

Universidad del Valle  
Facultad de Ingeniería  
Escuela de Ingeniería en Sistemas y Computación



Refinamiento de la planificación eco-eficiente de estiba en  
grandes buques portacontenedores.

Trabajo de grado presentado por  
Adolfo León Canizales Murcia

Santiago de Cali  
Abril 2013

Universidad del Valle  
Facultad de Ingeniería  
Escuela de Ingeniería en Sistemas y Computación



Refinamiento de la planificación eco-eficiente de estiba en  
grandes buques portacontenedores.

Estudiante: Adolfo León Canizales Murcia  
leon99adolfo@gmail.com

Director: Alberto Delgado

Codirector: Juan Francisco Díaz Frías, Ph.D.  
Profesor Titular  
Universidad del Valle

Santiago de Cali  
Abril 2013

---

# Índice general

---

<b>Índice general</b>	<b>I</b>
<b>1 Contexto y objetivos</b>	<b>3</b>
1.1. Desarrollo e investigación . . . . .	3
1.1.1. Planteamiento del problema . . . . .	3
1.2. Objetivos . . . . .	5
1.2.1. Objetivo general . . . . .	5
1.2.2. Objetivos específicos . . . . .	5
1.2.3. Productos esperados . . . . .	6
1.3. Justificación . . . . .	6
1.3.1. Justificación económica . . . . .	6
1.3.2. Justificación social . . . . .	7
1.3.3. Justificación académica . . . . .	7
<b>2 Marco Teórico</b>	<b>9</b>
2.1. Proceso de estiba . . . . .	9
2.1.1. Objetivos del proceso de estiba . . . . .	9
2.1.2. Procesos de estibación en buques portacontenedores . . . . .	9
2.2. CCP - Programación por restricciones concurrentes . . . . .	10
2.2.1. Modelo CCP . . . . .	10
2.2.2. Ejemplo CCP . . . . .	11
2.3. C++ . . . . .	12
2.4. Gecode . . . . .	12
<b>3 Estado de Arte</b>	<b>15</b>
3.1. Métodos basados en modelos únicos . . . . .	15
3.2. Métodos basados en la descomposición del problema en varios mo- delos de optimización . . . . .	16
3.3. Descripción del modelo en restricciones de Delgado et al (2012)[7].	17

<b>4</b>	<b>Introducción a Modelos</b>	<b>21</b>
4.1.	Abstracciones de restricciones . . . . .	22
4.2.	Conjunto de constantes . . . . .	24
<b>5</b>	<b>Modelo con contenedor virtual</b>	<b>27</b>
5.1.	Variables de decisión . . . . .	27
5.2.	Restricciones del modelo . . . . .	27
5.2.1.	Detalle de las restricciones . . . . .	27
5.2.2.	Propagación del modelo con respecto a la función objetivo .	32
5.3.	Estrategia de distribución . . . . .	32
5.3.1.	Estrategia de distribución por pila . . . . .	33
5.3.2.	Estrategia de distribución por niveles de cada pila . . . . .	33
<b>6</b>	<b>Modelo con espacios y contenedor virtuales</b>	<b>35</b>
6.0.3.	Variables de decisión . . . . .	35
6.0.4.	Restricciones del modelo . . . . .	35
6.0.5.	Propagación del modelo con respecto a la función objetivo .	40
	<b>Lista de Símbolos y Abreviaciones</b>	<b>41</b>
	<b>Índice de figuras</b>	<b>42</b>
	<b>Índice de cuadros</b>	<b>43</b>
	<b>Bibliografía</b>	<b>45</b>

---

# **Agradecimientos**

---



## Capítulo 1

---

# Contexto y objetivos

---

### 1.1. Desarrollo e investigación

#### 1.1.1. Planteamiento del problema

A lo largo de los últimos tiempos, en la sociedad, el transporte masivo de productos por vía marítima es de gran importancia, debido a que la movilización de objetos entre diferentes regiones cumple un factor de progreso, mejoramiento y bienestar. La industria del transporte marítimo de mercancías posee un gran impacto en la economía mundial. Acto que es reflejado en el incremento considerable y evolución de la flota mundial de buques mercantes y puertos, al pasar de los años.

El transporte por mar de los diferentes tipos de mercancía es realizada por diversos tipos de buques, en función de cómo es presentada esta mercancía para su embarque. Debido a notables ventajas con relación a la rapidez, eficacia y disminución de pérdidas por avería, uno de los más importantes embarques para carga seca son los buques portacontenedores. Estos buques están diseñados para estibar y transportar muchos tipos de mercancía, inclusive mercancía refrigerada.

Actualmente, los buques portacontenedores son el medio utilizado para las tareas de transporte de productos entre las diferentes regiones. Estos buques satisfacen un gran porcentaje, el 90 %, de la movilización de objetos en todo el mundo y es difícil imaginarse otro tipo de transporte que satisfaga dicha necesidad.

Por lo anterior, es de gran importancia que el proceso de transporte de productos en los buques portacontenedores sea rápido y efectivo. Por tal motivo, el proceso de transporte genera el problema de como transportar las mercancías cumpliendo determinadas características. Es decir, que la construcción de herramientas que implementen modelos de optimización podrían mejorar el proceso de embarque y desembarque de contenedores en un barco de una forma refinada y eficaz.

La optimización de procesos de estiba en los puertos costeros es un proceso

complejo que implica encontrar una buena forma de organizar un conjunto de cajas metálicas, contenedores, en un buque. Este ordenamiento debe estar enfocado a la búsqueda de objetivos como la seguridad del barco, para que el barco pueda soportar los viajes en alta mar sin que este sufra daños o se hunda, y la rapidez para carga y descarga de contenedores del buque en el menor tiempo posible en cada uno de los puertos que atraca.

De esta manera, uno de los más fuertes fundamentos del sector portuario es una buena planeación de estiba, la cual se ha estado produciendo por coordinadores que utilizan herramientas gráficas que les permite generar un plan para la embarcación, pero debido a la complejidad del problema se ha aumentado un creciente interés por la ampliación de estas herramientas fusionadas con algoritmos veloces de optimización que ayuden en la planeación de estiba.

El sistema mostrado en éste documento intenta ordenar de una forma factible y conveniente contenedores en un buque porta-contenedor. Por un lado, los contenedores son cajas de metal donde se puede almacenar muchos tipos de bienes. Cada contenedor tiene un ancho de 8 pies, un alto de 8.6 o 9.6 pies, un largo de 20, 40 o 45 pies y un puerto donde ha de ser descargado. Existen varios tipos de contenedores y éstos son ubicados en el barco dependiendo de un conjunto de reglas de separación (ver Figura 1.1). Por otro lado, Los buques porta-contenedores contienen espacios (bahías) en los que almacenan los contenedores. Las bahías se encuentran distribuidas en todo el barco, sobre y bajo cubierta separadas por tapas de escotilla a lo largo del buque. Éstas bahías se separan transversalmente en pilas que tienen el ancho de 8 pies, el largo de una 20 (TEU - Twenty-foot Equivalent Unit), 40 (FEU - Forty-foot Equivalent Unit) o 45 pies. La pila se encuentra dividida en celdas que se encuentra organizadas verticalmente e indexadas por niveles. Cada celda se divide en 2 ranuras que separan la celda en popa y proa. Adicionalmente, algunas ranuras de las celdas posee un enchufe que proporciona energía eléctrica a contenedores que necesitan refrigeración (ver Figura 2.1).



Figura 1.1: Imagen de un contenedor.

En la actualidad, existen diversas maneras de planear la estiba de contenedores en estos tipos de buques. Estos planes están dados por herramientas gráficas y recientemente, algoritmos de optimización que pretenden encontrar la mejor manera de organizar los contenedores en los buques, teniendo en cuenta factores como espacio, tiempo y seguridad en su transporte.

En este proyecto, se pretende ampliar la planeación de estiba por medio de mo-



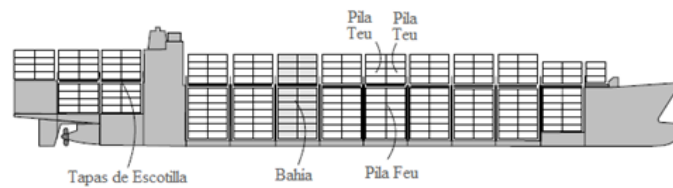


Figura 1.2: Imagen de un buque portacontenedores. [7]

delos de algoritmos de optimización a la carga localizada sobre la cubierta de los buques portacontenedores. Lo anterior, con el fin de mejorar el modelo presentado por Delgado et al(2012) [7] que hace parte como la segunda fase de un planteamiento propuesto para estibar contenedores en un buque.

Es decir, el enfoque utilizado para resolver el problema de estiba consta de dos partes. La primera parte es un modelo de programación entera para la asignación de grupos de contenedores en las áreas de almacenamiento de la embarcación a través de múltiples puertos [13] y la segunda parte se enfoca en un modelo de restricciones para estibar contenedores de forma individual en las locaciones de un buque y así poder estibar el buque en su totalidad [7], ésto es posible gracias a que la primera parte ya ha descompuesto el problema (ver Figura 3).

## 1.2. Objetivos

### 1.2.1. Objetivo general

Extender y refinar el modelo computacional presentado por Delgado et al (2012) [7].

### 1.2.2. Objetivos específicos

- Extender el modelo computacional de estibación teniendo en cuenta los siguientes aspectos:
  1. No provocar la desestabilización del buque.
  2. Contemplar las características principales del trincaje que se realiza en los contenedores.
  3. Minimización del número de contenedores que no se cargan en el buque.
- Implementación del modelo en GECODE.[9]
- Ejecución y documentación de un plan de pruebas del nuevo modelo computacional para estibación de contenedores.

### 1.2.3. Productos esperados

Objetivos Específicos	Producto(s) Esperados
Extender modelo computacional.	Modelo extendido que permita nuevas formas de estibación, intentando implementar características deseables, teniendo en cuenta el trincaje y la no desestabilización del buque.
Implementar modelo en GECODE.	Código que permita mostrar y corroborar el funcionamiento del modelo realizado.
Plan de pruebas.	Desarrollo y documentación de las pruebas a realizar del nuevo modelo con datos reales.

## 1.3. Justificación

El refinamiento y optimización de modelos encierra varios aspectos interesantes, la idea de representar de manera computacional una buena manera de estibar en los barcos portacontenedores es una buena perspectiva desde varios puntos de vista. El poder modelar una herramienta capaz de ordenar contenedores con ciertas reglas o restricciones, hace posible que los partícipes encargados del proceso tengan una visión más amplia de conocimiento, que les puede otorgar una gran ayuda, al momento incurrir en la acción de estibar contenedores.

### 1.3.1. Justificación económica

La industria del transporte marítimo de mercancías posee un gran impacto en la economía mundial, ya que la mayoría de las mercancías es transportada en barcos portacontenedores. Por lo anterior, se puede afirmar que aunque las mejoras en la optimización del proceso de estiba sean pequeñas éstas representan un gran impacto a nivel económico; la generación de planes óptimos de estiba representa grandes ganancias económicas tanto para los barcos como para los puertos.

Por una parte, tenemos que la generación de un buen plan de estiba permite a los buques una menor estadía en el puerto en el cual se encuentre anclado, debido a que el tiempo de estadía de un barco en el puerto es un cargo económico que el barco debe pagar; la minimización de este tiempo reduce los costos de cargar y descargar contenedores en el puerto. Adicionalmente, la reducción del tiempo en el puerto se puede utilizar en la disminución de la velocidad del barco para navegar en alta mar de un puerto a otro, lo que rebaja considerablemente la cantidad de combustible que éste debe utilizar.

Por otra parte, los buenos planes de estiba permiten a los puertos atender una mayor cantidad de buques en determinados periodos de tiempo y atender varios

buques es más satisfactorio económicamente que cobrar la estadía de unos cuantos buques.

### **1.3.2. Justificación social**

El impacto social estaría enfocado hacia la comunidad que se desempeña o estudia en el área relacionada con las actividades de estibación en los puertos costeros. Las personas en éste mercado obtendrían la posibilidad de encontrarse con un modelo y una herramienta que los ayudará a generar una idea más general al problema de ordenar contenedores tanto en el buque como en el puerto. Además, el proyecto también posee impactos ecológicos, ya que la posibilidad de que los barcos naveguen más lentamente en alta mar permite un consumo menor de combustible y por ende, una menor producción de  $CO_2$ .

### **1.3.3. Justificación académica**

Se pretende con este proyecto la extensión y el refinamiento de un modelo de estibación, lo que permitirá a personas que se desenvuelvan en estos ámbitos, estudiar y analizar los procesos de ordenamiento de contenedores en los buques y en los puertos. Como consecuencia, estas personas podrán entender y responder a las exigencias que esta área impone a la hora de incursionar en este mercado. El nuevo modelo estará en la capacidad de proporcionar los conocimientos claves del mercado costero.



## Capítulo 2

---

# Marco Teórico

---

### 2.1. Proceso de estiba

Se define como estiba a la técnica de colocar la carga a bordo para ser transportada con un máximo de seguridad para el buque y su tripulación, ocupando el mínimo espacio posible, evitando averías en la misma y reduciendo al mínimo las demoras en el puerto de descarga.[16]

El proceso de estiba se fundamenta en el hecho de que los buques portacontenedores viajan por el mundo de puerto en puerto cargando y descargando contenedores en cada puerto costero en el que atracan.

#### 2.1.1. Objetivos del proceso de estiba

En el proceso de estiba se debe tener en cuenta las siguientes metas u objetivos.

- La seguridad del buque y sus tripulantes.
- La minimización en el tiempo de carga y descarga de contenedores en el buque.
- Cargar la mayor cantidad de contenedores al buque.
- La estabilidad del buque.
- La integridad de la carga transportada.
- El menor impacto ecológico.

#### 2.1.2. Procesos de estibación en buques portacontenedores

Existen dos procedimientos bien diferenciados en la estiba de contenedores en los buques portacontenedores, la información sobre éstos tipos de procedimiento fue obtenida de un documento en la Web [6]:

### Mediante guías celulares en el buque

Las guías son un conjunto de raíles metálicos que se ubican de manera vertical y que forman celdas del tamaño en planta del contenedor, 20 x 8 pies o de 40 x 8 pies, estas guías permiten apilar en cada celda, sin necesidad de realizar un trincado sobre los contenedores. Adicionalmente, La limitación en altura la establece la resistencia del contenedor más bajo, a la carga de todos los contenedores apilados sobre él. Este es el procedimiento de estiba en bodegas de los buques dedicados exclusivamente al transporte de contenedores; para la estiba sobre cubierta y sobre tapas de escotilla se han buscado soluciones similares, e incluso se han construido buques con bodegas abiertas, "hatchless", para obviar el problema.

### Sin guías celulares

En este caso los contenedores se colocan agrupados en bloques, trincados para garantizar la resistencia de cada bloque a los movimientos del buque. Este procedimiento de estiba es mucho más laborioso que el de las guías celulares, y es típico de los buques polivalentes que transportan contenedores, éste procedimiento se ha usado frecuentemente sobre las escotillas en los porta-contenedores celulares.

(ALEONCM - ESPECIFICAR MAS EL PROCESO DE ESTIBA Y TRICAJE)

## 2.2. CCP - Programación por restricciones concurrentes

La programación por restricciones concurrentes es un formalismo bien construido para trabajar con procesos que interactúan entre sí, donde la comunicación está dada por unas variables compartidas y su interacción es tratada por restricciones sobre estas variables, en la tesis de V.Saraswat, este propuso La programación por restricciones concurrente como un modelo de concurrencia en la comunicación a través de variables compartidas y algunas ideas de la lógica.[?][?]

### 2.2.1. Modelo CCP

Un sistema concurrente se especifica mediante el modelo CCP en términos de restricciones sobre las variables del sistema. Una restricción es una fórmula en lógica de primer orden que representa información parcial acerca de los valores de las variables. Por ejemplo, para un sistema con variables  $x$  y  $y$  y cuyos dominio son los números naturales, la restricción  $x + y > 16$  especifica valores posibles para  $x$  y  $y$  (aquellos que satisfacen la inecuación). El modelo CCP está parametrizado con un sistema de restricciones, el cual especifica las restricciones pertinentes para el tipo de sistema en consideración, y una relación de derivación (entailment) entre las restricciones (por ejemplo,  $x + y > 16 \neq x + y > 0$ ). Durante una computación en CCP, el estado del sistema es especificado por una entidad llamada el almacén (store) en donde reside la información acerca de las variables del sistema. El almacén es representado como

una restricción, y así se puede proveer sólo información parcial acerca de las variables. Conceptualmente, el almacén en CCP es el medio (medium) por el cual los agentes interactúan entre sí. Un proceso en CCP puede actualizar el estado del sistema solo añadiendo (o diciendo - telling) información al almacén. Esto es representado como la conjunción lógica del almacén, el cual representa el estado previo, y la restricción que está siendo añadida. Por tanto, al actualizar no se cambian los valores de las variables como tal, sino que restringe algunos de los valores previamente posibles. Además, los procesos en CCP pueden sincronizarse al consultar (o preguntar - asking) información del almacén. Las preguntas están bloqueadas hasta que haya suficiente información en el almacén para derivar (entail) (responder afirmativamente) la consulta, es decir, la operación de pregunta determina si la restricción que representa el almacén deriva la consulta. Una computación en CCP termina cuando llegue a un punto, llamado de descanso - resting o inactivo - quiescent, en el cual no se puede agregar más información al almacén. La salida de la computación está definida como el punto final del almacén, también llamado almacén inactivo - quiescent store.

### 2.2.2. Ejemplo CCP

Para hacer la descripción del modelo CCP más clara, considere el escenario ilustrado en la figura. Se tienen cuatro agentes (o procesos) que desean interactuar a través del medio (inicialmente vacío). Nombrándolos, empezando desde el proceso en la esquina izquierda-arriba y siguiendo en el sentido de las manecillas del reloj, A1, A2, A3 y A4 respectivamente. Suponga que se han programado para ejecutar en el mismo orden en el que fueron nombrados. De esta manera A1 actúa primero y le dice a los demás a través del medio que la temperatura es mayor a 42 grados pero sin especificar un valor exacto. En otras palabras A1 da a los otros información parcial acerca de la temperatura. Esto hace que se añada la restricción "temperatura > 42" al almacén previamente vacío. Ahora A2 pregunta si la temperatura es exactamente 50 grados, y si es así el desea ejecutar un proceso P. Sin embargo, de la información actual en el almacén no se puede determinar cuál es el valor exacto de la temperatura. El agente A2 queda entonces bloqueado así como el agente A3, ya que tampoco se puede deducir si la temperatura está entre 0 y 100 grados (Figura 2.1).

El turno es ahora para A4, el cual dice que la temperatura es menor que 70 grados. El almacén se convierte en "temperatura > 42  $\wedge$  temperatura < 70". Ahora el proceso A3 puede ejecutar Q ya que su consulta puede ser deducida por medio de la información en el almacén. El otro agente que pregunta A2 está condenado a estar bloqueado por siempre, a menos que Q añada suficiente información al almacén de tal forma que la consulta pueda ser respondida.

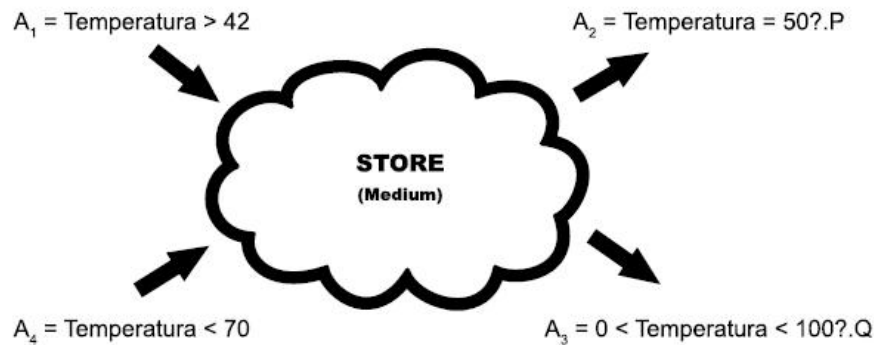


Figura 2.1: escenario CCP

## 2.3. C++

Es un lenguaje de programación que se cataloga como una evolución del lenguaje C. El lenguaje fue creado en el año 1980 por Bjarne Stroustrup y están son algunas de las características (ALEONCM - REFERENCIA):

- Lenguaje compilado: El lenguaje C++ compila directamente el código nativo sobre la maquina. Lo anterior, permite que sea uno de los lenguajes mas rápidos del mundo.
- Lenguaje abierto estandarizado por la ISO: Desde el año 1998 el lenguaje C++ fue estandarizado por un comité de la ISO.
- Lenguaje inseguro inflexible de tipos: Se espera que la persona que maneje el lenguaje sepa lo que esta haciendo. Por lo anterior, el lenguaje brinda una gran cantidad de controles como resultado.
- Varios paradigmas: El lenguaje permite trabajar con diferentes tipos de paradigmas tales como el procedimental, el genérico y el orientado a objetos.
- Portabilidad: Éste lenguaje tiene un amplio rango de compiladores que pueden corren sobre diferentes tipos de plataformas computacionales.
- Soporte a tipo de datos estáticos y dinámicos: Permite el control de tipos de datos ya sea en tiempo de compilación o en tiempo de ejecución.

## 2.4. Gecode

Gecode es un conjunto de herramientas para el desarrollo de sistemas y aplicaciones basados en restricciones. Gecode proporciona un solucionador de restricciones



con el más moderno desempeño, mientras es ser modular y extensible. El lenguaje de programación goza de las siguientes características (ALEONCM):

- Abierto: Gecode es totalmente abierto para programadores. Es fácil de acoplar con otros sistemas. Éste permite crear nuevas restricciones, nuevas estrategias de distribución y nuevos motores de búsqueda.
- Exhaustivo: Comprende un grupo de herramientas exhaustivas como restricciones sobre enteros, booleanos, conjuntos y flotantes; Modelamiento de capas por medio de C++; Implementación de avanzadas heurísticas; entre otras.
- Eficiencia: Gecode ofrece un excelente rendimiento con respecto al tiempo de ejecución y el uso de memoria.
- Libre: Es un software libre con licencia de la <Instituto Tecnológico de Massachusetts (MIT )>.
- Documentado: Contiene una documentación completa en su sitio web. (ALEONCM - REFERENCIA)
- paralelo: Tiene la capacidad de explotar múltiples núcleos.



## Capítulo 3

---

# Estado de Arte

---

Los trabajos académicos previamente realizados en el campo de planeación de estiba de contenedores, se centran en algoritmos de optimización acerca de la planificación de ranuras en los buques portacontenedores. Sin embargo, la inaccesibilidad a la información del problema de estibación ha desafiado los diferentes enfoques de solución, hasta ahora propuestos. Estos trabajos pueden ser divididos en dos grupos principales: métodos basados en un único modelo del problema y métodos basados en la descomposición del problema en varios modelos de optimización. Entre estos dos enfoques, el segundo ha sido más exitoso, aunque la mayoría de la literatura se centra en métodos heurísticos, usando un único modelo.

### 3.1. Métodos basados en modelos únicos

Dentro de los métodos basados en un único modelo, Avriel et al (1998)[4] presentan un modelo 0-1 IP (Integer Programming) y un procedimiento de heurística suspensiva para la estiba de buques, la cual se restringe a una sola bahía del buque y resuelve el problema de estiba de contenedores de un solo tamaño, sin tomar en cuenta la estabilidad del buque. Esta heurística apunta a minimizar la sobre estiba en un problema multi-puerto, donde solo la carga de contenedores es permitida.

Dubrosky & Penn (2002)[8] presentaron un algoritmo genético bajo los mismos supuestos de Avriel et al. Sin embargo, su propuesta fue suficientemente flexible para incluir nuevas restricciones y como resultado, instancias aleatorias de problemas con buques de 500 y 1000 TEUs, fueron resueltos para 10 puertos, en 30 minutos.

Ambrosio & Sciomanchen (1998) [2] presentaron un modelo de programación por restricciones optimizado con una estructura de ramificación y poda (branch & bound). Aunque este modelo tuvo por objetivo estibar un buque completo, restricciones de estabilidad entre bahías pudieron ser obviadas con el fin de asemejarse al problema de planeación de ranuras. Este método considera contenedores de 20 y 40

pulgadas, pero no tiene en cuenta los contenedores refrigerados, ni los tipo cúbicos. Su principal objetivo es minimizar la sobre estiba y el consumo, maximizando el número de contenedores cargados.

Otros trabajos sobre métodos basados en un único modelo, fueron propuestos por Aslidis (1984) [3], Botter & Brinati (1992)[5], Sciomachen & Tanfani (2003)[14] y Li et al (2008)[11], en donde se presentan modelos complejos que incluyen varias restricciones propuestas en el problema maestro de planeación. Aslidis introdujo heurísticas de apilamiento para minimizar sobre estiba, mientras Botter, Brinati y Li et al presentaron modelos 0-1 IP de programación entera. Adicionalmente, Botter y Brinati presentaron dos nuevas heurísticas para estibar contenedores debido a que su modelo IP no era escalable en instancias de la vida real. Sciomachen y Tanfani introdujeron un enfoque heurístico basado en problemas 3D de embalaje, en donde a diferencia de los demás métodos mencionados, se tuvo en cuenta los contenedores tipo cúbicos y no sólo los contenedores de 20 y 40 pulgadas.

### **3.2. Métodos basados en la descomposición del problema en varios modelos de optimización**

Por otra parte, dentro de los métodos basados en la descomposición del problema en varios modelos de optimización para la planeación de estiba de buques portacontenedores, Wilson & Roach (2000) [17], introdujeron la descomposición basada en bloques. Estos autores dividieron cada bahía de los buques en bloques, los que estaban distribuidos sobre y debajo de la cubierta y a menudo seguían el patrón de las tapas de escotilla. En resumen, Wilson y Roach propusieron un algoritmo tabú de búsqueda para solucionar una versión de la planeación de estiba, que incluía ranuras de refrigeración, restricciones de tamaño de los contenedores, minimización de sobre estiba y restricciones de pilas con el mismo puerto de descarga. Sin ninguna evidencia de peso, aparte de resultados experimentales, los autores afirmaron encontrar soluciones óptimas cercanas y de rápida computación a instancias de problemas para buques de 688 TEUs. Kang & King (2002)[10] también adoptaron el concepto de descomposición en bloques y describieron un método de enumeración para resolver una versión simple de planeación de ranuras, donde sólo es tenido en cuenta la minimización de sobre estiba, la violación de estabilidad y el ordenamiento de contenedores de 40 pulgadas después que los pesos son considerados. Tal como Wilson y Roach, ninguna evaluación experimental independiente del algoritmo fue proporcionada.

Ambrosino et al (2006)[1] describen un modelo 0-1 IP para estibar óptimamente subconjuntos de bahías de contenedores, en buques con el mismo puerto de descarga. Este modelo minimiza el tiempo de estiba de contenedores de 20 y 40 pulgadas, y éstos son ordenados de acuerdo a su peso en cada pila. En la sección experimental, planes completos de estiba para buques de 198 a 2124 TEUs fueron generados, en donde los tamaños máximos de las bahías fueron de 20 y 120 TEUs, respectivamente. No

fue dado tiempo computacional para la solución de estos sub problemas. En trabajos posteriores, Ambrosino et al, presentan una heurística constructiva para solucionar los mismos sub problemas como uno único. Usando esta heurística, son capaces de estibar un buque de 5632 TEUs, en donde se usan 11.8 segundos en promedio para estibar todas las bahías, pero la disposición física del buque no es descrita en detalle.

Otros autores como Zhang et al (2005)[15] y Yoke et al (2009)[12] presentaron enfoques multi-fases donde los problemas resueltos durante la fase de planeación de ranuras, no son independientes unos de los otros.

Por último, Pacino et al(2011)[13] presentan un modelo de programación entera para la estiba de contenedores en áreas de estibación de barco teniendo en cuenta múltiples puertos y Delgado et al (2012)[7] presentan un modelo de programación por restricciones para una óptima y veloz estiba de bahías en barcos contenedores. Este modelo se centra en el problema de estiba de contenedores ubicados por debajo de la cubierta.

### **3.3. Descripción del modelo en restricciones de Delgado et al (2012)[7].**

En este proyecto la idea es mejorar lo propuesto por Delgado et al (2012)[7] que solo se centra en la estiba de contenedores por debajo de cubierta. Es decir, el proyecto a presentar en éste documento se centra en estibar contenedores por encima de cubierta ya que el modelo a extender solo estiba contenedores por debajo de cubierta. Por lo tanto, se presentará una breve descripción del modelo general de restricciones presentado por Delgado et al (2012)[7].

(ALEONCM - FALTA EXPLICAR LAS FASES)

El modelo de restricciones cuenta con un conjunto de índices y de constantes utilizadas para la resolución del problema (Ver Cuadro 3.1).

Para facilitar el modelamiento de restricciones y objetivos del problema se adicionan unas variables auxiliares (Ver figura 3.2).

Adicionalmente, se tienen un conjunto de variables de decisión que representan un plan de estibación para un conjunto de contenedores que puede estibarse. Las variables se separan en dos subconjuntos. El primer subconjunto indica por cada variable de decisión un contenedor que se puede estibar, donde el dominio de éstas variables es definido por las ranuras o espacios. El segundo subconjunto indica por cada variable de decisión una ranura, donde el dominio de éstas variables es definido por los contenedores a estibar. Para utilizar el esquema de dos subconjuntos es necesario que los contenedores sean iguales a las ranuras, por lo cual, se utilizan unos contenedores virtuales para llenar las ranuras faltantes.

Las restricciones de (1) a (14) son reglas que se deben cumplir en el proceso de estiba (Ver Cuadro 3.3).

Indices y constantes	Descripción
$Stack$	Conjunto de índices de pilas.
$Slots$	Conjunto de índices de bahías para los contenedores.
$Cont$	Conjunto de índices de contenedores.
$Slot^{\{A,F\}}$	Conjunto de espacios de popa y proa.
$Slot_k$	Conjunto de espacios por pila k.
$Slot_k^{\{A,F\}}$	Conjunto de espacios de popa y proa por pila k.
$Slot^{\{R,\neg R\}}$	Conjunto de espacios refrigerados y no refrigerados.
$Slot^{\neg RC}$	Conjunto de celdas no refrigeradas.
$Slot^{\{20,40\}}$	Conjunto de espacios para contenedores de 20 y 40 pies.
$Cont^{\{V,L\}}$	Contenedores virtuales (V) y cargados (L).
$Cont^{\{20,40\}}$	Índice del contenedor de 20 y 40 pies.
$Cont^{40\{A,F\}}$	Contenedor de 40 pies separados por proa y popa.
$Cont^{R\{20,40\}}$	Índice del contenedor refrigerados de 20 y 40 pies.
$Cont^{\neg R}$	Conjunto de contenedores no refrigerados.
$Weight_i$	Peso del contenedor i.
$POD_i$	Puerto de descarga del contenedor i.
$Length_i$	Largo del contenedor i.
$Height_i$	Altura del contenedor i.
$Cont^{P=p}$	Número de contenedores con puerto de descarga p.
$Cont^{\{W=w,H=h\}}$	Número de contenedores con peso w y alto h.
$Cont^{\{NC,HC\}}$	Número de contenedores normales (NC) y número de contenedores cubicos (HC).
$Stack_i^{\{W,h\}}$	Peso y altura limite de la pila i.
$Classes$	Conjunto de clases de pila.
$Class^i$	Conjunto de pilas de clase i.

Cuadro 3.1: Tabla de constantes.

la restricción (1) canaliza dos subconjuntos de variables de decisión. La (2) garantiza para un contenedor de 40 pies que la parte popa y proa se encuentren en la misma celda. Las restricciones (3), (4), (5) y (6) tienen como objetivo enlazar las variables de ranura con las variables auxiliares de longitud, altura, peso y el puerto de descarga del contenedor estibado para una determinada ranura. La restricción (7) es para asignar contenedores cargados en su ranura predefinida. La (8) es una restricción regular para la popa y proa de cada pila. La restricción (9) y (10) modela la estiba de contenedores refrigerados. La (11) y (12) restringe los dominios de ranuras que sólo tienen capacidad de contenedores de 20 o 40 pies. La (13) limita la altura de cada pila en la ubicación es delimitada. Por último, la restricción (14) restringe el peso de todos los contenedores estibados en una pila para que estén dentro de los límites.

En el modelo de restricciones se calculan un conjunto de valores, con el fin de

Variables de decisión	Descripción
$C = \{C_1, \dots, C Cont \}$	$C_i \in Slots$ , índice de la espacios en el $Cont_i$ .
$S = \{S_1, \dots, S Slots \}$	$S_i \in Cont$ , índice del contenedor en el $Slot_i$ .
$L = \{L_1, \dots, L Slots \}$	$L_i \in Length$ , largo del contenedor estibado en el $Slot_i$ .
$H = \{H_1, \dots, H Slots \}$	$H_i \in Height$ , alto del contenedor estibado en el $Slot_i$ .
$W = \{W_1, \dots, W Slots \}$	$W_i \in Weight$ , peso del contenedor estibado en el $Slot_i$ .
$P = \{P_1, \dots, P Slots \}$	$P_i \in POD$ , puerto de descarga del contenedor estibado en el $Slot_i$ .
$HS = \{HS_1, \dots, HS Stack \}$	$HS_i \in Stack$ , Limite de altura por pila.
$O^V \in \{0, \dots,  Cont \}$	Número de contenedores sobre-estibados.
$O^U \in \{0, \dots,  Stack \}$	Número de pilas usadas.
$O^P \in \{0, \dots,  Stack  POD \}$	Número de contenedores diferentes por cada pila.
$O^R \in \{0, \dots,  Slot^R \}$	Número de contenedores no refrigerados estibados en espacios refrigerados.
$O \in \mathbb{N}$	Variable utilizada como costo total de estiba.
$C^V \subset C$	Contenedores virtuales.
$S_i^E \subset S$	espacios con las mismas características en la pila 1.

Cuadro 3.2: Tabla de variables de decisión.

Restricciones	
$channeling(C, S)$	(1)
$c_{fore(i)} = c_i + 1$	$\forall i \in \{1, \dots,  Cont^{40A} \}$ (2)
$element(S_i, L_i, Length)$	$\forall i \in Slots$ (3)
$element(S_i, H_i, Height)$	$\forall i \in Slots$ (4)
$element(S_i, W_i, Weight)$	$\forall i \in Slots$ (5)
$element(S_i, P_i, POD)$	$\forall i \in Slots$ (6)
$S_{pos(j)} = i$	$\forall i \in Cont^L$ (7)
$regular(Length_i^R)$	$\forall \pi \in \{A, F\}, i \in Stacks$ (8)
$S_i \notin Cont^{20R}$	$\forall i \in Slots^R$ (9)
$S_i \notin Cont^{40R}$	$\forall i \in Slots^{RC}$ (10)
$S_i \in Cont^{20}$	$\forall i \in Slots^{20}$ (11)
$S_i \in Cont^{40}$	$\forall i \in Slots^{40}$ (12)
$\sum_{j \in Slots_i} h_j \leq hs_i$	$\forall \pi \in \{A, F\}, i \in$ (13)
$\sum_{j \in Slots_i} w_j \leq Stack_i^j$	$\forall i \in Stacks$ (14)

Cuadro 3.3: Conjunto de restricciones

representar unos objetivos, estos son:

1. Calcular la cantidad total de re-manipulaciones (movimientos realizados para estibar un contenedor en la carga o descarga del mismo).

$$o^v = \sum_{i \in Slots^A} ov(i)$$

2. Encontrar la sumatoria de ranuras utilizadas en una pila y calcular la suma de los resultados obtenidos por cada pila.

$$o^u = \sum_{k \in Stacks} \left( \sum_{j \in Slots_k} p_j > 0 \right)$$

3. Calcular el número de contenedores estibados que se descargan en diferentes puertos en cada pila.

$$o^p = \sum_{k \in Stacks} \left( \sum_{\rho \in POD} \sum_{j \in Slots_k} (p_j = \rho) \right) > 0$$

4. Contar el número de contenedores no refrigerados en ranuras refrigeradas.

$$o^r = \sum_{i \in Slots^R} (s_i \in Cont^{\neg R})$$

En el modelo se pretende minimizar una función de costo. La función de costo es la suma de los valores que representan los objetivos multiplicados con un respectivo peso de costo.

$$o = 100o^v + 20o^p + 10o^u + 5o^r$$



## Capítulo 4

# Introducción a Modelos

En ésta sección se pretende dar una introducción a los dos nuevos modelos creados en ésta investigación. En los dos nuevos modelos para gestionar con más facilidad los contenedores de 40 pies; se han dividido en 2 contenedores de 20 pies, que irán ubicados en la proa y la popa de la misma celda. Adicionalmente, las bahías en el buque son indexadas de popa a proa y de abajo hacia arriba en la pila y continúan en las demás pilas de izquierda a derecha en una locación del barco (Figura 4.1 y 4.2).

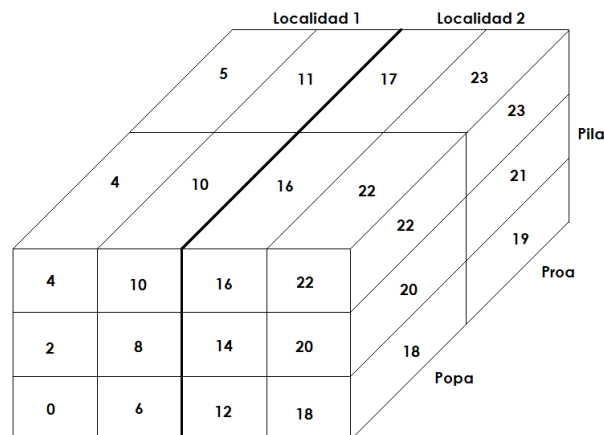


Figura 4.1: Índices de espacios (Slots)

Para obtener resultados factibles al problema de estibar contenedores se utiliza en los dos modelos restricciones que permiten imponer reglas al proceso de estiba. Este modelamiento por restricciones requiere básicamente de 3 pasos que son la definición de las variables de decisión y sus respectivos dominios, restricciones pertinentes y la estrategia de distribución.

A continuación, se muestra unas abstracciones de restricciones que serán utiliza-

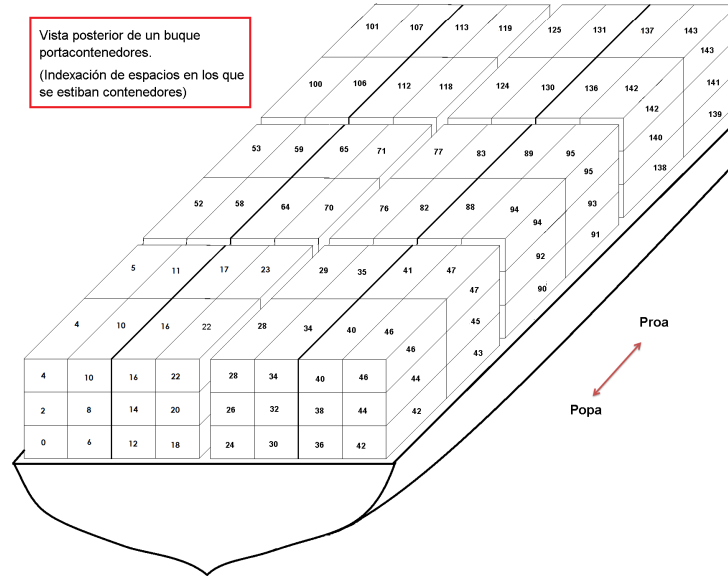


Figura 4.2: Indexación de espacios en el buque

das en los dos modelos.

#### 4.1. Abstracciones de restricciones

Para modelar algunas de las restricciones se hace uso de las siguientes abstracciones:

##### Restricciones elementales

Estas restricciones tienen la siguiente forma:

$$element(X, Y, C) = \{(e, f) | e \in D(X), f \in D(Y), f = C_e\}$$

Donde  $X$  es una variable entera,  $Y$  es una variable con dominio finito y  $C = \{C_1, \dots, C_n\}$  un conjunto de constantes. La restricción implica que  $Y$  es igual a la  $X$ -ésima constante de  $C$ .

##### Restricciones regulares

Este tipo de restricciones tienen la siguiente forma:

$$regular(X, M) = \{(d_1, \dots, d_n) | \forall i. d_i \in D(X_i), d_1 \dots d_n \in L(M)\}$$

Donde  $X$  es un conjunto de variables con dominio  $D(X_i)$  tal que  $1 < i < n$  y  $M$  es una expresión regular que reconoce  $L(M)$ .

### Restricciones Max

Este tipo de restricción tiene la siguiente forma:

$$\max(H, I, J) = \{(e, f, g) | e \in D(H), f \in D(I), g \in D(J), e > f \Rightarrow g = e, f \geq e \Rightarrow g = f\}$$

Donde la restricción implica que la variable  $J$  sera el valor más grande entre  $H$  e  $I$ .

### Restricciones Channel

Este tipo de restricción se puede expresar de las dos maneras:

- La primera abstracción es cuando  $H$  es una variable no escalar e  $I$  es una variable escalar. Esta abstracción tiene la siguiente forma:

$$\text{channel}(H, I) = \{(e, f) | e \in D(H), f \in D(I), e = f\}$$

Esta restricción se utiliza para realizar un cambio de una variable escalar a una variable no escalar.

- La segunda abstracción es cuando  $X$  y  $Y$  son conjuntos de variables. Esta abstracción se presenta formalmente de la siguiente forma:

$$\text{channel}(X, Y) = \{(e_1, \dots, e_n, f_1, \dots, f_n) | \text{forall } i, j. e_i \in D(X_i), f_i \in D(Y_j), e_i = j \Leftrightarrow f_j = i\}$$

Esta restricción enlaza consistentemente dos conjuntos de variables del mismo tamaño.

### Restricciones Expr

Este tipo de restricción tiene la siguiente forma:

$$R = \text{expr}(\text{Expres}) = \{h | h \in D(R), \text{Expres es una expresión sin ejecutarse}, R = \text{expres ejecutada}\}$$

Donde la restricción ejecuta la expresión  $\text{Expr}$  y el resultado lo guarda en  $R$ .

### Restricciones Ite

Este tipo de restricción se expresa formalmente así:

$$Ite(H, I, J, K) = \{(b, x, y, z) | b \in D(H), i \in D(I), j \in D(J), z \in D(K), z = b?x : y\}$$

En ésta restricción se asigna  $z = x$  si  $b$  es verdadero y se asigna  $z = y$  si  $b$  es falso.

### Restricciones Count

Este tipo de restricción tiene la siguiente forma:

$$Count(X, Y, R, Z) = \#\{i \in \{0, \dots, |X| - 1\} \mid X_i \in Y\} \sim_R Z.$$

Donde ésta restricción cuenta la cantidad de elementos  $X$  que concuerden con los elementos de  $Y$  y ejecuta una relación  $R$ , que puede ser ( $<$ ,  $>$ ,  $=$ ,  $\neq$ , *otras*), con respecto a la variable  $Z$ .

### Restricciones atmost

Este tipo de restricción tiene la siguiente forma:

$$atmost(X, Y, Z) = \#\{i \in \{0, \dots, |X| - 1\} \mid X_i = Y\} \leq Z.$$

Donde ésta restricción cuenta la cantidad de elementos  $X$  que concuerden con el valor  $Y$  y ejecuta una relación  $\leq$  con respecto al valor  $Z$ .

## 4.2. Conjunto de constantes

Para facilitar la construcción de las restricciones y de los objetivos a alcanzar en los dos modelos se plantean un conjunto de índices y constantes. Todos los conjuntos de índices que se presentarán son un subconjunto de enteros a excepción de la constante  $GCSY$  que puede ser un dato no escalar. Cada conjunto representa características ya sea de secciones del barco o de los contenedores (Cuadro 4.1).

Sobre los dos modelos se busca encontrar el centro de gravedad por cada una de las pilas del barco y se debe obtener la diferencia entre el centro de gravedad encontrado con respecto al centro de gravedad ideal de cada pila. El centro de gravedad ideal de una pila en el eje  $Y$  se encuentra dado por la fórmula  $(Stack_i^H/4)/(Stack_i^H/maxLevel(i))$  donde  $maxLevel(i)$  es el nivel máximo de la pila  $i$ . Solo se busca la coordenada  $Y$  del centro de gravedad debido a que la coordenada sobre el eje  $x$  no brinda información relevante para la estiba y trincaje de contenedores.

Índices y constantes	Descripción
$Slots$	Conjunto de índices de espacios para los contenedores.
$Cont$	Conjunto de índices de contenedores.
$Stack$	Conjunto de índices de pilas.
$Slot_k$	Conjunto de espacios por pila k.
$Slot_k^{A,F}$	Conjunto de espacios de popa y proa por pila k.
$Slot_k^R$	Conjunto de espacios refrigerados.
$Slot^{NR}$	Conjunto de espacios no refrigerados.
$Slot^{NRC}$	Conjunto de celdas no refrigeradas.
$Cont^L$	Conjunto de contenedores cargados.
$Cont_K$	Conjunto de contenedores en la pila k.
$Cont^{\{20,40\}}$	Índice del contenedor de 20 y 40 pies.
$Cont_{A,F}^{40}$	Índice del contenedor de 40 pies separados por proa y popa.
$Cont_R^{\{20,40\}}$	Índice del contenedor refrigerados de 20 y 40 pies.
$Length_i$	Largo del contenedor i.
$Weight_i$	Peso del contenedor i.
$Height_i$	Altura del contenedor i.
$POD_i$	Puerto de descarga del contenedor i.
$Stack_i^W$	Peso limite de la pila i.
$Stack_i^H$	Altura limite de la pila i.
$Cont^V$	Índice del contenedor virtual.
$Cont^{Load}$	Conjunto de contenedores a cargar.
$GCSY$	Centro de gravedad ideal en el eje Y.
$ContNonReefer$	Conjunto de variables booleanas del tamaño $ Cont $ , donde cada elemento representa la relación $ContNonReefer_i = Cont_i \in Cont_{NR}$
$OverCont$	Conjunto de variables booleanas del tamaño $ Cont  +  Cont^{Load} $ , donde cada elemento representa la relación $OverCont_i = i <  Slots $
$ContEW_i$	Número de contenedores con igual peso i.
$ContEP_i$	Número de contenedores con igual puerto de descarga i.
$ContEH_i$	Número de contenedores con igual altura i.
$ContEL_i$	Número de contenedores con igual largo i.

Cuadro 4.1: Tabla de constantes.

Para calcular el centro de gravedad de un conjunto de contenedores agrupados en una pila es necesario proponer un sistema de coordenadas sobre cada pila. El sistema de coordenadas para cada pila empezará con el punto  $(0, 0)$  sobre el centro del contenedor ubicado en la parte inferior de la pila en la sección de la popa. La coordenada aumentará en el eje  $X$  de la popa a la proa en la pila y en el eje  $Y$  de la parte inferior a la parte superior de la pila. (Ver Figura 4.3)

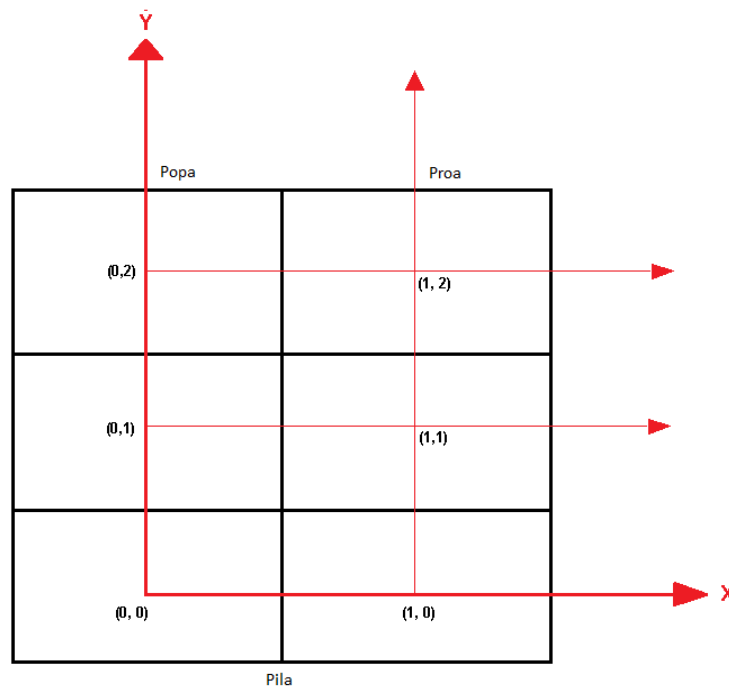


Figura 4.3: Coordenadas para cálculo de centro de gravedad

Anteriormente, se comentó que los contenedores de 40 pies estarían representados por medio de dos contenedores de 20 pies. Para evitar problemas al calcular datos que requiera el peso de los contenedores de 40 pies que ya estén estibados, la constante  $Weight_i$  del contenedor de 40 pies deberá ser dividida entre los dos contenedores de 20 pies. El contenedor de 20 pies que representan la división de la popa tendrá un peso de igual a  $Ceil(Weight_i/2)$  y el contenedor de 20 pies que representa la proa será igual a  $Weight_i - Ceil(Weight_i/2)$

Los capítulos 5 y 6 tienen como objetivo explicar a detalle los modelos realizados en ésta investigación. Para cada uno de los modelos en restricciones se explicarán las variables de decisión utilizadas y sus respectivos dominios, se explicarán las restricciones que se aplican sobre estas variables de decisión y la estrategia de distribución utilizada en cada uno de los modelos. El capítulo 7 mostrará los resultados obtenidos por cada modelo, éstos resultados se obtienen a partir de instancias reales utilizadas en la estiba de contenedores en los puertos costeros.

## Capítulo 5

---

# Modelo con contenedor virtual

---

En éste modelo se propone la idea de utilizar contenedor virtual que tiene como objetivo representar en el modelo que no se cargará un contenedor en el espacio que lo contenga. Las características como el largo, ancho, alto y puerto de descarga serán iguales a 0. Además, el índice del contenedor virtual es igual a 0 ( $Cont^V = 0$ ).

### 5.1. Variables de decisión

Sobre la variables de decisión actuarán las restricciones que se propondrán mas adelante. Adicionalmente, estas variables modelarán los posibles resultados obtenidos al final de proceso (Cuadro 5.1).

Las variables de decisión utilizadas para calcular el costo de las soluciones encontradas son utilizadas sobre el proceso de propagación. (Cuadro 5.2).

### 5.2. Restricciones del modelo

Para el modelo se plantean un conjunto de restricciones que tienen como objetivo disminuir el espació de búsqueda de posibles soluciones.

#### 5.2.1. Detalle de las restricciones

En ésta sección se presentan y se detallan el conjunto de restricciones modeladas (Cuadro 5.3 y 5.4).

La restricción (1) implica el hecho de que un contenedor no puede estar en diferentes slots a excepción del contenedor virtual. Las restricciones <elementales> que se encuentran referenciadas del (2) al (5) son para generar una asociación entre los slots y las características de longitud, altura, peso y puerto de descarga de los contenedores que se estibarán en dichos slots. La restricción (6) que también es <elemental> se

Variables de decisión	Descripción
$S = \{S_1, \dots, S Slots \}$	$S_i \in \text{Cont}$ , índice del contenedor en el $Slot_i$ .
$L = \{L_1, \dots, L Slots \}$	$L_i \in \text{Length}$ , largo del contenedor estibado en el $Slot_i$ .
$H = \{H_1, \dots, H Slots \}$	$H_i \in \text{Height}$ , alto del contenedor estibado en el $Slot_i$ .
$W = \{W_1, \dots, W Slots \}$	$W_i \in \text{Weight}$ , peso del contenedor estibado en el $Slot_i$ .
$WD = \{WD_1, \dots, WD Slots \}$	$WD_i \in \text{Weight}$ , peso del contenedor estibado en el $Slot_i$ . (Arreglo de variables no escalares).
$P = \{P_1, \dots, P Slots \}$	$P_i \in \text{POD}$ , puerto de descarga del contenedor estibado en el $Slot_i$ .
$NRSR$ $\{NRSR_1, \dots, NRSR Slots^R \}$	$NRSR_i \in \{0, 1\}$ , Indica si se carga un contenedor no refrigerado en un slot refrigerado.
$HS = \{HS_1, \dots, HS Stack \}$	$HS_i \in \text{Stack}$ , limite de altura por pila.
$CFEUA$ $\{CFEUA_1, \dots, CFEUA Slots/2 \}$	$CFEUA_i \in \{0, 1\}$ , contenedor de 40 pies estibado en la popa.
$CFEUF$ $\{CFEUF_1, \dots, CFEUF Slots/2 \}$	$CFEUF_i \in \{0, 1\}$ , contenedor de 40 pies estibado en la proa.
$OVT = \{OVT_1, \dots, OVT Slots \}$	$OVT_i \in \{0, 1\}$ , contenedor sobre-estibado en el $Slot_i$ .
$OPT = \{OPT_1, \dots, OPT Stack \}$	$OPT_i \in \text{Cont}_K$ , número de contenedores diferentes en la pila i.
$NVC = \{NVC_1, \dots, NVC Slots \}$	$NVC_i \in \{0, 1\}$ , contenedores no virtuales.
$WT = \{WT_1, \dots, WT Stack \}$	$WT_i \in \{0, \text{Stack}_i^W\}$ , peso total de los contenedores estibados en la pila i.
$GCenY$ $\{GCenY_{1,1}, \dots, GCenY Stack  Slots \}$	$GCenY_{i,s} \in \{0, \maxLevel(i)\}$ , centro de gravedad en eje Y para la pila i y el slot s.
$GCY = \{GCY_1, \dots, GCY Stack \}$	$GCY_i \in \{0, \maxLevel(i)\}$ , centro de gravedad en el eje Y.
$GCD = \{GCD_1, \dots, GCD Stack \}$	$GCD_i \in \{0, \maxLevel(i)\}$ , distancia al centro de gravedad ideal por pila.
$OUT = \{OUT_1, \dots, OUT Stack \}$	$OUT_i \in \text{Cont}^{\{20,40\}}$ , número de contenedores i de 20 y 40 pies en la pila k.
$OUTE$	$0 \leq OUTE \leq  Stack $ , número de pilas vacías.

Cuadro 5.1: Tabla de variables de decisión.



Variables de decisión	Descripción
$OV$	Número de contenedores sobre-estibados.
$OCNS$	Número de contenedores no estibados en el barco.
$OU$	Número de pilas usadas.
$OP$	Sumatoria del número de contenedores diferentes por cada pila.
$OR$	Número de contenedores no refrigerados estibados en slots refrigerados.
$OGCTD$	Sumatoria de los pesos otorgados por la distancia al centro de gravedad ideal por cada pila.
$O$	Variable utilizada como costo total de estiba.

Cuadro 5.2: Tabla de variables de decisión utilizadas para calcular el costo de la solución.

utiliza para identificar cuando un contenedor no refrigerado se estiba en un espacio refrigerado. Para poder realizar el cambio de una variable escalar, donde se guarda el peso del contenedor, a una variable no escalar se utiliza la restricción (7). Se presenta la restricción (8) para cargar los contenedores abordo del barco. Las restricciones (9) y (10) son para que la separación de los contenedores de 40 pies en uno de proa y otro de popa se estiben respectivamente en slots de proa y popa respectivamente. Se modelan las restricciones del (11) al (14) que tienen como función que los dos contenedores de 20 pies que representan a uno 40 pies queden estibados en la proa y popa de la misma celda, donde  $getCell(i)$  tiene como tarea obtener la celda del slot  $i$ . Las restricciones del (15) al (18) tienen como objetivo identificar cuando un contenedor ésta sobre-estibado. Para una mejor entendimiento sobre las restricciones de sobre-estiba, se detallará cada una de las restricción. Las restricciones (15) y (16) solo aplicará para los contenedores 40 pies y para los contenedores que se encuentre cumplan con la restricción  $overStowed40(i)$  que indica si existe  $P_i > MinP(x)$ , donde  $MinP(x)$  es el mínimo puerto de descarga de un conjunto  $x$  que contienen los puertos de descarga de los contenedores estibados en slots inferiores a  $i$  y que pertenecen a la misma pila. La restricciones (17) y (18) solo aplican para contenedores de 20 pies, éstas utilizan las restricciones  $overStowed20A(i)$  y  $overStowed20F(i)$  las cuales funcionan de la misma forma que  $overStowed40(i)$ . La diferencia es que  $overStowed20A(i)$  actúa solo sobre los contenedores estibados en la popa de una misma pila y  $overStowed20F(i)$  actúa solo sobre los contenedores estibados en la proa de una misma pila. La restricción (19) se utiliza para obtener el peso total de los contenedores estibados por pila. Para calcular el centro de gravedad de los slots de una pila en el eje  $Y$  se utiliza la restricción (20); no se calcula el centro de gravedad en eje  $x$  debido a que brinda no brinda información relevante. El calculo del centro de

Restricciones		
$atmost(S, i, 1)$	$\forall i \in Cont   i \neq 0$	(1)
$element(S_i, L_i, Length)$	$\forall i \in Slots$	(2)
$element(S_i, H_i, Height)$	$\forall i \in Slots$	(3)
$element(S_i, W_i, Weight)$	$\forall i \in Slots$	(4)
$element(S_i, P_i, POD)$	$\forall i \in Slots$	(5)
$element(S_i, NRSR_i, ContNonReefer)$	$\forall i \in Slot^R$	(6)
$channel(WD_i, W_i)$	$\forall i \in Slots$	(7)
$S_{pos(i)} = i$	$\forall i \in Cont^L$	(8)
$S_i \notin Cont_A^{40}$	$\forall i \in Slot_k^F, \forall k \in Stacks$	(9)
$S_i \notin Cont_F^{40}$	$\forall i \in Slot_k^A, \forall k \in Stacks$	(10)
$CFEUA_{getCell(i)} = expr(L_i == 40)$	$\forall i \in Slot_k^A$	(11)
$CFEUA_{getCell(i)} = 1 \Rightarrow S_{i+1} = S_i + 1$	$\forall i \in Slot_k^A$	(12)
$CFEUF_{getCell(i)} = expr(L_i == 40)$	$\forall i \in Slot_k^F$	(13)
$CFEUF_{getCell(i)} = 1 \Rightarrow S_{i-1} = S_i - 1$	$\forall i \in Slot_k^F$	(14)
$Ite(CFEUA_{getCell(i)} overStowed40(i), 1, 0, OVT_i)$	$\wedge \forall i \in Slot_k^A$	(15)
$Ite(CFEUA_{getCell(i)} overStowed40(i), 1, 0, OVT_{(i+1)})$	$\wedge \forall i \in Slot_k^A$	(16)
$Ite(!CFEUA_{getCell(i)} overStowed20A(i), 1, 0, OVT_i)$	$\wedge \forall i \in Slot_k^A$	(17)
$Ite(!CFEUA_{getCell(i)} overStowed20F(i), 1, 0, OVT_{(i+1)})$	$\wedge \forall i \in Slot_k^A$	(18)
$WT_i = \sum_{j \in Slots_i} WD_j$	$\forall i \in Stacks$	(19)
$GCenY_{i,s} = WD_s * getPosY(s)$	$\forall i \in Stacks, s \in Slots_i$	(20)
$GcY_i = (\sum GCenY_{i,s}) / (WT_i + 0,001)$	$\forall i \in Stacks, s \in Slots_i$	(21)
$PenY_i = GcY_i - GCSY$	$\forall i \in Stacks$	(22)
$Max(PenY_i, 0, GCD_i)$	$\forall i \in Stacks$	(23)
$regular(Length_i^\alpha, R)$	$\forall \alpha \in \{A, F\}, i \in Stack$	(24)
$\sum_{j \in Slots_i^\alpha} H_j \leq HS_i$	$\forall \alpha \in \{A, F\}, i \in Stacks$	(25)
$\sum_{j \in Slots_i} W_j \leq Stack_i^W$	$\forall i \in Stacks$	(26)
$W_i + W_{i+1} \geq W_{i+2} + W_{i+3} \geq \dots \geq W_{i+n} + W_{i+n+1}$	$\forall k \in Stack \text{ donde } i = Min(Slot_k^A), n = Max(Slot_k^A)$	(27)
$OPT_k = distinctPOD(Slot_k)$	$\forall k \in Stacks$	(28)
$S_i \notin Cont_R^{20}$	$\forall i \in Slot^{NR}$	(29)
$S_i \notin Cont_R^{40}$	$\forall i \in Slot^{NRC}$	(30)
$S_i \in Cont^{20}$	$\forall i \in Slot^{20}$	(31)
$S_i \in Cont^{40}$	$\forall i \in Slot^{40}$	(32)
$Count(L_i, \{20, 40\}, =, OUT_k)$	$\forall i \in Slot_K^{40}$	(33)
$Count(OUT_k, \{0\}, =, OUTE)$	$\forall k \in Stack$	(34)
$atmost(W, VW, N)$	$\forall VW \in ContEW   i \neq 0$	(35)
$atmost(P, VP, N)$	$\forall VP \in ContEP   i \neq 0$	(36)
$atmost(H, VH, N)$	$\forall VH \in ContEH   i \neq 0$	(37)
$atmost(L, VL, N)$	$\forall VL \in ContEL   i \neq 0$	(38)

Cuadro 5.3: Conjunto de restricciones

Restricciones para calcular el costo	
$OV = \sum_{i \in Slots} OVT_i$	(39)
$OCNS =  Cont^L  +  Cont^{Load}  - getCountNVC()$	(40)
$OU = getNumStack() - OUTE - getMinUseStack()$	(41)
$OP = \sum_{i \in Stacks} OPT_i$	(42)
$OR = \sum_{i \in Slots^R} NRSR_i$	(43)
$OGCTD = (\sum_{i \in Stacks} GCD_i) * 100$	(44)
$O = 1000 * OCNS + 100 * OV + 20 * OP + 10 * OU + 5 * OR + OGCTD$	(45)

Cuadro 5.4: Conjunto de restricciones

gravedad de la pila en eje  $Y$  se realiza por medio de la restricción (21). La restricción (22) se utiliza para calcular la penalización que corresponde a la distancia entre el centro de gravedad calculado para la pila en el eje  $Y$  y el centro de gravedad ideal de la pila el cual es representado por medio de  $GCSY$ . La distancia del centro de gravedad para cada pila se obtiene del máximo valor entre la penalización por pila y la constante 0, ésta distancia se obtiene por medio de la restricción (23). La restricción (24) plantea que las variables de longitud que pertenecen a una misma pila deben cumplir con la expresión regular  $R = 20*40*0^*$ . La restricción (25) especifica el limite de altura para cada pila. El limite de peso para pila es especificado por medio de la restricción (26). Se modela la restricción (27) la cual implica que los contenedores de una pila deben estar ordenados de abajo hacia arriba y de mayor a menor peso, donde  $Min(x)$  y  $Max(x)$  especifica el mínimo y máximo índice de conjunto de slot que pertenecen a una pila y que aún no se han cargado. La restricción (28) tiene como objetivo contar por cada pila la cantidad de puertos de descarga diferentes. La restricción (29) y (30) implican que no es posible estibar contenedores refrigerados en slot no refrigerados. La restricción (31) y (32) restringen el dominio de los slot de 20 y 40 pies indicando que solo se pueden cargar contenedores de 20 y 40 pies respectivamente. La restricción (33) cuenta la cantidad de contenedores de 20 y 40 pies estibados por pila. La restricción (34) se encarga de obtener la cantidad de pilas vacías. Las restricciones (35), (36), (37) y (38) tienen como objetivo limitar el número de contenedores a estibar de acuerdo a características como al peso, el puerto de descarga, la altura y el largo de los contenedores (Cuadro 5.3).

Las restricciones utilizadas para calcular el costo de la soluciones se encuentran referenciadas del (39) al (45). Se debe tener en cuenta que para éste modelo las restricciones de costo para los contenedores de 40 pies es el doble de uno de 20 pies. A continuación, se detalla cada unas de las restricciones de costo.

La restricción (39) tiene como objetivo calcular el costo de los contenedores sobre-estibados. Un contenedor se considera sobre-estibado en una pila cuando este contiene un puerto de descarga mayor a cualquiera de los puertos de descarga de los contenedores que se encuentren por debajo del mismo. La restricción (40) cuenta el costo de contenedores no estibados en el buque. El costo de contenedores no esti-

bados se obtiene del total de contenedores menos la cantidad de contenedores no virtuales estibados, obtenido mediante la función *getCountNVC()*. Para obtener la cantidad de pilas utilizadas se utiliza la restricción (41) que obtiene la cantidad de pilas y resta éste valor con la cantidad de pilas no usadas; la función *getNumStack()* es la encargada de obtener la cantidad de pilas, menos *getMinUseStack()* que es el mínimo número de pilas a utilizar dependiendo de la cantidad de contenedores estibados. La restricción (42) suma la cantidad de contenedores con puerto de descarga diferente en una misma pila. La restricción (43) suma la cantidad de contenedores no refrigerados estibados en espacios refrigerados. La restricción (44) se encarga de obtener el peso del centro de gravedad, éste calculo consiste en la suma de los centros de gravedad de cada pila multiplicado por la constante 100. Por último, la restricción (45) obtiene el costo total de la solución encontrada (Cuadro 5.4).

### 5.2.2. Propagación del modelo con respecto a la función objetivo

La propagación del modelo se realiza por medio de una función de costo que tiene en cuenta los siguientes objetivos:

- Minimización de los contenedores no estibados, variable **OCNS**. Se adicionará un costo de 1000 unidades por cada contenedor no estibado.
- Minimización de la sobre-estiba, variable **OV**. Se adicionará un costo de 100 unidades por cada contenedor sobre-estibado.
- Minimización de los puertos de descarga de los contenedores en cada pila, variable **OP**. Se agregará por cada puerto de descarga adicional de los contenedores en la pila un costo de 20 unidades.
- Minimización de las pilas vacías, variable **OU**. Se adicionará por cada nueva pila usada un costo de 10 unidades.
- Minimizar la carga de contenedores no refrigerados en celdas refrigeradas, variable **OR**. Se adicionará un costo de 5 unidades por cada contenedor no refrigerado estibado en una celda refrigerada.
- Minimización de la distancia desde el centro de gravedad ideal de una pila al centro de gravedad real de la pila estibada, variable **OGCTD**. El valor de la variable **OGCTD** se encuentra dado en centímetros.

## 5.3. Estrategia de distribución

Para éste modelo se utilizan dos estrategias de distribución. En ambas estrategias se hace uso de las variables auxiliares  $P, L, WyH$  y por último se usa la variable  $S$ .

### 5.3.1. Estrategia de distribución por pila

La estrategia de distribución por pila consiste en agrupar conjuntos de valores donde cada conjunto representa una pila y los valores representan los índice de unos espacios en el buque portacontenedores. Los índices son utilizados para poder asignar el puerto de descarga, el largo, el peso, la altura y el contenedor en un espacio en el buque. (Ver figura 5.1)

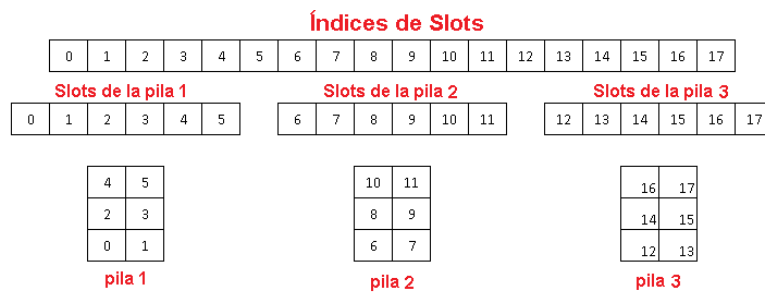


Figura 5.1: Índices por pila

La estrategia de distribución de agrupar por pilas permite fácilmente asignar valores de contenedores a una pila y la estrategia no continua con la siguiente pila hasta que la primera pila, en la que se están asignando contenedores, cumpla con todas las restricciones del modelo.

Sobre cada pila, la estrategia de distribución escoge los índices  $i$  del espacio o ranura a estibar, ésta selección toma primero los índices del nivel inferior al nivel superior de la pila y en cada nivel de la pila avanza de popa a proa. Como se menciono anteriormente, se hace uso de las variables auxiliares  $P, L, WyH$  para disminuir el espacio de búsqueda. Primero, se asignan  $P_i$  de la pila, asignando al principio el máximo valor de  $P$  para el espacio  $i$ . Segundo, se asignan las variables  $L_i$ , intentado cumplir con la restricción (24), escogiendo primero los de largo 20, luego los de 40 y por último los de largo 0. Tercero, se asignan la variables  $W_i$ , intentando cumplir con la restricción (27), dando al principio el máximo valor de  $W$  para el espacio  $i$ . Cuarto, se asignan las variables  $H_i$ , asignando el máximo valor de  $H$  para el espacio  $i$ . Por último, se intenta asignar la variable  $S$  buscando así obtener resultado factibles para la pila.

### 5.3.2. Estrategia de distribución por niveles de cada pila

La estrategia de distribución por niveles consiste en agrupar conjuntos de valores donde cada conjunto contiene de cada una de las pilas los indices de un nivel. (Ver figura 5.2)

Sobre cada nivel, se asignan las variables  $P, L, W, HyS$  de la misma forma en que se asignan para la estrategia de distribución por pila. A diferencia de la estrategia

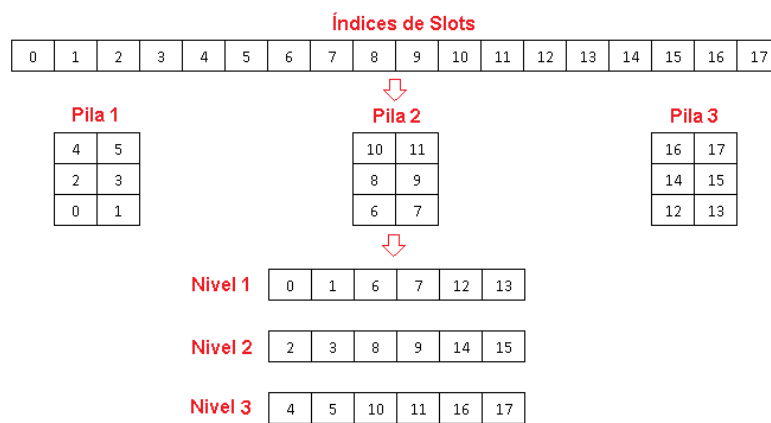


Figura 5.2: Índices por niveles

de distribución por pila, la distribución por niveles otorga la posibilidad de estibar la mayor cantidad de contenedores en un periodo mas corto.

## Capítulo 6

---

# Modelo con espacios y contenedor virtuales

---

Éste modelo propone la idea de utilizar espacios y contenedores virtuales. Los espacios virtuales se utilizarán para que los contenedores no virtuales puedan ser siempre estibados y los contenedores virtuales se utilizan para indicar que un espacio no fue estibado por un contenedor real. La cantidad de contenedores y espacios, virtuales y no virtuales, deben tener la misma cardinalidad. Adicionalmente, los contenedores virtuales tienen características como el largo, ancho, alto y puerto de descarga iguales a 0.

### 6.0.3. Variables de decisión

Sobre las variables de decisión actuarán las restricciones que se propondrán mas adelante. Adicionalmente, estas variables modelarán los posibles resultados obtenidos al final de proceso (Cuadro 6.1).

Las variables de decisión utilizadas para calcular el costo de las soluciones encontradas son utilizadas sobre el proceso de propagación. (Cuadro 6.2).

### 6.0.4. Restricciones del modelo

Para el modelo se plantean un conjunto de restricciones que tienen como objetivo disminuir el espacio de búsqueda de posibles soluciones.

#### Detalle de las restricciones

En ésta sección se presentan y se detallan el conjunto de restricciones modeladas (Cuadro 6.3 y 6.4).

Variables de decisión	Descripción
$C = \{C_1, \dots, C_{ Slots + ContLoad }\}$	$C_i \in Cont^{Load} + Slot$ , índice del espacio en el $Cont_i$ .
$S = \{S_1, \dots, S_{ Slots + ContLoad }\}$	$S_i \in Cont^{Load} + Slot$ , índice del contenedor en el $Slot_i$ .
$L = \{L_1, \dots, L_{ Slots + ContLoad }\}$	$L_i \in Length$ , largo del contenedor estibado en el $Slot_i$ .
$H = \{H_1, \dots, H_{ Slots + ContLoad }\}$	$H_i \in Height$ , alto del contenedor estibado en el $Slot_i$ .
$W = \{W_1, \dots, W_{ Slots + ContLoad }\}$	$W_i \in Weight$ , peso del contenedor estibado en el $Slot_i$ .
$WD = \{WD_1, \dots, WD_{ Slots + ContLoad }\}$	$WD_i \in Weight$ , peso del contenedor estibado en el $Slot_i$ . (Arreglo de variables no escalares).
$P = \{P_1, \dots, P_{ Slots + ContLoad }\}$	$P_i \in POD$ , puerto de descarga del contenedor estibado en el $Slot_i$ .
$NRSR = \{NRSR_1, \dots, NRSR_{ Slots^R }\}$	$NRSR_i \in \{0, 1\}$ , Indica si se carga un contenedor no refrigerado en un slot refrigerado.
$HS = \{HS_1, \dots, HS_{ Stack }\}$	$HS_i \in Stack$ , Limite de altura por pila.
$OPT = \{OPT_1, \dots, OPT_{ Stack }\}$	$OPT_i \in Cont_K$ , Número de contenedores diferentes en la pila i.
$WT = \{WT_1, \dots, WT_{ Stack }\}$	$WT_i \in \{0, Stack_i^W\}$ , Peso total de los contenedores estibados en la pila i.
$GCenY = \{GCenY_{1,1}, \dots, GCenY_{ Stack  Slots }\}$	$GCenY_{i,s} \in \{0, maxLevel(i)\}$ , Centro de gravedad en eje Y para la pila i y el slot s.
$GCD = \{GCD_1, \dots, GCD_{ Stack }\}$	$GCD_i \in \{0, maxLevel(i)\}$ , Distancia al centro de gravedad ideal por pila.
$OUT = \{OUT_1, \dots, OUT_{ Stack }\}$	$OUT_i \in Cont^{\{20,40\}}$ , Número de contenedores i de 20 y 40 pies en la pila k.
$OUTE$	$0 \leq OUTE \leq  Stack $ , Número de pilas vacías.
$VSC = \{VSC_1, \dots, VSC_{ ContLoad }\}$	$VSC_i \in \{0, 1\}$ , Indica verdadero si se ha estibado un contenedor no virtual en un espacio virtual, de lo contrario indica falso.

Cuadro 6.1: Tabla de variables de decisión.



Variables de decisión	Descripción
<i>OCNS</i>	Número de contenedores no estibados en el barco.
<i>OU</i>	Número de pilas usadas.
<i>OP</i>	Sumatoria del número de contenedores diferentes por cada pila.
<i>OR</i>	Número de contenedores no refrigerados estibados en slots refrigerados.
<i>OGCTD</i>	Sumatoria de los pesos otorgados por la distancia al centro de gravedad ideal por cada pila.
<i>O</i>	Variable utilizada como costo total de estiba.

Cuadro 6.2: Tabla de variables de decisión utilizadas para calcular el costo de la solución.

La restricción (1) implica el enlace de los contenedores con los espacios. Las restricciones <elementales> que se encuentran referenciadas del (2) al (5) son para generar una asociación entre los slots y las características de longitud, altura, peso y puerto de descarga de los contenedores que se estibarán en dichos slots. La restricción (6) que también es <elemental> se utiliza para identificar cuando un contenedor no refrigerado se estiba en un espacio refrigerado. La restricción (7) y (8) se utilizan para identificar la cantidad de contenedores estibados en espacios virtuales. Para poder realizar el cambio de una variable escalar, donde se guarda el peso del contenedor, a una variable no escalar se utiliza la restricción (9). Se presenta la restricción (10) para cargar los contenedores abordó del barco. La restricción (11) es para asegurar que los dos contenedores de 20 pies utilizados para representar uno de 40 pies queden estibados de forma continua. Las restricciones (12) y (13) son para que la separación de los contenedores de 40 pies en uno de proa y otro de popa se estiben de forma continua respectivamente. La restricción (14) se utiliza para obtener el peso total de los contenedores estibados por pila. Para calcular el centro de gravedad de los espacios de una pila en el eje  $Y$  se utiliza la restricción (15); no se calcula el centro de gravedad en eje  $x$  debido a que brinda no brinda información relevante. El calculo del centro de gravedad de la pila en eje  $Y$  se realiza por medio de la restricción (16). La restricción (17) se utiliza para calcular la penalización que corresponde a la distancia entre el centro de gravedad calculado para la pila en el eje  $Y$  y el centro de gravedad ideal de la pila el cual es representado por medio de  $GCSY$ . La distancia del centro de gravedad para cada pila se obtiene del máximo valor entre la penalización por pila y la constante 0, ésta distancia se obtiene por medio de la restricción (18). La restricción (19) plantea que las variables de longitud que pertenecen a una misma pila deben cumplir con la expresión regular  $R = 20*40*0^*$ . La restricción (20) especifica el límite de altura para cada pila. El límite de peso para pila es especificado por medio de la

Restricciones		
$channel(S, C)$		(1)
$element(S_i, L_i, Length)$	$\forall i \in Slots$	(2)
$element(S_i, H_i, Height)$	$\forall i \in Slots$	(3)
$element(S_i, W_i, Weight)$	$\forall i \in Slots$	(4)
$element(S_i, P_i, POD)$	$\forall i \in Slots$	(5)
$element(S_i, NRSR_i, ContNonReefer)$	$\forall i \in Slot^R$	(6)
$element(C_i, VSC_i, OverCont)$	$\forall i \in Slots \wedge i \notin Cont_F^{40}$	(7)
$VSC_i = 0$	$\forall i \in Cont_F^{40}$	(8)
$channel(WD_i, W_i)$	$\forall i \in Slots$	(9)
$S_{pos(i)} = i$	$\forall i \in Cont^L$	(10)
$C_{i+1} = C_i + 1$	$\forall i \in Cont_A^{40}$	(11)
$S_i \notin Cont_A^{40}$	$\forall i \in Slot_k^F, \forall k \in Stacks$	(12)
$S_i \notin Cont_F^{40}$	$\forall i \in Slot_k^A, \forall k \in Stacks$	(13)
$WT_i = \sum_{j \in Slots_i} WD_j$	$\forall i \in Stacks$	(14)
$GCenY_{i,s} = WD_s * getPosY(s)$	$\forall i \in Stacks, s \in Slots_i$	(15)
$GcY_i = (\sum GCenY_{i,s}) / (WT_i + 0,001)$	$\forall i \in Stacks, s \in Slots_i$	(16)
$PenY_i = GcY_i - GCSY$	$\forall i \in Stacks$	(17)
$Max(PenY_i, 0, GCD_i)$	$\forall i \in Stacks$	(18)
$regular(Length_i^\alpha, R)$	$\forall \alpha \in \{A, F\}, i \in Stack$	(19)
$\sum_{j \in Slots_i^\alpha} H_j \leq HS_i$	$\forall \alpha \in \{A, F\}, i \in Stacks$	(20)
$\sum_{j \in Slots_i} W_j \leq Stack_i^W$	$\forall i \in Stacks$	(21)
$W_i + W_{i+1} \geq W_{i+2} + W_{i+3} \geq \dots \geq W_{i+n} + W_{i+n+1}$	$\forall k \in Stack$ donde $i = Min(Slot_k^A), n =$ $Max(Slot_k^A) - 1$	(22)
$P_i \geq P_{i+2} \geq \dots \geq P_{i+n}$	$\forall k \in Stack$ donde $i = Min(Slot_k^A), n =$ $Max(Slot_k^A)$	(23)
$P_i \geq P_{i+2} \geq \dots \geq P_{i+n}$	$\forall k \in Stack$ donde $i = Min(Slot_k^F), n =$ $Max(Slot_k^F)$	(24)
$OPT_k = distinctPOD(Slot_k)$	$\forall k \in Stacks$	(25)
$S_i \notin Cont_R^{20}$	$\forall i \in Slot^{NR}$	(26)
$S_i \notin Cont_R^{40}$	$\forall i \in Slot^{NRC}$	(27)
$S_i \in Cont^{20}$	$\forall i \in Slot^{20}$	(28)
$S_i \in Cont^{40}$	$\forall i \in Slot^{40}$	(29)
$Count(L_i, \{20, 40\}, =, OUT_k)$	$\forall i \in Slot_K^{40}$	(30)
$Count(OUT_k, \{0\}, =, OUTE)$	$\forall k \in Stack$	(31)

Cuadro 6.3: Conjunto de restricciones

Restricciones para calcular el costo	
$OCNS = \sum_{i \in Stacks} VSC_i$	(32)
$OU = getNumStack() - OUTE$	(33)
$OP = \sum_{i \in Stacks} OPT_i$	(34)
$OR = \sum_{i \in Slots^R} NRSR_i$	(35)
$OGCTD = (\sum_{i \in Stacks} GCD_i) * 100$	(36)
$O = 1000 * OCNS + 20 * OP + 10 * OU + 5 * OR + OGCTD$	(37)

Cuadro 6.4: Conjunto de restricciones

restricción (21). Se modela la restricción (22) la cual implica que los contenedores de una pila deben estar ordenados de abajo hacia arriba y de mayor a menor peso, donde  $Min(x)$  y  $Max(x)$  especifica el mínimo y máximo índice de conjunto de slot que pertenecen a una pila y que aún no se han cargado. Las restricciones (23) y (24) tienen como objetivo evitar contenedor sobre-estibados, un contenedor se considera sobre-estibado en una pila cuando este contiene un puerto de descarga mayor a cualquiera de los puertos de descarga de los contenedores que se encuentren por debajo del mismo. La restricción (25) tiene como objetivo contar por cada pila la cantidad de puertos de descarga diferentes. La restricción (26) y (27) implican que no es posible estibar contenedores refrigerados en espacios no refrigerados. La restricción (28) y (29) restringen el dominio de los espacios de 20 y 40 pies indicando que solo se pueden cargar contenedores de 20 y 40 pies respectivamente. La restricción (30) cuenta la cantidad de contenedores de 20 y 40 pies estibados por pila. La restricción (31) se encarga de obtener la cantidad de pilas vacías (Cuadro 6.3).

Las restricciones utilizadas para calcular el costo de la soluciones se encuentran referenciadas del (32) al (37). Se debe tener en cuenta que para éste modelo las restricciones de costo para los contenedores de 40 pies es el doble de uno de 20 pies. A continuación, se detalla cada unas de las restricciones de costo.

La restricción (32) cuenta el costo de contenedores no estibados en el buque. El costo de contenedores no estibados se obtiene de los contenedores no virtuales estibados en espacios virtuales. Para obtener la cantidad de pilas utilizadas se utiliza la restricción (33) que obtiene la cantidad de pilas y resta éste valor con la cantidad de pilas no usadas; la función  $getNumStack()$  es la encargada de obtener la cantidad de pilas. La restricción (34) suma la cantidad de contenedores con puerto de descarga diferente en una misma pila. La restricción (35) suma la cantidad de contenedores no refrigerados estibados en espacios refrigerados. La restricción (36) se encarga de obtener el peso del centro de gravedad, éste calculo consiste en la suma de los centros de gravedad de cada pila multiplicado por la constante 100. Por último, la restricción (37) obtiene el costo total de la solución encontrada (Cuadro 6.4).

### 6.0.5. Propagación del modelo con respecto a la función objetivo

La propagación del modelo se realiza por medio de una función de costo que tiene en cuenta los siguientes objetivos:

- Minimización de los contenedores no estibados, variable **OCNS**. Se adicionará un costo de 1000 unidades por cada contenedor no estibado.
- Minimización de los puertos de descarga de los contenedores en cada pila, variable **OP**. Se agregará por cada puerto de descarga adicional de los contenedores en la pila un costo de 20 unidades.
- Minimización de las pilas vacías, variable **OU**. Se adicionará por cada nueva pila usada un costo de 10 unidades.
- Minimizar la carga de contenedores no refrigerados en celdas refrigeradas, variable **OR**. Se adicionará un costo de 5 unidades por cada contenedor no refrigerado estibado en una celda refrigerada.
- Minimización de la distancia desde el centro de gravedad ideal de una pila al centro de gravedad real de la pila estibada, variable **OGCTD**. El valor de la variable **OGCTD** se encuentra dado en centímetros.

---

## Lista de símbolos y abreviaciones

---

Abreviaciones	Descripciones	Definición
TEU	veinte pies de unidades equivalentes	page 4
FEU	cuarentea pies de unidades equivalentes	page 4
MIT	Instituto Teconologico de Massachusetts	page 13
ISO	Organización Internacional de Normalización	page 12

---

# Índice de figuras

---

1.1. Imagen de un contenedor. . . . .	4
1.2. Imagen de un buque portacontenedores. [7] . . . . .	5
2.1. escenario CCP . . . . .	12
4.1. Índices de espacios (Slots) . . . . .	21
4.2. Indexación de espacios en el buque . . . . .	22
4.3. Coordenadas para cálculo de centro de gravedad . . . . .	26
5.1. Índices por pila . . . . .	33
5.2. Índices por niveles . . . . .	34

---

# Índice de cuadros

---

3.1. Tabla de constantes. . . . .	18
3.2. Tabla de variables de decisión. . . . .	19
3.3. Conjunto de restricciones . . . . .	19
4.1. Tabla de constantes. . . . .	25
5.1. Tabla de variables de decisión. . . . .	28
5.2. Tabla de variables de decisión utilizadas para calcular el costo de la solución. . . . .	29
5.3. Conjunto de restricciones . . . . .	30
5.4. Conjunto de restricciones . . . . .	31
6.1. Tabla de variables de decisión. . . . .	36
6.2. Tabla de variables de decisión utilizadas para calcular el costo de la solución. . . . .	37
6.3. Conjunto de restricciones . . . . .	38
6.4. Conjunto de restricciones . . . . .	39





---

# Bibliografía

---

- [1] Daniela Ambrosino, Davide Anghinolfi, Massimo Paolucci, and Anna Sciomachen. A new three-step heuristic for the master bay plan problem. *Maritime Economics & Logistics*, 11(1):98–120, 2009.
- [2] Daniela Ambrosino and Anna Sciomachen. A constraint satisfaction approach for master bay plans. *Water studies series*, pages 175–184, 1998.
- [3] Anastasios Haralampos Aslidis. *Optimal container loading*. PhD thesis, Massachusetts Institute of Technology, 1984.
- [4] Mordecai Avriel, Michal Penn, Naomi Shpirer, and Smadar Witteboon. Stowage planning for container ships to reduce the number of shifts. *Annals of Operations Research*, 76:55–71, 1998.
- [5] RC Botter and MA Brinati. Stowage container planning: A model for getting an optimal solution. 1991.
- [6] Lucas Antón De Luca, Nicolas German & Neller. Introducción a la ingeniería naval - buques portacontenedores.
- [7] Alberto Delgado, Rune Møller Jensen, Kira Janstrup, Trine Høyer Rose, and Kent Høj Andersen. A constraint programming model for fast optimal stowage of container vessel bays. *European Journal of Operational Research*, 220(1):251–261, 2012.
- [8] Opher Dubrovsky, Gregory Levitin, and Michal Penn. A genetic algorithm with a compact solution encoding for the container ship stowage problem. *Journal of Heuristics*, 8(6):585–599, 2002.
- [9] GECODE. Generic constraint development environment.
- [10] Jun-Gyu Kang and Yeong-Dae Kim. Stowage planning in maritime container transportation. *Journal of the Operational Research Society*, 53(4):415–426, 2002.
- [11] Feng Li, Chunhua Tian, Rongzeng Cao, and Wei Ding. An integer linear programming for container stowage problem. In *Computational Science-ICCS 2008*, pages 853–862. Springer, 2008.

- [12] Malcolm Yoke Hean Low, Xiantao Xiao, Fan Liu, Shell Ying Huang, Wen Jing Hsu, and Zhengping Li. An automated stowage planning system for large containerships.
- [13] Dario Pacino, Alberto Delgado, Rune Møller Jensen, and Tom Bebbington. Fast generation of near-optimal plans for eco-efficient stowage of large container vessels. In *Computational Logistics*, pages 286–301. Springer, 2011.
- [14] Anna Sciomachen and Elena Tanfani. The master bay plan problem: a solution method based on its connection to the three-dimensional bin packing problem. *IMA Journal of Management Mathematics*, 14(3):251–269, 2003.
- [15] Zhang Wei-Ying, Lin Yan, and Ji Zhuo-Shang. Model and algorithm for container ship stowage planning based on bin-packing problem. *Journal of Marine Science and Application*, 4(3):30–36, 2005.
- [16] enciclopedia informatica Wikipedia. Definición de estiba.
- [17] ID Wilson and PA Roach. Container stowage planning: a methodology for generating computerised solutions. *Journal of the Operational Research Society*, pages 1248–1255, 2000.