.

Por último, para el algoritmo GRASP que integra las dos técnicas, se estableció un tiempo máximo de 300 segundos (5 minutos) para cada una de las 25 pruebas. Esto le permite realizar extensamente tanto la construcción aleatoria de soluciones iniciales en la primera fase, como la prolongada búsqueda local de mejoras incrementales en la segunda fase, que es la más demandante computacionalmente.

No obstante, es importante recalcar que estos tiempos máximos no implican que los algoritmos requieran el límite completo para encontrar mejores soluciones en todos los casos. Simplemente se dejó correr hasta el máximo para permitir la convergencia total. En implementaciones prácticas con instancias de gran escala, se podrían utilizar criterios dinámicos de parada más temprana.

## 6.1. ANOVA y Prueba Tukey

La comparación inicial de los RPD promedio arrojó ganancias sustanciales al incorporar aleatorización y búsqueda local frente a los enfoques más simples. Sin embargo, para confirmar que estas mejoras no se deben al azar y poseen significancia estadística, es necesario aplicar pruebas estadísticas inferenciales robustas. En esta sección se lleva a cabo un análisis de varianza (ANOVA) de un factor para determinar si existen diferencias significativas entre los RPD de los cuatro algoritmos. Posteriormente, para identificar entre qué pares de algoritmos se presentan divergencias, se realiza una comparación empleando la prueba de Tukey. De esta manera, a través de sólidas técnicas estadísticas, se busca proveer una validación concluyente de las ganancias en calidad de solución obtenidas con la aleatorización y el algoritmo GRASP propuesto. Los resultados de los contrastes estadísticos brindan un soporte inferencial a las conclusiones iniciales.

Como se observa en la Tabla 6.2, el ANOVA arrojó un valor p de  $5.568 \times 10^{-12}$ , el cual es mucho menor que el nivel de significancia  $\alpha = 0,05$ . Esto permite rechazar la hipótesis nula de que los grupos tienen medias iguales, concluyendo que al menos entre dos algoritmos existen diferencias significativas en los valores de RPD.

**Tabla 6.2:** Resultados del ANOVA

Estadístico F	Valor p	Conclusión
24.841	$5,568 \times 10^{-12}$	Se rechaza la hipótesis nula, hay diferencias significativas entre los grupos.

Dado que el ANOVA detectó que hay diferencias entre grupos pero no especifica entre cuáles, se procedió a realizar una prueba de Tukey para realizar comparaciones por pares de algoritmos. En la Tabla 6.3 se reportan los resultados de esta prueba:

**Tabla 6.3:** Resultados de la Prueba de Tukey

Grupo 1	Grupo 2	Diferencia Media	Valor p	Límite Inferior	Límite Superior	Rechazo de $H_0$
RPD_Aleatorización	RPD_GRASP	-0.793	0.7821	-2.9935	1.4074	No
RPD_Aleatorización	RPD_LocalSearch	3.9252	0,0001	1.7247	6.1256	Sí
RPD_Aleatorización	RPD_Proximidad	5.3065	0,0000	3.1060	7.5069	Sí
RPD_GRASP	RPD_LocalSearch	4.7182	0,0000	2.5177	6.9187	Sí
RPD_GRASP	RPD_Proximidad	6.0995	0,0000	3.8991	8.3000	Sí
RPD_LocalSearch	RPD_Proximidad	1.3813	0.3606	-0.8191	3.5818	No

Al comparar el RPD del algoritmo con Aleatorización vs el RPD de GRASP, el valor p fue 0.7821, mayor que 0.05, por lo que no se rechaza la hipótesis nula de medias iguales entre este par de grupos. Esto era de esperarse dado que ambos obtuvieron un RPD promedio muy cercano a 0.

Por otro lado, en la comparación del RPD con Aleatorización vs el RPD con Local Search, el valor p resultó menor a 0.05, por lo que se rechaza la hipótesis nula y se concluye que existen diferencias estadísticamente significativas entre estos dos algoritmos en términos del RPD. Como era de esperarse, la Aleatorización controlada obtuvo un desempeño significativamente mejor que la búsqueda local, con una diferencia media de RPD de 3.9252.

Análogamente, al contrastar el RPD de Aleatorización contra el RPD de Proximidad, nuevamente se rechaza la hipótesis nula con un valor p menor a 0.05 y una amplia diferencia media de 5.3065 en el RPD. Esto valida la mejora sustancial en términos de calidad de solución que aporta la aleatorización frente al enfoque puramente voraz de Proximidad.

Al comparar GRASP con Local Search y Proximidad también se rechaza la hipótesis nula en ambos casos, comprobando estadísticamente que GRASP logra un RPD significativamente menor que ambos métodos, gracias a la integración de aleatorización y búsqueda local.

Los resultados de la prueba también indican que la incorporación de búsqueda local en el algoritmo de Proximidad no condujo a mejoras estadísticamente significativas en el RPD. Al comparar los grupos de Local Search y Proximidad, se obtuvo un valor p de 0.3606, mayor al nivel de significancia de 0.05. Por lo tanto, no se rechaza la hipótesis nula de medias iguales entre ambos algoritmos.

Si bien numéricamente el RPD promedio de Local Search fue menor que el de Proximidad, esta mejora no fue lo suficientemente grande como para ser estadísticamente relevante, según la prueba de Tukey. Esto sugiere que el operador de búsqueda local por sí solo no aporta una mejora sustancial sobre el enfoque voraz puro.

En conclusión, a través de técnicas estadísticas inferenciales como ANOVA y Tukey se valida que la incorporación de aleatorización controlada en la construcción de soluciones conduce a una reducción significativa del RPD en comparación con enfoques estrictamente greedy; empero, la búsqueda local por sí sola no es estadísticamente significativa. Aún así, la integración de aleatorización y búsqueda local en GRASP potencia aún más las mejoras en general, superando ampliamente en desempeño a los algoritmos de Proximidad y Local Search. Los análisis estadísticos refuerzan las conclusiones encontradas inicialmente mediante la comparación numérica de los RPD promedio.

# 7. Conclusiones y futuras líneas de investigación

---

El presente trabajo de investigación abordó el problema de asignación óptima de contenedores entrantes en el patio de una terminal portuaria. Esta asignación determina la ubicación de cada contenedor dentro de los distintos bloques rectangulares disponibles, buscando minimizar el tiempo total del proceso sujeto a múltiples restricciones operativas.

Para resolver de manera eficiente este problema NP-Hard, se propuso desarrollar un novedoso algoritmo basado en la metodología GRASP (Greedy Randomized Adaptive Search Procedure). GRASP combina una construcción greedy aleatorizada de soluciones iniciales diversas con una intensificación mediante búsqueda local, en un proceso iterativo e interactivo.

Los resultados experimentales obtenidos permiten extraer diversas conclusiones relevantes. En primer lugar, se comprobó que el algoritmo GRASP propuesto, integrando la construcción greedy aleatorizada y búsqueda local intensiva, constituye una solución computacional efectiva para asignar contenedores entrantes, minimizando el tiempo total requerido por el proceso.

Los experimentos computacionales realizados sobre un amplio conjunto de 25 instancias aleatorias permitieron validar la efectividad del algoritmo GRASP propuesto. En todas las instancias, GRASP fue capaz de encontrar la mejor solución, mejorando los resultados del algoritmo greedy de proximidad original que tiene un RPD promedio de 6%.

Incluso bajo condiciones adversas como capacidad limitada en bloques y comparaciones en base al peor caso de GRASP contra el mejor de proximidad, la

metodología propuesta logró consistentemente igualar la solución de mejor calidad posible. Esto demuestra la robustez de GRASP para explorar diversamente el espacio de soluciones y explotar las mejores a través de la búsqueda local.

Por otra parte, el análisis estadístico realizado mediante ANOVA y la prueba Tukey confirmó que las mejoras en RPD obtenidas con la aleatorización controlada y el algoritmo GRASP no se deben al azar ni a fluctuaciones muestrales, sino que representan ganancias altamente significativas en rendimiento frente a enfoques puramente greedy e individuales. Las sólidas técnicas estadísticas aplicadas brindaron un soporte inferencial y contundente a las conclusiones extraídas.

En conjunto, los aportes teóricos y prácticos del trabajo de investigación son variados. En el ámbito teórico, se realiza una contribución al proponer y validar una metodología algorítmica basada en la integración de GRASP, sentando las bases para el desarrollo de nuevas variantes y aplicaciones. En cuanto a la parte práctica, se provee una herramienta computacional con gran potencial para optimizar procesos logísticos claves en la operación de puertos reales.

En conclusión, este trabajo de grado cumplió el objetivo general planteado de desarrollar un algoritmo de asignación de contenedores entrantes más eficiente que los existentes, mediante la exploración innovadora de modernas metaheurísticas como GRASP, obteniendo resultados prometedores tanto en términos de calidad de soluciones como eficiencia computacional. Asimismo, se sientan las bases para continuar desarrollando algoritmos y herramientas que permitan mejorar la competitividad de terminales en Colombia y América Latina.

## **7.1. Futuras Líneas de Investigación**

Las técnicas y enfoques novedosos incorporados en este trabajo sientan las bases para continuar cultivando una fructífera línea de investigación en el área de optimización de procesos portuarios, tanto perfeccionando el algoritmo GRASP

desarrollado como extendiendo su aplicación a otros problemas relevantes en la gestión de terminales como localización de contenedores, planeamiento de grúas, enrutamiento de vehículos, etc.

Como trabajo futuro se identifican múltiples oportunidades para continuar extendiendo y fortaleciendo esta línea de investigación:

- Profundizar en el análisis del impacto de variar el parámetro alpha que controla el grado de aleatorización en la fase constructiva de GRASP. Determinar un rango óptimo para este parámetro.

- Evaluar vecindarios más complejos e intensos en la búsqueda local, como inserción de cadenas o secuencias de contenedores en lugar de simples intercambios.

- Validar el desempeño de GRASP en instancias de referencia y terminales reales de gran escala, con miles de contenedores y bloques. Incorporar restricciones adicionales como balanceo de carga.

- Comparar el rendimiento de GRASP con otras metaheurísticas como algoritmos evolutivos, recocido simulado, búsqueda tabú, entre otras. Determinar fortalezas y diferencias.

- Aplicar la metodología GRASP desarrollada para optimizar otros problemas relevantes en gestión de terminales portuarias, como localización de contenedores dentro de bloques, enrutamiento de vehículos en patio, asignación de grúas, etc.

- Complementar GRASP con técnicas de aprendizaje automático y algoritmos hiper-heurísticos para seleccionar de manera inteligente los movimientos más prometedores.

Las futuras líneas de investigación propuestas permitirían consolidar y expandir los aportes realizados en este trabajo, con el fin último de desarrollar herramientas de optimización avanzadas que impulsen la competitividad del sector portuario.

# Bibliografía

---

- [1] A. Bartošek, O. M. (2013). Quay cranes in container terminals. 6. Recuperado de:  
<https://tots.upol.cz/pdfs/tot/2013/01/02.pdf>.
- [2] Acero, A. (2021). Cargue y descargue de contenedores como factor limitante de la capacidad portuaria y determinante del nivel de competitividad del puerto de buenaventura en el marco de alianza del pacífico. Recuperado de:  
<https://repository.ucc.edu.co/server/api/core/bitstreams/0cb20c3b-39ff-4f56-b0df-e5677656dff8/content>.
- [3] Alonso M. et al. (2014). A reactive grasp algorithm for the container loading problem with load-bearing constraints. *European J. Industrial Engineering*, pages 669–694. DOI:  
<https://doi.org/10.1504/EJIE.2014.065734>.
- [4] Alvarez et al. (2005). A tabu search algorithm for the pallet loading problem. *Operations Research-Spektrum*, pages 43–61. Recuperado de:  
[https://www.researchgate.net/publication/225889500\\_A\\_tabu\\_search\\_algorithm\\_for\\_the\\_pallet\\_loading\\_problem](https://www.researchgate.net/publication/225889500_A_tabu_search_algorithm_for_the_pallet_loading_problem).
- [5] Bierwirth, C. and Meisel, F. (2015). A follow-up survey of berth allocation and quay crane scheduling problems in container terminals. *European Journal of Operational Research*, 244(3):675–689. Recuperado de:  
<https://www.sciencedirect.com/science/article/pii/S0377221714010480>.
- [6] Bottasso A. et al. (2013). The impact of port throughput on local employment: Evidence from a panel of european regions. *Transport Policy*, 27(C):32–38. Recuperado de:  
[https://econpapers.repec.org/article/eeetrapol/v\\_3a27\\_3ay\\_3a2013\\_3ai\\_3ac\\_3ap\\_3a32-38.htm](https://econpapers.repec.org/article/eeetrapol/v_3a27_3ay_3a2013_3ai_3ac_3ap_3a32-38.htm).

- [7] Camargo, Y. (2019). Historicidad del transporte en Colombia, un proceso de transición y rupturas. *Revista de Estudios Históricos*, pages 193–217. Recuperado de: <https://www.redalyc.org/journal/898/89860728007/html/#fn10>.
- [8] Cardozo, O. (2015). Historia del río Magdalena desde el municipio de Girardot a la ciudad de Barranquilla. *PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO*. Recuperado de: [https://wiki.ead.pucv.cl/images/7/73/4FUNDAMENTO\\_TEORICO\\_HISTORIA\\_DEL\\_RIO\\_MAGDALENA\\_GIRARDOT-BARRANQUILLA.pdf](https://wiki.ead.pucv.cl/images/7/73/4FUNDAMENTO_TEORICO_HISTORIA_DEL_RIO_MAGDALENA_GIRARDOT-BARRANQUILLA.pdf).
- [9] Carga (2020). Historia de los tres puertos más importantes de Colombia. Recuperado de: <https://carga.com.co/historia-de-los-tres-puertos-mas-importantes-de-colombia/>.
- [10] Carlan V. (2017). Digital innovation in the port sector: Barriers and facilitators. *Competition and Regulation in Network Industries*, 18:178359171773479. Recuperado de: [https://www.researchgate.net/publication/320454812\\_Digital\\_innovation\\_in\\_the\\_port\\_sector\\_Barriers\\_and\\_facilitators](https://www.researchgate.net/publication/320454812_Digital_innovation_in_the_port_sector_Barriers_and_facilitators).
- [11] Carlo, H. J., Vis, I. F., and Roodbergen, K. J. (2014). Storage yard operations in container terminals: Literature overview, trends, and research directions. *European Journal of Operational Research*, 235(2):412–430. Maritime Logistics.
- [12] Caserta M. et al. (2009). A new binary description of the blocks relocation problem and benefits in a look ahead heuristic. pages 37–48. Recuperado de: [https://www.researchgate.net/publication/220866784\\_A\\_New\\_Binary\\_Description\\_of\\_the\\_Blocks\\_Relocation\\_Problem\\_and\\_Benefits\\_in\\_a\\_Look\\_Ahead\\_Heuristic](https://www.researchgate.net/publication/220866784_A_New_Binary_Description_of_the_Blocks_Relocation_Problem_and_Benefits_in_a_Look_Ahead_Heuristic).
- [13] Chen, L. et al. (2007). A tabu search algorithm for the integrated scheduling problem of container handling systems in a maritime terminal. *European Journal of Operational Research*, 181:40–58.
- [14] Correcher, J. (2017). *Models and algorithms for berth allocation problems in port terminals*. PhD thesis.

- [15] Dirección General Marítima (2023). Publicaciones estadísticas anuales de transporte marítimo. Recuperado de:  
<https://www.dimar.mil.co/operaciones-estadisticas/publicaciones-estadisticas-de-transporte-maritimo>.
- [16] Double-trolley quayside container crane (2013). Recuperado de:  
<https://chec.my/double-trolley-quayside-container-crane/>.
- [17] Díaz, M. (2015). Afectación y protección de ecosistemas marino-costeros en colombia. pages 95–116. Recuperado de:  
<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjK69GQ5oCEAxW1hYkEHU9iBPKQFnoECAgQAQ&url=https%3A%2F%2Frevistas.usergioarboleda.edu.co%2Findex.php%2Fverbum%2Farticle%2Fdownload%2F469%2F413&usg=A0vVaw1LqZ5u4oAsvIFsCyl20qQq&opi=89978449>.
- [18] Editorial Etecé (s.f.). Transporte marítimo. *Enciclopedia Concepto*. Recuperado de:  
<https://concepto.de/transporte-maritimo/>.
- [19] El Espectador (2023). Grupo puerto de cartagena: el quinto mejor del mundo. Recuperado de:  
<https://www.elespectador.com/contenido-patrocinado/grupo-puerto-de-cartagena-el-quinto-mejor-del-mundo/>.
- [20] El ruido del tráfico marítimo es una amenaza para la vida marina (2023). *Portal Ambiental*. Recuperado de:  
<https://www.portalambiental.com.mx/sabias-que/20230531/el-ruido-del-trafico-maritimo-es-una-amenaza-para-la-vida-marina>.
- [21] Encinas, J. E. (2017). La evolución de los puertos marítimos. *Instituto de investigaciones estratégicas de la armada de México*. Recuperado de:  
[https://cesnav.uninav.edu.mx/cesnav/links\\_acc\\_progr/doctorado\\_damp\\_site/productividad\\_academica/CAPENCINASVALENZUELA/da\\_65-17-laevoluciondelospuertosmaritimos.pdf](https://cesnav.uninav.edu.mx/cesnav/links_acc_progr/doctorado_damp_site/productividad_academica/CAPENCINASVALENZUELA/da_65-17-laevoluciondelospuertosmaritimos.pdf).

- [22] Expósito-Izquierdo, C. et al. (2012). Pre-marshalling problem: Heuristic solution method and instances generator. *Expert Systems with Applications*, 39(9):8337–8349. Recuperado de:  
<https://www.sciencedirect.com/science/article/pii/S0957417412002151>.
- [23] Feo, T. Resende, M. (1995). "greedy randomized adaptive search procedures". *Journal of Global Optimization*, pages 109—133. Recuperado de:  
[doi:10.1007/BF01096763](https://doi.org/10.1007/BF01096763).
- [24] Feo T. Resende, M. (1989). "a probabilistic heuristic for a computationally difficult set covering problem". pages 67–71. DOI:  
<https://doi.org/10.1016%2F0167-6377%2889%2990002-3>.
- [25] Gaviria, C. (2018). ¿qué pasó después de la eliminación de colpuertos? *Semana*. Recuperado de:  
<https://www.semana.com/que-paso-despues-de-la-eliminacion-de-colpuertos/593737/>.
- [26] Günther, H.-O. and Kim, K. (2006). Container terminals and terminal operations. *OR Spectrum*, 28:437–445. Recuperado de:  
[https://www.researchgate.net/publication/226429109\\_Container\\_terminals\\_and\\_terminal\\_operations](https://www.researchgate.net/publication/226429109_Container_terminals_and_terminal_operations).
- [27] Herrera, C. (2022). Treinta años de la reestructuración del sector portuario colombiano; eslabón clave en el comercio exterior y la reactivación de la economía. *Acuerdos. Revista de Derecho Económico Internacional*. Recuperado de:  
<https://acuerdosrevista.mincit.gov.co/articulos/treinta-anos-de-la-reestructuracion-del-sector-por>.
- [28] Hwan, K. Park, Y. (2004). A crane scheduling method for port container terminals. *European Journal of Operational Research*, pages 752–768. DOI:  
[https://doi.org/10.1016/S0377-2217\(03\)00133-4](https://doi.org/10.1016/S0377-2217(03)00133-4).

- [29] Jin X. et al. (2023). Container stacking optimization based on deep reinforcement learning. Recuperado de:  
<https://www.sciencedirect.com/science/article/pii/S0952197623006929>.
- [30] Kemme, N. (2012). Effects of storage block layout and automated yard crane systems on the performance of seaport container terminals. *Or Spektrum*, 34:1–29. Recuperado de:  
<http://dx.doi.org/10.1007/s00291-011-0242-7>.
- [31] Lam, J. S. L. and Yap, W. (2011). Container port competition and complementarity in supply chain systems: Evidence from the pearl river delta. *Maritime Economics and Logistics*, 13:102–120. Recuperado de:  
<http://dx.doi.org/10.1057/mel.2011.5>.
- [32] Larrea, M. (2022). El papel de los puertos en el transición energética. Recuperado de:  
[https://www.orquestra.deusto.es/images/investigacion/publicaciones/informes/cuadernos-orkestra/220022\\_Papel\\_puertos\\_transicion\\_energetica\\_ES.pdf](https://www.orquestra.deusto.es/images/investigacion/publicaciones/informes/cuadernos-orkestra/220022_Papel_puertos_transicion_energetica_ES.pdf).
- [33] Legato, P. and Mazza, R. M. (2017). A simulation model for designing straddle carrier-based container terminals.
- [34] Lehnfeld, J. Knust, S. (2014). Loading, unloading and premarshalling of stacks in storage areas: Survey and classification. *European Journal of Operational Research*. Recuperado de:  
<https://www.sciencedirect.com/science/article/abs/pii/S0377221714002252>.
- [35] Levinson, M. (2008). Introduction to the box: How the shipping container made the world smaller and the world economy bigger. Recuperado de:  
<https://search-ebSCOhost-com.ez.urosario.edu.co/login.aspx?direct=true&AuthType=ip&db=edsrep&AN=edsrep.h.pup.chapts.8131.1&lang=es&site=eds-live&scope=site>.

- [36] Lucenti, K. Mendes, J. (2019). Cómo lograr puertos más eficientes. *BID*. Recuperado de:  
<https://blogs.iadb.org/integracion-comercio/es/eficientes-puertos/>.
- [37] MarGerencia (s.f.). El puerto de rotterdam (holanda). Recuperado de:  
<https://marygerencia.com/2010/05/31/el-puerto-de-rotterdam/>.
- [38] Mejía, A. (2003). Sistemas de transporte en colombia en los siglos xix y xx. Recuperado de:  
<https://repository.eia.edu.co/server/api/core/bitstreams/cc63bf3a-7d61-49fd-b0eb-7582c0e03be3/content>.
- [39] Moon, D. (2014). The impact of port operations on efficient ship operation from both economic and environmental perspectives. *Maritime Policy Management*, 41. DOI:  
[10.1080/03088839.2014.931607](https://doi.org/10.1080/03088839.2014.931607).
- [40] Murty, K. et al. (2005). Hongkong international terminals gains elastic capacity using a data-intensive decision-support system. *Interfaces*, 35:61–75. Recuperado de:  
<http://dx.doi.org/10.1287/inte.1040.0120>.
- [41] Muñoz, V. (2008). Optimización de la producción en una terminal marítima de contenedores. umbrales y punto de equilibrio. Recuperado de:  
<https://www.tdx.cat/bitstream/handle/10803/7009/01VEmc01de01.pdf?sequence=1&isAllowed=y>.
- [42] Naciones Unidas (1992). "declaración de río sobre el medio ambiente y el desarrollo". Recuperado de:  
<https://www.un.org/spanish/esa/sustdev/agenda21/riodeclaration.htm>.
- [43] Ng, W. (2005). Crane scheduling in container yards with inter-crane interference. *European Journal of Operational Research*, pages 64–78. DOI:  
<https://doi.org/10.1016/j.ejor.2003.11.025>.
- [44] Nishimura E. et al. (2009). Container storage and transshipment marine terminals.

*Transportation Research Part E: Logistics and Transportation Review*, pages 771–786. DOI: <https://doi.org/10.1016/j.tre.2009.03.003>.

- [45] Ocampo, R. (2021). Puerto de Buenaventura mueve 35 % del comercio exterior no minero-energético. *La República*. Recuperado de: <https://www.larepublica.co/economia/puerto-de-buenaventura-mueve-35-del-comercio-exterior-no-minero-energetico-3190357>.
- [46] OCEANA (s.f.). Contaminación por la industria naval. Recuperado de: <https://europe.oceana.org/es/contaminacion-por-la-industria-naval-0/#:~:text=Al%20igual%20que%20otros%20modos,contaminantes%20que%20agravan%20el%20problema>.
- [47] ONU (2020). Review of maritime transport 2020. *United Nations Conference on Trade and Development (UNCTAD)*. Recuperado de: [https://unctad.org/system/files/official-document/rmt2020\\_en.pdf](https://unctad.org/system/files/official-document/rmt2020_en.pdf).
- [48] Petering, M. E. and Murty, K. G. (2009). Effect of block length and yard crane deployment systems on overall performance at a seaport container transshipment terminal. *Computers Operations Research*, 36(5):1711–1725. Recuperado de: <https://www.sciencedirect.com/science/article/pii/S0305054808000798>.
- [49] Portafolio (2019). Puertos de Colombia, los cuartos más movidos de Latinoamérica. Recuperado de: <https://www.portafolio.co/economia/puertos-de-colombia-los-cuartos-mas-movidos-de-latinoamerica-528183>.
- [50] Prieto, D. (2019). Estimación de las emisiones atmosféricas de buques en el puerto de Barranquilla. Recuperado de: <https://manglar.uninorte.edu.co/handle/10584/8692?show=full>.
- [51] R., B. K. G. (2005). A critical review of conventional terminology for classifying seaports. *Transportation Research Part A: Policy and Practice*, 39(1):75–92. Recuperado

de:

<https://www.sciencedirect.com/science/article/pii/S0965856404001089>.

[52] Ramírez, O. (2023). Mejoras en infraestructura, la clave para impulsar el crecimiento del sector portuario. *Semana*. Recuperado de:

<https://www.semana.com/mejor-colombia/articulo/mejoras-en-infraestructura-la-clave-para-impulsar-el-crecimiento-del-sector-portuario/202300/>.

[53] Rashidi, H. and Tsang, E. P. (2013). Novel constraints satisfaction models for optimization problems in container terminals. *Applied Mathematical Modelling*, 37(6):3601–3634. Recuperado de:

<https://www.sciencedirect.com/science/article/pii/S0307904X12004544>.

[54] Redacción (2017). Transporte marítimo. *Gerente*. Recuperado de:

<https://gerente.com/co/guias/transporte-maritimo/>.

[55] Rodrigue, J. (2020). The geography of transport systems. Recuperado de:

<https://transportgeography.org/media/e-book/>.

[56] Romero, C. (2020). Optimización del problema de asignación de contenedores en una terminal portuaria. Recuperado de:

<https://riunet.upv.es/handle/10251/152165>.

[57] Rosales, M. (2019). ¿cuánto contaminan los buques de carga? *La Verdad*. Recuperado de:

<https://laverdadnoticias.com/ecologia/Cuanto-contaminan-los-buques-de-carga-20191109-0115.html>.

[58] Saidi, M. et al. (2015). An ant colony algorithm (aca) for solving the new integrated model of job shop scheduling and conflict-free routing of agvs. *Computers Industrial Engineering*, 86:2–13. Applications of Computational Intelligence and Fuzzy Logic to Manufacturing and Service Systems.

[59] Santos, A. (1970). *Por caminos de mar, tierra y aire: evolución del transporte en Colombia*.

Italgraf. Recuperado de:

<https://books.google.com.co/books?id=2NQPAQAAIAAJ>.

[60] Scheithauer, G. (1992). Algorithms for the container loading problem. *Springer Berlin Heidelberg*, pages 445–452. Recuperado de:

[https://link.springer.com/chapter/10.1007/978-3-642-46773-8\\_112#citeas](https://link.springer.com/chapter/10.1007/978-3-642-46773-8_112#citeas).

[61] Sikorra J. et al. (2021). Allocation of container slots based on machine learning. Recuperado de:

[https://hhla.de/fileadmin/module/its/ITS\\_Tech\\_Paper-HPC\\_machine\\_learning.pdf](https://hhla.de/fileadmin/module/its/ITS_Tech_Paper-HPC_machine_learning.pdf).

[62] Stahlbock, R. and Voss, S. (2008). Operations research at container terminals: A literature update. *Operations Research-Spektrum*, 30:1–52. Recuperado de:

[https://www.researchgate.net/publication/225323734\\_Operations\\_research\\_at\\_container\\_terminals\\_A\\_literature\\_update](https://www.researchgate.net/publication/225323734_Operations_research_at_container_terminals_A_literature_update).

[63] Steenken, D., Voss, S., and Stahlbock, R. (2004). Container terminal operation and operations research - a classification and literature review. *OR Spectrum*, 26:3–49. Recuperado de:

<http://dx.doi.org/10.1007/s00291-003-0157-z>.

[64] Two New Hybrid Rubber-Tired Gantry Cranes (RTGs) to be installed in Yidong Terminal, Port of Shanghai (s.f.). Recuperado de:

<https://corvusenergy.com/two-new-hybrid-rubber-tired-gantry-cranes-rtgs-powered-by-corvus-energy-to-be-installed-in-yidong-terminal-port-of-shanghai/>.

[65] Vidal, I. (2023). Análisis teórico-práctico de las emisiones de un buque oceanográfico con propulsión diésel-eléctrica. Recuperado de:

[https://upcommons.upc.edu/bitstream/handle/2117/392147/179450\\_TFM\\_Isaac\\_Vidal\\_DefinitivoOK.pdf?sequence=2&isAllowed=y](https://upcommons.upc.edu/bitstream/handle/2117/392147/179450_TFM_Isaac_Vidal_DefinitivoOK.pdf?sequence=2&isAllowed=y).

[66] Viloría, J. (s.f.). Los puertos marítimos de Colombia en la década de los noventa. *Banco de la República Sucursal Cartagena*. Recuperado de:

<https://www.banrep.gov.co/sites/default/files/publicaciones/archivos/puertos-maritimos.pdf>.

[67] Vis, I. and Harika, I. (2004). Comparison of vehicle types at an automated container terminal. *OR Spectrum*, 26:117–143. Recuperado de:  
<http://dx.doi.org/10.1007/s00291-003-0146-2>.

[68] Zlatař, Y. (1994). Los derrames de petróleo y su impacto ambiental. *Revista Museo*. Recuperado de:  
<http://sedici.unlp.edu.ar/handle/10915/47221>.

[69] Şafak Ö. Erdoğan, G. (2023). A large neighbourhood search algorithm for solving container loading problems. Recuperado de:  
<https://www.sciencedirect.com/science/article/pii/S0305054823000631>.