Computers & Industrial Engineering 216 (2026) 111983

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Enhancing pallet load stability: A MILP model for the Manufacturer's Pallet Loading Problem with interlocking constraints

João Araújo [a], António G. Ramos [b,*], Elsa Silva [c], Ana Moura [d]

[a] INESC TEC, Porto, Portugal
[b] INESC TEC, ISEP, Polytechnic of Porto, Porto, Portugal
[c] ALGORITMI Research Center/LASI, University of Minho, Braga, Portugal
[d] University of Aveiro, Aveiro, Portugal

ARTICLE INFO

Keywords:
Cutting & packing
Stability
Interlocking

ABSTRACT

The Manufacturer's Pallet Loading Problem involves optimising the packing of a maximal number of identical rectangular boxes onto a single rectangular pallet. This problem arises in various logistic operations that involve the storage and transportation of boxed products, where efficient packing can result in substantial cost reductions and improved operational efficiency. Logistics managers anticipate that some boxes can be damaged during handling and transport, so the stability of the pallet load is essential to avoid such damage. The interlocking method is commonly used in practice to improve stability when loading pallets, minimising product damage and reducing the risk of injury to personnel handling the pallet. This study introduces a Mixed Integer Linear Programming model that addresses the Manufacturer's Pallet Loading Problem, promoting static stability through interlocking. Stability is evaluated with respect to the relationship between successive layers of the loading plan, with three types of interlocking incorporated into the mathematical model. Computational experiments with real-world instances were conducted to assess the model's performance using different objective functions and post-optimisation heuristics that target real-world requirements. Three stability metrics were used to evaluate the load plans generated by the mathematical model. The results show the interlocking method's benefits on the pallet loads' stability while maximising the pallet volume usage.

1. Introduction

The Pallet Loading Problem (PLP) is a challenge in the manufacturing and logistics sectors that focusses on optimising the loading of boxes onto pallets in order to maximise volume usage. Two main constraints of the PLP are the maximum weight capacity of the pallet and the stability of the loaded boxes. An efficient arrangement is crucial for space usage and cargo stability, reducing the number of pallets needed. PLP significantly impacts transportation and warehousing costs, as these are directly related to space usage and the quantity of pallets shipped.

The PLP can be classified as follows.

- Manufacturer's Pallet Loading Problem, when there are identical boxes to be efficiently placed onto a pallet. It is a repetitive problem, and the loading of boxes onto pallets can be done using automated handling systems.
- Distributor's pallet loading problem, which requires arranging heterogeneous boxes on a pallet. This problem is characterised by its non-repetitive nature, attributable to the diversity of boxes requiring palletisation.

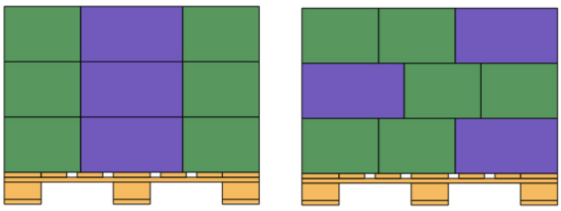
This work tackles the Manufacturer's Pallet Loading Problem (MPLP). The motivation for this problem arises from the need to develop a decision support system as part of an intralogistics automation cell for automatic unloading and palletising of boxes transported in shipping containers while guaranteeing that pallets are stable and the patterns built follows customers' preferences.

Based on the typology introduced by Wäscher et al. (2007) for cutting and packing problems, we address MPLP as a two-dimensional rectangular Identical Item Packing Problem (IIPP).

In this problem, a box is defined by length a, width b, and height c. The box can only rotate about its vertical axis. As such, the pallet arrangement will consist of a set of layers with identical heights. The 3-dimensional boxes are converted into rectangles to determine the layers' arrangements, and the 3-dimensional MPLP is transformed into a 2-dimensional problem. For a review of off-line and on-line solution

* Corresponding author.
E-mail addresses: joao.araujo@inesctec.pt (J. Araújo), agr@isep.pt (A.G. Ramos), elsa@dps.uminho.pt (E. Silva), ana.moura@ua.pt (A. Moura).

https://doi.org/10.1016/j.cie.2026.111983
Received 23 June 2025; Received in revised form 3 March 2026; Accepted 19 March 2026
Available online 26 March 2026
0360-8352/© 2026 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).



(a) Column Stacking (b) Interlocking Stacking

Fig. 1. Column Stacking vs. Interlock Stacking.

approaches to 3D packing problems, see Ali et al. (2022). The objective is to place the maximum number of identical small rectangles within the dimensions of the pallet base, length L and width W, ensuring that there is no overlap and the edges of the rectangles align with those of the pallet.

A set of stacked layers forms a pallet loading plan. The total number of layers depends on the height of the boxes and the maximum height of the pallet. When solving the MPLP, in addition to the geometry, there are other constraints to consider, such as the stability, the fragility of the boxes, and the load bearing.

In this work, in addition to dealing with the geometric constraints related to overlapping in the MPLP we also tackle stability using the interlocking method.

Interlocking stacking uses a type of rotation method by placing boxes so that the vertical edges of the boxes in one layer are not aligned with the edges of adjacent layers, so that more than one box below supports the box above it, as opposed to the column stacking method, where the vertical edges are aligned. Fig. 1 illustrates the two methods.

A Mixed Integer Linear Programming (MILP) model is proposed to ensure that the boxes positioned at the corners of the pallet in two consecutive layers have different orientations. This constraint prevents the creation of vertical stacks, which have an important negative impact on stability. Different objective functions were proposed for a MILP model and a set of post-optimisation heuristics that mimic the planner's goals in practice. The combination of the different objective functions and post-optimisation heuristics was evaluated with respect to metrics from the literature to evaluate stability.

This paper is organised as follows. In Section 2, a review of the literature concerning MPLP is presented with a classification of the works considering the dimensionality of the problem, the type of solution approach (exact methods or heuristics) and what concerns stability. In Section 3, the problem addressed in this study is presented, describing all its specificities, particularly the interlocking rules that must be followed. In Section 4, the mathematical models implemented in this study are introduced. Then, Section 5 includes the proper clarification and description of the novel static stability metric developed during this study. Section 6 introduces a series of post-processing heuristics designed to rearrange the loading patterns. Section 7 presents all computational experiments carried out during this study and the results obtained during these experiments. Finally, Section 8 unveils the main conclusions derived from this work.

2. Manufacturer's pallet-loading problem - state of art

The MPLP is crucial for logistics activities, particularly in efficient storage and transportation of products packed in boxes and loaded onto pallets and trucks (Morabito & Morales, 1998). However, due to the complexity of MPLP, only a few researchers have proposed exact solution methods to tackle it. This highlights the challenge of finding optimal solutions to such real-world logistics problems.

2.1. Overview of solution approaches to MPLP

The MPLP has been the focus of 26 studies, most of them addressing it as a two-dimensional (2D) problem and using exact approaches (as outlined in Table 1), such as Dowsland (1987a), Dowsland (1987b), Tarnowski et al. (1994), Mascarenhas (2005), Alvarez-Valdes et al. (2005b), Kocjan and Holmström (2010), Ahn et al. (2015), Aljuhani and Papageorgiou (2021), Sawicki and Sawicka (2023). Others have proposed heuristic methods (see Table 1), for example Farago and Morabito (2001), Amaral and Wright (2001), Yamassaki (2003).

Has can be seen in Table 1 other works address the MPLP as a three-dimensional (3D) loading problem, like Liu et al. (2010), Taylor et al. (2017), Almasarwah et al. (2023) and Aylak et al. (2021).

The positive results obtained using heuristics approaches caused the interest of other researchers to develop hybrid methods. It can be observed in Table 1, that the majority of the works use hybrid methods, such as exact algorithms with the incorporation of heuristics (Almasarwah et al., 2023; Alvarez-Valdes et al., 2005a; Birgin et al., 2010, 2005; Gunawardena et al., 2021; Liu et al., 2010; Martins & Dell, 2008; Morabito & Morales, 1998; Pureza & Morabito, 2006; Ribeiro & Lorena, 2007; Taylor et al., 2017; Wu & Ting, 2007; Young-Gun & Kang, 2001). Some of these methods include the conformation of 2D block patterns, referred to as G-structure (guillotine structure). Scheithauer and Terno (1999) presented an heuristic, called the G4-heuristic, which is based on the definition of the so-called G4 structure of packing patterns. The G4 structure developed by the authors is a generalisation of the commonly used block structure of packing patterns, which requires the same orientation of the packed boxes within each block.

As seen in the literature, heuristics are usually used within these solution methods because they allow the possibility of reducing computational time for the MPLP, NP-hard problem. However, even considering this issue, some authors solve the MPLP problem using exact approaches, such as Dowsland (1987a), Dowsland (1987b), Tarnowski et al. (1994), Mascarenhas (2005), Ahn et al. (2015), Kocjan and Holmström (2010), Aljuhani and Papageorgiou (2021) and Sawicki and Sawicka (2023). Other researchers such as Farago and Morabito (2001), Amaral and Wright (2001), Yamassaki (2003) and Aylak et al. (2021) focused their research only on the use of specific heuristic techniques. The last one (Aylak et al., 2021), uses block heuristics to build horizontal layers. The layers in a pallet have the same pattern and overlap between layers is not allowed. Therefore, the horizontal layers are identical and have the same height. Then the number of horizontal layers is computed based on the maximum height and weight of the pallet. It is important to state that although all methods have proven to be efficient in solving the MPLP in one way or another, they have limitations, many of which are related to computational time. For example, block or pattern methods, in certain instances, generate spaces that cannot be completely filled, leading to lower efficiency. However, regardless of the method, especially for large instances, it requires significant computational time and often fails to produce results within a reasonable timeframe. These computational limitations highlight the need for further research into more time-efficient approaches to solving the MPLP.

The constraints inherent in MPLP must be considered in conjunction with resolution methodologies. These constraints can potentially impact the quality of solutions and make certain computational times prohibitive. A review of exact approaches to the MPLP reveals that numerous researchers have accounted for only a limited subset of the extensive considerations and constraints that characterise real-world problems while formulating their models and/ or solution methods.

Numerous constraints must be considered to represent MPLP as authentically as possible. Height and weight invariably impose significant constraints during transport or handling, limiting the number of boxes that can be packed. A maximum height often stems from stringent packing standards set by customers, who dictate the maximum permissible height of a pallet, making it mandatory compliance for all

Table 1
PLP manufacturer's problem Approaches and constraints.

Studies	Dimensionality		Approaches		Constraints			
	2D	3D	Exact methods	Heuristics	Stability	Load bearing	Dynamic stability	Unspecified
Dowsland (1987a)	✓		✓		✓			
Dowsland (1987b)	✓		✓					✓
Tarnowski et al. (1994)	✓		✓					
Morabito and Morales (1998)	✓		✓	✓				✓
Farago and Morabito (2001)	✓			✓				✓
Amaral and Wright (2001)	✓			✓				✓
Young-Gun and Kang (2001)	✓		✓	✓				✓
Yamassaki (2003)	✓			✓				✓
Alvarez-Valdes et al. (2005a)	✓		✓	✓	✓			
Birgin et al. (2005)	✓		✓	✓				✓
Mascarenhas (2005)	✓		✓					✓
Alvarez-Valdes et al. (2005b)	✓		✓					✓
Pureza and Morabito (2006)	✓		✓	✓				✓
Wu and Ting (2007)	✓		✓	✓				✓
Ribeiro and Lorena (2007)	✓		✓	✓				✓
Birgin et al. (2010)	✓		✓	✓				✓
Martins and Dell (2008)	✓		✓	✓		✓		
Liu et al. (2010)		✓	✓	✓	✓		✓	
Ahn et al. (2015)	✓		✓					✓
Kocjan and Holmström (2010)	✓		✓		✓			
Taylor et al. (2017)		✓	✓	✓	✓	✓	✓	
Aljuhani and Papageorgiou (2021)	✓		✓					✓
Gunawardena et al. (2021)	✓		✓	✓	✓			
Aylak et al. (2021)		✓		✓	✓		✓	
Almasarwah et al. (2023)		✓	✓	✓	✓			
Sawicki and Sawicka (2023)	✓		✓			✓		

Table 2
Loading schemes and stability metrics.

	Loading schemes	Metrics	Instances
Dowsland (1987a)	-	- CPU time	Cover I
Alvarez-Valdes et al. (2005a)	- Waste between boxes - Overlapping	- CPU time - N° of nodes	Cover I; and Cover II
Yi et al. (2009)	- Pinwheel pattern	- N° of boxes - Hole size	Random generated instances based on Cover I and Cover II
Kocjan and Holmström (2010)	- Base support - Base contact - Non guillotine cutting pattern	- Stability - Optimality	Cover I; and 500 randomly generated instances
Taylor et al. (2017)	- Layer pattern - Stacking rule	- Coverage area - N° of boxes	Random generated instances
Gunawardena et al. (2021)	- Overhanging - Block, H-Box or V-Box	- N° of boxes - Overhanging - Stability	15 real instances
Aylak et al. (2021)	- Layer pattern - Block building - Dynamic compression	- Pallets height - Stability	15 real datasets
Almasarwah et al. (2023)	- Guillotine cutting pattern - Dynamic compression	- Stability - Volume utilisation - N° of boxes/pallets per layer	52 datasets randomly generated

suppliers involved. Height is crucial because transportation vehicles, storage racks, and other handling devices have specific dimensions that cannot be exceeded. Similarly, the maximum weight imposes another critical limitation for palletising, influenced by transportation conditions, customer requirements, and handling capacities, among other factors.

The practical importance of load stability contrasts sharply with its limited treatment in MPLP literature. As shown in Table 1, only 8 of the 26 reviewed studies explicitly incorporate stability constraints (Almasarwah et al., 2023; Alvarez-Valdes et al., 2005a; Aylak et al., 2021; Dowsland, 1987a; Gunawardena et al., 2021; Kocjan & Holmström, 2010; Liu et al., 2010; Taylor et al., 2017). This gap is concerning because unstable pallet loads create significant operational challenges during handling, transportation, and storage, including product damage and safety risks.

Among stability-aware approaches, Liu et al. (2010), Taylor et al. (2017), and Aylak et al. (2021) consider both static and dynamic stability constraints, while Taylor et al. (2017), and Sawicki and Sawicka (2023) additionally incorporate load-bearing capacity constraints that account for compression strength limitations of boxes and pallets.

2.2. Stability constraints

Beyond volume maximisation, load stability represents a critical practical constraint in pallet loading operations. Damage to products during handling and transportation imposes significant costs on supply chains, which makes stability optimisation essential for industrial applications. This section reviews approaches that explicitly model stability constraints.

In short, very few works have studied this problem, including the stability constraint (Table 1; Table 2). The seminal work by Dowsland

(1987a) addresses the two-dimensional packing problem, focusing on the generation of various box layouts intended for the subsequent placement of pallets. Employing an exact tree search methodology grounded in a graph-theoretical model, Dowsland introduced a class of six data sets derived from various global pallet standards. These data sets are known as Cover I and can accommodate up to 50 boxes. The performance of this approach was assessed using CPU time as an evaluation metric.

Another pioneering study that considers the stability of three-dimensional packing is presented in Bischoff (1991). This study considers the stability in the context of pallet loading and examines approaches to generate stable stacking patterns that are also optimal from the point of view of space use. It uses three criteria, based on Carpenter and Dowsland (1985) work, for the evaluation of packing stability:

1. Each box in a layer has to be supported by at least two boxes in the layer below, not counting contact areas of less than 5% of the base area of a box.
2. At least 85% of the base area of a box in a layer needs to be in contact with the boxes in the layer below.
3. Each box positioned on the perimeter of the pattern has to be supported by at least two boxes in the layer below, not counting contact areas of less than 5% of the base area of a box.

The authors use problem instances made up of a random sample of box/ pallet combinations, generating different layouts using the algorithm of Bischoff and Dowsland (1982). The approach was tested with 5000 randomly generated pallet and box combinations (layouts) using a similar generating routine described by Bischoff (1991) and Carpenter and Dowsland (1985).

Alvarez-Valdes et al. (2005a) adapted the binary linear programming model for the 2D non-guillotine cutting problem proposed by Bischoff and Dowsland (1985). They used a branch-and-cut algorithm that explores the structure of the problem to define the solution graph with an efficient separation procedure. They tested the approach using the Cover I and Cover II problem classes using the CPU time and the number of nodes as metrics. Another work presented in Table 2 is Yi et al. (2009), which presented a pinwheel pattern as an alternative solution to the pallet-loading problem. They propose a uniform notation and the practical advantages and applications of this alternative. The authors state that the most relevant practical advantages lie in: stability, dynamic stability, balance, contour, and multiface. Due to some ambiguities detected by the authors in the Cover I and Cover II class problem tests and already pointed out by other authors in other works, the pinwheel pattern approach was tested using a data set generated and proposed by the authors.

Kocjan and Holmström (2010) presented an integer linear programming model to solve MPLP using several stability criteria to generate stable loading patterns. They tested the model using Cover I problem classes and the B-Type problem sets by Liu and Hsiao (1997) and proved that the proposed model is very effective for moderate-size problem instances. The same authors showed that the stability criteria can be treated as the main objective without compromising the utilisation of the pallet. The scheme used to generate stable two-dimensional loading patterns was the same scheme proposed by Carpenter and Dowsland (1985):

- supportive criterion: each box must be supported by at least two boxes from the layer below, disregarding boxes with contact area less than X%;
- base contact criterion: at least Y% of the base area of each box is supported by boxes from the layer below;
- non-guillotine criterion: no straight or jagged cuts are running across the length or width of the pallet longer than Z% of the length/width of a pallet.

Taylor et al. (2017) deals with stability using a two-stage heuristic where the layers are built in the first stage, and in the second stage, a heuristic layering approach is used to pack the pallets. To stack the layers into pallets, the authors use a staking rule that starts with the tallest layers in the base of the pallet. Gunawardena et al. (2021) deals with the stability caused by the overhanging of items, and that is one of the most critical constraints and also the most neglected in MPLPs. However, the authors treat the instability as a result of the overhanging of the base of the items in the configuration of the pallet itself. This overhanging is a cause of the reduced strength of the pallet. However, due to practicalities, it is common for warehouses to allow these protrusions of the items on the pallets in order to minimise costs and reduce the number of required pallets. Thus, the authors consider an item to be stable when at least more than 50% of the base is supported by the pallet base.

The stability of a pallet in the work presented by Aylak et al. (2021) is evaluated according to the dynamic compression, which is also taken into account when building the pallet. Dynamic compression is measured considering that the average load on a box at the bottom layer of the pallet should be less than its dynamic compression strength, considering humidity, interlock stacking pattern, box dimensions, and storage time.

The most recent work, published in 2023, that considers load stability is from Almasarwah et al. (2023) (Table 2). Like in the previous work, here, the authors also use dynamic compression strength to evaluate pallet stability. This work deals with three-dimensional MPLP, where a second phase of the proposed approach improves the solution generated in the first phase, not only in terms of volume utilisation but also in terms of the stability of the pallets. However, the measure of load stability is related to the pallet's height and the boxes' dynamic compression strength, which is maximised by creating an interlocking stacking pattern.

For the MPLP, depending on the type of load that is palletised for transport and storage, an additional metric can be considered: the load-bearing capacity of boxes/pallets and their dynamic compression strength. The study by Singh et al. (2019) shows that the load-bearing capacity and stability of items/ pallets are significantly affected by factors such as relative humidity and storage time. High humidity and prolonged storage can reduce the load-bearing capacity by more than 50%. Additionally, they find that dynamic compression resistance is weakest when there is minimal weight at the bottom, which can occur with a higher number of boxes per horizontal layer. Therefore, metrics to assess the stability of palletised loads can vary depending on the type of item, environmental conditions, and operational processes. In practical applications, additional metrics beyond the three used in this study may be required to ensure a comprehensive evaluation of load stability.

2.2.1. Load stability analysis

Product losses during transit are common and often perceived as unavoidable, leading logistics managers to anticipate that certain products will suffer damage during handling and transportation and potentially cause injury to employees. A critical factor in ensuring the integrity of the products upon arrival is the stability of the pallet loads. When companies palletise loads, they typically consider the following factors to optimise the stability of the packaging.

- Pallet Weight Limits: The stability of the pallet load will be compromised if the load weight exceeds the weight limit of the pallet.
- Weight Distribution: The way in which the weight of the products is distributed on a pallet is crucial.
- Pallet Utilisation: Products loaded onto a pallet should use as much space as possible without overhanging at the edges.
- Stack Height: The maximum safe height at which products can be stacked on a pallet is related to the forklift and the operator's ability to handle the pallet.

- Pallet Wrapping: Once a pallet is loaded, the cartons will have to be tightly attached to each other and to the pallet with plastic wrap.

As discussed earlier, few theoretical approaches address pallet loading with a focus on stability. Due to this, there is a scarcity of information on key performance indicators to evaluate load stability. The measurement system used in this study to measure the stability of the cargo is derived from Carpenter and Dowsland (1985) and was subsequently adopted by Bischoff (1991), Bischoff and Ratcliff (1995), and Moura and Oliveira (2005). In the latter two studies, the stability criteria were also modified and applied to the 3D container loading problem (CLP).

According to the heuristic's constructive nature, in a loading pattern, the boxes may be allowed to be placed partially (or fully) suspended, or all boxes have their base fully supported, either on the pallet/ container floor or on top of one or more boxes. In this case, the stability measures required are as follows:

- **Metric 1 - Interlocking** - Average number of boxes by which the boxes are supported (except those positioned directly on the pallet/container). Carpenter and Dowsland (1985), suggest that for a load to be stable, each box should be supported by at least two boxes. This measure does not indicate the possibility that a box has lateral movements, so another measure is also required.
- **Metric 2 - Lateral support** - Average percentage of boxes surrounded on at least three sides. The lateral support of a box is understood as the face of the other boxes or the plastic wrap of the pallet (or the container walls in a CLP).

With these two measures, we can evaluate the performance of the approach with respect to the stability of the cargo. The higher the value, the better.

2.3. MSI and computational efficiency considerations

An important computational consideration in MPLP research is the concept of the Minimum Size Instance (MSI), introduced by Martins and Dell (2007). The MSI represents the smallest instance within an equivalence class of MPLP problems that share the same optimal packing pattern. Two instances (L, W, a, b) and (L', W', a', b') belong to the same equivalence class if their optimal loading patterns are identical up to scaling. Martins and Dell (2008) demonstrated that the MSI can be computed by finding the greatest common divisor (GCD) of the pallet and box dimensions and dividing all dimensions by this GCD. This reduction significantly decreases the computational burden of solving MPLP instances, as the complexity of integer programming formulations is directly related to the magnitude of the coefficients in the constraints.

Lu and Cha (2014) further refined this approach by proposing a fast algorithm for MSI identification that operates in O(log(min(L,W,a,b))) time complexity. Their method enables rapid preprocessing of MPLP instances before applying optimisation algorithms, making it particularly valuable for industrial applications where numerous instances must be solved in sequence.

However, it is crucial to note that MSI reduction is not appropriate for problems involving stability constraints such as interlocking. While MSI preserves the combinatorial box placement pattern, it does not preserve the geometric relationships between boxes in stacked layers that determine stability. The degree of overlap between boxes in adjacent layers, the support polygon geometry, and the effectiveness of interlocking mechanisms all depend on the precise dimensional ratios of the original instance. Therefore, our interlocking-based approach requires working with the actual pallet and box dimensions rather than reduced MSI representations.

In addition to the geometric constraints intrinsic to MPLPs, this study addresses the load stability constraint through interlocking. In this way of dealing with stability and using three types of interlocking, the creation of vertical stacks is prevented, which has a negative effect on load stability. To assess the stability of the solutions, the aforementioned metrics are used.

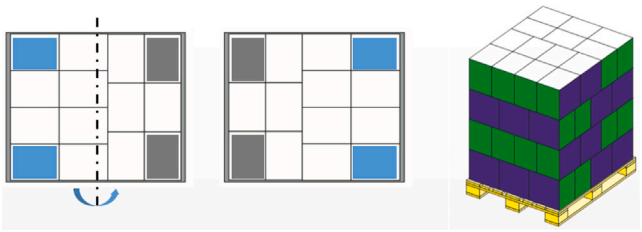


Fig. 2. Type 1 interlocking.

3. Problem definition

The problem can be formulated as follows. Consider a large rectangle, denoted as a pallet, with specified length L and width W, with $L \geq W$, and a smaller rectangle, referred to as a box, with length a and width b, with $L \geq a$, $W \geq b$, $a \geq b$. The objective is to maximise the number of boxes placed on the pallet. Each instance of the MPLP is represented by a quadruple (L, W, a, b).

A horizontally orientated box is defined as a H-box, whereas a vertically orientated box is defined as a V-box. The positions of the boxes are defined with reference to a coordinate system that originates in the bottom left corner of the pallet. Geometric constraints are imposed to ensure that the boxes remain non-overlapping and entirely within the boundaries of the pallet.

The arrangement of the boxes on the pallet is called a layer. A set of stacked layers forms a pallet loading plan. The total number of layers depends on the height of the boxes and the maximum height defined for the pallet. When the layers are vertically stacked to form a loading plan, stability also becomes a constraint. The concept of interlocking is used to deal with stability. Interlocking ensures that the boxes that are placed in the corners of the pallet in two consecutive layers have different orientations, preventing the creation of vertical stacks, which have a negative impact on stability.

In this work, three types of interlocking configurations are considered. Fig. 2 illustrates a Type 1 interlocking. The left side of Fig. 2 shows an initial layer that, when vertically mirrored, results in a transformed layer where the orientation of the corner boxes is reversed (shown in the centre of Fig. 2). Specifically, in the original layer, the boxes at the left corners are H-boxes (horizontally orientated), while those at the right corners are V-boxes (vertically orientated). After vertical mirroring, the orientation switches: the left corner boxes become V-boxes, and the right corner boxes become H-boxes. The final pallet loading plan consists of four layers, ensuring that the orientation of the corner boxes alternates between consecutive layers, enhancing stability (as shown on the right side of Fig. 2).

Fig. 3 illustrates a Type 2 interlocking configuration. The initial layer on the left side of Fig. 3, mirrored horizontally, results in a transformed layer where the orientation of the corner boxes is reversed (shown in the centre of Fig. 3). In the original layer, the top corners contain V-boxes (vertically orientated), while the bottom corners contain H-boxes (horizontally orientated). After horizontal mirroring, the top corners become H-boxes, and the bottom corners become V-boxes. The final pallet loading plan, similar to Type 1 interlocking, comprises four layers, with alternating corner box orientations between consecutive layers, ensuring stability (as shown on the right side of Fig. 3).

Fig. 4 presents a representation of a Type 3 interlocking configuration. The initial layer, shown on the left side of Fig. 4, is mirrored horizontally or vertically, resulting in a transformed layer in which the orientation of the corner boxes is reversed (as shown in the centre of Fig. 4). In the original layer, the top left and bottom right corners contain V-boxes (vertically orientated), while the top right and bottom left corners contain H-boxes (horizontally orientated). After simultaneous horizontal or vertical mirroring, the top-left and bottom-right corners

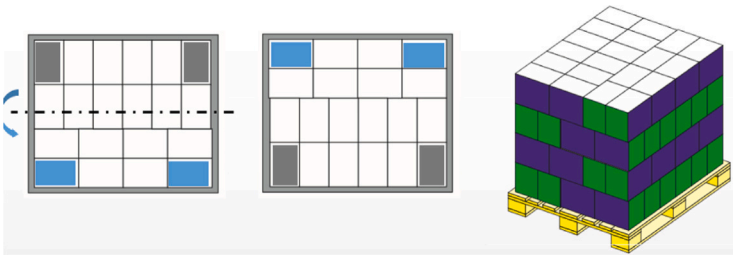


Fig. 3. Type 2 interlocking.

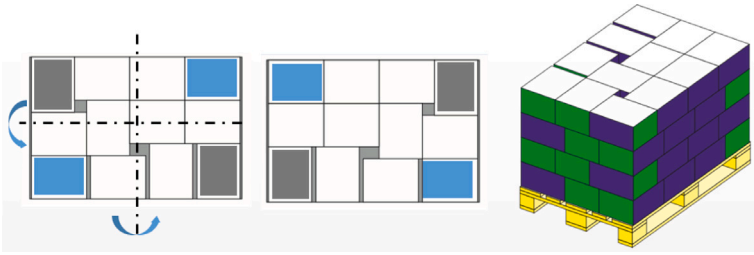


Fig. 4. Type 3 interlocking.

contain H-boxes, while the top-right and bottom-left corners contain V-boxes. As with interlocking Types 1 and 2, the final pallet loading plan consists of four layers, with alternating corner box orientations between consecutive layers to enhance stability (as shown on the right side of the figure).

Our interlocking-based formulation differs fundamentally from traditional MPLP approaches that optimise solely for volume maximisation. While the combinatorial structure of box placements may be preserved under dimensional scaling through MSI reduction, the geometric relationships that determine stability are not scale-invariant. Consequently, our formulation must operate on actual dimensional values to accurately capture and optimise stability characteristics.

4. A mixed-integer linear programming model

This section is dedicated to mathematical models. All the mathematical models developed in this work and presented in the following were based on the formulation proposed by Alvarez-Valdes et al. (2005a). Additional objective functions and constraints were added to address the problem constraints and objectives.

4.1. Base formulation with interlocking constraints

Decision Variables:

$$h_{kj} = \begin{cases} 1 & \text{if a H-box is placed in position } (k,j), k = 0,\dots,L-a; j = 0,\dots,W-b, \\ 0 & \text{otherwise.} \end{cases}$$

$$v_{kj} = \begin{cases} 1 & \text{if a V-box is placed in position } (k,j), k = 0,\dots,L-b; j = 0,\dots,W-a, \\ 0 & \text{otherwise.} \end{cases}$$

Fig. 5 represents the possible orientations of a box and the respective decision variables.

The mathematical formulation for the MPLP proposed considering interlocking is the following:

$$F1 = \text{Maximise} \sum_{k=0}^{L-a}\sum_{j=0}^{W-b} h_{kj} + \sum_{k=0}^{L-b}\sum_{j=0}^{W-a} v_{kj} \tag{1}$$

Subject to:

$$\sum_{k=max\{0,r-a+1\}}^{min\{r,L-a\}}\sum_{j=max\{0,s-b+1\}}^{min\{s,W-b\}} h_{kj} + \sum_{k=max\{0,r-b+1\}}^{min\{r,L-b\}}\sum_{j=max\{0,s-a+1\}}^{min\{s,W-a\}} v_{kj} \le 1$$

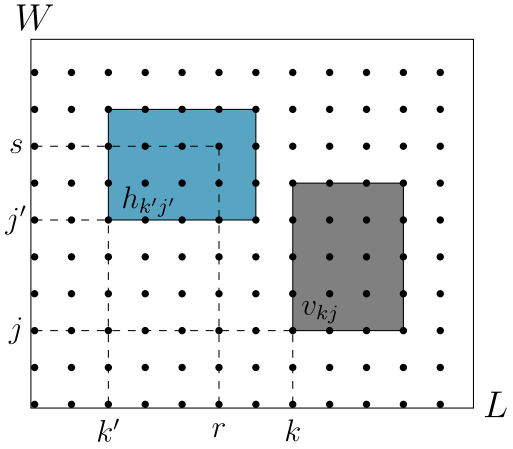


Fig. 5. Possible Boxes Orientations.

$$\forall(r = 0,\dots,L-b; s = 0,\dots,W-b) \tag{2}$$

$$\sum_{k=0}^{min\{b-1,L-a\}}\sum_{j=0}^{min\{b-1,W-b\}} h_{kj} + \sum_{k=0}^{min\{b-1,L-a\}}\sum_{j=max\{0,W-2b+1\}}^{W-b} h_{kj} + \sum_{k=max\{0,L-b-a+1\}}^{L-a}\sum_{j=max\{0,W-2b+1\}}^{W-b} h_{kj} + \sum_{k=max\{0,L-b-a+1\}}^{L-a}\sum_{j=0}^{min\{b-1,W-b\}} h_{kj} = 2 \tag{3}$$

$$\sum_{k=0}^{min\{b-1,L-b\}}\sum_{j=0}^{min\{b-1,W-a\}} v_{kj} + \sum_{k=0}^{min\{b-1,L-b\}}\sum_{j=max\{0,W-b-a+1\}}^{W-a} v_{kj} + \sum_{k=max\{0,L-2b+1\}}^{L-b}\sum_{j=max\{0,W-b-a+1\}}^{W-a} v_{kj} + \sum_{k=max\{0,L-2b+1\}}^{L-b}\sum_{j=0}^{min\{b-1,W-a\}} v_{kj} = 2 \tag{4}$$

$$h_{kj} \in \{0,1\}, \quad \forall(k = 0,\dots,L-a; j = 0,\dots,W-b) \tag{5}$$

$$v_{jk} \in \{0,1\}, \quad \forall(k = 0,\dots,L-b; j = 0,\dots,W-a) \tag{6}$$

In the original model proposed by Alvarez-Valdes et al. (2005a), which consists of the mathematical model without interlocking ((1), (2), (5) and (6)), there was only one objective function ($F1$), which

maximises the number of boxes placed, through the sum of the number of boxes placed horizontally with the number of boxes placed vertically.

This objective function is inherited from the original model and is pivotal in pallet-loading models for improving the efficiency of pallet-loading operations. In this regard, decision-makers typically select patterns with the highest number of loaded boxes.

Constraints (2) are cover constraints that avoid box overlapping. For each pair $(r; s)$, this constraint guarantees that the corresponding unit square is covered by one box at most.

To incorporate interlocking into the MILP model, constraints (3) and (4) are introduced. These constraints ensure the interlocking mechanism by requiring that at the four corners of the pattern, two boxes are positioned in one orientation and the other two are rotated by 90 degrees. This arrangement guarantees that at least one of the three types of interlocking is achieved, thus inhibiting stack formation at the corners of the pallet.

The last set of constraints (5)–(6) define the domain of the decision variables.

4.2. Lexicografic multi-objective extensions

However, when implemented in practical scenarios, the model previously presented may exhibit certain limitations. For example, in most cases, solutions are produced in which boxes are dispersed across the pallet, generating empty spaces. Even though this was not important for the objective goal of the previous model, which was to maximise the number of placed boxes, in a real-world situation, this is not desired as it facilitates the movement of boxes during transportation.

Taking into account the context above, four additional objective functions were formulated in addition to the main objective function. The new objective functions were included in a multiobjective lexicographic function, with the original objective function ($F1$) always being the most important. Consequently, the main goal is always to maximise the number of boxes placed in the pallet, but additional goals would be included.

4.2.1. Optimisation of the compactness directed downwards and to the left of the pallet

During pallet loading operations, efficiency is not the only factor to consider, stability is equally essential for the reasons outlined above. Yet, when $F1$ was the only objective function implemented in the model, the generated patterns often showed multiple idle spaces between boxes, which reduced pattern compactness and, consequently, the stability of the cargo. Therefore, the objective function $F2$ was added to maximise the compactness of the generated pattern, which typically leads to more stable arrangements. As such, to enhance stability during loading and transportation, patterns with greater compactness are usually chosen.

This objective function maximises the compaction of the placed boxes, forcing all of them to be the most downward and left that they can possibly be. Therefore, all boxes placed are compact to the bottom left corner of the pallet.

$$
\begin{aligned}
F2 = \text{Maximise}\Bigg(& \sum_{k=max\{0,L-b-a+1\}}^{L-a} \sum_{j=0}^{W-b} h_{kj} \cdot (L-k-a) \cdot b + \\
& + \sum_{k=max\{0,L-2b+1\}}^{L-b} \sum_{j=0}^{W-a} v_{kj} \cdot (L-k-b) \cdot a + \\
& + \sum_{k=0}^{L-a} \sum_{j=max\{0,W-2b+1\}}^{W-b} h_{kj} \cdot (W-j-b) \cdot a + \\
& + \sum_{k=0}^{L-b} \sum_{j=max\{0,W-b-a+1\}}^{W-a} v_{kj} \cdot (W-j-a) \cdot b \Bigg) \Big/ \Big(b \cdot W + b \cdot (L-b) \Big)
\end{aligned} \tag{7}
$$

To achieve compactness, the objective function seeks to maximise the proportion of idle space located on the top and right margins of the pallet, thus compelling the model to concentrate all idle spaces in these areas. This is illustrated in Fig. 6, where the objective function strives to optimise the idle spaces (represented in green) within the red polygon.

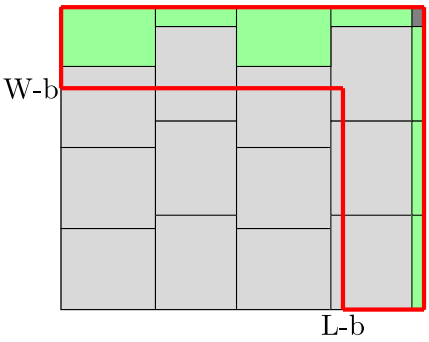


Fig. 6. F2 components.

As illustrated in Fig. 7(a), prior to the application of additional objective functions, the unused space was dispersed across the pallet, resulting in gaps between the boxes placed. In contrast, upon implementing the objective function $F2$, the unused space is relocated to the upper and right ends, thus improving the compaction of the cargo and optimising the use of the available areas in the upper and right margins (Fig. 7(b)).

4.2.2. Optimisation of the compactness directed to the left of the pallet

The objective function $F3$ was incorporated with the goal of consolidating the load on the left side of the pallet. To achieve this, the objective function maximises the unused space between the rightmost box and the right edge of the pallet. However, to achieve the desired results, two additional constraints (9)–(10) have to be integrated into the model (2)–(6), together with an additional decision variable, C^H_{max}, which denotes the maximum value on the x-axis across all loaded boxes.

$$F3 = \text{Maximise}(L - C^H_{max}) \tag{8}$$

Subject to:

$$h_{kj} \cdot (k+a) \le C^H_{max}, \quad \forall (k = 0, \dots, L-a; j = 0, \dots, W-b) \tag{9}$$

$$v_{kj} \cdot (k+b) \le C^H_{max}, \quad \forall (k = 0, \dots, L-b; j = 0, \dots, W-a) \tag{10}$$

Although this objective function may seem quite similar to $F2$, different aspects differentiate them. Primarily, $F2$ exerts influence in two directions, pushing the boxes both downward and to the left simultaneously, whereas $F3$ only forces the boxes to move to the left. In addition, $F2$ is also characterised by trying to compact all boxes on the pallet, forcing all to the left and downward. On the other hand, $F3$ only acts directly on one box, the one occupying the rightmost position on the pallet. Therefore, this function does not directly impose the compaction of all the boxes; nevertheless, this outcome is indirectly achieved by exerting force on the rightmost box.

Fig. 7(c) shows the impact of incorporating the objective function $F3$.

Introduced as an alternative to $F2$, this objective function serves precisely the same purpose: increasing compactness to enhance the stability of the arrangement. However, $F3$ compacts the cargo only in one direction, longitudinally, which is often a requirement. Moreover, by adding this objective function, it also enabled an interesting comparison with the patterns generated with $F2$, revealing which objective function leads to the generation of more stable arrangements.

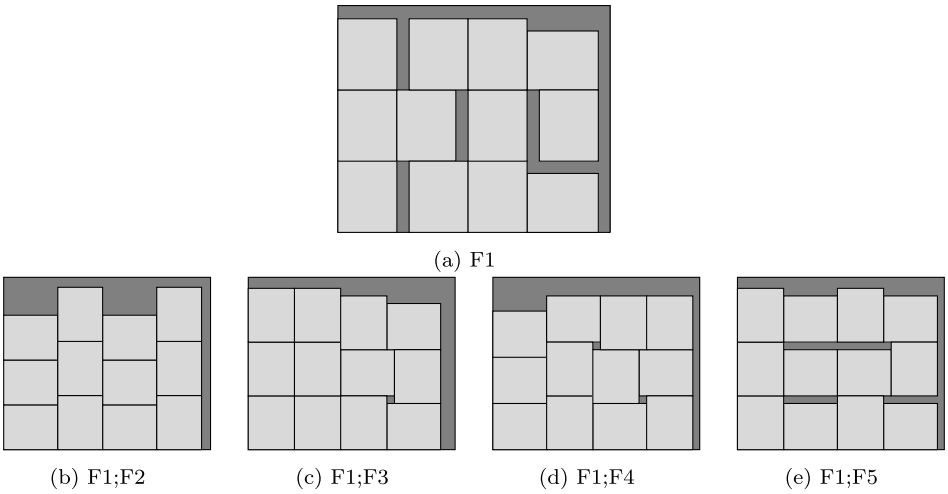


Fig. 7. Before and After new objective functions implementation.

4.2.3. Optimisation of the compactness directed to the bottom of the pallet

$$F4 = \text{Maximise}(W - C^V_{max}) \tag{11}$$

Subject to:

$$h_{kj} \cdot (j + b) \leq C^V_{max}, \quad \forall (k = 0, \dots, L - a; j = 0, \dots, W - b) \tag{12}$$

$$v_{kj} \cdot (j + a) \leq C^V_{max}, \quad \forall (k = 0, \dots, L - b; j = 0, \dots, W - a) \tag{13}$$

The objective function $F4$ was incorporated into the original model to compact the load downward on the pallet. This function maximises the idle space between the box positioned highest on the pallet and the top margin of the pallet. Additionally, it requires the incorporation of a new pair of constraints (12)–(13) into the model, as well as of another decision variable, C^V_{max}, which represents the maximum value on the y-axis across all loaded boxes.

Provides an additional alternative objective function to compact the cargo, since there are also situations where it is preferable to compact the cargo only transversally, and the addition of this objective function allows that. Moreover, it also grants an interesting complement to $F3$, as it compacts the cargo in a distinct direction. For that reason, during the testing stage, different combinations of these two objective functions were tested to determine which produced better stability results.

Fig. 7(d) illustrates the outcome of incorporating this function.

It is imperative to note that not only this function, but also the two previous ones, are invariably executed in conjunction with the interlocking constraints, which, in most cases, restrain the compaction of the load, thereby preventing the generation of more compacted patterns by these functions.

4.2.4. Optimisation of the number of boxes placed horizontally and vertically

The objective function $F5$, aims to minimise the difference between the number of boxes placed horizontally and the number of boxes placed vertically.

$$F5 = \text{Minimise} \left| \sum_{k=0}^{L-a} \sum_{j=0}^{W-b} h_{kj} - \sum_{k=0}^{L-b} \sum_{j=0}^{W-a} v_{kj} \right| \tag{14}$$

As such, when this objective function is combined with the original one $F1$, the model is expected first to maximise the number of boxes to be placed on the pallet and then to find the arrangement where the number of boxes placed horizontally is closer to the number of boxes placed vertically. This objective function was added to the model with the intention of offering an additional strategy to improve the level of interlocking and thereby potentially contribute to the generation of more stable arrangements.

Fig. 7(e) illustrates a scenario in which the incorporation of the objective function $F5$ performed as expected. Firstly and more importantly, the number of boxes placed on the pallet remained unchanged. Furthermore, the arrangement was successfully reorganised, resulting in a configuration in which the number of boxes positioned horizontally equals that of the boxes positioned vertically.

All the patterns presented in Fig. 7 were generated for a real-world instance, which encompasses a 1200 × 1000 mm^2 pallet to be loaded with identical boxes exhibiting length equal to 314 mm and width equal to 260 mm. This instance can be found in Araújo et al. (2025) under the name *ReadData_1_145*.

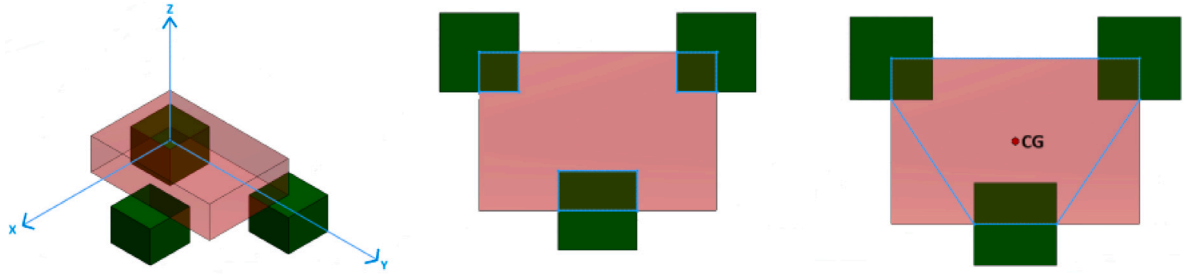
5. New static stability metric

In addition to the two metrics previously discussed in Section 3 for evaluating the static stability of the placed boxes, a third metric has been developed based on the concept of the support polygon. This concept posits that for a box to be considered stable, its centre of gravity must lie within its support polygon, as illustrated in Fig. 8.

In Fig. 8, it is possible to see that there is a red box supported by three other green boxes, those of their contact areas drawn in blue (Fig. 8(b)). The convex hull generated using the vertices of the contact areas corresponds to the support polygon of the red box, highlighted in blue (Fig. 8(c)). Considering that the centre of gravity (indicated by the red dot labelled 'CG') of the red box is positioned within the support polygon, the red box can be deemed stable. In contrast, in scenarios where the centre of gravity lies outside the support polygon, the red box should be regarded as unstable.

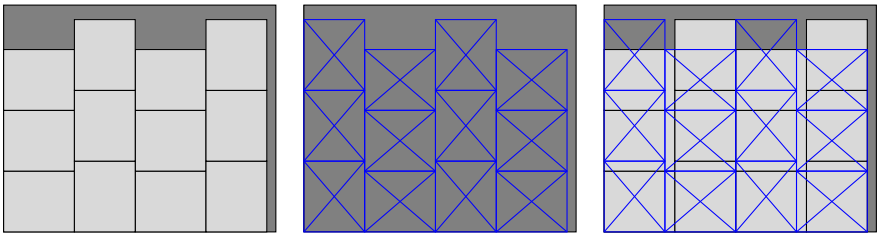
Identifying the support polygon for a specific box is contingent on prior knowledge of the boxes providing support and their respective configurations. Therefore, it was assumed that during the loading procedures, the two potential configurations (original and mirror) would alternate in successive layers, ensuring interlocking. This means that on top of a layer with the original pattern, there will always be a layer with the mirror pattern, and the same also happens the other way around. As such, to assess the relative position of each box compared to the boxes in the layer below, both patterns should be overlapped, as shown in Fig. 9.

Metric 3 - Support Polygon (Percentage) -The average percentage of support polygon area. The support polygon is always defined within the limits of the box in cause, so its area will always be less than or



(a) Isometric View (b) Contact Areas (c) Support Polygon

Fig. 8. Support Polygon Explanation.



(a) Original (b) Mirror (c) Overlap

Fig. 9. Vertical Mirror.

equal to the area of that box. As such, to determine the value of this metric, the support polygon is defined for every single one of the placed boxes, one at a time, being checked afterwards to determine which percentage of this box area is within its own support polygon area. The same procedure is repeated for every box and the average is calculated.

This procedure is undertaken only after the centre of gravity is verified to be within the support polygon. In contrast, if the centre of gravity is outside the support polygon, a value of 0%

This procedure is done for both possible patterns, first checking all the percentages of support polygon area for each box in the original patterns. Afterwards, the same is done for the boxes in the mirror pattern. In the end, the average of all of them is determined and that value is defined as the final value of this metric.

$$\textbf{Metric3} = \frac{\sum_{k=0}^{L-a}\sum_{j=0}^{W-b}\frac{h_{kj}\cdot sh_{kj}\cdot A_{kj}}{(a\cdot b)} + \sum_{k=0}^{L-b}\sum_{j=0}^{W-a}\frac{v_{kj}\cdot sv_{kj}\cdot A_{kj}}{(a\cdot b)}}{N} * 100 \tag{15}$$

N = Number of boxes in each layer

$$sh_{kj} = \begin{cases} 1 & \text{if the CG of a H-box placed in position } (k,j), k = 0,\dots,L-a; \\ & j = 0,\dots,W-b, \text{is inside its own support polygon} \\ 0 & \text{otherwise.} \end{cases}$$

$$sv_{kj} = \begin{cases} 1 & \text{if the CG of a V-box placed in position } (k,j), k = 0,\dots,L-b; \\ & j = 0,\dots,W-a, \text{is inside its own support polygon} \\ 0 & \text{otherwise.} \end{cases}$$

A_{kj} = Support Polygon Area of the box placed in position$(k,j), k = 0,\dots,L-b; j = 0,\dots,W-b$

We deliberately present three separate stability metrics instead of combining them into a composite score. Each metric addresses a distinct failure mode—compression and vertical stability (Metric 1), sliding (Metric 2), and toppling risk and fundamental geometric stability (Metric 3). The relative importance of each depends on the operational context: transport would prioritise Metric 2, storage would prioritise Metric 1, and manual handling would prioritise Metric 3. Separate presentation enables informed, context-appropriate pattern selection, rather than optimising an arbitrary weighted combination. This approach is consistent with established practise in the stability assessment literature (Bischoff, 1991; Carpenter & Dowsland, 1985).

6. Post-processing heuristics

After obtaining optimal box placement patterns from the MILP models described in Section 4, we apply post-processing heuristics to generate types of spatial configurations that are commonly required or preferred in real-world warehouse and logistics operations. In practice, decision-makers and customers frequently request specific layout types — for instance, a centred load for balanced forklift handling, evenly distributed boxes across the pallet surface for uniform weight distribution, or loads compacted within the minimum bounding rectangle. These are practical operational requirements that arise independently of stability optimisation. The heuristics presented in this section address these requirements by transforming the MILP-generated patterns into the desired configurations. The stability of the resulting patterns is subsequently evaluated (Section 7.4) to quantify the impact of these practically-motivated configurations on load stability, thereby providing decision-makers with the information needed to make informed trade-offs between layout requirements and stability.

The seven heuristics presented in this section perform spatial redistribution of empty space within the pallet. Specifically, they:

- Adjust the absolute coordinates of boxes to redistribute gap spaces.
- Create patterns ranging from highly compact (all boxes tightly grouped) to evenly distributed (gap spaces spread uniformly).
- Generate multiple alternative arrangements from a single MILP solution.

It must be clear that the post-processing heuristics do not:

- change box orientations: A box placed horizontally (H-box) by the MILP remains horizontal; a vertical box (V-box) remains vertical;
- alter relative positions: Boxes cannot be moved past each other; their topological arrangement (which boxes are adjacent to which) remains fixed;

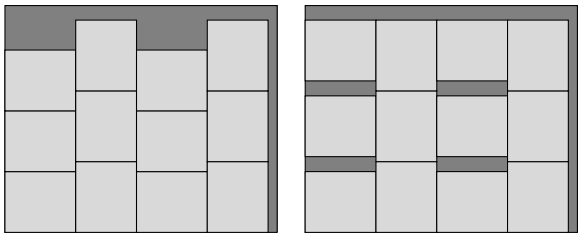


Fig. 10. Spread Empty Spaces in Vertical (Across Occupied Perimeter).

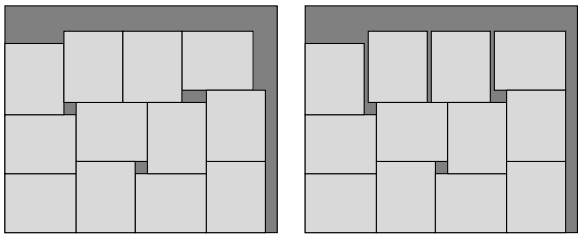


Fig. 11. Spread Empty Spaces in Horizontal (Across Occupied Perimeter).

- change box count: The number of boxes on the pallet is determined by the MILP and cannot be increased by heuristics;
- restructure the packing pattern: The fundamental combinatorial structure (row/column configuration, placement topology) is preserved.

In essence, these heuristics are gap space management strategies, not pattern redesign algorithms. They redistribute empty space within a fixed combinatorial structure determined by the MILP solver to produce layout types that reflect real-world operational preferences. It is important to note that these heuristics are not intended to improve stability; their purpose is to provide configurations that practitioners require in practice. The stability analysis in Section 7.4 quantifies how these configurations affect load stability, enabling informed decision-making.

6.1. Spread empty spaces along the pallet width (across MBR)

In order to illustrate this heuristic, it is first essential to comprehend the concept of the Minimum Bounding Rectangle (MBR). In most scenarios, the perimeter of the pallet diverges from the perimeter actually occupied by the cargo. Consequently, the MBR is defined as the perimeter occupied by the cargo, specifically, the smallest rectangle that fully encapsulates the cargo.

Thereby, this heuristic intends to take the original layout and then spread the boxes along the width of the pallet, without, however, going beyond the width of the MBR. As such, after applying this heuristic, it is expected to obtain a pattern in which all columns end at the same point, and in each of the columns, the empty spaces are spread homogeneously between the boxes, as in the example shown in Fig. 10.

6.2. Spread empty spaces along the pallet length (across MBR)

This heuristic exhibits a behaviour analogous to the previous one, with the primary distinction that it distributes the boxes along the length of the pallet rather than its width. The sole constraint in this approach is that the boxes must not exceed the original length of the MBR, as illustrated in Fig. 11.

6.3. Spread empty spaces along the pallet width (across pallet)

As the name suggests, this heuristic resembles the first one, once it also spreads the boxes along the pallet width. However, in this heuristic, the limit is not the original width of the MBR, but the width of the pallet. Therefore, after applying this heuristic, the empty spaces are homogeneously spread between the boxes in each of the columns. Moreover, like with *Spread Empty Spaces along the Pallet Width (Across MBR)*, it is also pretended to have all columns ending at the same point, but in this case that point will always be the upper limit of the pallet. Fig. 12 shows an example of a pattern generated by this heuristic.

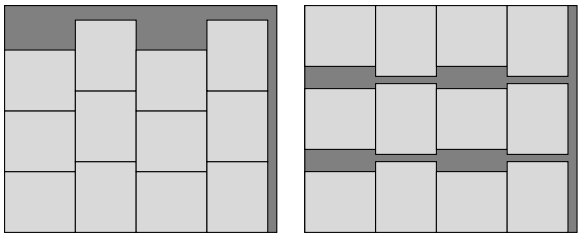


Fig. 12. Spread Empty Spaces in Vertical (Across Pallet).

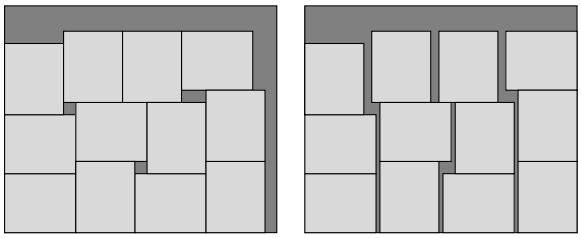


Fig. 13. Spread Empty Spaces in Horizontal (Across Pallet).

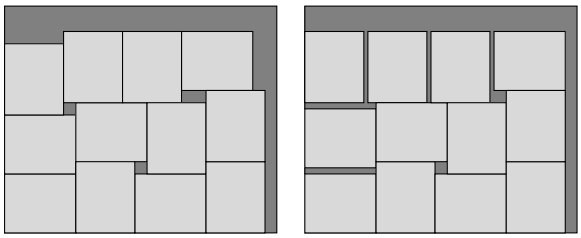


Fig. 14. Spread Empty Spaces Within Occupied Perimeter.

6.4. Spread empty spaces along the pallet length (across pallet)

Similar to the preceding heuristic, this method also aims to distribute the boxes placed on the pallet across its entire area. However, rather than dispersing the boxes along the width of the pallet, this technique distributes them along the length of the pallet, as illustrated in Fig. 13.

6.5. Spread empty spaces within MBR

This heuristic appears as a conjunction between the heuristics *Spread Empty Spaces along the Pallet Width (Across MBR)* and *Spread Empty Spaces along the Pallet Length (Across MBR)*. By doing so, the goal is to spread all boxes homogeneously inside the MBR in both ways, along the pallet width and along the pallet length. As a result of the application of this heuristic, a new arrangement is obtained where all boxes are still inside the original length and width of the MBR, but the idle spaces between the boxes are homogeneous both along the width of the pallet and along the length of the pallet, which is possible to see in the pattern exhibited on the right side of Fig. 14.

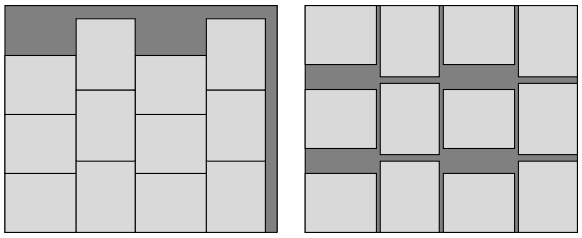


Fig. 15. Spread Empty Spaces Within Pallet.

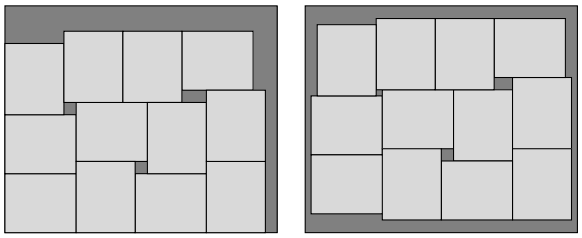


Fig. 16. Centralise Load.

6.6. Spread empty spaces within pallet

In a similar manner, *Spread Empty Spaces Within Pallet* is the heuristic responsible for joining *Spread Empty Spaces along the Pallet Width (Across Pallet)* and *Spread Empty Spaces along the Pallet Length (Across Pallet)*, in order to spread all boxes homogeneously throughout the entire pallet, creating a layout that occupies the whole pallet and where the idle spaces between boxes are homogeneous in both directions, along the pallet width and along the pallet length, as can be seen in Fig. 15.

6.7. Centralise load

The *Centralise Load* heuristic distinguishes itself significantly from other heuristics by seeking to centralise the load distributed on the pallet, rather than dispersing the idle spaces inherent in the initial pattern. This centralisation is applied to both the columns and the rows. As a result of applying this heuristic, a pattern is obtained where each row is exactly at the same distance from the left limit of the pallet as it is from the right limit of the pallet. In the same way, for each column, the distance to the bottom limit of the pallet must also match the distance to the top limit of the pallet. Fig. 16 contains a perfect example of the application of this heuristic.

Nevertheless, it is essential to notice that the centralisation of the rows and columns is commonly limited by the positions of the boxes belonging to other rows or columns, which does not allow for perfect centralisation in some situations, in order to prevent overlap.

It is also important to mention that this heuristic is generally implemented as a complement to all the other six heuristics, being applied on top of them.

7. Computational experiments

This section provides a detailed description of the comprehensive computational experiments conducted on a set of real-world instances from the warehouse operations of a retail company.

The set comprises 532 instances, each defined by the dimensions of the boxes to be arranged on the pallet, as well as the dimensions of the pallet itself. Each instance features homogeneous boxes, indicating that all boxes within a given instance have the same dimensions.

Table 3
Statistics regarding the Dimensions of the Box Sides.

	Group 1		Group 2	
	Length	Width	Length	Width
Range	194-610	110-485	194-610	95-485
Average	331.48	236.21	327.48	229.96
Mode	330	244	330	244
Standard Deviation	83.30	68.56	83.26	71.34

Table 4
Distribution of the Dimensions of the Box Sides.

	Group 1		Group 2	
	Length	Width	Length	Width
]0,100]	0	0	0	5
]100,200]	1	97	1	108
]200,300]	94	115	103	115
]300,400]	119	39	126	39
]400,500]	26	7	26	7
]500,600]	15	0	15	0
]600,700]	3	0	3	0

An important characteristic of the set of instances utilised in these experiments is their categorisation into two distinct groups. The fundamental difference between these groups is that Group 1 includes pallets measuring 1200×1000 mm^2. Conversely, Group 2 contains smaller pallets with dimensions of 1200×800 mm^2. Group 1 includes 258 instances, and Group 2 274.

Tables 3 and 4 display some statistics on the instances of each of the two groups. The range, average, mode, and standard deviation of the length and width of the sets of instances.

In order to conduct these experiments, a C++ algorithm was developed to generate the loading patterns corresponding to each instance. Initially, the algorithm processes the instance by reading the dimensions of the boxes and pallets, subsequently invoking CPLEX to implement the mathematical models. Fourteen combinations of objective functions (see Table 5) were employed, culminating in the creation of 14 patterns. The algorithm also ensures the elimination of duplicated patterns.

Post-processing heuristics are then applied, creating additional patterns. In this stage, only heuristics *Spread Empty Spaces Within Pallet* and *Centralise Load* were applied individually. The remaining heuristics were applied, followed by *Centralise Load*, and only the final results of both were stored.

Subsequently, the algorithm retains only the distinct patterns and proceeds to generate their mirrored counterparts for each pattern. This process is carried out while taking into account the specific type of interlocking configuration identified (Type 1, Type 2, or Type 3 as referenced in Section 3). With mirror development, the algorithm calculates the three stability metrics for each pattern, as explained in Sections 2 and 5. Finally, the algorithm generates an output file for each instance, storing all possible patterns, their mirrors, and the corresponding metric values.

The computational experiments were conducted on an Intel Core i7-12700H CPU operating at 2.30 GHz with 16.0 GB of RAM, using CPLEX 22.1.1 and Microsoft Visual Studio 2022, with a time constraint of 1 h imposed on each MILP model.

To ensure full reproducibility of our results, all test instances have been made publicly available through Mendeley Data in Araújo et al. (2025).

Table 5 presents for each MILP model considered during the experiments the acronyms, the indication of the enforcement of the interlocking constraints, the compaction strategy used and the objective functions and constraints. The sequence of objective functions delineates the sequence of lexicographic criteria employed within the computational experiments. For example, model S\F(1 ·2) represents a

Table 5
Set of models implemented during the testing stage.

Model	Interlocking	Compactation	Objective functions	Constraints
F(1)	–	–	(1)	(2) (5)-(6)
S\F(1)	Y	–	(1)	(2)-(6)
F(1·2)	–	downward and left	(1)(7)	(2) (5)-(6)
S\F(1·2)	Y	downward and left	(1)(7)	(2)-(6)
S\F(1·3)	Y	left	(1)(8)	(2)-(6) (9)-(10)
S\F(1·4)	Y	downward	(1)(11)	(2)-(6) (12)-(13)
S\F(1·5)	Y	–	(1)(14)	(2)-(6)
S\F(1·2·5)	Y	downward and left	(1)(7)(14)	(2)-(6)
S\F(1·3·4)	Y	left then downward	(1)(8)(11)	(2)-(6) (9)-(10) (12)-(13)
S\F(1·4·3)	Y	downward then left	(1)(11)(8)	(2)-(6) (9)-(10) (12)-(13)
S\F(1·3·5)	Y	left	(1)(8)(14)	(2)-(6) (9)-(10)
S\F(1·4·5)	Y	downward	(1)(11)(14)	(2)-(6) (12)-(13)
S\F(1·3 · 4·5)	Y	left then downward	(1)(8)(11)(14)	(2)-(6) (9)-(10) (12)-(13)
S\F(1·4 · 3·5)	Y	downward then left	(1)(11)(8)(14)	(2)-(6) (9)-(10) (12)-(13)

Table 6
Impact of the interlocking constraints.

Deviation	# of Instances	% of Instances
0	494	92.86
1	38	7.14

model that enforces interlocking (represented by the letter S) and has as its primary objective function to maximise the number of boxes (F1), followed by the secondary objective of maximising the compaction of the load towards the downward-left corner of the pallet (F2).

Model F(1) represents the original formulation proposed by Alvarez-Valdes et al. (2005a), which was used in this study as a benchmark to assess the efficiency of the distinct novel formulations presented in this paper.

7.1. A comparative analysis of the results pertaining to instances in group 1 and group 2

As stated previously, Groups 1 and 2 use different pallet dimensions: Group 1 uses pallets of size 1200×1000 mm^2, whereas Group 2 uses pallets measuring 1200×800 mm^2. However, the variation in the dimensions of the pallet does not significantly affect the maximal number of items that can be arranged per layer in all instances of each group. To assess this, each of the 532 instances was analysed using the aforementioned C++ algorithm. It was verified that the maximum number of items that could be accommodated per layer in all the generated layer patterns for the instances in Group 1 was 30, a value that is equivalent to that obtained for the instances in Group 2.

7.2. Analysis of the impact of the interlocking constraints

The initial phase of the experiments involved a comprehensive analysis of how interlocking constraints influence the total quantity of boxes positioned on the pallet, with a comparative evaluation of conditions absent of interlocking constraints F(1) and those incorporating them S\F(1).

Table 6 illustrates the impact of interlocking constraints on the total number of boxes in the optimal solution. A zero deviation indicates that the same number of boxes was achieved in the optimal solution for both F(1) and S\F(1). In turn, a deviation of one indicates that incorporating interlocking constraints into the model results in a solution with one less box. As presented in Table 6, in 92.86% of the cases, there is no reduction in the number of boxes per layer when interlocking constraints are applied. Furthermore, there is no instance where the deviation in the number of boxes exceeds a single unit. Notably, only 38 out of 532 instances demonstrate a reduction in the number of boxes per layer due to the enforcement of interlocking constraints.

This result can be explained by the nature of the MPLP solution space. The base formulation with only the non-overlapping constraints (2) typically admits a large number of alternative optimal solutions — that is, many distinct box arrangements achieve the same maximum box count. The interlocking constraints (3)-(4) restrict the feasible space by requiring alternating corner box orientations, but because the set of optimal solutions is typically rich, at least one arrangement satisfying the interlocking requirements almost always exists among the alternative optima. This is consistent with the findings of Yi et al. (2025), who demonstrated that MPLP instances generally admit numerous distinct optimal arrangements. Only in 38 instances (7.14%) is the feasible space sufficiently constrained by the interlocking requirements that no compatible arrangement achieves the original optimal count, resulting in a reduction of exactly one box.

7.3. Analysis of the different lexicographic objective functions

During computational experiments, 14 different lexicographic combinations of objective functions were implemented as defined in Table 5.

It is important to clarify the role of the secondary objective functions (F2–F5) within the lexicographic framework. Since F1 (box count maximisation) is always solved first and its optimal value is preserved as a constraint before optimising the subsequent objectives, the secondary objectives never reduce the number of boxes placed. Instead, they act as a principled selection mechanism among the multiple optimal solutions that achieve the maximum box count under the interlocking constraints. As discussed in Section 7.2, the MPLP typically admits numerous alternative optima, and without secondary objectives, the solver returns an arbitrary arrangement from this set, which, as illustrated in Fig. 7(a), may exhibit dispersed boxes with poor compactness and stability. The secondary objectives guide the solver towards specific patterns within the set of alternative optima: F2 selects the most compact arrangement towards the bottom-left corner, F3 and F4 select patterns compacted in a single direction, and F5 selects arrangements with balanced horizontal and vertical box counts. This approach provides a structured method for navigating the solution multiplicity inherent in the MPLP, directing the solver towards the most practically useful patterns rather than leaving the choice arbitrary. The patterns shown in Fig. 7 illustrate this mechanism: all five patterns achieve the same box count, but each secondary objective steers the arrangement towards a different configuration with distinct compactness and stability properties.

Starting with Metric 1 (Fig. 17), three conclusions stand out. First, F(1) was notably the worst-performing model, confirming that interlocking constraints have a positive effect on this metric. Second, S\F(1 ·2) and S\F(1 ·2 ·5) clearly outperformed all other models, indicating that combining interlocking with bottom-left compaction is the most effective strategy for improving static stability. Third, the addition of

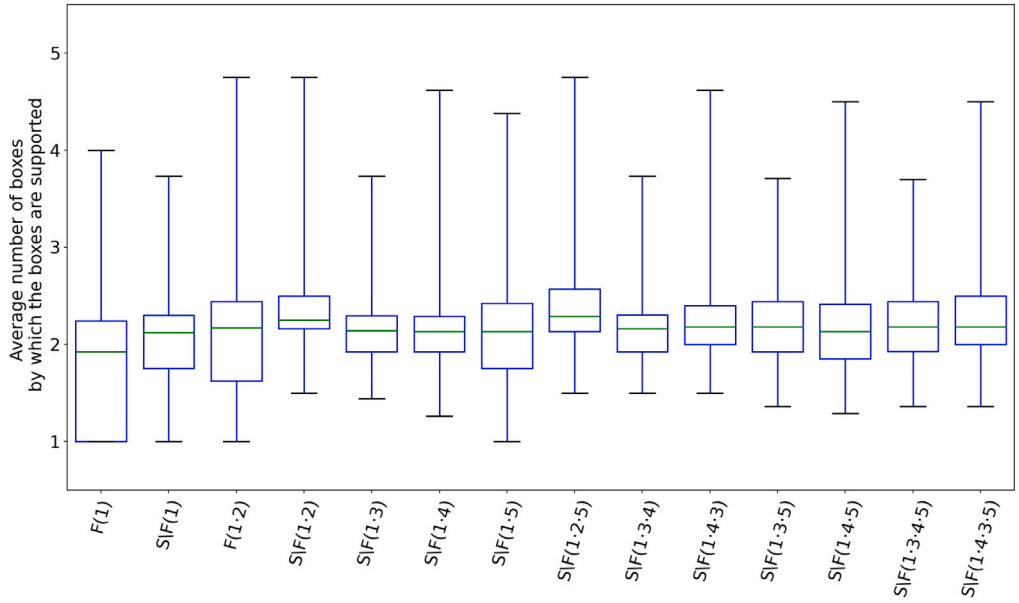


Fig. 17. Metric 1 - Interlocking comparison between all combinations of functions.

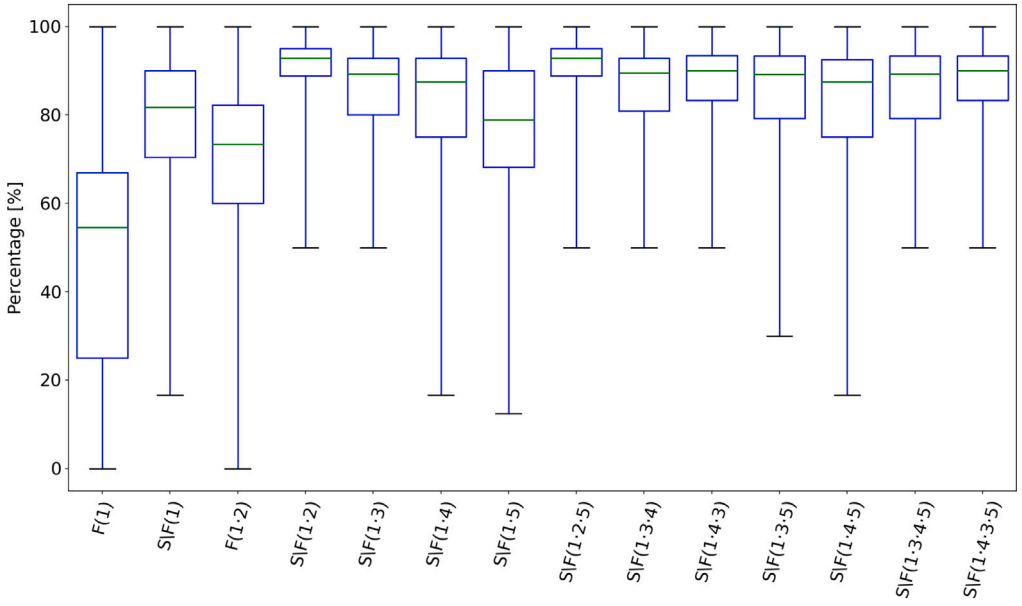


Fig. 18. Metric 2- Lateral support comparison between all combinations of functions.

F5 did not produce significant improvements, though it increased the dispersion of results.

Concerning Metric 2 (Fig. 18), the same pattern emerges: S\F(1 ·2) and S\F(1 ·2 ·5) performed best while F(1) produced the worst results, confirming that interlocking combined with bottom-left compaction is also the most effective strategy for lateral support. The addition of F5 had no significant impact on this metric.

For Metric 3 (Fig. 19), all models performed similarly, with averages exceeding 95% across all formulations. No meaningful differentiation between models was observed for this metric.

Finally, concerning the computational time necessary to obtain the optimal solution, it is possible to visualise in Fig. 20 that all models presented similar performance; it is noticeable that all these formulations are able to reach the optimal solution in extremely reduced amounts of time.

All models achieved proven optimal solutions in less than 1.3 s per instance, with mean times ranging from 0.033s (F(1)) to 0.224s (S\F(1 ·4 ·3 ·5)) and median times from 0.025s (S\F(1)) to 0.186s (S\F(1 ·4 ·3 ·5)). The maximum solution time across all 7448 optimisation runs was 1.22 s. The compact boxplots demonstrate consistent, predictable performance suitable for real-time automated warehouse operations. Adding interlocking constraints (comparing F(1) to S\F(1)) increases mean solution time by only 33%, from 0.033s to 0.044s, confirming minimal computational overhead for stability considerations.

In sum, the results presented show that the 12 novel formulations outlined in this study exhibit a superior performance to that of the formulation proposed by Alvarez-Valdes et al. (2005a) regarding the metrics Interlocking and Lateral support. Furthermore, in relation to the third metric considered in this study, the Support Polygon (Percentage), the proficiency of the novel formulations suggested was identical to that displayed by the formulation presented in Alvarez-Valdes et al.

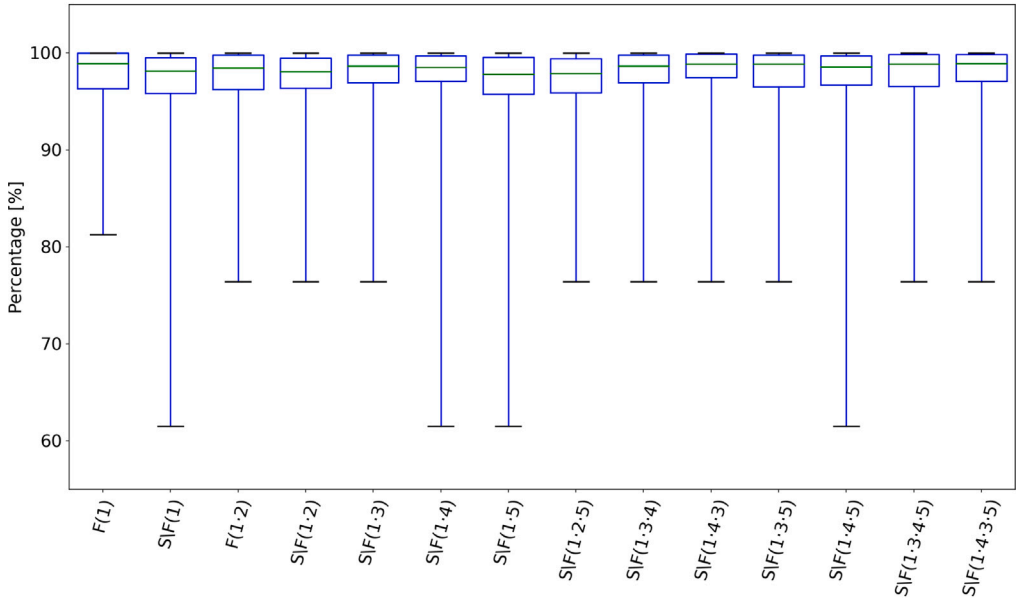


Fig. 19. Metric "Support Polygon (Percentage)" comparison between all combinations of functions.

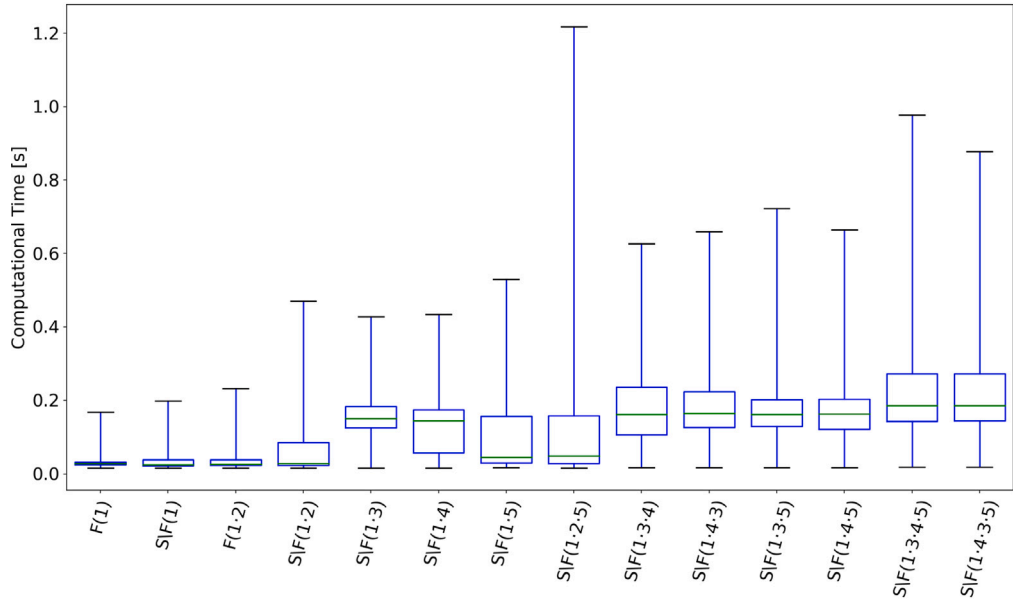


Fig. 20. Computational Time comparison between all combinations of objective functions across 532 instances.

(2005a). Finally, with respect to computational time, as mentioned above, all the models presented similar performances, which means that the novel formulations demonstrated a promptness equivalent to that portrayed by the formulation of Alvarez-Valdes et al. (2005a).

Taking into account that S\F(1 ·2) and S\F(1 ·2 ·5) outperformed the other models in the Metric 1 and Metric 2 metrics and also presented exceptional results in Metric 3 and computational times, it is plausible to consider that, overall, these two are the best models.

It is worth noting that the diversity of patterns generated by the 14 lexicographic combinations of objective functions demonstrates that the MILP model itself is a powerful mechanism for producing multiple distinct loading patterns with varying stability profiles. Each combination yields patterns that are optimal with respect to a different priority ordering, providing a systematic and principled exploration of the solution space. This observation is consistent with findings in the broader PLP literature — notably Yi et al. (2025), already discussed in Section 2, who demonstrated that many MPLP instances admit numerous distinct optimal arrangements. In our framework, the multi-objective lexicographic approach harnesses this solution multiplicity to offer decision-makers a curated set of patterns, each with known stability characteristics, from which they can select the most appropriate configuration for their specific operational context.

7.4. Analysis of the impact of post-processing Heuristics on stability metrics

This subsection evaluates how the practically-motivated configurations generated by the post-processing heuristics (Section 6) affect load stability. Since these heuristics are designed to produce layout types required in practice — not to improve stability — the analysis below quantifies the stability cost or preservation associated with each configuration, providing decision-makers with the information

needed to balance operational layout requirements against stability considerations.

As previously mentioned, it was determined at this stage that the application of *Spread Empty Spaces along the Pallet Width (Across MBR)*, *Spread Empty Spaces along the Pallet Length (Across MBR)*, *Spread Empty Spaces along the Pallet Width (Across Pallet)*, *Spread Empty Spaces along the Pallet Length (Across Pallet)*, and *Spread Empty Spaces Within MBR* would invariably be succeeded by the implementation of *Centralise Load*.

Therefore, the patterns obtained following the implementation of these six heuristics are then always subject to the application of *Centralise Load*, leading to subsequent rearrangement of those patterns. The metrics results shown in Table 7 correspond to the patterns obtained afterwards, i.e., obtained after applying the two heuristics in succession.

On the other hand, *Spread Empty Spaces Within Pallet* was always applied alone, since the application of *Centralise Load* would not change the patterns. Furthermore, *Centralise Load* was also applied individually, without any heuristic before or after.

Taking into account the analysis presented in Section 7.3, which indicated that models S\F(1 ·2) and S\F(1 ·2 ·5) outperformed the other models, these two models were selected to compare the performance of the various post-processing heuristics.

Table 7 presents the stability results for both MILP models (S\F(1 ·2) and S\F(1 ·2 ·5)) across all post-processing heuristics and the three stability metrics. Regarding Metric 1 (Interlocking), the results after applying the heuristics are very similar to those of the original patterns, indicating that the interlocking structure established by the MILP model is robust to spatial redistribution of empty spaces.

Regarding Metric 2 (Lateral Support), the heuristics produce a notable decline compared to the original patterns. This is expected, as the heuristics create gaps between boxes — either by spreading them across the pallet or within the minimum bounding rectangle — which reduces the number of boxes enclosed on at least three sides. Among the heuristics, Centralise Load yields the least degradation, as it repositions the load without explicitly distributing empty spaces between boxes. Heuristics that spread boxes across the full pallet dimensions produce worse results than those constrained to the minimum bounding rectangle, since the former distribute gaps over a larger area. Similarly, heuristics that act in both directions simultaneously (Spread Empty Spaces Within MBR and Spread Empty Spaces Within Pallet) produce lower Metric 2 values than those acting in a single direction, as they introduce gaps along both axes. These results quantify the lateral support cost of adopting each layout configuration and allow decision-makers to select the configuration that best balances their operational requirements against stability.

Regarding Metric 3 (Support Polygon), the heuristics do not produce a considerable impact, with all configurations maintaining averages above 95%. Notably, Spread Empty Spaces Within MBR and Spread Empty Spaces Within Pallet achieve averages of 100%, indicating that uniform gap distribution in both directions slightly improves the geometric stability of individual boxes. In summary, the post-processing heuristics preserve interlocking (Metric 1) and support polygon stability (Metric 3) while reducing lateral support (Metric 2) to varying degrees. This information enables practitioners to make informed choices when operational requirements demand a specific layout configuration.

7.5. Trade-off analysis: Efficiency vs. Stability

A critical question for industrial decision-making is: When should stability be prioritised over loading efficiency? Our results enable quantitative analysis of this trade-off. Table 6 shows that interlocking constraints reduce box count in only 7.14% of instances (38 of 532), with a maximum reduction of one box per layer. This translates to:

- Efficiency loss: Maximum 3.3–8.3% reduction (depending on baseline box count);
- Stability gain: Substantial improvements in Metrics 1 and 2 (Figs. 17–18).

For the 493 instances (92.86%) where box count is maintained, stability improvements come at zero efficiency cost, making the decision straightforward: always apply interlocking constraints. For the 38 instances where interlocking reduces box count by one, decision-makers should consider:

When to accept the efficiency loss (prioritise stability):

1. High-value products: When product damage costs exceed transportation savings;
2. Long-distance transport: When exposure to dynamic forces is prolonged;
3. Multi-handling operations: When pallets undergo frequent transfers;
4. Regulatory/customer requirements: When stability standards are mandated.

When to forgo interlocking (prioritise efficiency):

1. Low-value commodity products: When damage costs are minimal;
2. Short-distance controlled transport: When stability risks are minimal;
3. Robust packaging: When boxes themselves provide structural integrity.

8. Conclusions

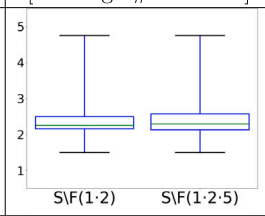
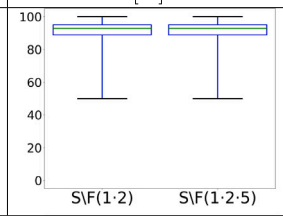
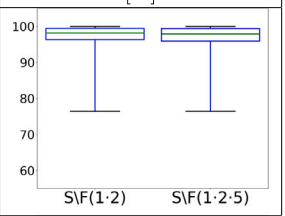
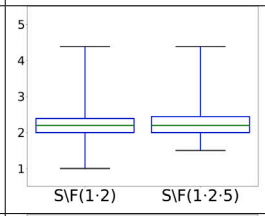
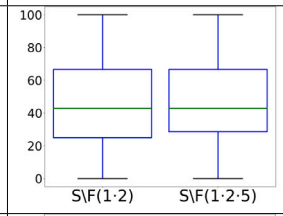
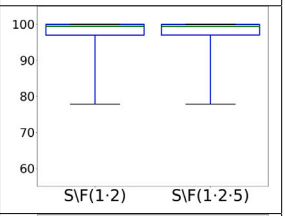
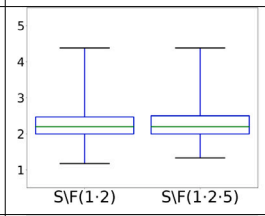
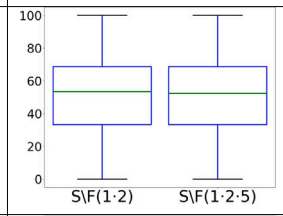
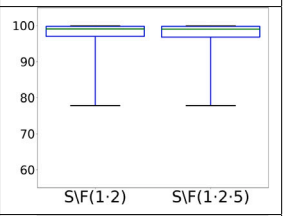
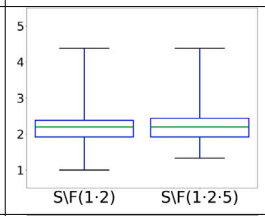
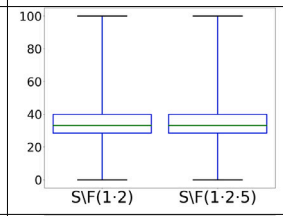
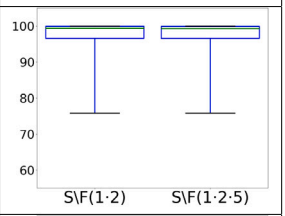
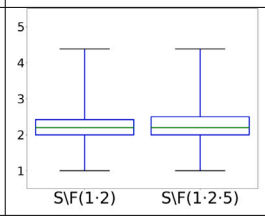
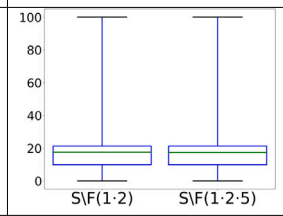
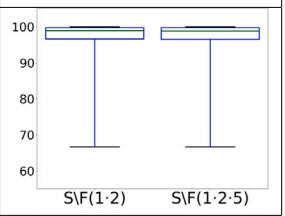
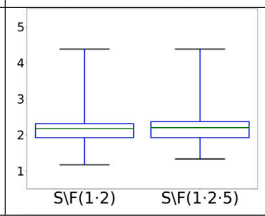
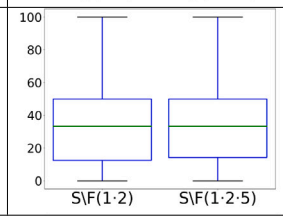
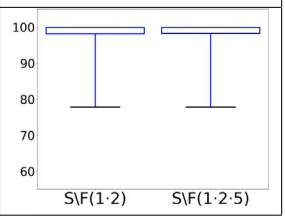
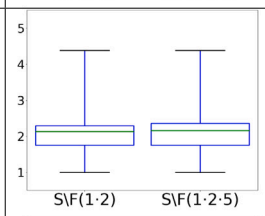
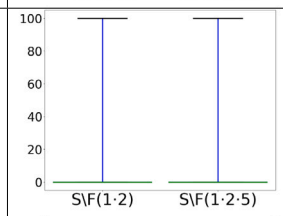
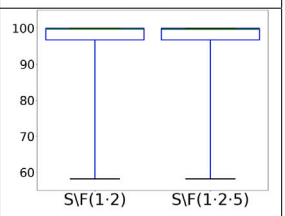
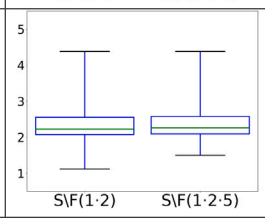
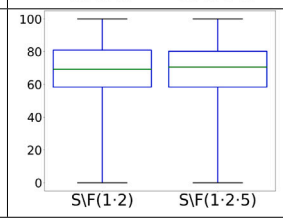
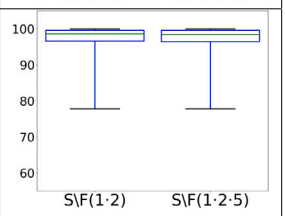
In this paper, we have strived to integrate interlocking strategies in the generation of loading patterns for the Manufacturer's Pallet Loading Problem, aiming to secure a loading that is not only efficient but also stable. Our approach started with the development of different versions of the MILP model that incorporated constraints to guarantee interlocking during the loading process. Those different versions were the result of distinct combinations of lexicographic objectives functions that intended to maximise the efficiency in the loading process, that is, maximise the number of boxes loaded per layer, but also to generate diverse patterns, with alternative dispositions of the boxes. Furthermore, we also developed a series of post-processing heuristics with the aim of reorganising the patterns generated previously, creating new solutions, and therefore leading to a greater diversity of loading patterns.

To identify which combination produced the best patterns, in terms of both loading efficiency and stability, all patterns were evaluated using a set of three metrics. Two of those were already previously used in the literature, while the third was a novel metric developed during this study, also one of the major contributions of this work.

Following the comprehensive computational experiments conducted, it can be concluded that:

- Interlocking strategies exhibited a beneficial effect on stability, as the patterns produced with interlocking constraints demonstrated improved outcomes in this aspect. Conversely, the implementation of these strategies resulted in a reduction of loading efficiency in certain instances, allowing a smaller number of boxes to be loaded per layer. However, this occurrence was limited to 7% of the cases with a maximum of one box, indicating that in 93% of the experiments, efficiency remained unaffected.
- Regarding the various combinations of objective functions tested, the combination between maximisation of the number of boxes per layer and compacting to the bottom-left corner proved to be the one that generated the best results in terms of efficiency and stability.

Table 7
Results obtained by each post-processing heuristic regarding the three stability metrics.

Post-Processing Heuristics	Metric 1 [Average # of Boxes]	Metric 2 [%]	Metric 3 [%]
Original Results (Without the Application of any Heuristic)			
Spread Empty Spaces along the Pallet Width (Across MBR)			
Spread Empty Spaces along the Pallet Length (Across MBR)			
Spread Empty Spaces along the Pallet Width (Across Pallet)			
Spread Empty Spaces along the Pallet Length (Across Pallet)			
Spread Empty Spaces Within MBR			
Spread Empty Spaces Within Pallet			
Centralize			

- The post-processing heuristics were designed to generate types of spatial configurations commonly required in real-world operations (e.g., centred loads, evenly distributed layouts). The stability analysis revealed that these configurations preserve interlocking and support polygon stability while reducing lateral support to varying degrees, quantifying the stability cost of meeting specific operational layout requirements and enabling informed trade-off decisions within a decision support system.

CRediT authorship contribution statement

João Araújo: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation. **António G. Ramos:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Elsa Silva:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis. **Ana Moura:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis.

Acknowledgments

This work is co-financed by Component 5 - Capitalization and Business Innovation, integrated in the Resilience Dimension of the Recovery and Resilience Plan within the scope of the Recovery and Resilience Mechanism (MRR) of the European Union (EU), framed in the Next Generation EU, for the period 2021–2026, within project Produtech_R3, with reference 60.

The third author is financed by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia, within project UIDP/50014/2020.

Data availability

There is a link in the paper to the instances.

References

Ahn, S., Park, C., & Yoon, K. (2015). An improved best-first branch and bound algorithm for the pallet-loading problem using a staircase structure. *Expert Systems with Applications*, *42*(21), 7676–7683.

Ali, Sara, Ramos, António Galrão, Carravilla, Maria Antónia, & Oliveira, José Fernando (2022). On-line three-dimensional packing problems: A review of off-line and on-line solution approaches. *Computers & Industrial Engineering*, *168*, Article 108122.

Aljuhani, Deemah, & Papageorgiou, Lazaros (2021). Improved layout structure with complexity measures for the manufacturer's pallet loading problem (MPLP) using a block approach. *Journal of Industrial Engineering and Management*, *14*(2), 231–249.

Almasarwah, Najat, Abdelall, Esraa, Suer, Gursel, Egilmez, Gokhan, Singh, Manjeet, & Ramadan, Saleem (2023). Pallet loading optimization considering storage time and relative humidity. *Journal of Industrial Engineering and Management*, *16*(2), 453–471.

Alvarez-Valdes, R., Parreño, F., & Tamarit, J. M. (2005a). A branch-and-cut algorithm for the pallet loading problem. *Computers & Operations Research*, *32*(11), 3007–3029.

Alvarez-Valdes, R., Parreño, F., & Tamarit, J. M. (2005b). A tabu search algorithm for the pallet loading problem. *OR Spectrum*, *27*, 43–61.

Amaral, André R. S., & Wright, Mike (2001). Experiments with a strategic oscillation algorithm for the pallet loading problem. *International Journal of Production Research*, *39*(11), 2341–2351.

Araújo, João, Ramos, António Galrão, Silva, Elsa, & Moura, Ana (2025). Instances for the manufacturer's pallet loading problem [data set].

Aylak, Batin Latif, İnce, Murat, Oral, Okan, Süer, Gürsel, Almasarwah, Najat, Singh, Manjeet, & Salah, Bashir (2021). Application of machine learning methods for pallet loading problem. *Applied Sciences*, *11*(18).

Birgin, Ernesto G., Lobato, Rafael D., & Morabito, Reinaldo (2010). An effective recursive partitioning approach for the packing of identical rectangles in a rectangle. *Journal of the Operational Research Society*, *61*(2), 306–320.

Birgin, Ernesto G., Morabito, Reinaldo, & Nishihara, Fabio H. (2005). A note on an L-approach for solving the manufacturer's pallet loading problem. *Journal of the Operational Research Society*, *56*(12), 1448–1451.

Bischoff, Eberhard E. (1991). Stability aspects of pallet loading. *Operations-Research-Spektrum*, *13*(4), 189–197.

Bischoff, E., & Dowsland, W. B. (1982). An application of the micro to product design and distribution. *Journal of the Operational Research Society*, *33*(3), 271–280.

Bischoff, E., & Dowsland, W. B. (1985). An exact two-dimensional non-guillotine cutting tree search procedure. *Operations Research*, *33*(1), 49–64.

Bischoff, E. E., & Ratcliff, M. S. W. (1995). Loading multiple pallets. *Journal of the Operational Research Society*, *46*(11), 1322–1336.

Carpenter, H., & Dowsland, W. B. (1985). Practical considerations of the pallet-loading problem. *Journal of the Operational Research Society*, *36*(6), 489–497.

Dowsland, Kathryn A. (1987a). An exact algorithm for the pallet loading problem. *European Journal of Operational Research*, *31*(1), 78–84.

Dowsland, Kathryn A. (1987b). A combined data-base and algorithmic approach to the pallet-loading problem. *Journal of the Operational Research Society*, *38*(4), 341–345.

Farago, R., & Morabito, R. (2001). Um método heurístico baseado em relaxação Lagrangiana para o problema de carregamento de paletes do produtor. *Pesquisa Operacional*, *20*(2), 197–212.

Gunawardena, Kavindu, Wijayanayake, Annista, & Kavirathna, Chathumi (2021). Solve manufacturer's pallet loading problem (MPLP) with practical warehouse constraints. In *2021 6th international conference on information technology research* (pp. 1–6).

Kocjan, W., & Holmström, K. (2010). Computing stable loads for pallets. *European Journal of Operational Research*, *207*(2), 980–985.

Liu, Ya, Chu, Chengbin, & Wang, Kanliang (2010). A dynamic programming-based heuristic for the variable sized two-dimensional bin packing problem. *International Journal of Production Research*, *49*(13), 3815–3831.

Liu, Fuh-Hwa F., & Hsiao, C. J. (1997). A three-dimensional pallet loading method for single-size boxes. *Journal of the Operational Research Society*, *48*(7), 726–735.

Lu, Yiping, & Cha, Jianzhong (2014). A fast algorithm for identifying minimum size instances of the equivalence classes of the pallet loading problem. *European Journal of Operational Research*, *237*(3), 794–801.

Martins, Gustavo H. A., & Dell, Robert F. (2007). The minimum size instance of a pallet loading problem equivalence class. *European Journal of Operational Research*, *179*(1), 17–26.

Martins, Gustavo H. A., & Dell, Robert F. (2008). Solving the pallet loading problem. *European Journal of Operational Research*, *184*(2), 429–440.

Mascarenhas, Walter F. (2005). Two aspects of the pallet loading problem. *Electronic Notes in Discrete Mathematics*, *19*, 381–387, Proceedings of GRACO2005.

Morabito, Reinaldo, & Morales, S. R. (1998). A simple and effective recursive procedure for the manufacturer's pallet loading problem. *Journal of the Operational Research Society*, *49*, 819–828.

Moura, Ana, & Oliveira, José Fernando (2005). A GRASP approach to the container-loading problem. *IEEE Intelligent Systems*, *20*(4), 50–57.

Pureza, Vitória, & Morabito, Reinaldo (2006). Some experiments with a simple tabu search algorithm for the manufacturer's pallet loading problem. *Computers & Operations Research*, *33*(3), 804–819.

Ribeiro, Glaydston Mattos, & Lorena, Luiz Antonio Nogueira (2007). Lagrangean relaxation with clusters and column generation for the manufacturer's pallet loading problem. *Computers & Operations Research*, *34*(9), 2695–2708.

Sawicki, Piotr, & Sawicka, Hanna (2023). Design optimization of stacked pallet load units. *Applied Sciences*, *13*(4), 2153.

Scheithauer, Guntram, & Terno, Johannes (1999). The G4-Heuristic for the pallet loading problem. *Journal of the Operational Research Society*, *47*.

Singh, Manjeet, Almasarwah, Najat, & Süer, Gürsel (2019). A two-phase algorithm to solve a 3-dimensional pallet loading problem. *Procedia Manufacturing*, *39*, 1474–1481, 25th International Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing August 9-14, 2019 | Chicago, Illinois (USA).

Tarnowski, Alexander G., Terno, Johannes, & Scheithauer, Guntram (1994). A polynomial time algorithm for the guillotine pallet loading problem. *INFOR. Information Systems and Operational Research*, *32*(4), 275–287.

Taylor, Gregory S., Chan, Yupo, & Rasool, Ghulam (2017). A three-dimensional bin-packing model: exact multicriteria solution and computational complexity. *Annals of Operations Research*, (251), 397–427.

Wäscher, Gerhard, Haußner, Heike, & Schumann, Holger (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, *183*(3), 1109–1130.

Wu, K. C., & Ting, C. J. (2007). A two-phase algorithm for the manufacturer's pallet loading problem. In *2007 IEEE international conference on industrial engineering and engineering management* (pp. 1574–1578).

Yamassaki, Vitoria (2003). Um refinamento do algoritmo tabu de dowsland para o problema de carregamento de paletes do produtor. *Pesquisa Operacional*, *13*(3).

Yi, Junmin, Chen, Xing-Guang, & Zhou, Jing (2009). The pinwheel pattern and its application to the manufacturer's pallet-loading problem. *International Transactions in Operational Research*, *16*(6), 809–828.

Young-Gun, G., & Kang, Maing-Kyu (2001). A fast algorithm for two-dimensional pallet loading problems of large size. *European Journal of Operational Research*, *134*(1), 193–202.