**Análisis del problema de carga de contenedores de tipo problema sencillo de la mochila**

Jhon Casas, Valeria Diaz

1. **Introducción**

El Problema de Carga de Contenedor Único (CLP) consiste en organizar caja u objetos pequeños de forma perpendicular a las paredes de un objeto más grande o contenedor. De acuerdo con Wäscher et al. (2006) existen distintas formas de clasificar este tipo de problema, 3D-SLOPP cuando hablamos de objetos tridimensionales y el conjunto de cajas es débilmente heterogéneo, y como 3D-SKP cuando hablamos de objetos tridimensionales que son fuertemente heterogéneos. En este caso se estará trabajando sobre el 3D-SLOPP según la tipología propuesta por Wäscher (Ilustración 1). Este problema consiste en un solo contenedor de tres dimensiones en el cual se busca organizar la mayor cantidad posible de cajas, maximizando su valor (Wäscher, 2006). Sin embargo, en este caso se estará trabajando sobre un solo contenedor en el cual se busca organizar una cantidad de cajas con características diferentes, buscando maximizar el volumen de las cajas dentro del contenedor, sin solapamiento entre cajas, con una restricción de peso cuando se coloca una caja sobre otra y las cajas deben encontrarse completamente dentro del contenedor en la solución final.

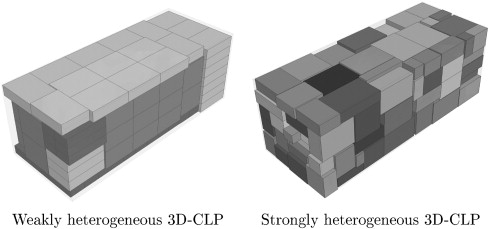


Figura 1. Tomado de Gonçalves & Resende (2012)

De acuerdo con como se muestra en la imagen a continuación (Ilustración 2), es posible ubicar una caja encima de otra, por lo tanto, existe una restricción del peso máximo que puede haber sobre una caja dependiendo la posición de esta misma. Por ejemplo, si se está transportando un televisor o equipos electrónicos, no se pueden poner cargas muy pesadas sobre estos. Además, dependiendo de la posición en la que se ubique la caja, la cantidad de peso máxima varía porque hay áreas en donde los electrodomésticos son más delicados o están más cerca al borde la caja. Esto es a lo que se refiere la restricción de peso.

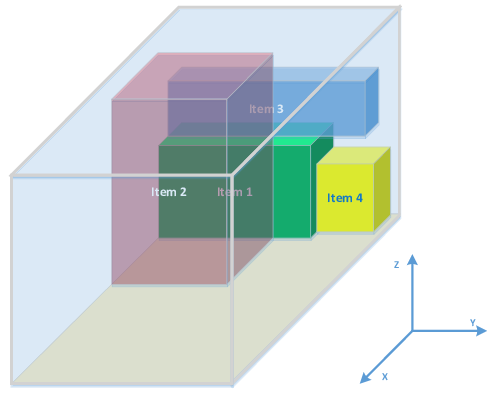


Figura 2. Tomado de Ha et. al. (2017)

3D-SLOPP en general se enfrenta a varias características particulares tanto de los

contenedores como de las cajas como lo son: dimensiones del contenedor, restricciones del peso, posición, tamaño y formas de las cajas. De la misma forma, debido a que se puede transformar un problema de la mochila la dificultad que mejor describe al problema de las cajas y el contenedor es un NP-hard (Lim y Zhang, 2005). Este problema escala en dificultad a medida que aumentan las cajas que hay que organizar debido a que las posibles soluciones aumentan exponencialmente dependiendo de la cantidad de cajas, dado esto, este problema se resuelve usando heurísticas o metaheurísticas (Parreño et. al., 2008).

Se busca resolver el Problema de Carga de Contenedores de tipo *single knapsack* de manera eficiente con el fin de maximizar la utilización del contenedor. Sin embargo, el CLP ya genera unas consecuencias en términos económicos y ambientales que también se verían impactadas por una solución de este. En términos económicos, sabemos que hay costos fijos relacionados al alquiler de contenedores, transporte, mantenimientos, entre otros que se distribuyen por m3 ocupado por paquetes heterogéneos. Por otro lado, en la parte ambiental se espera que al tener que transportar más contenedores, mas es la huella de carbono generado por el uso de combustible fósil distribuido por los contenedores usados. Siguiendo la lógica propuesta, con el objetivo de la solución, se busca reducir los costos por m3 ocupado asumidos por el contenedor y reducir la huella de carbono al poder maximizar la utilización de los contenedores reduciendo así la cantidad de contenedores usados.

Algunos ejemplos muy usuales de este problema y para los que están diseñados estos algoritmos son para los contenedores de carga marítima y terrestres. La logística encargada de ingresar los productos en los contenedores siempre busca ingresar la mayor cantidad de productos con el fin generar un ahorro dado el volumen usado. Empresas como lo es Mercado Libre quienes trabajan en mensajería nacional e internacional son objeto de un gran volumen de productos y quienes deben contar con una amplia logística en la que debe existir el apartado de la carga de los contenedores para transporte internacional. Si se implementara una solución eficiente del CLP en su logística de carga seguramente podrían mejorar la utilización del contenedor pudiendo transportar un mayor volumen por contenedor y mejorar su logística operativa. De la misma forma, empresas como Inter rapidísimo quienes transportan más a nivel nacional, usan también contenedores para poder llegar a los puntos más lejanos del país por lo que la carga normalmente se realiza en los centros logísticos de despacho y se envía en camiones de carga. Es así que un algoritmo que soluciona el CLP seguramente les permitiría reducir costos fijos de mantenimiento, alquiler entre otros; mejorar la velocidad de despacho al tener ya un modelo sobre las posiciones de los productos y su principal objetivo, maximizar el volumen usado por contenedor.

1. **Estado del arte**

Debido a que el problema de contenedor único es un problema muy conocido, existen diversos trabajos que tratan el problema. A continuación, en la *Tabla 1* se ve un resumen de los casos tratados anteriormente que han trabajado el problema de contenedor único.

Tabla 1. Revisión de la literatura.

|  |  |  |  |
| --- | --- | --- | --- |
| Autores | Tipología (según Wäscher) | Metodología | Restricciones adicionales |
| Álvarez et. al. (2015) | 3D-SLOPP y 3D-SKP | Algoritmo GRASP | Múltiples paradas para dejar cajas. |
| Araujo y Armentano (2007) | 3D-SLOPP y 3D-SKP | Algoritmo constructivo aleatorio con múltiples inicios. | Orientación y estabilidad. |
| Bortfeldt y Gehring (2001) | 3D-SKP | Metaheurística genética híbrida. | Orientación, estabilidad, apilamiento, peso y balance. |
| Lim et. al. (2013) | SCLP | Algoritmo GRASP de construcción de paredes junto con problemas de programación lineares enteros. | Peso individual de las cajas, peso total de las cajas |
| Phongmoot et. al. (2023) | 3D-SKP | Modelo matemático y algoritmo artificial de colonia de abejas (ABC) | Estabilidad, apilamiento, balance, no solapamiento, facilitar el manejo de carga y las prioridades de los clientes. |
| Tian et. al. (2023) | SCLP | Algoritmo constructivo de dos fases | Orientación, soporte de la caja, no solapamiento. |
| Zhang et. al. (2022) | 3D-SLOPP | Algoritmo heurístico hibrido | Orientación, soporte de la caja, capas de cajas. |
| Pisinger, D. (2002) | 3D-SKP | Un algortimo que usa una heurística de búsqueda de árboles con muros. | las cajas se pueden girar en cualquier dirección ortogonal, Sin tener en cuenta el soporte de los artículos. |
| Ramos et. al. (2016) | 3D-SLOPP | Un algoritmo Genético con hibridación de restricciones de estabilidad de equilibrio mecánico estático. | Caras Paralelas  No Solapamiento  Dentro del contenedor•  Estabilidad estática  Todas las cajas son rígidas.  Centro geométrico. |
| Ramos, A. G., Silva, E., & Oliveira, J. F. (2018). | 3D-SLOPP | algoritmo genético de clave aleatoria sesgado en múltiples poblaciones (BRKGA) con estabilidad estática. | Caras Paralelas  no solapamiento  dentro del contenedor•  Estabilidad estática  Todas las cajas son rígidas.  Centro geométrico.  función de adecuación |
| Huang, Y. H., Hwang, F. J., & Lu, H. C. (2016). | 3DCLP | Algoritmo con heurística de colocación de carga simple | Orientacion de caras |
| Kurpel, D. V., Scarpin, C. T., Junior, J. E. P., Schenekemberg, C. M., & Coelho, L. C. (2020). | 3D-SLOPP y 3D-SKP | Metodos exactos | Cajas rectangulares, de forma ortogonal y sin superposición, dentro de contenedores |
| Şafak, Ö., & Erdoğan, G. (2023). | 3D-SKP | Algoritmo de búsqueda de vecindario grande | La restricción del límite de peso,  las restricciones de distribución del peso. prioridades de carga.  Orientaciones, Restricción de apilaciones. |
| Kuo et. al. (2023) | 3D-SLOPP | Algoritmo genético de clasificación no dominado II (NSGA II) y el algoritmo de optimización de enjambre de partículas multiobjetivo (MOPSO). | Dimensiones, la restricción de orientación, la distribución del peso y la restricción de localidad |
| Nascimiento (2021) | SCLP | Algoritmo exacto | No solapamiento, envío completo, ítems conflictivos, prioridades, peso limite, estabilidad vertical y horizontal, soporte de la carga, multidrop, balance de la carga, carga manual, agrupamiento, separación, y múltiples orientaciones. |
| Zhang et. al. (2012) | SCLP | Algoritmo heurístico de carga de bloques, algoritmo de búsqueda multilayer | No solapamiento,  dependiendo del tipo de caja algunas cajas sólo pueden ser orientadas de una forma, todas cajas que no estén ubicadas sobre el suelo del contenedor deben soportadas al 100% por otras cajas. |
| Xavier Do Nascimento et. al. (2018) | SCLP | Método exacto | Condiciones prioritarias, carga horizontal y vertical, estabilidad vertical y horizontal. |
| Bayraktar et. al. (2021) | SCLP | Algoritmo de colonia de abejas artificial | Orientación, restricción de peso sobre la caja, restricción de peso total, estabilidad, balance. |
| Layeb et. al. (2017) | SCLP | Algoritmo voraz de dos pasos | Distribución del peso, posicionamiento y apilamiento. |
| Wang et. al. (2013) | SCLP | Método de múltiples rondas con búsqueda parcial de haz | Orientación, soporte completo, prioridad. |
| Araya et. al. (2017) | SCLP | Nueva función evaluativa | Orientación, estabilidad. |

Seguidamente se realizará una breve explicación de algunos de los artículos mencionados anteriormente.

* Algoritmo grasp con restricción multidrop

Álvarez (2015) propone un algoritmo GRASP que, de acuerdo con la literatura de los últimos 20 años, contempla las restricciones geométricas de las cajas en el problema y agrega restricciones más realistas sobre el manejo de estas cajas. Es así como basado con las ultimas publicaciones de Bortfeldt & Wäscher, agrega las restricciones de multidrop en su problema teniendo en cuenta también restricciones como la rotación, límites de carga, entre otras. Su método GRASP es la combinación de un procedimiento constructivo aleatorio para encontrar una solución inicial y un proceso de mejora a esta solución llegando una solución valida y optima.

* Heurística de múltiples inicios con subconjuntos

Araujo & Armentano (2007) usan una Metaheurística constructiva de la misma familia que GRASP para resolver CLP donde partiendo de un método constructivo se crean varias soluciones iniciales desde las que se parten para después en un proceso de mejora converger a un espacio con las mejores soluciones. Sin embargo, en este algoritmo las cajas se organizan en bloques o como subconjuntos de cajas con un volumen definido de tal forma que no se busca acomodar las cajas individualmente, sino que se busca ubicar los subconjuntos.

* Solución con metaheurística genética híbrida

Bortfeldt y Gehring (2001) publicaron un método genético híbrido que buscaba dar solución al CLP. Este modelo genético utiliza como método de inicialización una heurística voraz y busca crear capas verticales de cajas que no se superponen y estas paralelas a una pared del contenedor. Según sus resultados, este método es bastante efectivo para un contenedor superando los rendimientos de otros métodos.

* VCS: Nueva función heurística para seleccionar cajas en el problema de carga de contenedor único

Araya et. al. (2017) propuso una nueva función evaluativa para escoger las cajas en el problema de carga de contenedor único cuando se realiza una construcción de bloques. Esta función tiene en cuenta el volumen del bloque, pero también la superficie de este que estaría cubierta dependiendo de la posición en la que sea ubicado.

* Efectos de memoria y operadores genéticos en un algoritmo de colonia de abejas artificial para problema de contenedor único (ABC)

Bayraktar et. al. (2021) plantea un algoritmo ABC con memoria integrada para pasos de búsqueda útiles en búsqueda local y un algoritmo ABC basado en un operador genético con el cual se generan los siguientes pasos de búsqueda para la búsqueda global. A costo plazo, el método de memoria dio mejores resultados. Sin embargo, a largo plazo el método genético presentó mejores resultados y es más aplicable para casos de cajas fuertemente heterogéneas.

* Un algoritmo heurístico de carga de bloques en búsqueda multilayer para el problema de carga de contenedores

Zhang et. al. (2012) sugiere un algoritmo de carga de bloques para el problema de carga de contenedores. Este algoritmo presenta bloques compuestos, y evalúa el estado actual para proponer bloques que sean apropiador para la carga. Evaluado en 1500 instancias, muestra que este algoritmo tiene un mejor resultado que el mejor algoritmo para problemas con y sin restricción de soporte.

1. **Formulación**

Función objetivo: maximizar el volumen de las cajas dentro del contenedor.

S-t:

Restricciones de sobreposición:

Las restricciones (1-6) buscan cumplir con la superposición de unas cajas sobre otras. Para ello se deben tener en cuenta 2 supuestos: I. El peso de una caja no afecta el centro de masa de una caja que esta debajo. II. El peso de una caja se distribuirá uniformemente sobre el peso de la caja de abajo. Sabiendo lo anterior, la restricción (1) busca encontrar el peso soportado por una caja j dadas unas cajas i que están encima de j. Eso significa sumar los pesos de las cajas que están arriba de j (. (2) busca definir la posición de la caja i que esta encima de j en la coordenada Z. En la restricción (3) se quiere definir una condición donde la caja i solo puede estar encima de la caja j ( o debajo (. La restricción (4) define que el peso acumulado de las cajas i que están arriba de la caja j no puede ser mayor al peso soportado de la caja j. La restricción (7) define una condición donde las cajas i encima de la caja j deben tener un área menor. Finalmente, la restricción (5) y (6) buscan que, dada las 6 posiciones, la caja i este dentro del contenedor en solo una posición, además de que la suma de las cajas de cada tipo en sus diferentes posiciones debe ser menor o igual al número de cajas disponibles de ese tipo.

Restricciones de contención:

La restricción de contención busca que la posición de las cajas se defina dentro de los límites del contenedor. Es así como las restricciones (8), (9) y (10) buscan que los artículos empacados no estén superpuestos. (11), (12) y (13) definen que las dimensiones de las cajas que están colocadas dentro del contenedor estén completamente dentro de estos límites.

Restricciones de naturaleza:

Estas restricciones definen la naturaleza de las variables. (15) define variables binarias y (16) variables reales positivas.

1. **Heurística**

La heurística busca encontrar una solución por medio de la realización de dos pasos, primeramente, se selecciona el espacio maximal de acuerdo con una esquina del contenedor, es decir, seleccionamos qué espacio estaremos utilizando. Seguidamente se seleccionará la rotación de la caja que se ubicarán en este espacio. En esta heurística se va comprobando las rotaciones y la ubicación de las cajas dentro del contenedor, es decir, se le van añadiendo cajas al contenedor intentando maximizar el volumen de cajas dentro de este, asegurando el cumplimiento de las restricciones. Además, cada iteración implica una mejora de la función objetivo. Dado esto, esta heurística sería considerada una heurística constructiva. A continuación, se muestra una descripción de esta.

La heurística comienza escogiendo un espacio maximal en el que ubicar la caja, este espacio de escoge priorizando el espacio que esté más cercano al suelo. Luego, se escoge el tipo de caja escogiendo la que tenga mayor volumen. Luego, se verifica que alguna rotación de la caja pueda estar dentro del espacio maximal. Si hay una caja que pueda entrar en el espacio, la caja se mete al contenedor. Si hay múltiples cajas, se escoge la que mejor ocupe el espacio, es decir, la que más se acerque a las dimensiones del espacio maximal.

La caja seleccionada se ubica en la esquina más cercana a las coordenadas de origen (0, 0, 0) del contenedor. Cuando la caja esta dentro del contenedor, se actualizan las cantidades de tipos de cajas, la utilización del contenedor y los espacios maximales disponibles.

Si ninguna caja entra, se pasa al siguiente tipo de caja de mayor volumen. Cuando ninguna rotación de ningún tipo de caja puede ser ubicada en ningún espacio maximal, concluye la heurística.

Este enfoque permite una selección cuidadosa de cajas y rotaciones, optimizando el llenado del contenedor y priorizando la utilización eficiente del espacio, especialmente al considerar los espacios maximales más cercanos al suelo para minimizar la altura total del contenedor.

1. **Instancias y diseño de experimentos**

Las instancias por utilizar para el análisis de los resultados próximamente en este documento corresponden a series de datos ampliamente utilizados en la literatura, estos datos corresponden a datos de industrias y también son consideradas instancias clásicas. Ratcliff & Bischoff (1998) fueron los primeros en usar estos datos con restricciones de peso, asimismo, Bischoff (2006) y David (2000) también utilizaron estas instancias. Estas instancias se pueden encontrar en la página de OR-Library con el link <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/conloadinfo.html>. Cada instancia tiene en la primera línea las dimensiones del contenedor, es decir, el largo, ancho y alto del contenedor en centímetros. En la segunda línea, están el número de tipos de caja y el volumen total del contenedor. Las siguientes líneas corresponden a las características de cada tipo de caja, donde se encuentran el largo, ancho y alto de la caja en centímetros y después de cada medida hay un indicador (1 o 0) que sugiere si es posible poner esa dimensión en el eje vertical y se asume que se pueden ubicar en ambas orientaciones 90 grados con respecto al eje vertical. Adicionalmente, también se encuentra información del número de cajas de este tipo y el peso de la caja en kilogramos, el peso se asume como la multiplicación de las medidas de las cajas. Finalmente, se encuentran los pesos sobre metro cuadrado que la caja puede soportar en cada orientación. Existen siete categorías de los datos, cada una con un número distintos de tipos de caja y cada categoría cuenta con 100 instancias.

Tabla 2. Características de las instancias.



En la *Tabla* 2 se encuentra la información resumida de las instancias a utilizar. Allí se puede ver la cantidad máxima de cajas que hay en una instancia por categoría, así como también la cantidad mínima. Además, podemos ver que el número de tipos de cajas que hay por categoría. Finalmente, en la tabla se muestra la promedio utilización de cada categoría. Aunque existan pequeñas diferencias respecto a la utilización, en todos los casos los valores se acercan al 100%. Es decir, si se utilizara el espacio dentro del contenedor de la forma más eficiente posible, prácticamente todas las cajas podrían estar dentro del mismo. Sin embargo, esto es un caso complicado debido a que, con las medidas de las cajas, la organización y las restricciones peso se tienden a dejar espacios en blanco.

En este documento se utilizará un algoritmo VNS, la cual consiste en realizar una búsqueda por una variedad de vecindarios, los cuales son variaciones de soluciones encontradas previamente. En los vecindarios hallados se encuentra la mejor solución y luego se cambia de vecindario. Esto se repite por un número determinado de iteraciones.

La metaheurística será probada utilizando las 10 primeras instancias de cada una de las 7 categorías de estas. Esto se hará debido a las restricciones con el tiempo que se tienen. Dado que las instancias son bastante similares entre sí, utilizar 10 de cada categoría representa un buen acercamiento al comportamiento esperado real de los datos. Es decir, las respuestas dadas se espera que se acerquen a las encontradas con todo el conjunto de datos. Este diseño experimental nos permite tener una compresión de los resultados al utilizar las instancias, a pesar de no usar todas las disponibles. Esto se hace con el objetivo de comprender en un principio el rendimiento de la metaheurística.

Los resultados finales de la metaheurística serán evaluados calculando la utilización promedio de cada categoría de las instancias. Asimismo, se utilizará también el tiempo de cómputo como evaluación de la metaheurística. Se decidió utilizar estos dos valores dado que son una métrica común en artículos sobre este tipo de problema, por ejemplo, la utilización promedio se utiliza en artículos como Bortfeldt & Gehring (2001), Ratcliff & Bischoff (1998) y Bischoff (2006). Asimismo, el tiempo de cómputo se utiliza en Li et. al. (2022).

1. **Metaheurística**

En la metaheurística se comienza a partir de una solución proporcionada por la heurística constructiva. Para crear un vecindario, de cada solución se escoge un número n de cajas consecutivas de forma aleatoria que deben ser cambiadas por otras que puedan ocupar ese espacio. Luego de escoger las cajas a cambiar, se escoge también aleatoriamente qué cambiar, lo cual puede ser reemplazar un tipo de caja por otra, cambiar la rotación de la caja y/o cambiar el espacio de la caja en el contenedor. Después de realizar el movimiento, le aplica nuevamente la lógica de la heurística constructiva para organizar el resto de las cajas en el contenedor.

Seguidamente, se repite este procedimiento hasta generar un vecindario con al menos 50% de las cajas que se introdujeron en la solución determinística. Este vecindario representa diferentes configuraciones obtenidas al cambiar una caja y realizar movimientos aleatorios.

Luego, se encuentra la mejor solución dentro del vecindario actual, para luego moverse a un nuevo vecindario donde se cambiarán cajas. Esto corresponde a un *shake*.

El algoritmo sigue explorando diferentes vecindarios, aumentando gradualmente la complejidad al cambiar más cajas a medida que avanza en la búsqueda. La combinación de la heurística constructiva determinística inicial y los movimientos aleatorios en los vecindarios proporciona una estrategia para explorar soluciones de manera eficiente en el espacio de búsqueda.



En el anterior pseucódigo se presentó de forma general el funcionamiento de la metaheurística.

Con el objetivo de ver los resultados de la metaheurística se probaron 10 instancias de cada categoría, eso se decidió debido al tiempo limitado. Teniendo en cuenta esto, el promedio de los resultados representa una muestra significativa debido a que las instancias son bastante similares.

A continuación, se hará una muestra de la comparación de resultados con otros autores que han utilizado las mismas instancias que en este documento. En el caso de Bischoff (2006) se utilizaron los mejores resultados de la búsqueda aleatoria, en el caso de los demás se utilizaron los resultados proporcionados.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Bischoff (2006) | | Radcliff & Bischoff (1998) | | Davis (2000) | | VNS | |
| Categorías | Utilización | Tiempo(seg) | Utilización | Tiempo(seg) | Utilización | Tiempo(seg) | Utilización | Tiempo (seg) |
| 1 | 79.35% | 26.40 | 75.56% | ----- | 78.99% | ----- | **81.36%** | 75.25 |
| 2 | 82.97% | menos de 210 | 78.74% | ----- | **83.04%** | ----- | 81.28% | 80.79 |
| 3 | **85.33%** | menos de 210 | 80.91% | ----- | 84.62% | ----- | 82.08% | 98.45 |
| 4 | **85.44%** | menos de 210 | 80.96% | ----- | 84.69% | ----- | 83.52% | 167.42 |
| 5 | **85.61%** | menos de 210 | 80.35% | ----- | 83.73% | ----- | 82.98% | 103.40 |
| 6 | **85.59%** | menos de 210 | 79.90% | ----- | 84.12% | ----- | 83.64% | 131.07 |
| 7 | **85.83%** | 321.00 | 78.93% | ----- | 82.74% | ----- | 83.35% | 222.31 |

Tabla 3. Comparación de resultados.

Como se puede observar, se obtuvieron resultados similares en la utilización de los contenedores. Respecto a Radcliff & Bischoff (1998) los resultados de la metaheurística tienen mejor utilización. Sin embargo, con respecto a Bischoff (2006) a partir de la tercera categoría este obtiene los mejores resultados en comparación con los demás. En la primera instancia la metaheurística gana en términos de utilización, esto debido a que para instancias pequeñas esta tiende a tener mejores resultados dado que con los métodos que se utilizan para escoger cajas, requiere menos esfuerzo computacional cuando existe una menor cantidad de tipos de cajas. En términos de tiempo podemos observar que el tiempo de implementación aumenta proporcionalmente con el número cajas, como se esperaba debido a las condiciones de escogencia de cajas, aún así, los tiempos son similares a los reportados por Bischoff (2006).

1. **Referencias**

Gonçalves, J. F. & Resende, M. G. (2012). *compared WH 3D-CLP and SH 3D-CLP* (3D-CLP) [Illustration 1]. A parallel multi-population biased random-key genetic algorithm for a container loading problem. Revista de Investigación en Algoritmos Genéticos. *Computers & Operations Research, 39(2)*. <https://doi.org/10.1016/j.cor.2011.03.009>

Ha, C. T., Nguyen, T., Bui, L., & Wang, R. (2017). An online packing heuristic for the three-dimensional container loading problem in dynamic environments and the physical internet [Illustration 2]. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Vol. 10200 LNCS, pp. 140–155). Springer Verlag. <https://doi.org/10.1007/978-3-319-55792-2_10>

Álvarez Martínez, D., Alvarez-Valdes, R., & Parreño, F. (2015). A GRASP algorithm for the container loading problem withmulti-drop constraints. Pesquisa Operacional, 35(1), 1–24. <https://doi.org/10.1590/0101-7438.2015.035.01.0001>

Araújo, O. C. B. D., & Armentano, V. A. (2007). A multi-start random constructive heuristic for the container loading problem. *Pesquisa Operacional*, *27*, 311-331. <https://doi.org/10.1590/S0101-74382007000200007>

Wäscher, G., Haußner, H., & Schumann, H. (2007). An improved typology of cutting and packing problems. European Journal of Operational Research, 183(3), 1109–1130. <https://doi.org/10.1016/j.ejor.2005.12.047>

BORTFELDT A & GEHRING H. 2001. A hybrid genetic algorithm for the container loading problem. European Journal of Operational Research, 131: 143-161. <https://doi.org/10.1016/S0377-2217(00)00055-2>

Lim, A., & Zhang, X. (2005). The container loading problem. In Proceedings of the ACM Symposium on Applied Computing (Vol. 2, pp. 913–917). Association for Computing Machinery (ACM). <https://doi.org/10.1145/1066677.1066888>

Lim, A., Ma, H., Qiu, C., & Zhu, W. (2013). The single container loading problem with axle weight constraints. International Journal of Production Economics, 144(1), 358–369. <https://doi.org/10.1016/j.ijpe.2013.03.001>

Parreño, F., Alvarez-Valdes, R., Tamarit, J. M., & Oliveira, J. F. (2008). A maximal-space algorithm for the container loading problem. INFORMS Journal on Computing, 20(3), 413–422. <https://doi.org/10.1287/ijoc.l070.0254>

Phongmoo, S., Leksakul, K., Charoenchai, N., & Boonmee, C. (2023). Artificial Bee Colony Algorithm with Pareto-Based Approach for Multi-Objective Three-Dimensional Single Container Loading Problems. Applied Sciences (Switzerland), 13(11). <https://doi.org/10.3390/app13116601>

Tian, T., Zhu, W., Zhu, Y., Qiang, L. & Lijun, W. (2023) A two-phase constructive algorithm for the single container mix-loading problem. Ann Oper Res. <https://doi-org.ezproxy.uniandes.edu.co/10.1007/s10479-023-05542-9>

Zhang, D., Gu, C., Fang, H., Ji, C., & Zhang, X. (2022). Multi-strategy hybrid heuristic algorithm for single container weakly heterogeneous loading problem. Computers and Industrial Engineering, 170. <https://doi.org/10.1016/j.cie.2022.108302>

Pisinger, D. (2002). Heuristics for the container loading problem. *European journal of operational research*, *141*(2), 382-392. <https://doi.org/10.1016/S0377-2217(02)00132-7>

Ramos, A. G., Oliveira, J. F., Gonçalves, J. F., & Lopes, M. P. (2016). A container loading algorithm with static mechanical equilibrium stability constraints. *Transportation Research Part B: Methodological*, *91*, 565-581. <https://doi.org/10.1016/j.trb.2016.06.003>

Ramos, A. G., Silva, E., & Oliveira, J. F. (2018). A new load balance methodology for container loading problem in road transportation. *European Journal of Operational Research*, *266*(3), 1140-1152. <https://doi.org/10.1016/j.ejor.2017.10.050>

Huang, Y. H., Hwang, F. J., & Lu, H. C. (2016). An effective placement method for the single container loading problem. *Computers & Industrial Engineering*, *97*, 212-221. <https://doi.org/10.1016/j.cie.2016.05.008>

Kurpel, D. V., Scarpin, C. T., Junior, J. E. P., Schenekemberg, C. M., & Coelho, L. C. (2020). The exact solutions of several types of container loading problems. *European Journal of Operational Research*, *284*(1), 87-107. <https://doi.org/10.1016/j.ejor.2019.12.012>

Şafak, Ö., & Erdoğan, G. (2023). A Large Neighbourhood Search Algorithm for Solving Container Loading Problems. *Computers & Operations Research*, *154*, 106199 <https://doi.org/10.1016/j.cor.2023.106199>

Kuo, R. J., Ho, P. C., & Zulvia, F. E. (2023). Application of metaheuristics algorithm on a multi-objective container loading problem considering container’s utilization and vehicle’s balance. *Applied Soft Computing*, *143*, 110417. <https://doi.org/10.1016/j.asoc.2023.110417>

Nascimento, O. X. do, Alves de Queiroz, T., & Junqueira, L. (2021). Practical constraints in the container loading problem: Comprehensive formulations and exact algorithm. Computers and Operations Research, 128. <https://doi.org/10.1016/j.cor.2020.105186>

Zhang, D., Peng, Y., & Leung, S. C. H. (2012). A heuristic block-loading algorithm based on multi-layer search for the container loading problem. Computers and Operations Research, 39(10), 2267–2276. <https://doi.org/10.1016/j.cor.2011.10.019>

Xavier Do Nascimento, O., Vieira De Melo, L., & Alves De Queiroz, T. (2018). Exact method for the container loading problem with priority and stability. In Proceedings - 2018 44th Latin American Computing Conference, CLEI 2018 (pp. 176–183). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/CLEI.2018.00029>

Bayraktar, T., Ersöz, F., & Kubat, C. (2021). Effects of memory and genetic operators on Artificial Bee Colony algorithm for Single Container Loading problem. Applied Soft Computing, 108. <https://doi.org/10.1016/j.asoc.2021.107462>

Bortfeldt, A., & Wäscher, G. (2013). Constraints in container loading-A state-of-the-art review. European Journal of Operational Research, 229(1), 1–20. <https://doi.org/10.1016/j.ejor.2012.12.006>

Layeb, S. B., Jabloun, O., & Jaoua, A. (2017). New Heuristic for the Single Container Loading Problem. International Journal of Economics & Strategic Management of Business Process (ESMB), 8(1), 1–7.

Wang, N., Lim, A., & Zhu, W. (2013). A multi-round partial beam search approach for the single container loading problem with shipment priority. International Journal of Production Economics, 145(2), 531–540. <https://doi.org/10.1016/j.ijpe.2013.04.028>

Araya, I., Guerrero, K., & Nuñez, E. (2017). VCS: A new heuristic function for selecting boxes in the single container loading problem. Computers and Operations Research, 82, 27–35. <https://doi.org/10.1016/j.cor.2017.01.002>

Ratcliff, M. S. W., & Bischoff, E. E. (1998). Allowing for weight considerations in container loading. OR Spectrum, 20(1), 65–71. <https://doi.org/10.1007/BF01545534>

Bischoff, E. E. (2006). Three-dimensional packing of items with limited load bearing strength. In European Journal of Operational Research (Vol. 168, pp. 952–966). <https://doi.org/10.1016/j.ejor.2004.04.037>

Li, Y., Chen, M., & Huo, J. (2022). A hybrid adaptive large neighborhood search algorithm for the large-scale heterogeneous container loading problem. Expert Systems with Applications, 189. <https://doi.org/10.1016/j.eswa.2021.115909>

Davies, A.P., 2000. Approaches to the container loading problem. Unpublished Ph.D. dissertation, University of Wales Swansea.