Supporting Your Basic Needs - A Base Support Approach for Static Stability Assessments in Air Cargo

Diego Enrico Longhitano
University of Cologne
dlonghi1@smail.uni-koeln.de

Philipp Gabriel Mazur
University of Cologne
mazur@wim.uni-koeln.de

Simon Maximilian Wolf
University of Cologne
wolf@cs.uni-koeln.de

Detlef Schoder
University of Cologne
schoder@wim.uni-koeln.de

Abstract

Static stability is one of the most important constraints in the design and efficient calculation of safe air cargo pallets. To calculate the static stability of a cargo layout, base-focused methods such as full or partial base support are often used. Compared to mechanical or simulation-based methods, they offer high performance and simplicity. However, these methods currently reach their limits when dealing with the practical complexity of air cargo, making them difficult to apply in practice. In this research, we extend and generalize these support point methods by modeling irregular and multilevel cargo shapes, which enables improved practical applications. We follow a design-oriented approach to capture air cargo requirements, design an artifact, and evaluate its performance. Our results show a generalized approach that covers a greater practical complexity while maintaining its efficiency.

Keywords: Pallet Loading, Static Stability, Full Base Support, Partial Base Support, Logistics

1. Introduction

Throughout the recent global pandemic, air cargo has played a vital role in transporting critical products like medical equipment and pharmaceuticals. Additionally, air cargo remains an essential component to global supply chains, enabling international businesses to partake in e-commerce (IATA, 2021). Despite its importance, air cargo operations like packing and loading of cargo on pallets are still largely carried out without IT support (Lee et al., 2021), which can result in the inefficient construction of packed pallets and low volume utilizations, depending on the palletizer's individual skills (Lee et al., 2021).

To support the packing of pallets and containers, many approaches tackled this optimization problem called Pallet Loading Problem (PLP) (Dowsland, 1987). The complexity of computing optimal solutions varies with the complexity of input items and the number of constraints that reflect practical requirements. A multitude of constraints exist (Bortfeldt & Wäscher, 2013; Pollaris et al., 2015; Zhao et al., 2016). One of the most important constraints is static stability (Bortfeldt & Wäscher, 2013; Ramos, Oliveira, & Lopes, 2016), which guarantees that items maintain their position during loading (Bischoff, 1991; Junqueira et al., 2012; Parreño et al., 2008; Ramos & Oliveira, 2018), thereby reducing the risk of damaged cargo, injuries of personnel, and unsafe operations (Bortfeldt & Wäscher, 2013; Zhao et al., 2016).

To compute static stability, authors use base support-related constraints like full base support (FBS) or partial base support (PBS), which are the focus of this research. These two approaches are tightly related as they both assess static stability by calculating the percentage of the item's base support (Ramos, Oliveira, Gonçalves, and Lopes, 2016). Generally, base support approaches are criticized to be very restrictive and not to reflect real-world complexity (Kurpel et al., 2020; Ramos, Oliveira, Gonçalves, and Lopes, 2016).

However, the criteria perform well for homogeneous input sets only containing boxes of similar size (Ramos, Oliveira, Gonçalves, and Lopes, 2016) and promise lower runtimes compared to more sophisticated alternatives like physical simulations, which cope with extensive runtimes (Mazur et al., 2020).

Although most items in air cargo are of cuboid shape, they are often characterized by strong heterogeneity up to irregularity (Brandt & Nickel, 2019; Lee et al., 2021). So far, base-related approaches primarily focus on boxes, neglecting other non-regular item shapes or items with multiple bases, which limits their applicability to more complex and realistic problem instantiations, such as air cargo. Further, to meet strict deadlines and flight schedules, constraints like static stability must be efficiently evaluated (Mazur et al., 2022).

Due to their simplicity and efficiency, base-related approaches might fill this gap and represent a new

URI: https://hdl.handle.net/10125/102775
978-0-9981331-6-4

solution to this application domain. To the best of our knowledge, no previous study has investigated applications of base-related static stability approaches to highly heterogeneous, irregular problem contexts such as air cargo.

This research tackles the research problem of extending current base-related approaches to the novel application domain of air cargo and tries to answer the research question:

How can the base support approach be applied to static stability assessments in air cargo?

To answer this research question, we carve out requirements for FBS and PBS in air cargo problem contexts, implement an artifact that fulfills our requirements and demonstrate its functionality as well as its efficiency. We follow a Design Science Research (DSR) approach that aims to generate knowledge by developing novel IS-artifacts to solve practical problems.

The remainder of this work is structured as follows. Section 2 presents an overview about related studies employing base-related approaches. Section 3 sketches our design-oriented approach. In section 4, we present our results, which are then discussed in section 5.

2. Related Work

Static stability is relevant during loading and unloading operations of cargo pallets (Bortfeldt & Wäscher, 2013) and measures how well the arrangement of cargo items can withstand the force of gravity (Junqueira et al., 2012). Multiple approaches exist that reflect practical complexity and are therefore applicable in practice. Apart from FBS and PBS, simulation-based (SIM) approaches focus on the static stability evaluation using real-time physics engines (Bracht et al., 2016; Mazur et al., 2020, 2022). Further, static mechanical equilibrium approaches use an equilibrium-based application of Newton's laws of motion (Krebs & Ehmke, 2021; Ramos, Oliveira, Gonçalves, and Lopes, 2016).

With respect to base support, Gehring & Bortfeldt (1997) present an early application. The authors propose to calculate stability for one item using "the ratio of the bottom area in contact with the boxes below to the total bottom area of the box" (Gehring & Bortfeldt, 1997, p. 402). This ratio α is termed base support and frequently considered in related studies, e.g., in Junqueira et al., (2012), Kurpel et al. (2020), and Nascimento et al. (2021). It reflects the percentage of an item's supported area. Support means that the item's lower face is in contact with either other items's top faces or the floor (Ramos, Oliveira, Gonçalves, and Lopes, 2016).

The value for α can range from 0% to 100% (Kurpel et al., 2020). Depending on the value assigned to α one can distinguish between two cases. If α equals one, then full support of the item's base is imposed (Nascimento et al., 2021). This implies that "no overhanging boxes are allowed" (Ramos, Oliveira, Gonçalves, and Lopes, 2016, p. 569). This case is termed FBS. If the required base support is set to values below 100%, it is termed PBS (Bortfeldt & Wäscher, 2013).

For PBS, the top surface of item *s* is at the same height as the bottom surface of item *i*, for item *i* to be stable over item *s* (Deplano et al., 2021). Additionally, the bottom face of item *i* is in direct contact with the top face of item *s* (Paquay et al., 2016). If the item lays on the floor, it is automatically declared stable (Nascimento et al., 2021; Ramos, Oliveira, Gonçalves, and Lopes, 2016). Often authors distinguish between different base support for different item types. In such cases, the parameter $\alpha_i \in [0; 1]$ is used to describe the desired amount of static stability for all boxes of type *i* (Junqueira et al., 2012).

The minimal required α value is frequently discussed in related studies, since adjusting α implies a tradeoff between space utilization and correct reflection of static stability. Only 100% guarantees static stability (Ramos & Oliveira, 2018). However, setting α to 100% is also the most restrictive approach (Ramos, Oliveira, Gonçalves, and Lopes, 2016), since it strictly limits placement options for optimization algorithms and therefore limits the solution space. So, FBS is "very costly for algorithm efficiency, especially in the strongly heterogeneous instances" (Ramos, Oliveira, Gonçalves, and Lopes, 2016, p. 569).

To relax the constraint and achieve better space utilization, authors set α to smaller values, risking unstable loads in return. Elhedhli et al. (2019) impose base support of 70%, while Gajda et al. (2022) allow an overhanging of 20%, thus requiring 80% of the base area to be supported.

Besides adjusting α, some approaches define a minimal number of supported corners. Deplano et al. (2021) enforce at least two bottom corners of every box to be on the top surface of another item. Similarly, Olsson et al. (2020) claim that a box achieves base support if either "all four corners have support below […] or two corners and 80% of the base area have support below" (Olsson et al., 2020, p. 1026).

To the best of our knowledge, no related study considers items other than boxes. Problem instances that contain items other than boxes are characterized as irregular, while instances only containing boxes of the same size are called homogeneous (Bortfeldt & Wäscher, 2013). If only boxes are considered, but with varying dimensions, this is termed heterogeneous. A high frequency of different dimensions is termed strongly heterogeneous, while a low frequency is regarded as weakly heterogeneous (Bortfeldt & Wäscher, 2013).

With respect to air cargo contexts, most items are of cuboid shape (Brandt & Nickel, 2019; Lee et al., 2021), while being "strongly heterogeneous in terms of their dimensions" (Lee et al., 2021, p. 1407). Often, the cargo is attached underneath a wooden pallet to simplify handling. Apart from boxes, other shapes like sacks, barrels, or furniture occur (Brandt & Nickel, 2019; Lee et al., 2021).

Studies facing practical air cargo complexity and irregular shapes are scarce. Mazur et al. (2020) first introduced the integration of simulations using real-time physics engines for static stability assessments in an air cargo context. In a subsequent work, the authors extended their prototype by using graphical processing unit (GPU)'s compute capabilities to parallelize stability simulations (Mazur et al., 2022). Although faster in most cases, the GPU-accelerated simulations still suffer from weak performances compared to the CPU-only version. Consequently, both studies report weak performances with runtimes ranging from minutes to hours, which also negatively impact overall solution quality. The trade-off between realistic modeling of practical complexity and performance remains an issue.

3. Approach

Since our goal is to both raise requirements and implement a novel artifact, we follow a design-oriented approach (Vaishnavi et al., 2017). In doing so, we ensure that the resulting implementation can cope with real life complexity and is relevant for practical use. In the IS field, the DSR methodology is well established for knowledge generation through the design and development of novel IT-artifacts (Hevner et al., 2004). In this research, we adapt the iterative DSR methodology by Peffers et al. (2007). Our adapted approach consists of three phases: (1) requirements analysis, (2) design & development, and (3) demonstration.

In the requirements analysis phase, we first describe limitations of the classic base support definition using practical examples. Out of this problem analysis, we then derive artifact design requirements, which include an adapted base support definition.

We then create our instantiated artifact based on the requirements. Finally, we demonstrate the artifact's functionality, which includes the demonstration of the artifact's suitability to solve multiple problem instances. In this research, we apply our new method to calculate FBS and PBS to multiple exemplary item arrangements.

Furthermore, we measure how well the artifact solves the problem, which includes quantifying its performance in terms of runtime. Further, we compare the performance of our approach to the already established SIM approach for air cargo, presented by Mazur et al. (2020). Finally, we compare and contrast our observations to the previously defined objectives.

4. Development of the Artifact

4.1 Requirements Analysis

We derive our artifact requirements (AR) from our main goal, that is, the efficient and reliable assessment of FBS and PBS for air cargo. We summarize our artifact design requirements in Table 1.

The application of FBS and PBS in air cargo implies additional challenges. From a theoretical perspective, we notice that FBS and PBS have almost exclusively been considered for items of cuboid shape, which is also reflected by their definitions. Classic base support definitions are not suitable to model shapes that have non-rectangle bases, multiple bases (or: height levels), or multiple top surfaces.

Since air cargo includes the loading of irregular items, adapted definitions are needed that also take into account shapes other than boxes. Classically, in related air cargo literature, shapes are modeled through boxes, cylinders, polygon prisms, and L-shapes (Mazur et al., 2020). Boxes, cylinders, and polygon prisms all expose a single top surface parallel to one single bottom surface (base shape) (AR1). In contrast, when looking at multi-level shapes (i.e., shapes with multiple height levels, such as L-shapes), complexity is added through its orientation and cut-out position, as multiple top surfaces or multiple base shapes exist (Figure 1). The classic definition, according to which an item lying on the floor has full support, is not sufficient when considering complex, multi-level shapes. As illustrated in Figure 1, it is possible that an item with multiple bases can be in contact with the floor but has no FBS. For this reason, we develop an adapted version of the base support approach that can deal with any three-dimensional shape and correctly compute the base support (AR2).

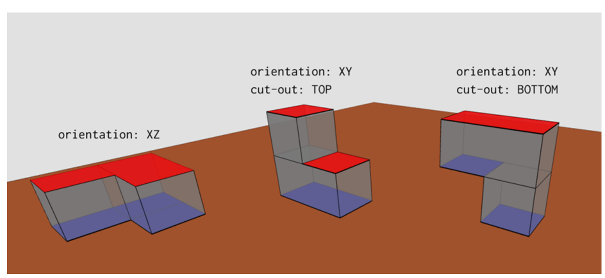

Figure 1. L-shape orientations and cut-out.

Our criterion should be able to deal with full base support and partial base support. Therefore, we resort to the base support factor α ∈ [0; 1] (AR3) that drives an item's amount of support.

In the context of this research, we define that an item arrangement is statically stable, if it is stable in every sequence of its construction (AR4). This implies that loading an item on the pallet, the entire arrangement should remain stable. Further, when facing the constructed item arrangement, there might exist multiple in-between states that are not stable, which would lead to practically infeasible and endangering constructions.

To determine the supporting area, we need to calculate the intersections between items lying above each other. In order to do so, we must consider all intersecting surfaces between the respective two-dimensional top and base shapes (AR5).

Finally, high performances are desirable due to multiple reasons. The faster a static stability approach operates, the more solutions can be assessed in the same amount of time. Especially, when considering metaheuristic approaches to the PLP that hold an entire population of candidate solutions, high performance implies that more candidate solutions are assessed in the same time span. In turn, this can positively impact the overall solution quality. Furthermore, due to short term changes to the flight plan, high performance boosts the flexibility and responsiveness of the system (AR6).

Table 1. Artifact design requirements.

AR	Description
AR1	Support irregular shapes
AR2	Support multi-level shapes
AR3	Allow variable definition of base support factor
AR4	Consider stability for every sequence
AR5	Calculate intersection area for two-dimensional shapes
AR6	High performance

4.2 Design and Development: Instantiation

4.2.1 Adapted Base Support Definition. In line with previous studies, our adapted base support factor is noted as α. This factor α ∈ [0; 1] describes the percentage of an item's base surface that must be supported either by the ground or by other items for the item to be classified as stable.

Every item can be mapped to one or multiple two-dimensional shapes, the items' *base shapes*. We define a base shape to be any surface that can be seen when looking at the item directly from underneath. Thus, the entire *base* of an item is defined by the union of all its *base shapes*. Equivalently, the top surface(s) of an item can be defined as every two-dimensional area that can be seen when looking at the item from a bird's eye view (see Figure 2).

For an item to be supported by another item, two conditions must hold. First, the two items have to be in a top-bottom relationship. This means that for item *s* to give support to item *i*, item *s* must be in a level below item *i*. For item *s* to be considered in a level below item *i*, any top surface of *s* must be on the same altitude as any base shape of *i*. This restricts hovering or overlapping situations with another item. Second, when visualizing top and base shapes in a two-dimensional plane, the two shapes must intersect. If both conditions hold, we define that item *s* supports item *i* (see Figure 3).

In Figure 3, we observe that the L-shape and box are at the same level, although the L-shape reaches higher. The decisive condition is that one of the top surfaces of the L-shape is at the same altitude as the base of the box and the shapes intersect on a two-dimensional plane. In this case, the box is receiving base support from the L-shape. The degree of support is equal to the percentage of base area covered from the top surface. One item can receive support from multiple other items. The item's overall base support factor is then defined as the sum of all support factors the item receives from any other item or the floor.

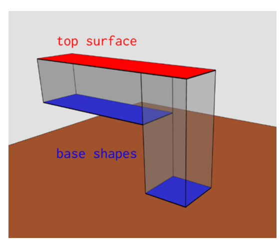

Figure 2: Base and top shapes.

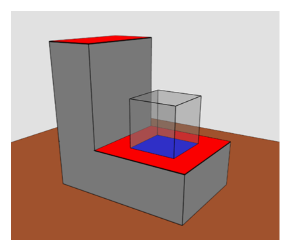

Figure 3: Supporting item.

Formally, we define the base support factor (BSF) for an item $i \in I$ as follows:

- I: set of all items.
- J: set of all items except *i* together with the floor. The floor is modeled as a box with the same dimensions as the pallet.
- B_i: set of all bases for item *i*.
- T_i: set of all top surfaces for item *i*.
- γ_{s_1,s_2}: equal to 1, if the two-dimensional shapes s1 and s2 are located at the same

altitude in three-dimensional space; equal to 0 otherwise.

- $area$(i): area of the two-dimensional shape *i*.
- $intersection(s_1, s_2)$: intersection area of two-dimensional shapes s_1 and s_2.
- $totalBaseArea$(i): the total base area of item *i* ($= \sum_{b \in B_i} area(b)$)
- BSF_i: the base support factor for item *i*.

$$BSF_i = \sum_{j \epsilon J} \sum_{b \epsilon B_i} \sum_{t \epsilon T_j} \frac{intersection(b,t)}{totalBaseArea(i)} \times \gamma_{b,t}$$

If $BSF_i >= \alpha$ for every item $i \in I$ then the entire item arrangement has base support of α.

4.2.2 Inputs and Outputs. We designed our algorithm such that it can be used stand-alone or in optimization processes (e.g., metaheuristics). It evaluates a given item arrangement and returns a static stability score. The input entails an arrangement of items which includes information about all items, their packing sequence, their shape and packing coordinates. The returned score reflects the arrangement's degree of stability. A score of 1.0 means that every item is stable, while a score of $\frac{k}{n}$ indicates that the first *k* out of *n* items are stable, and the first unstable item occurs at sequence *k*. Our algorithm is parameterized by α, the desired base support factor. We employ the same support factor α for every shape type. Consequently, α = 1 equals FBS while 0 < α < 1 implies PBS.

4.2.3 Base Support Calculation. Our algorithm to compute the base support proceeds as follows. For each item *i,* we iterate through all already placed items (i.e., that have a lower sequence) and check the respective base support *i* receives from *s*. If an item is located on the floor (y = 0), we consider the floor as well by appending it to the set of items to be checked.

The core base support calculation algorithm comprises the following stages: (1) x/z-overlap check, (2) y-level comparison, (3) two-dimensional intersection area calculation, and (4) BSF calculation. The process is depicted in Figure 4. It takes two items *i* and *s* as input and outputs the base support item *i* receives from item *s*.

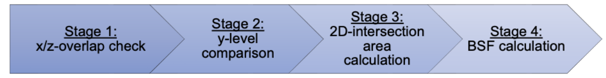

Figure 4. Base support calculation process.

The first stage (1) implements an overlap check on the x and z axes (i.e., when looking from a bird's eye view) and therefore abstracts the y-dimension. If we do not detect any overlap between the two items, we can terminate the computation prematurely, since no base support is possible. This is an analogy to the early bounding box tests in physics engines' broad phase collision tests.

In the y-level comparison (2) we compare the y-coordinates of all base shapes of *i* to all top shapes of *s*. All matching y-coordinates imply potential base contact. With this mapping, we are able to reduce the three-dimensional problem to a two-dimensional computation of intersections between contacting base shapes of *i* and top surfaces of *s*.

Facing our two-dimensional intersection calculations (3), we used proven approaches for simple cases such as the intersection between two rectangles (e.g., using the mins of maxes and maxes of mins principle). For more complicated cases such as polygon intersections, we employ a geometry library called *locationtech*[1]. In terms of circle intersections, we need to account for accuracy loss by allowing a certain error margin when checking the satisfaction of the BSF. We evaluated intersection calculations to have a percentage error of 0.06% and thus allowed an error margin of 0.1%.

Forming the sum of intersections (4), we obtain the supported base area of item *i* by item *s*. The quotient of supported base area to total base area then represents the BSF that item *i* receives from item *s*.

When reiterating this process for all potential supporting items for *i* and forming the sum, we obtain BSF_i. In a last step, the obtained BSF_i is compared to the desired base support factor α considering the allowed percentage error. Formally, the condition

$$|BSF_i - \alpha| \leq \varepsilon$$

is checked, with ε being the allowed percentage error. If this condition is satisfied, we classify item *i* to be stable and proceed with the next item.

To illustrate some insights, we visually depict a loading situation (Figure 5). Displayed is a set of items numbered by their placement sequence. First, consider item 1 for the support of item 4. When executing the first step, we observe no overlap on the x/z axes and thus no

[1] https://locationtech.github.io/jts/javadoc

base support. When considering item 4 and item 2, they pass stage 1 as they overlap on the x/z axes. However, item 4's base shape is located on a different altitude than item 2's top surface, thus no base support is provided (y-level comparison).

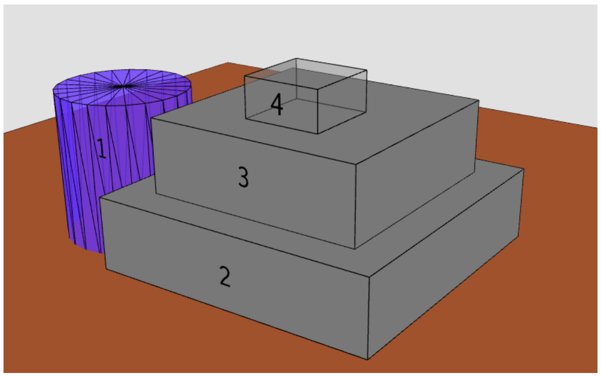

Figure 5. Item support structure.

4.3 Demonstration: Functionality

4.3.1 Example Arrangement. We illustrate our algorithm using an exemplary item arrangement that consists of multiple irregular shapes. Figures 6 and 7 display this exemplary arrangement.

Since the first three items are all placed on the floor and only reveal one base shape, they have FBS and thus are statically stable (Figure 6). In our algorithm, we calculate the intersection of the items' bases with the floor. Consequently, this intersection area equals the base area of the item, so FBS holds.

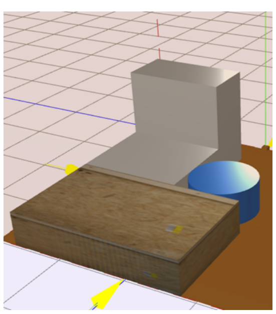

Figure 6. Example arrangement without polygon prism.

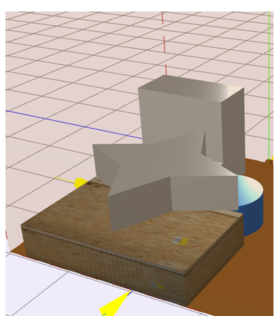

Figure 7. Example arrangement with polygon prism.

In contrast, the polygon prism's support differs from the other items, as it is not placed on the floor, but on top of the L-shape, box, and cylinder (Figure 7). In our algorithm, we calculate the individual areas that are supported by the items below. Starting with the L-shape, the algorithm computes the polygon prism's base area that is covered by any top surface of the L-shape.

To achieve this, we first check the polygon prism's base shape and the L-shapes's top shape for overlap (stage 1). Since an overlap is present, we proceed.

In stage 2, the L-shape compares its top shapes' y-coordinate to the passed base shape's y-coordinate. The first top shape is on a different altitude; thus no support is provided by this surface. However, the second top shape's y-coordinate equals the base shape's y-coordinate, and is therefore positioned at the same height as the base shape of the polygon prism.

In the next step (stage 3), our algorithm calculates the intersection area between those two matching shapes and returns it to the polygon prism. This intersection area is depicted in Figure 8.

In the last stage (stage 4), the polygon prism now divides the returned supported area by its entire base area. This results in the BSF the polygon prism receives from the L-shape, which is approximately 42.84%.

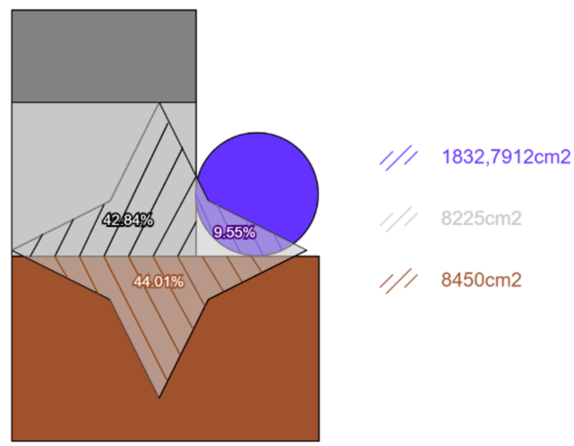

Figure 8. Intersection areas example.

Subsequently, the same procedure is repeated with the other items below (i.e., box and cylinder). We respectively add 44.01% and 9.55% to the BSF. In total, we obtain a BSF of 96,39%, which implies for every α <= 0.9639 the item would be regarded as statically stable. Using the FBS approach, this item would not be marked as statically stable, which implies a score of 0.75.

4.3.2 Unstable Item on Floor. Figure 2 displays an L-shape with two base shapes. Additionally, the item is located on the floor. The example demonstrates the advantage of the generalized base support definition: Regarding previous definitions, the item would be classified as statically stable, since it lies on the pallet floor. However, the item exposes multiple base shapes, which are together only partly supported. So, the item is not stable and would simply tip over towards its longer side. Instead of simply declaring the item as stable for being located on the floor, we extend this notion and consider the floor as an additional item that can give support to other items. As illustrated in Figure 9, a smaller percentage of its base shapes is supported by the floor. In contrast, the larger base area located above the ground is not supported. In total, this item only reaches a BSF of 0.25 and is only classified as stable, if α <= 0.25 is chosen.

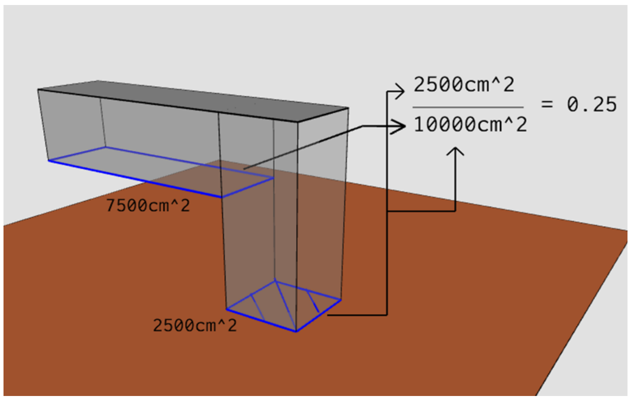

Figure 9: Unstable item on floor.

4.4 Demonstration: Performance

To loop back to our requirements, we opted to design a static stability approach that not only better reflects practical complexity, but also reaches a high performance (AR6). In this study, we operationalize performance through runtime. In detail, we compare runtimes of our new approach to the SIM approach by Mazur et al. (2022). To the best of our knowledge, no other approach for static stability equally takes into account air cargo complexity in terms of shapes.

We randomly generated item arrangements using the dataset of Brandt and Nickel (2019) with varying irregularity factors. An irregularity factor measures the share of irregular (that is: non-boxes) item shapes to the total number of item shapes (from 0 to 1). We generated a total of 2852 item arrangements using a genetic algorithm metaheuristic that assigns items to a placement position and loading sequence, thereby ensuring basic constraints like balancing and maximum weight, but without static stability. The latter condition ensures that our item arrangements are not randomly placed but come close to realistic item arrangements. We then evaluate every single item arrangement using the FBS and SIM approach. The results are aggregated in Table 2 and displayed in Figure 10.

Table 2. Performance evaluation results.

Number of arrangements	Mean irregularity	Mean runtime FBS	Mean runtime SIM
2852	0.2438	0.3172 ms	261.9891 ms

The mean irregularity over all item arrangements was approximately 0.2438. With respect to runtimes, we observe a large discrepancy between both approaches. For FBS,it takes an average of 0.3172 milliseconds to assess one item arrangement. On the one hand, the SIM approach requires substantially more runtime, on average 262 milliseconds for one assessment.

Furthermore, we observe multiple peak runtimes of the SIM approach. We illustrate this finding in Figure 10. For certain assessments, the runtime climbed up to 1252 milliseconds. In contrast, FBS remained nearly constant throughout all tested assessments, which makes it a more robust approach far less sensitive to input complexity.

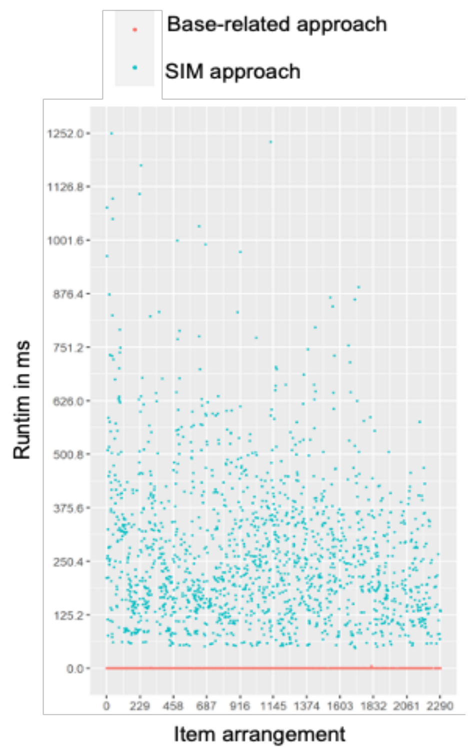

Figure 10: Performances for FBS and SIM.

5. Discussion and Conclusion

In this study, we presented the adaption of FBS and PBS to static stability assessment in air cargo. Both approaches are already well established in related works but are hardly applicable to real-world cases as air cargo. We tackled our goal using a DSR approach that included both the formulation of an adapted base support definition and its implementation within an artifact. Thereby, we raised and covered six artifact design requirements. The key insights of this research are as follows: (1) A more generalized definition of base support is necessary to tackle the complexity of irregular and multi-level shapes and (2) our generalized FBS and PBS approaches perform well in terms of runtime when compared to the state of the art simulation-based approach.

The demonstration illustrated how our developed artifact met the raised requirements. The criterion supports irregular (AR1) and multi-level shapes (AR2). Our adapted base support definition (AR3) uses a minimal support factor $\alpha \in [0; 1]$, which implies the flexibility for FBS and PBS. Further, (AR4) is covered by the identification of unstable sequences and its confluence in a compound score. Moreover, we sketched the intersection area calculation (AR5). Finally, our comparison against the SIM approach illustrated that our artifact achieves a high performance (AR6). We presented an algorithm that comprises four stages that can cope with the identification of intersecting items, a y-level comparison to identify candidate intersection shapes, a dimensional reduction to 2D and a final intersection area calculation.

Regarding the limitations, our study lacks a profound evaluation of performances of FBS and PBS in the context of overall solution generation and optimization. When implemented and integrated in solution-generating optimizations (e.g., genetic algorithms), an entire population of item arrangements is evaluated. A less restrictive constraint (such as FBS or PBS) extends the solution space and simplifies the search for good solutions. Also, performances of constraint assessments impact the overall optimization performance. As demonstrated, the mean runtime for the SIM approach largely exceeds the mean base support runtimes. It remains unclear how much this performance improvement affects overall solution quality (e.g., space utilization).

Another neglected aspect is PBS. We refrained from evaluating our results using PBS, since no statement about the actual static stability of the generated solutions is possible (Ramos, Oliveira, Gonçalves, and Lopes, 2016). This unclear impact of minimal base support on cargo stability might be the focus of future works. In detail, it could be evaluated, how much overall solution quality improvements can be achieved when relaxing the minimal required base support factor α. Further, it could be quantified how often PBS wrongly classifies an arrangement as stable when varying the minimal base support factor. These insights could then help to decide how much minimal base support is necessary regarding the trade-off between solution quality and restrictiveness.

6. References

Bischoff, E. E. (1991). Stability aspects of pallet loading. OR Spektrum, 13(4), 189–197. https://doi.org/10.1007/BF01719394

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

Bracht, E. C., de Queiroz, T. A., Schouery, R. C. S., & Miyazawa, F. K. (2016). Dynamic cargo stability in loading and transportation of containers. 2016 IEEE International Conference on Automation Science and Engineering (CASE), 227–232. https://doi.org/10.1109/COASE.2016.7743385

Brandt, F., & Nickel, S. (2019). The air cargo load planning problem-A consolidated problem definition and literature review on related problems. European Journal of Operational Research, 275(2), 399–410. https://doi.org/10.1016/j.ejor.2018.07.013

Deplano, I., Lersteau, C., & Nguyen, T. T. (2021). A mixed-integer linear model for the multiple heterogeneous knapsack problem with realistic container loading constraints and bins' priority. International Transactions in Operational Research, 28(6), 3244–3275. https://doi.org/10.1111/itor.12740

Dowsland, K. A. (1987). An exact algorithm for the pallet loading problem. European Journal of Operational Research, 31(1), 78–84. https://doi.org/10.1016/0377-2217(87)90140-8

Elhedhli, S., Gzara, F., & Yildiz, B. (2019). Three-Dimensional Bin Packing and Mixed-Case Palletization. INFORMS Journal on Optimization, 1(4), 323–352. https://doi.org/10.1287/ijoo.2019.0013

Gajda, M., Trivella, A., Mansini, R., & Pisinger, D. (2022). An optimization approach for a complex real-life container loading problem. Omega, 107(C). https://ideas.repec.org/a/eee/jomega/v107y2022ics0305048321001687.html

Gehring, H., & Bortfeldt, A. (1997). A genetic algorithm for solving the container loading problem. International Transactions in Operational Research, 4(5–6), 401–418. https://doi.org/10.1016/S0969-6016(97)00033-6

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. MIS Quarterly, 28(1), 75–75. https://doi.org/10.2307/25148625

IATA. (2021). Annual Review-2021. International Air Transport Association.

https://www.iata.org/contentassets/c81222d96c9a4e0bb4ff6ced0126f0bb/iata-annual-review-2021.pdf

Junqueira, L., Morabito, R., & Sato Yamashita, D. (2012). Three-dimensional container loading models with cargo stability and load bearing constraints. Computers and Operations Research, 39(1), 74–85. https://doi.org/10.1016/j.cor.2010.07.017

Krebs, C., & Ehmke, J. F. (2021). Vertical Stability Constraints in Combined Vehicle Routing and 3D Container Loading Problems. In M. Mes, E. Lalla-Ruiz, & S. Voß (Eds.), Computational Logistics (pp. 442–455). Springer International Publishing. https://doi.org/10.1007/978-3-030-87672-2_29

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

Lee, N.-S., Mazur, P. G., Bittner, M., & Schoder, D. (2021). An Intelligent Decision Support System for Air Cargo Palletizing. Proceedings of the 54th Hawaii International Conference on System Sciences, January 5-8, 2021, Hawaii, USA. https://doi.org/10.24251/HICSS.2021.170

Mazur, P. G., Lee, N.-S., & Schoder, D. (2020). Integration of Physical Simulations in Static Stability Assessments for Pallet Loading in Air Cargo. In K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, & R. Thiesing (Eds.), Proceedings of the 2020 Winter Simulation Conference, December 13-16, Orlando, Florida, USA. https://doi.org/10.1109/WSC48552.2020.9383878

Mazur, P. G., Lee, N.-S., & Schoder, D. (2022). A GPU-Accelerated Approach to Static Stability Assessments for Pallet Loading in Air Cargo. Proceedings of the 55th Hawaii International Conference on System Sciences (2022). https://doi.org/10.24251/HICSS.2022.204

Nascimento, O. X. do, Alves de Queiroz, T., & Junqueira, L. (2021). Practical constraints in the container loading problem: Comprehensive formulations and exact algorithm. Computers & Operations Research, 128, 105186. https://doi.org/10.1016/j.cor.2020.105186

Olsson, J., Larsson, T., & Quttineh, N.-H. (2020). Automating the planning of container loading for Atlas Copco: Coping with real-life stacking and stability constraints. European Journal of Operational Research, 280(3), 1018–1034. https://doi.org/10.1016/j.ejor.2019.07.057

Paquay, C., Schyns, M., & Limbourg, S. (2016). A mixed integer programming formulation for the three-dimensional bin packing problem deriving from an air cargo application. International Transactions in Operational Research, 23(1–2), 187–213. https://doi.org/10.1111/itor.12111

Parreño, F., Alvarez-Valdes, R., Tamarit, J. M., & Oliveira, J. F. (2008). A Maximal-Space Algorithm for the Container Loading Problem. INFORMS Journal on Computing, 20(3), 412–422. https://doi.org/10.1287/ijoc.1070.0254

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. Journal of Management Information Systems, 24(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Pollaris, H., Braekers, K., Caris, A., Janssens, G. K., & Limbourg, S. (2015). Vehicle routing problems with loading constraints: State-of-the-art and future directions. OR Spectrum, 37(2), 297–330. https://doi.org/10.1007/s00291-014-0386-3

Ramos, A. G., & Oliveira, J. F. (2018). Cargo Stability in the Container Loading Problem-State-of-the-Art and Future Research Directions. In A. I. F. Vaz, J. P. Almeida, J. F. Oliveira, & A. A. Pinto (Eds.), Operational Research (Vol. 223, pp. 339–350). Springer International Publishing. https://doi.org/10.1007/978-3-319-71583-4_23

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

Ramos, A. G., Oliveira, J. F., & Lopes, M. P. (2016). A physical packing sequence algorithm for the container loading problem with static mechanical equilibrium conditions. International Transactions in Operational Research, 23(1–2), 215–238. https://doi.org/10.1111/itor.12124

Vaishnavi, V., Kuechler, B., & Petter, S. (2017). Design Science in Information Systems Research. Design Science Research in Information Systems: http://desrist.org/desrist/content/design-science-research-in-information-systems.pdf

Zhao, X., Bennell, J. A., Bektaş, T., & Dowsland, K. (2016). A comparative review of 3D container loading algorithms. International Transactions in Operational Research, 23(1–2), 287–320. https://doi.org/10.1111/itor.12094