The Air Cargo Load Planning Problem

Zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

bei der Fakultät für Wirtschaftswissenschaften
des Karlsruher Instituts für Technologie (KIT)

genehmigte

Dissertation

von

Dipl.-Inform. Felix Brandt

Tag der mündlichen Prüfung: 13.09.2017
Hauptreferent: Prof. Dr. Stefan Nickel
Korreferent: Prof. Dr. Kai Furmans

Abstract

Felix Brandt

The Air Cargo Load Planning Problem

A major operational planning problem in the air cargo industry is how to arrange cargo in an aircraft to fly safely and profitably. Therefore, a challenging planning puzzle has to be solved for each flight. Besides its complexity, the planning is mostly done manually today, which is a time consuming process with uncertain solution quality. The literature on loading problems in an air cargo context is scarce and the term is used ambiguously for different subproblems like selecting containers, packing items into containers, or loading containers into aircraft. All of the presented models only focus on certain aspects of what is in practice a larger planning problem. Additionally, some practical aspects have not been covered in the literature.

In this work, we provide a comprehensive overview of the air cargo load planning problem as seen in the operational practice of our industrial partner. We formalize its requirements and the objectives of the respective stakeholders. Furthermore, we develop and evaluate suitable solution approaches. Therefore, we decompose the problem into four steps: aircraft configuration, build-up scheduling, air cargo palletization, and weight and balance. We solve these steps by employing mainly mixed-integer linear programming. Two subproblems are further decomposed by adding a rolling horizon planning approach and a Logic-based Benders Decomposition (LBBD). The actual three-dimensional packing problem is solved as a constraint program in the subproblem of the LBBD.

We evaluated our approaches on instances containing 513 real and synthetic flights. The numerical results show that the developed approaches are suitable to automatically generate load plans for cargo flights. Compared to load plans from practice, we could achieve a 20 percent higher packing density and significantly reduce the handling effort in the air cargo terminal. The achieved costs of additional fuel burn due to aircraft imbalances and reloading operations at stop-over airports are almost negligible. The required runtimes range between 13 and 38 minutes per flight on standard hardware, which is acceptable for non-interactive planning.

Cargo airlines can significantly profit from employing the developed approaches in their operational practice. More and especially the profitable last-minute cargo can be transported. Furthermore, the costs of load planning, handling effort, and aircraft operations can be significantly reduced.

Contents

1 Introduction

> *"A recession is when you have to tighten your belt; depression is when you have no belt to tighten. When you've lost your trousers – you're in the airline business."*
>
> — Sir Adam Thomson, founder of British Caledonian

The airline business is highly competitive. Over the past decades this fierce competition has lead to numerous innovations to utilize the most expensive assets, the aircraft, to the maximum and to reduce costs wherever possible. Operations research has played an important role in this process by providing theory and tools for creating flight and crew schedules, optimizing revenue management, and supporting many more application areas. Staying in the airline business today without such support is not imaginable.

Besides the well-known passenger airline business, a lot of airlines also transport cargo and many passenger flights carry cargo next to baggage in their lower decks. Furthermore, as of 2015 there are around 1,770 large cargo-only aircraft operated by airlines worldwide (Crabtree et al., 2016). Although all these aircraft transport less than 1 percent of world trade tonnage, this accounts for around 35 percent of world trade by value (Crabtree et al., 2016). Albeit its economical importance, air cargo has gained far less R&D attention than the passenger airline business and therefore has considerable potential to be improved by applying current technologies and developing new tools like decision support systems.

One major operational problem present specifically in air cargo is how to arrange the cargo in an aircraft. At first, loading cargo into an aircraft might seem to be a simple task. But, in reality the loading must be carefully planned. For each single flight an airline that transports cargo has to answer a set of delicate planning questions to operate safely and profitably. And there are more challenges: Cargo rates have fallen by around 35 percent over the last 20 years (Crabtree et al., 2014). Therefore, airlines must aim for a high utilization of the aircraft on each flight. On the other hand, there is a soaring overcapacity in the air cargo market, see Murray (2016) and The Economist (2016), which leads to historically low average load factors of the aircraft. Accordingly, airlines are forced to also reduce their operating cost for ground handling and fuel wherever possible to stay in business.

In the past, there have been a couple of scientific contributions addressing "air cargo loading" problems. However, the term is used ambiguously for different decisions to be made in the load planning process. This ranges from selecting the containers to load (Mongeau and Bes, 2003), over solving three-dimensional packing problems (Paquay et al., 2016), to balancing the aircraft for fuel efficiency (Lurkin and Schyns, 2015). All of the presented models only focus on certain aspects of what is in practice a larger planning problem. In this work, we consider the overall problem of operational load planning for airlines that transport cargo and therefore we coin the term *Air Cargo Load Planning Problem* (ACLPP).

This work was partially supported by our industrial partner Lufthansa Cargo AG, the cargo division of the Lufthansa Group. Therefore, the described planning problems are derived from the current operational practice of our partner, but apply to other airlines transporting cargo nonetheless. Besides an insight into practice and access to its planning experts, our partner also provided benchmark data for the analysis of the problem at hand and the evaluation of our approaches.

1.1 Problem statement

Air cargo load planning today is often a manual task that has to be performed by experienced load planners (Amadeus, 2015). Only for parts of the planning process there exist first commercially available decision support systems, like the SABLE Weight & Balance system (de Cleyn et al., 2014). Solving the full ACLPP in practice still involves a lot of pen-and-paper or spreadsheet-based planning as well as trial and error during the actual packing. On the one hand, this leads to high labor costs but often also to suboptimal results, as there is a constant pressure of time and the problem complexity can be quite high.

Accordingly, in today's highly competitive air cargo market, the introduction of automated decision support systems into the load planning process can provide a significant advantage for airlines. First, by increasing the productivity of its load planning staff and also by producing high quality solutions tailored for each individual flight.

The aim of this thesis is to establish a comprehensive view of the load planning tasks in an air cargo terminal, collect the practical relevant aspects, formalize the underlying planning problems, and develop suitable solution approaches. Our scope includes all the planning tasks regarding the cargo loading operations of a flight from the arrival of the individual shipments at the terminal up to the fully loaded and ready-to-go aircraft. Therefore, we follow the planning processes seen in practice. Furthermore, we identify gaps between research and practice. In the end, we develop a decision support system for the ACLPP and two extensions for improving the solutions. Our goal is to achieve a solution quality that is competitive with the planning experts' manual approach and can be computed within a reasonable amount of time on readily available commodity hardware. Ideally, this decision support system can then be used to provide the planners with good and feasible solutions to start their day-to-day load planning tasks.

1.2 Research questions

To structure our work, we follow three research questions:

1) What are the requirements and drivers of air cargo load planning in practice? In the literature, the term "air cargo loading" is used ambiguously and no general models exist. Therefore, we develop a unified nomenclature of the covered aspects and a comprehensive model of the problem as seen in practice. We describe the decisions to be made, which restrictions have to be adhered, and what the overall objective is.

2) Can valid and competitive load plans be generated with reasonable effort? Describing an optimization model is only halfway. We develop and implement a solution approach based on the current manual planning process of our industrial partner. Therefore, we split the problem into multiple steps and solve them one after the other. We evaluate the approach on numerous test instances, stemming from real flights and extreme scenarios to get insights into the quality achieved and the runtimes required.

3) Does splitting the problem lead to adverse effects and what can we do about it? From a practical point of view, the steps performed during load planning should be as independent and self-contained as possible. However, this limits the scope of optimization, which might lead to local optimization and to results that represent extreme cases. This in turn can lead to adverse results in the solution process. We analyze, where such effects may occur in our approach and develop ways to mitigate them.

1.3 Organization of the thesis

Following the research questions outlined in the previous section we structure this thesis as follows:

In Chapter 2 we provide a basic introduction and required background information about the air cargo business. In Chapter 3 we focus on the loading aspects present in an air cargo environment. There, we introduce the Air Cargo Load Planning Problem, identify the load planning stakeholders and their objectives in practice, and describe the current planning procedures and the related challenges.

In the next four Chapters 4 to 7, we introduce the load planning problem and the related model in depth, accompanied by a review of the related literature. The respective chapters each deal with a different subproblem: In Chapter 4 we describe the aircraft configuration, which selects the air cargo containers (ULDs) to use for each flight. In Chapter 5 we introduce the build-up scheduling, which determines the starting time each ULD is assembled. In Chapter 6 we present the air cargo palletization, which distributes items across ULDs and arranges the items in the ULDs. As the last subproblem, we describe the weight and balance in Chapter 7. It determines the arrangement of the assembled ULDs in the aircraft.

In Chapter 8 we introduce the benchmark instances that we use to evaluate our approach. Furthermore, we analyze the problem characteristics seen in practice and define performance indicators to measure the quality of a load plan.

In Chapter 9 we present our main solution approach: SeqACLPP, a sequential planning approach for the Air Cargo Load Planning Problem. We evaluate this approach in Chapter 10, discuss the results found, and identify areas for improvement. After that, we present our implementation of two extensions of the SeqACLPP approach and evaluate their performance in Chapter 11.

In Chapter 12 we summarize the contributions of this work, discuss the results with regard to the considered research questions, and point out areas of future work.

2 Air cargo primer

In this chapter, we present a basic introduction into the world of air cargo and the operational planning problems faced by airlines that handle cargo. The topics are selected with the required aspects of load planning in mind. In Section 2.1, we describe the business of airlines that provide cargo services. Afterwards, we characterize the typical cargo in Section 2.2, the used loading devices in Section 2.3, the airlines' modes of transport in Section 2.4, and the relevant airport processes in Section 2.5. For a more in depth introduction we refer to the *Air Cargo Guide* by Grandjot et al. (2007).

2.1 Air cargo business

Cargo airlines offer the very basic service to transport goods between different airports at a certain price, very similar to the service provided by passenger airlines. But the world of air cargo is more complex than it is for passengers. In Table 2.1, we sketch the main differences between the transport of passengers and cargo. In the remainder of this section, we discuss these differences one by one.

Besides the shipper and the airline, there is an important additional party in air cargo: the forwarder. He is typically involved in transporting the cargo from the original shipper to the departure airport and from the arrival airport to the consignee. Accordingly, the forwarders are by far the most important direct customers of a cargo airline, not the original shippers. The market of forwarders is more concentrated than the airline market. This gives bigger forwarders a lot of power in negotiations. According to Burnson (2013), the six largest air cargo forwarders with their approximate market share as of September 2013 are: DHL Global Forwarding (9.3%), DB Schenker (4.4%), Kuehne+Nagel (4.4%), Kintetsu World Express (4.3%), UPS Supply Chain Solutions (3.4%), and Panalpina (3.2%). Together they already represent nearly 30 percent of the global forwarding market.

There is a range of different business models among airlines that offer cargo services. The majority of airlines can be placed into one of these five categories:

- **Passenger airlines** operate only passenger aircraft and sell the cargo capacity in the lower deck that is not used for baggage. As this so-called *belly space* is quite limited and some goods are not allowed in there, cargo is more a by-product for those airlines. Known carriers in this category include Delta Airlines, United Airlines, or SAS Scandinavian Airlines. However, some passenger airlines, especially low-cost-carriers, even do not sell the free belly space. This way, they achieve shorter turn-around times at the airports and higher operational flexibility.
- **Integrated carriers**, also called *express carriers*, combine the role of an airline and the forwarder. Large carriers are for example DHL, UPS, FedEx, or TNT. They operate their own network of trucks, aircraft, and transition points. Thereby,

Passenger	Air cargo
the **passenger** is the customer	a **forwarder** is an intermediary between the airline and the customer
standard business model, sell seats on scheduled flights	**diverse business models** including express, cargo-only, combination carriers
one seat, one passenger	**complex 3D puzzle** of volume, weight, and handling requirements
active, passengers will watch for their flight, board on their own, and contact staff on problems	**passive**, all (un)packing has to be done by airline staff, cargo will not notify anybody if it is forgotten
bidirectional flows, most passengers will return to their origin	**unidirectional flows**, cargo flows are highly imbalanced as cargo seldom returns to its origin
highly predictable, customers book and pay months in advance	**hard to predict**, most bookings are made at short notice, transport is usually post-paid

Table 2.1: Main differences between the passenger airline and air cargo business

they provide a one-stop-shop solution for shippers. These carriers provide a more standardized and automated service specifically designed for small shipments and documents. Therefore, these carriers usually apply limits on the maximum size, weight, and handling requirements of the shipments.

- **Combination carriers** operate both passenger aircraft and cargo aircraft. Similar to the passenger airlines, they transport cargo on their passenger aircraft as a by-product. However, they couple this service with their backbone network of cargo-only flights. This way, the smaller passenger flights can feed cargo into the cargo-only network to increase its overall utilization. Additionally, some combination carriers also provide trucking services, to transport cargo from/to smaller airports that are not serviced by cargo-only aircraft. Shipments in the cargo-only network and trucking services can be of almost any size respecting the aircraft dimensions and also contain hazardous goods. Carriers from this category often operate as a special cargo division of large airlines. Examples are Lufthansa Cargo, Emirates Sky Cargo, or Cathay Pacific Cargo.
- **Cargo-only carriers** operate exclusively cargo aircraft. They have a significantly smaller network than the combination carriers. Typically, they operate between large markets, like China and the USA, to fill their capacity. Thus, they leave the main part of feeding and consolidation to their forwarders. Examples for large cargo-only carriers are Cargolux, AirBridgeCargo, or Nippon Cargo Airlines.
- **Charter carriers** provide the most flexible and the most expensive air cargo transportation service. With them, the shipper can transport nearly any good between any pair of airports as required by his own schedule. Therefore, these carriers are

typically used for large machine parts, for example, power transformers or drilling equipment, or for emergency cases requiring an entire aircraft. There are few dedicated charter carriers like Atlas Air, Southern Air, or Antonov Airlines, but most other carriers will also offer their aircraft for charter.

This work mostly applies to combination and cargo-only carriers. The loading problems faced by the other types of carriers are usually easier to solve. The shipments transported by express carriers and passenger airlines are mostly small, have little requirements, and do not interfere with other shipments. Charter carriers often transport only a single bulky item or mostly homogeneous pallets, like in humanitarian aid.

Besides the business models, the passenger and cargo business differ in several ways:

In the passenger business one seat equates to one passenger. Managing the cargo capacity of a flight is much harder. To estimate the remaining cargo capacity, the airline has to track weight, volume, and a variety of handling requirements, e.g., incompatibilities between shipments. Furthermore, it must solve a three-dimensional planning puzzle, defining how all items can be safely arranged inside the aircraft.

Transporting air cargo involves extensive ground handling operations. First, the cargo needs to be transported by conveyors or fork lifts inside the terminal. Furthermore, it must be packed onto ULDs, lashed, and tracked on its way. Especially the packing part gets more difficult the more heterogeneous the cargo is.

Another difference between the air cargo and passenger business is the geographic structure of transport demands. Passenger flows are usually bidirectional as most passengers return to their origin. Therefore, passenger aircraft are typically operated as shuttles between an airline's hubs and other airports. In contrast, cargo flows are highly imbalanced and cargo airlines often operate flights with *stop-overs*, i.e., only a part of the payload is (un)loaded at a stop-over airport and the aircraft continues its journey under the same flight number.

To be able to describe certain subsections of a flight with stop-overs we define three terms that appear frequently throughout this work. We define a *flight* as any sequence of aircraft movements performed under the same flight number. A *leg* is a nonstop aircraft movement between two airports. A *segment* is a subsequence of legs of a flight on which cargo can be transported. We note, that these terms are not used consistently in the airline world, but we stick to the above definitions for simplicity.

Flights contain stop-overs mostly for two reasons. First, there might not be enough cargo demand between two airports to justify the use of a large cargo aircraft. By adding more stop-overs in between, the airline can attract more shipments and thus increase the utilization of the aircraft. Lufthansa Cargo for example operates flight LH8272, depicted in Figure 2.1, from Frankfurt to Santiago de Chile with stop-overs in Dakar, Senegal, and Viracopos, Brazil. The second reason is that splitting a long flight into two shorter legs with a stop-over for refueling allows the airline to load less fuel on each leg and instead to load more cargo into the aircraft. In the latter case, the amount of loaded/unloaded cargo at the stop-over airports is usually smaller than in the former case. An example is the Lufthansa Cargo flight LH8475 from Hong Kong to Frankfurt with a 60 minute stop-over in Almaty, Kazakhstan, at half-distance.

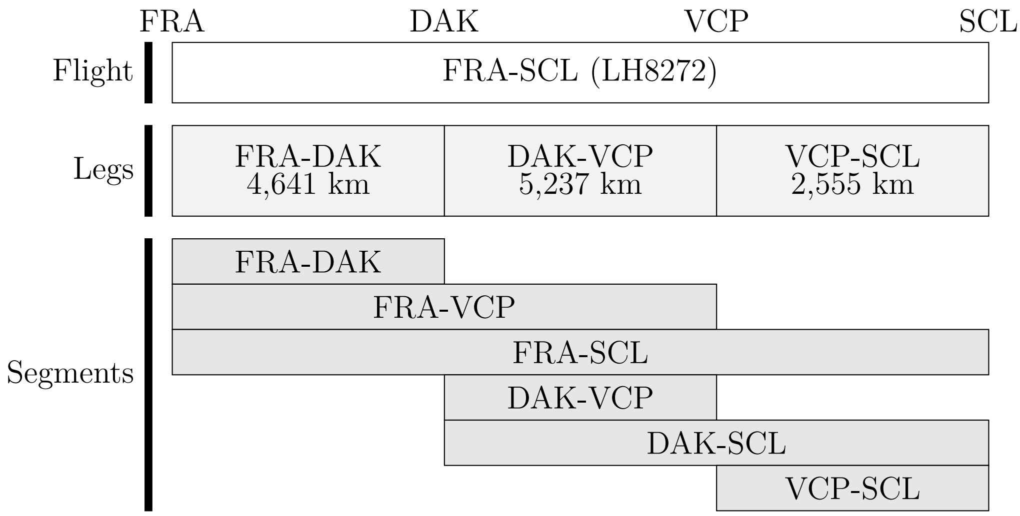


Figure 2.1: Illustration of the definition of flights, legs, and segments: Flight LH8272 consists of three successive legs and technically cargo could be transported on each of the six segments.

From a sales perspective, the air cargo business is split into two phases. In the first phase, customers with a regular demand, typically forwarders, negotiate *long-term contracts* (LTC) with the airline. An LTC allows the customer to ship a certain amount of cargo on a defined set of flights for a fixed rate. This way, the airline increases its planning reliability and the customer gets a better price. Nevertheless, the actual cargo that will be on a flight is only booked a few days in advance. Depending on the contract the customer is able to resign from a flight at short notice, sometimes even without any cost.

The second phase of capacity sales is the regular booking process, which works similar to the one for passengers. The customer requests a quote, books capacity for a certain route, and specifies the cargo properties like size, weight, and handling requirements. Most bookings only take place a few days before the flight, much shorter than in the passenger business.

To conclude, the air cargo business has some major differences to passenger airlines. Unidirectional cargo flows and late bookings make it hard to predict and plan ahead. When it comes to loading an aircraft, a lot of handling manpower is involved to solve a complex three-dimensional planning puzzle in short time.

2.2 The cargo

The types of goods that are transported by air is manifold. Shipments can be anything as small as a pocketbook, as large as an aircraft engine, or as long as the mast of a sailing vessel.

Air cargo is a fast and safe way of transport and primarily attracts shipments that profit from these features. The transport by air is expensive, often 10 to 50 times the price of land transport. Therefore, it is usually only chosen for goods with at least one of the following four properties:

- **urgent:** The goods need to be at a distant place within short time. Typical goods include spare parts, fashion, or living animals.
- **perishable:** The goods degrade when they are not properly handled, for example cooled. Therefore, they should be in a controlled environment and arrive at the destination as fast as possible. Typical goods include flowers, drugs, or fresh food.
- **valuable:** The goods are of high value and have to be transported in a secure way. Typical goods include semiconductors, banknotes, or works of art.
- **dangerous:** The goods may harm their environment when they are not properly handled, which could mean tilting, too high surrounding temperature, or shock. Therefore, they have to be handled with care and the airline needs to be prepared for accidents by training its crews and, for example, installing an automated fire-extinguishing system on the aircraft. Typical goods include chemicals, batteries, or radioactive materials.

Although some shipments can be quite small, a significant share of the cargo transported by combination or cargo-only carriers is at least the size of a wooden pallet, like those defined in the ISO 6780 standard (ISO, 2003). Depending on the originating world region typical pallet sizes are 1,016x1,219 mm (North America), 800x1,200 mm (Europe), or 1,100x1,100 mm (Asia) (Wikipedia, 2016). Throughout this work we refer to the European pallet format 800x1,200 mm as EUR-pallet. Most shipments are box-shaped but irregular shapes, barrels, or just sacks occur frequently. Often multiple boxes are already consolidated onto a wooden pallet by the shipper and should not be separated by the airline. The packaging is often made of cardboard, wood, or styrofoam and optionally wrapped with plastic foil or protected by reinforced metal edges.

Accordingly, one challenging task in load planning is to find a good combination of these strongly heterogeneous shipments to fill the aircraft capacity in a secure and efficient way.

2.3 Unit load devices

To streamline the ground handling processes, the cargo is most often assembled onto special air cargo pallets or containers, generally named *unit load devices* (ULD), see Figure 2.2. Inside the aircraft the ULDs are placed on designated loading positions and locked into position by latches on the floor. As the aircraft fuselage has a near circular cross section, there exist ULD types with different shapes to efficiently use the aircraft's interior space. Figure 2.3 shows a few frequently used ULD types.

Tables 2.2 and 2.3 present popular types of ULDs, which can be loaded into many aircraft. However, most aircraft require certain types of ULDs for specific loading positions. There are general purpose ULDs, like plain aluminum pallets and containers of different formats, as well as special purpose ULDs for transporting cars, horses, or frozen goods.

Pallets are generally preferred for larger shipments, because they can be easier packed than containers and their contour can be freely chosen. It is also common to pack pallets with overhanging items to fill the full width of an aircraft's lower deck. However, each

Figure 2.2: Loaded PMC pallet with main deck contour (left). Lower deck AKE container (right). Source: Lufthansa Cargo AG, Photographer: Stefan Wildhirt (left), Ingrid Friedl (right)

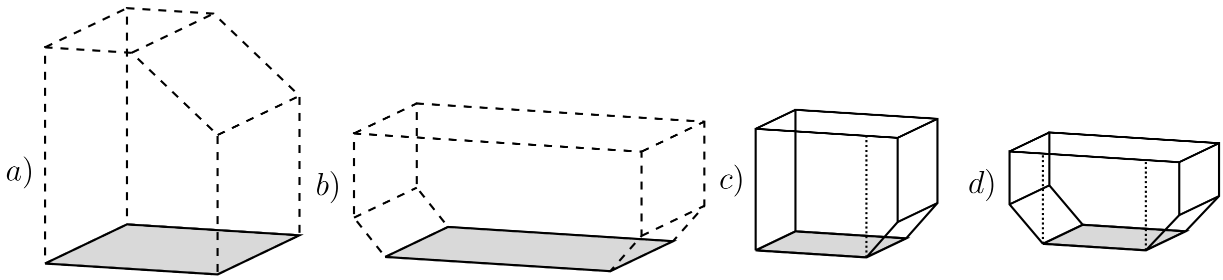


Figure 2.3: Example ULD types in original proportions. Dashed lines mark the allowed contour of a filled pallet while solid lines mark floor sheets and walls of containers. a) main deck pallet (PMC) used in freighter aircraft b) lower deck pallet (PMC) and c) half-size lower deck container (AKE) used in freighter and wide-body passenger aircraft d) lower deck container (AKH) used in narrow body passenger aircraft.

Type	**Volume**	**Dimensions**			**Tare**
		Width	**Depth**	**Height**	
AKH	3.6 m^3	146-239 cm	144 cm	111 cm	85 kg
AKE	4.3 m^3	146-195 cm	142 cm	160 cm	66 kg
ALF	9.0 m^3	305-400 cm	142 cm	160 cm	230 kg
AMP	11.6 m^3	223 cm	305 cm	154 cm	305 kg
AMJ	16.5 m^3	230 cm	306 cm	240 cm	320 kg

Table 2.2: Common ULD container types. The given dimensions are the usable space inside the ULD. Tare is the empty weight of the ULD.

Type	Width	Depth	Tare	Remarks
PAG	210 cm	304 cm	115 kg	MD, or rotated 90° in LD
PMC	230 cm	304 cm	115 kg	MD, or rotated 90° in LD
PGE	230 cm	592 cm	530 kg	2xPMC for large shipments
PMW	305-400 cm	230 cm	167 kg	LD, with side extensions
PKC	146-239 cm	144 cm	80 kg	LD, with side extensions

Table 2.3: Common ULD pallet types. Tare is the empty weight of the pallet including net and straps. The loadable volume depends on the constructed contour.

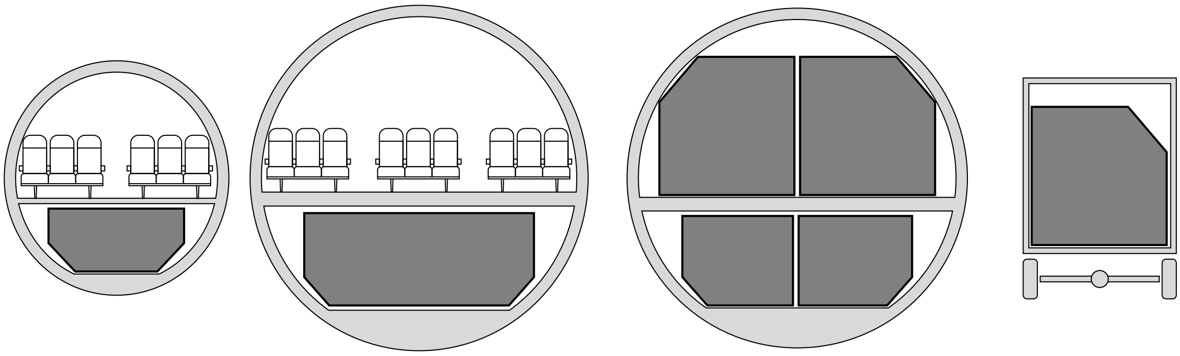


Figure 2.4: Cross sections of typical modes of transport (from left to right): lower deck of narrow-body aircraft (A320), lower deck of wide-body aircraft (A350), main deck and lower deck of freighter aircraft (MD11F), and road feeder trucks.

pallet needs to be covered by a net and straps after it is built, which requires extra handling effort. Containers on the other hand have a predefined contour and side walls such that usually no net or straps are required. Accordingly, they can only be placed at positions where their contour fits in and less handling effort for securing the ULD contents is required. They are preferred for smaller shipments or baggage.

2.4 Modes of transport

We distinguish four different modes of transport, shown in Figure 2.4, that combination carriers usually employ. On short and less frequently booked routes, trucks are used to transport cargo between airports. These systems are known as *road feeder service* (RFS). The trucks can either be loaded with loose items or, if they have a roller bed installed, with ULDs.

Furthermore, we identify three categories of aircraft with respect to the cargo capacity. We give an overview and list their technical details in Table 2.4. The smallest are narrow-body passenger aircraft like the Airbus A320-family or the Boeing B737/B757. They are used mostly on short-haul routes and have only little cargo payload of up to 3 tonnes. Some of these aircraft can only load loose cargo. The next category by size are wide-body passenger aircraft like the Airbus A330/A340/A350/A380 or the Boeing B747/B767/B777/B787. They can be used on long-haul intercontinental routes, transport

Type	Model	PAX	Payload max	Payload net	Cargo volume	ULDs MD	ULDs LD
narrow-body	A320	180	19 t	2 t	37 m^3	0	0
	B737	122	15 t	2 t	23 m^3	0	0
	B757	186	25 t	6 t	51 m^3	0	0
wide-body	A330	300	109 t	17 t	150 m^3	0	32
	A340	335	112 t	15 t	158 m^3	0	32
	A350	325	76 t	15 t	170 m^3	0	36
	A380	555	89 t	12 t	175 m^3	0	36
	B747	412	67 t	13 t	177 m^3	0	26
	B767	269	44 t	15 t	114 m^3	0	30
	B777	370	70 t	27 t	214 m^3	0	44
	B787	242	43 t	12 t	137 m^3	0	28
freighter	MD11F	0	93 t	93 t	535 m^3	26	32
	B747F	0	113 t	113 t	615 m^3	29	32
	B777F	0	103 t	103 t	580 m^3	27	32

Table 2.4: Common aircraft types and their cargo capacities. The columns give (from left to right): The aircraft type, model name, maximum number of passengers, the maximum payload (passengers + cargo) the aircraft can bear, the typical cargo payload after loading (passengers + baggage + fuel), the usable volume of the cargo compartments, the number of standard PMC pallet equivalents on the main deck and the number of standard AKE container equivalents on the lower deck. All numbers are only indicative and can vary between different aircraft versions and even airlines. (Source: Airport planning manuals of Airbus and Boeing)

ULDs, and have a medium payload between 10 and 27 tonnes. The largest cargo aircraft are cargo-only aircraft, like the freighter versions of the McDonnell Douglas MD-11 or the Boeing B747/B777. They carry no passengers, are loaded with ULDs on the *lower deck* (LD) and *main deck* (MD), and have a typical payload of 90 to 130 tonnes.

2.5 Airport processes

Air cargo is transported between airports and usually processed in separate cargo terminals. If no direct connection between the origin and destination airport exists, the cargo will be transshipped between its legs at one or multiple hubs. The general workflow at all the involved cargo terminals is shown in Figure 2.5.

Typically, an air cargo terminal has two interfaces: one towards the customer and another towards the airline's network of aircraft and RFS. The customer drops his shipment at the export point either as single items or pre-built ULDs. Depending on the scheduled departure time of the flight on which the cargo is booked it is then either stored tem-

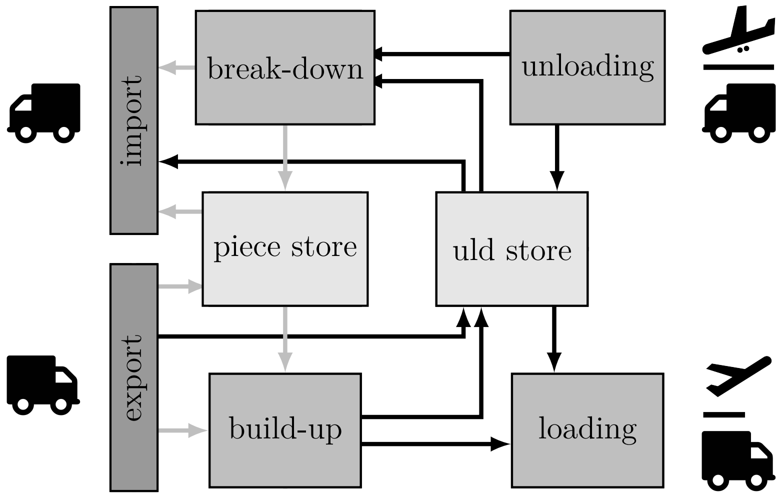


Figure 2.5: Typical material flow in an air cargo terminal (gray arcs represent items, black arcs ULDs). Cargo is unloaded at the terminal from incoming flights or trucks, or it is dropped by the customer at the export interface. Inside the terminal the cargo is broken down from ULDs into single items, stored, and build up into ULDs. The cargo leaves the terminal to be loaded into an aircraft or truck, or to be picked up by the customer at the import interface.

porarily or directly sent to the build-up process. During build-up the items are packed onto ULDs. The ULDs are weighed and then loaded into an aircraft or RFS truck.

After unloading at the next station the ULDs are again stored as a whole or broken down into their items. These can then be picked up by the customer through the import point or continue their journey towards their final destination.

In the following, we describe the two processes relevant for load planning, namely the build-up and aircraft loading. The corresponding unload and break-down processes are simpler and do not require as much description.

Build-up process

The build-up process is the most time consuming physical step done in an air cargo terminal. Here the ULDs are packed with cargo. See Figure 2.6 for a sample view into the terminal build-up area. The workers start with one or more empty ULDs and an assortment of items assigned to the same transport segment. They have to pack the items onto the ULDs in a legal and safe way. Large or heavy items are placed by using a fork lift. Smaller items are usually placed by hand.

This task is challenging for several reasons: The majority of ULDs is built several hours before the flight departs. Therefore, the items at the workstation are generally a subset of the cargo that already arrived at the terminal for the considered flight. As this amount of cargo can be quite large, see the stated volumes in Table 2.4, it is hard to keep track of and to combine the shipments such that all items fit onto the ULDs. Furthermore, the workers have to make sure that the weight limit for each ULD is not exceeded and the

Figure 2.6: Build-up workstations in a cargo terminal. Source: Lufthansa Cargo AG, Photographer: Werner Bartsch

given maximum contours for each ULD are respected. Single over-sized items might be placed on a ULD with overhang and another ULD, which then has to be placed on an adjacent position in the aircraft, needs to leave some free space at the adverse position.

The items have to be packed in a stable way, i.e., no tipping, slipping, or breaking because of too much top load is acceptable. This load stability is also required during takeoff, flight, and landing, when the aircraft is tilting or exposed to accelerating forces. To secure the items on pallet ULDs the workers use nets and straps. Furthermore, dunnage materials like wooden boards or empty wooden pallets are used to distribute point loads across larger surfaces, to lock items into position, or to provide an even loading surface for further items.

In addition, there is a constant pressure of time. Each aircraft has a scheduled departure time and will rarely wait for a single missing ULD. Therefore, the decisions once made during build-up should not be revised too often.

Aircraft loading

The final physical step of the loading process is to load the built ULDs into the aircraft. The containers and pallets are loaded into special ULD compartments equipped with rollers and latches on the floor, see Figure 2.7. Most aircraft have two separate compartments on the lower deck, one in front of and another behind the center wing box, see Figure 2.8. Cargo aircraft also have a main deck compartment spanning along the fuselage. Most compartments only have a single door for the ULDs to enter and leave. Accordingly, loading is often a first-in-last-out (FILO) process. This is especially important for cargo aircraft as they frequently fly on multi-leg flights, i.e., they are not fully unloaded at each airport and some ULDs will continue their journey on the same aircraft.

Main deck compartments often have two separate lanes. Nevertheless, lateral movements and rotations of ULDs can only be made near the door, where rotatable rollers are installed on the floor. Lower deck compartments usually have a single full width lane that can be

Figure 2.7: Aircraft loading: Empty main deck of a cargo aircraft with rollers and latches on the floor (left). Loading of packed ULDs into the aircraft (right). Source: Lufthansa Cargo AG, Photographer: Jannah Baldus (right)

split to accommodate two half-width contoured containers. Each compartment is divided into distinct loading positions. An example of the loading positions of an MD11F is given in Figure 2.8.

In most cases, there is a standard configuration for each compartment, i.e., a valid assignment of ULD types to loading positions that maximizes the space utilization. Depending on the installed floor latches other configurations might be possible. A typical change is to replace two standard ULDs by one large ULD to accommodate larger items. Other reasons to deviate from the standard configuration might be a shortage of a certain ULD type at the origin or destination airport, or to reduce the number of handling operations. To keep things simple, airlines therefore prefer a certain configuration and deviate from it only if it is necessary.

One important aspect of aircraft loading is called *weight and balance* (WAB). As shown in Table 2.4, the total payload an aircraft can carry is limited. Furthermore, the aircraft structure has certain stress limits. There is a maximum weight for each loading position and there also exist weight limits for various subsets of loading positions that are effectively smaller than the sum of their individual weight limits. Typically, there are limits for adjacent positions or positions above each other in the main and lower deck.

Besides the weight limits there is a balance restriction. As each laden ULD has its individual weight and most ULDs could be placed at different loading positions in the aircraft, the loading assignment influences the aircraft's *center of gravity* (CG). The CG always has to stay within an allowed range, which depends on the aircraft type, to remain maneuverable. If the aircraft is too nose-heavy, the front gear would not lift for takeoff. Or, if it is too tail-heavy, it might tip over on the ground. The lateral balance must also be respected but is usually less important as the lever is rather small compared to the longitudinal balance. Additionally, the pilots are able to pump fuel between the left and right wing tanks to distribute the load evenly during flight. Besides the aircraft's hard CG limits, there is an optimal CG individual to the aircraft at which it consumes the least fuel. On most modern large commercial aircraft the optimal CG is close to the aft hard CG limit. To get an impression of the CG impact on fuel consumption, Mongeau

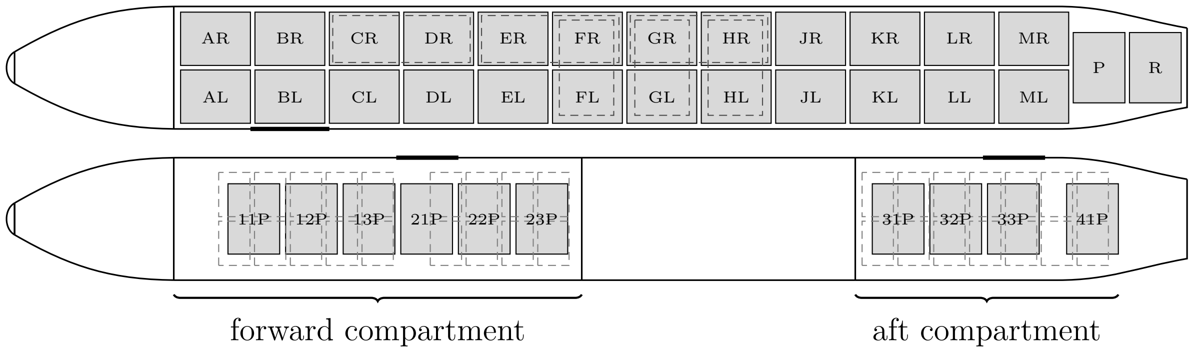


Figure 2.8: Layout of the loading positions of an MD11F freighter aircraft (top: main deck, bottom: lower deck). Doors are marked as thick lines along the fuselage. Standard loading positions are marked in gray. Alternative loading positions for larger ULDs (e.g., CR+DR or GR+GL) and smaller ULDs (e.g., 11L/R) are marked as dashed lines.

and Bes (2003) report that a CG shift of 75 cm on an Airbus A340 might lead to 4 tonnes of wasted fuel per 10,000 km flown.

2.6 Summary

In this chapter, we provided a short introduction into the air cargo business and the components one needs to understand the loading problems faced by cargo carriers.

We started with pointing out the differences between the passenger and air cargo business, introduced the typical cargo characteristics, the container types (ULDs) into which the cargo is packed, as well as the types of aircraft used.

After that, we introduced the outbound cargo handling processes, namely the build-up and aircraft loading. In the next chapter we define the general load planning problem in more detail and the steps the carriers have to undertake before a cargo flight can start.

3 Air cargo load planning

In the last chapter, we introduced the two main physical processes conducted before a cargo flight: the ULD build-up and aircraft loading. In this chapter, we look at the planning tasks that go along with these physical actions.

Our contribution here is twofold. First, we describe the full operational load planning problem as faced by airlines that transport cargo. Only separate parts of it have yet been studied in the literature. Second, we discuss the challenges of current manual planning approaches, which will lead us to our computerized planning approach proposed in Chapter 9.

In the following, we introduce the decisions made during load planning in Section 3.1 as well as the airline's stakeholders and their respective objectives in Section 3.2. We discuss the current state of the planning practice in Section 3.3 and identify current challenges in Section 3.4.

3.1 Planning tasks

In the daily practice of ULD build-up and aircraft loading at our industrial partner and from the present literature (Mongeau and Bes (2003), Rong and Grunow (2009), Paquay et al. (2016), Lurkin and Schyns (2015)) we identify four types of decisions to make:

1. **What** ULDs should be built?
2. **When** should the ULDs be built?
3. **How** to pack items on the ULDs?
4. **Where** to load the ULDs inside the aircraft?

The parameters and constraints to make these decisions properly are quite extensive. Therefore, we split the general load planning into four subproblems. Each tackles one type of decision:

1. **Aircraft Configuration Problem (ACP)**:
   Select the types and number of the ULDs to be built for a flight.
2. **Build-up Scheduling Problem (BSP)**:
   Decide at what time and workstation each ULD is built.
3. **Air Cargo Palletization Problem (APP)**:
   Decide the assignment of items to and their placement inside the ULDs.
4. **Weight and Balance Problem (WBP)**:
   Decide the loading positions of the ULDs inside the aircraft.

We denote the union of these four subproblems as the Air Cargo Load Planning Problem (ACLPP). Parts of these subproblems and their constraints have already been studied

Figure 3.1: Load planning stakeholders and their main objectives

separately in the literature. To our knowledge, there is no published work yet that describes the planning problems and related processes in the practically required level of detail. We introduce the ACLPP by its four subproblems in detail in the upcoming Chapters 4 to 7.

3.2 Stakeholders and objectives

We identify three stakeholders whose objectives need to be considered during load planning. These are the departments of sales, handling, and aircraft operations. An overview of their main objectives is given in Figure 3.1.

The sales department offers the cargo capacity of all flights to the customers. Its main objective is to maximize the revenue of each flight. Besides maximizing the space utilization, the desired load plans should therefore also allow to transport as much special cargo as possible. Outsized items, dangerous goods, or living animals might have special loading requirements but usually provide much higher yields than standard cargo.

The handling department executes all cargo operations inside the cargo terminal. They receive (deliver) cargo from (to) the customers at the ramp, store it in and retrieve it from the warehouse, and perform the ULD build-up and break-down. As the build-up and aircraft loading processes have tight deadlines, the primary handling objective is to minimize loading effort and keep things simple. Furthermore, the workload should be evenly distributed throughout the day to reduce peak demands for workers and workstations.

The aircraft operations department prepares the aircraft for each flight. In particular, it plans how the aircraft is loaded and fueled. Besides loading complete set of built ULDs, it has two main objectives. First, to reduce the amount of required fuel by distributing the load inside the aircraft in the best possible way. Second, the ULDs should be arranged in a way that only a small number of reloading operations are necessary at each stop-over

	Sales	Handling	Operations
Aircraft Configuration Problem (ACP)			
maximize load/revenues	$+$	$-$	$-$
minimize handling effort	$-$	$+$	$+$
Build-up Scheduling Problem (BSP)			
even workload	$-$	$+$	$\circ$
late build-ups	$+$	$-$	$-$
Air Cargo Palletization Problem (APP)			
maximize load/revenues	$+$	$-$	$-$
minimize handling effort	$-$	$+$	$\circ$
Weight and Balance Problem (WBP)			
minimize fuel consumption	$-$	$\circ$	$+$
minimize handling operations	$\circ$	$+$	$+$

Table 3.1: Overview of subproblem objectives and how they align with the goals of the stakeholders, i.e., sales, ground handling, and flight operations. We mark three levels: conformity ($+$), conflict of interest ($-$), and no influence ($\circ$).

airport. This reduces the turnaround time as well as the wear and tear of the aircraft, and results in less risk to damage the cargo.

In Table 3.1 we give an overview where the stakeholders' goals echo in the objectives of the defined subproblems and how they affect the goals of the other stakeholders. For example, the minimization of effort during palletization might have a negative impact on the sales goals ($-$) as it reduces the aircraft utilization. It has a positive impact on the handling goals ($+$) as it reduces the number of handling operations. The goals of flight operations are not influenced ($\circ$) by the handling effort

From our experience, there is no clear hierarchy of objectives. What is most important strongly depends on the overall situation. On a highly booked flight load maximization seems to be the top priority. However, if the flight departs during a rush hour, it might also be important to keep the handling effort low to be able to meet all deadlines. Finally, weight and balance might become a driving factor on long-haul flights that operate at the aircraft limits.

3.3 Traditional planning practice

In this section, we describe the traditional planning practice at our industrial partner. We refrain from a more detailed description for confidentiality. The ACLPP is an operational problem and needs to be solved roughly throughout the last 24 hours before each flight. Flights departing at around the same time compete for the same build-up resources. Therefore, their planning problems are interwoven. When the load planning begins, most parameters like the booked cargo and the workforce in the terminal are mostly fixed.

Corresponding to the four subproblems introduced in Section 3.1, the planning process is split into four stages each performed by appropriately trained workers.

In the first stage, denoted as load planning, the planners select the ULD types and their respective number to built for each destination airport of the flight. They do this based on the booked cargo weight and volume for each destination, the aircraft specifications, internal regulations, as well as their personal experience. Furthermore, they add some extra ULDs that should be filled with low priority cargo. These ULDs will be built and sent to the aircraft but will only be loaded into it if an originally planned ULD is not loadable. This might happen if a ULD arrives too late or is not properly packed and thus rejected by the ramp agent at the aircraft. Usually, items are not assigned to specific ULDs yet, but the planners might already suggest a certain ULD or ULD type for special cargo. The list of ULDs to build and the suggested assignments are forwarded to the next stage.

In the second stage, denoted as build-up planning, the planners decide about the schedule of build-ups for all outgoing flights. They trigger the build-up of groups of ULDs and assign an available workstation to them. Only nonstandard ULDs are scheduled on an individual basis. The basic weekly workstation schedule, defining which flights are built when and where, is predetermined usually for a full six-month flight plan. Accordingly, the planners mostly supervise the task queue at each workstation and manually intervene when disruptions occur. Furthermore, the planners make sure that enough cargo is available at the workstations. They release batches of shipments from the warehouse and send them to the workstations. Except for the suggested assignments from the first stage, still no individual items are picked at this stage. Thus, the result of the second stage are workstations set up with empty ULDs and an assortment of cargo.

In the third stage, denoted as palletization, the workers pack the cargo into the ULDs. They decide which item is placed onto which ULD and where. The goal is generally to put as much cargo as possible onto the ULDs. On overbooked flights, as few items as possible should be left behind at the cargo terminal. Even if a flight is not fully booked, dense packing is a goal as the remaining space can be filled with shipments booked on later flights to the same destination that already arrived at the terminal. The workers usually decide ad-hoc about the item placement based on their experience and the available cargo within visual range. To get an impression of what is stackable and what is not, the workers frequently touch and feel the items. Usually, they start with large, heavy, or other critical items and finish with small undemanding cargo. Furthermore, they might use dunnage material to provide an even base for further items or to protect and secure items. During the entire build-up process, the workers have to ensure the load safety and loading restrictions, like those stemming from *dangerous goods regulations* (DGR) defined by IATA (2016). The workers are responsible for a safe and legal build-up. At the very last, the workers close the ULDs. ULD pallets are covered with a net and lashed with straps. Inside ULD containers the items are locked into position by straps if necessary before the container's doors are closed. The finished ULDs are sent to a weigh station and stored until they are sent out to the aircraft.

In the fourth stage, denoted as weight and balance, the planners assign the assembled ULDs to loading positions inside the aircraft. They usually apply a greedy strategy starting to place heavy ULDs at the center of the aircraft and then moving outwards.

After all ULDs have been placed, the planner might apply local modifications changing the aircraft's center of gravity to reduce its fuel consumption. During this manual process the planner is supported by tools that calculate the aircraft's performance indicators. Tools that check the feasibility of a load plan are usually already provided by the aircraft manufacturer. The final plan is delivered to the aircraft, where the ramp agent loads the ULDs accordingly. He can still make final adjustments if a ULD cannot be loaded and has to be replaced by one of the available low priority ULDs.

3.4 Challenges of current practice

The procedures described in the last section provide several challenges. Most obvious, many decisions are made manually by the planners although there is a constant pressure of time and the problems at hand are demanding. One result of this situation is, that the planners often stop at the first feasible solution they find. Thus, they only solve the satisfiability part of what is an optimization problem. Furthermore, as safety is a major issue in the airline industry, the planner's decisions are repeatedly checked in the later stages to make sure no constraint is violated. Altogether, a lot of manpower is involved in a process that seems to be automatable.

We attribute the current lack of automation to two reasons. First, the data needed for automated planning are just becoming gradually available and data quality still needs to improve. In the past, booking data only contained the total shipment weight and volume, not individual items. Today, the shipper can provide item dimensions, but sometimes these might not be defined at the time of booking or they change afterwards. To improve data quality, cargo airlines start using item scanners at their terminals to update the shipment data as soon as they get hold of it. Throughout this work, we therefore assume that these data challenges can be solved in the future.

The second reason is the lack of suitable planning models. Only parts of the process like the weight and balance stage have recently been subject to research, see Lurkin and Schyns (2015). The palletization stage is at its core a classical optimization problem, the multidimensional Knapsack Problem. First articles are examining it in an air cargo context, see Paquay et al. (2016), but the full set of its constraints has not been studied in the literature in its entirety. For the first two stages, only loosely related articles can be found.

Besides the automation issues, there are other challenges present at the interfaces of the planning stages. The stages are processed strictly sequential (see Figure 3.2) and each stage has its own constraints and objective. In practice, this approach leads to only local optimization or worse, to infeasible problem states in later stages. Infeasibility typically leads to offloads, i.e., some of the shipments have to be rebooked on another flight causing lost revenue or even penalties. In the following, we describe a few situations that can be found in practice if the stages are not properly coordinated.

First, the build-up planners often have to schedule a ULD build-up several hours before the flight due to a limited workstation capacity later. But, at the time of build-up no shipments to pack might be physically available. Thus, the ULD cannot be built or is delayed and might cause a shortage of handling capacity later on.

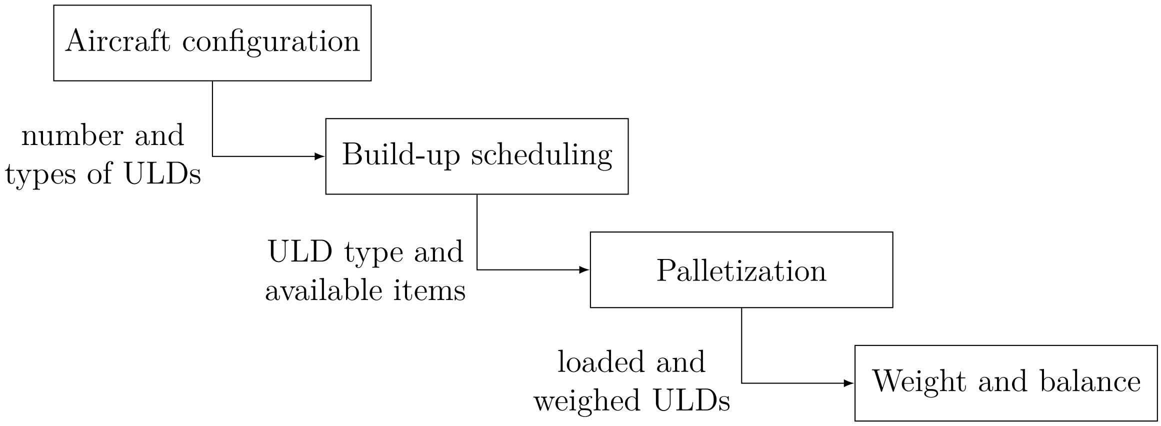


Figure 3.2: Sequential workflow of load planning in practice

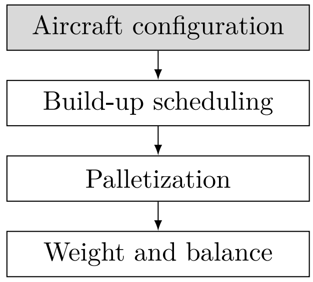
Another issue in practice is that the palletization workers only see the shipments that are present at their workstation. They continuously pack the items onto the ULDs while further shipments are released by the build-up planners and arrive at the workstation. If a large item arrives late, the worker might not find a suitable empty space for it although the total volume and weight capacity of the ULD is not fully utilized. So, the large item must be rebooked.

In the weight and balance stage, the weights of all the ULDs are already fixed. They are a result of the palletization. As the palletization workers try to fully utilize the ULDs, they typically build some heavy ULDs first and lighter ones later. This might result in an unfavorable weight distribution between the ULDs. So, the weight and balance planner might only find suboptimal solutions that consume more fuel or need more ULD reloading operations at the stop-over airports than necessary. Both results in higher cost.

3.5 Summary

In this chapter, we provided an overview of the load planning problems faced by cargo airlines. We introduced the three stakeholders: sales, handling, and aircraft operations, and their objectives: maximizing revenue, minimizing physical handling effort, and minimizing operational cost of the flight. Furthermore, we discussed the challenges of the currently implemented planning approaches. The identified challenges motivate the work of this thesis and lead to our solution approaches presented in Chapters 9 and 11.

Prior to that we define the identified subproblems one by one in the next four chapters, discuss the related literature and present a comprehensive mathematical formulation.



4 Aircraft configuration

The first decision of air cargo load planning we address is what types of ULDs and their respective amounts should be used for each transport segment of a flight. We denote this subproblem of the ACLPP as the *Aircraft Configuration Problem* ACP.

As an introductory example, we consider a cargo flight from airport A to airport C with a stop-over stop at airport B where some cargo is loaded and unloaded. The task is to select a set of ULDs for each transport segment (A-B, A-C, B-C) of the flight, into which the cargo can later be packed. The demand volume and weight for each segment is given as well as the minimum required number of special ULD types like double-sized ULDs for bulky items. The primary restriction is that on each individual flight leg (A-B, B-C) the chosen ULDs must fit into the given aircraft. The objective is to be able to load as much cargo of each segment as possible or, if everything can be loaded, to select the least expensive set of ULDs.

This task becomes challenging when the aircraft can be loaded with different types of ULDs and when multiple types of ULDs could be placed at a certain loading position in the aircraft. Both is the case for most commercial passenger and cargo aircraft. These aircraft typically contain multiple cargo compartments and each compartment can usually accommodate different layouts of ULD types. However, there is normally a preferred configuration which maxes out the aircraft volume and requires the least handling effort.

For our example, we take a McDonnell Douglas MD-11 freighter aircraft (MD11F), whose specification can be found in Boeing (2011). Let us assume that only two ULD types (PMC, PGE) can be loaded in the main deck and two types (PMC, AKE) in the lower deck. Table 4.1 gives exemplary transport demands for a flight with two legs and presents a valid aircraft configuration. Figure 4.1 visualizes the solution, i.e., the utilization of the decks on both flight legs. In our example, we assume PMC ULDs should be used as a default, because it is the standard ULD pallet type with the largest volume capacity. However, there are also some over-sized items requiring the larger PGE ULD type and items that require a closed container, like the AKE ULD.

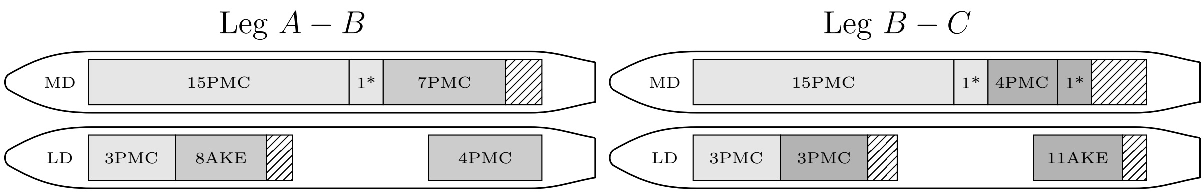


Figure 4.1: Visualization of the aircraft configuration resulting from Table 4.1. Different colors depict the different transport segments. Striped areas represent remaining capacity. The * marks double-sized PGE ULDs.

ULD type	MD		LD fwd		LD aft	
	PMC	PGE	PMC	AKE	PMC	AKE
max volume (m^3)	19	39	15	4	19	4
max weight (t)	6.8	11.3	5.1	1.5	5.1	1.5
#ULDs per segment						
$A \to C$ (380 m^3, 66 t)	16	1	3	0	0	0
$A \to B$ (240 m^3, 42 t)	8	0	0	8	4	0
$B \to C$ (210 m^3, 37 t)	4	1	3	0	0	14
#ULDs on leg						
$A - B$	24	1	3	8	4	0
$B - C$	20	2	6	0	0	14

Table 4.1: Example aircraft configuration for a flight with two legs. The "#ULDs per segment" rows give the number of selected ULDs per compartment. The total demand (volume, weight) is given inside the parentheses. The last two rows give the total number of ULDs flown on each leg.

From a theoretical perspective the ACP combines two nested packing problems. The inner problem is that enough ULDs must be reserved for each segment of the flight to cover the cargo demand. To fully solve it, one would also need to solve the three-dimensional packing problem that is the subject of the palletization stage presented in Chapter 6. From the operational practice of our industrial partner we learned that this level of detail is not necessary to find suitable aircraft configurations. Instead, a simplified one-dimensional consideration, with respect to cargo volume and weight, seems sufficient in practice.

The outer problem is that on each leg the ULDs of all segments using this leg must fit into the aircraft. This problem must be solved in parallel for all legs of a flight as the set of loaded ULDs might change at each stop-over airport. We assume that each compartment's capacity can be described by a set of linear restrictions of the present ULD types.

In the following, we give a few examples of typical compartment combination restrictions. In the simplest form the compartment has a number of capacity units and each ULD type consumes a certain amount of them. On our sample aircraft the main deck has 26 units. Here, a PMC ULD consumes one unit and a PGE ULD two units. So, the restriction can be stated as $2|PGE| + |PMC| \leq 26$. Depending on the valid ULD combinations more than one linear restriction might be needed to express the configuration space. Let us assume in our example that we can choose one of the configurations shown in Figure 4.2 for the lower deck aft compartment (not all ULDs must be present). Note that the configuration space is not convex with respect to the loadable number of ULDs per type. Then, besides our general restrictions, $|AKE| \leq 14$ and $|PMC| \leq 4$, either $|AKE| \leq 4$ and $16 - |AKE| \leq 4|PMC|$ must hold or $14 - |AKE| \leq 4|PMC|$.

In the next section, we discuss the present literature related to the ACP followed by its mathematical formulation in Section 4.2.

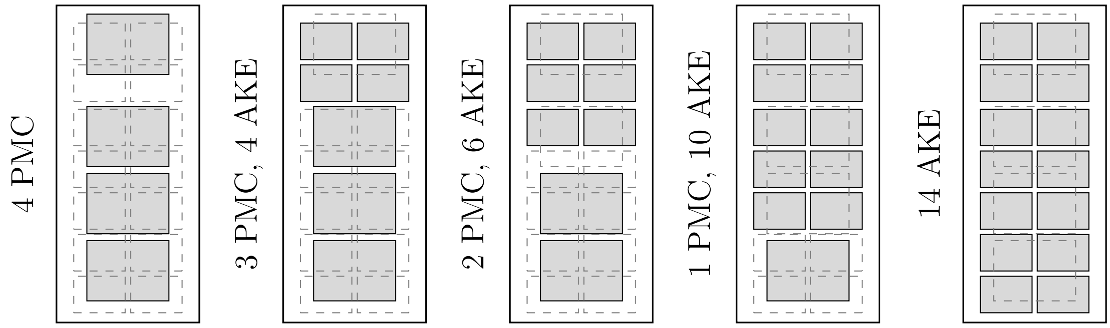


Figure 4.2: Exemplary ULD configurations inside a compartment. Starting on the left with 4 PMC pallets and replacing them by AKE containers until the compartment is filled with 14 AKEs.

4.1 Related literature

To the best of our knowledge there is no work regarding the ACP as introduced in the last section. However, aspects of the ACP have already been studied in the literature.

Yan et al. (2006) present a model to decide for a whole flight network which cargo streams should be loaded and combined into which ULD type and compartment. Their objective is to minimize the total ULD handling cost. The main idea is to combine cargo for the same final destinations early such that these single-destination ULDs must not be handled again. The model was formulated as a non-linear mixed integer program and evaluated on the Asia Pacific network of FedEx using IBM's CPLEX solver. All instances contained only two ULD types, one for main deck positions and another for lower deck positions. That means only a single aircraft configuration is considered, since ULD types cannot be exchanged for another. Furthermore, the model applies to single-leg flights only.

Besides, aircraft configuration restrictions are also considered in some papers dealing with aircraft weight and balance. Here, the cargo is already packed onto ULDs of different types and the ULDs have to be assigned to loading positions in the aircraft. If multiple ULD types have to be loaded and more ULDs are present than loading positions exist, then these models might also decide about the configuration. Mongeau and Bes (2003) present a problem with different ULD types and an elaborate description of the constraints limiting the combinations of ULD types for each compartment of an Airbus A340. To model the constraints, they introduce multiple linear combinations of the amounts per ULD type for each compartment and put a limit on these. Limbourg et al. (2012) present a model assigning ULDs to loading positions in the aircraft. They implicitly decide about the configuration by restricting the usage of partially overlapping positions. The same model is extended by Lurkin and Schyns (2015) for multi-leg flights.

4.2 The Aircraft Configuration Model

In this section, we define a mathematical model of the ACP, which we denote as the Aircraft Configuration Model (ACM) in the following. The goal of the ACP is to find a set of ULDs for each transport segment of a flight that minimizes the total cost, i.e., the sum of the cost of all selected ULDs plus the penalties for offloaded cargo.

We start by defining the input parameters and decision variables followed by the problem's constraints and objective. At last, we present a short overview of the model.

4.2.1 Input parameters

The ACM considers four sets of entities. Let L denote the set of flight legs, K the set of transport segments, V the set of ULD types, and F the set of sets of ULD combination restrictions of the aircraft. We use the respective lower case indices to represent single entities: $f \in G$ for each $G \in F$, $k \in K$, $l \in L$, $v \in V$.

Flight parameters

- $\mathrm{on}_k \subseteq L$: set of flight legs of transport segment k
- $\mathrm{start}_l \subseteq K$: set of transport segments starting with flight leg l

Aircraft parameters

- $\mathrm{limit}_f \in \mathbb{R}$: total ULD limit in combination restriction f
- $\mathrm{m}_{f,v} \in \mathbb{R}$: factor of ULD type v in combination restriction f

ULD parameters

- $\mathrm{avail}_{l,v} \in \mathbb{N}$: available units of ULD type v at departure airport of leg l
- $\mathrm{cap}_v^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$: volume capacity per ULD of type v
- $\mathrm{cap}_v^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$: weight capacity per ULD of type v
- $\mathrm{cost}_v \in \mathbb{R}$: handling cost per ULD of type v (preparation, loading, unloading)

Demand parameters

- $\mathrm{dem}_k^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$: total volume of shipments on segment k
- $\mathrm{dem}_k^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$: total weight of shipments on segment k
- $\mathrm{min}_{k,v} \in \mathbb{N}$: minimum required number of ULDs of type v on segment k
- $\mathrm{offload}_k^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$: penalty for offloaded shipment volume on segment k
- $\mathrm{offload}_k^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$: penalty for offloaded shipment weight on segment k

4.2.2 Decision variables

The primary decision made by the model is the number of selected ULDs for each transport segment. If not all cargo is loaded, we also need to decide the offloaded weight and volume.

- $x_{k,v} \in \mathbb{N}$: number of selected ULDs on segment k of type v
- $y_k^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$: offloaded volume on segment k
- $y_k^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$: offloaded weight on segment k

4.2.3 Constraints

In the following, we introduce the constraints that define a valid aircraft configuration. We note that a solution to this subproblem alone does not guarantee a feasible ULD loading since only a simplification of the three-dimensional packing constraints is considered. Further constraints to guarantee the feasibility will be added in Chapter 6.

Capacity reservation For each transport segment k the capacity of the selected ULDs must be large enough to accommodate the transported shipments by volume and weight. The transported amount is the difference of the demand and the offloaded shipments:

$$\sum_{v \in V} x_{k,v} \mathrm{cap}_v^{\mathrm{wgt}} \geq \mathrm{dem}_k^{\mathrm{wgt}} - y_k^{\mathrm{wgt}} \qquad k \in K \tag{4.1}$$

$$\sum_{v \in V} x_{k,v} \mathrm{cap}_v^{\mathrm{vol}} \geq \mathrm{dem}_k^{\mathrm{vol}} - y_k^{\mathrm{vol}} \qquad k \in K \tag{4.2}$$

Combination restrictions On each flight leg l the ULDs of all present segments must add up to a valid aircraft configuration. We model the valid configurations implicitly by the allowed ULD type combinations. The given restrictions F are organized in groups. Similar to a logic formula given in conjunctive normal form, for all groups $G \in F$ at least one restriction $f \in G$ must hold. Thereby, alternative non-convex combinations can be modeled. Each restriction f represents a valid linear combination of ULDs, which can be loaded into the aircraft. Each ULD type v is weighted by a parameter $\mathrm{m}_{f,v}$ and the sum is limited by parameter limit_f. Furthermore, we introduce an auxiliary variable $b_{l,f}$ indicating if restriction f is fulfilled on leg l and make sure that at least one $b_{l,f}$ equals 1 for each group:

$$\sum_{k \in K : l \in \mathrm{on}_k} \sum_{v \in V} \mathrm{m}_{f,v} x_{k,v} \leq \mathrm{limit}_f + (1 - b_{l,f}) M \qquad l \in L; G \in F; f \in G \tag{4.3}$$

$$\sum_{f \in G} b_{l,f} \geq 1 \qquad l \in L; G \in F \tag{4.4}$$

Required ULD types For each segment some ULDs of certain types might be required to load special cargo. In practice this requirement arises when very large, heavy or fragile items are among the shipments. Large items might require a larger ULD type, heavy items a ULD type with a reinforced floor area, or fragile items should only be loaded into a closed ULD container. Another source of required ULD types is that in practice cargo might already be loaded on a ULD and this ULD should be used for further build-up.

All these requirements can be precalculated and the number of the respective ULD types limited accordingly:

$$\mathrm{min}_{k,v} \leq x_{k,v} \qquad k \in K; v \in V \tag{4.5}$$

Airport ULD type limits The airline might not have an unlimited supply of all ULD types at all departure airports. Therefore, we need to respect the availability limits for

ULDs that should be build-up at an airport. This restriction applies only to transport segments whose first leg starts at the given airport:

$$\sum_{k \in K: k \in \text{start}_l} x_{k,v} \leq \text{avail}_{l,v} \qquad l \in L; v \in V \tag{4.6}$$

4.2.4 Objective

While satisfying all the given constraints, we want to load as much cargo as possible and choose the least expensive set of ULDs for it. The parameter cost_v for ULD type v represents the fixed handling cost for build-up and break-down, the transport to and from the aircraft, the loading into and out of the aircraft, as well as the wear and tear of the ULD itself. The penalty for offloaded volume $\text{offload}_k^{\text{vol}}$ and weight $\text{offload}_k^{\text{wgt}}$ comprises the lost revenues and the impact on customer satisfaction for shipments on segment k. The penalties can only be estimated, but reasonable values can be chosen based on the segment's travel distance or the effective delay of offloaded shipments based on the frequency at which the airline serves this route.

Thus, the objective is to minimize the total cost:

$$\text{minimize} \sum_{k \in K} \left(y_k^{\text{wgt}} \text{offload}_k^{\text{wgt}} + y_k^{\text{vol}} \text{offload}_k^{\text{vol}} + \sum_{v \in V} x_{k,v} \text{cost}_v \right) \tag{4.7}$$

Note that this formulation can tackle both bin packing and knapsack style problems. If there is little demand the number of ULDs is reduced. If the demand exceeds the aircraft capacity the least important cargo is offloaded.

4.2.5 Model overview

After introducing all components of the ACM in the last sections, we now give a consolidated view of the full model:

$$\text{minimize} \sum_{k \in K} \left(\sum_{v \in V} x_{k,v} \text{cost}_v + y_k^{\text{wgt}} \text{offload}_k^{\text{wgt}} + y_k^{\text{vol}} \text{offload}_k^{\text{vol}} \right) \tag{4.8}$$

subject to

$$\text{dem}_k^{\text{wgt}} \leq \sum_{v \in V} x_{k,v} \text{cap}_v^{\text{wgt}} + y_k^{\text{wgt}} \qquad k \in K \tag{4.9}$$

$$\text{dem}_k^{\text{vol}} \leq \sum_{v \in V} x_{k,v} \text{cap}_v^{\text{vol}} + y_k^{\text{vol}} \qquad k \in K \tag{4.10}$$

$$\text{limit}_f \geq \sum_{k \in K: l \in \text{on}_k} \sum_{v \in V} \text{m}_{f,v} x_{k,v} + (b_{l,f} - 1)M \qquad l \in L; f \in F \tag{4.11}$$

$$1 \leq \sum_{f \in G} b_{l,f} \qquad l \in L; G \in F \tag{4.12}$$

$$\text{min}_{k,v} \leq x_{k,v} \qquad k \in K; v \in V \tag{4.13}$$

$$\text{avail}_{l,v} \geq \sum_{k \in K: k \in \text{start}_l} x_{k,v} \qquad l \in L; v \in V \qquad (4.14)$$

$$x_{k,v} \in \mathbb{N} \qquad k \in K; v \in V \qquad (4.15)$$

$$y_k^{\text{wgt}}, y_k^{\text{vol}} \in \mathbb{R}_{\geq 0} \qquad k \in K \qquad (4.16)$$

$$b_{l,f} \in \{0,1\} \qquad l \in L; G \in F \qquad (4.17)$$

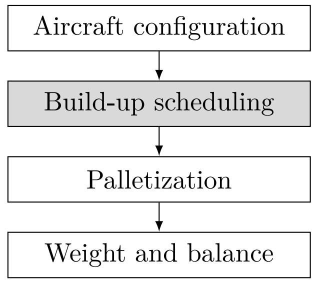
The objective, given in Equation (4.8), is to minimize the total cost from the used ULDs and offloaded cargo. Equation (9.3) and (9.4) model the capacity reservation, Equations (9.5) and (9.6) the ULD combination restrictions, Equation (9.7) the minimum required ULDs, and Equation (9.8) the airport ULD type limit.

4.3 Summary

In this chapter, we introduced the Aircraft Configuration Problem (ACP) in detail, which is to decide what ULD types and amounts should be selected for each transport segment. Furthermore, we analyzed the related literature and presented a formal model, denoted as Aircraft Configuration Model (ACM), of its features.

The literature containing aspects of aircraft configuration from a load planning perspective is very scarce. In most works on aircraft loading, only a single configuration is considered or only one ULD type can be placed on each loading position. In both cases the problem becomes much easier, but significant parts of the configuration space of a typical cargo aircraft are not covered. Aspects of aircraft configuration have also been present in papers dealing with weight and balance. Here, the ULDs are already built and weighed. Therefore, only one part of the problem remains, which is to combine different ULD types into the compartments.

The presented model contains two nested packing problems. The inner is a vector packing problem, with respect to weight and volume, that assigns demand to ULDs. The outer matches the ULDs to aircraft compartments and fulfills aircraft specific loading restrictions. The overall objective is to minimize the costs from offloaded demand and used ULDs.



5 Build-up scheduling

The second step of air cargo load planning we discuss is to decide when to actually pack the cargo into ULDs. We denote this subproblem of the ACLPP as the BSP.

Throughout the day an air cargo terminal handles many outgoing flights. All these flights compete for the same scarce build-up resources, namely workstations and workers. Therefore, some planning is needed to build-up all ULDs on time. As speed is a major sales argument for air cargo, shipment itineraries often only have a short transfer time between their incoming and outgoing flight. So, part of the cargo only becomes physically available for the build-up shortly before its outgoing flight. Therefore, we need to take the cargo arrivals into account to decide when to pack cargo into ULDs.

As shown in Figure 2.5, cargo arrives at the terminal in two ways. Either it is dropped at the export point by the shipper or his forwarder, or it arrives on an inbound flight. On both ways the arrival times are usually predictable. If a time window management system is used at the export point, a rough handover time is known. Similarly, the arrival time of an inbound flight is known and even unlikely to change during the last hours when it is already on its way.

On the other end, the built ULDs need to be transported to the aircraft on time. At major airports, where the distance between terminal and aircraft can be quite long, this ready-for-delivery due date might be an hour or more ahead of departure. Within the window of item arrivals and ULD due dates the ULDs have to be built.

For economic reasons, there is usually only a limited capacity of workstations and workers available at the terminal. Both should be used evenly over time to limit stress and provide flexibility. During outbound rush hours the build-up capacity is usually too low to handle all outgoing ULDs. For example Figure 5.1 sketches the total outbound capacity of the Lufthansa Cargo Hub in Frankfurt during one day. There is a peak at noon where more than 15 percent of the day's capacity departs within one hour. Under the assumption of a constant build-up rate, the workload of the peak hour alone takes nearly four hours.

Accordingly, some ULDs have to be built earlier than their due date. But, packing ULDs before all shipments arrived leaves fewer options in combining items. In a previous study we have shown that the best possible bin utilization reduces if a part of the items arrives after the first bins are built (Brandt et al., 2014). The study considered one-dimensional packing only. We measured a loss of utilization if less than 70 percent of the remaining cargo volume was available on average when a bin is built. For each 10 percent less available cargo the best possible utilization reduced by around 1 percent. To give an example, we would expect a loss of 3 percent in utilization if on average only 40 percent of the cargo is available at each build-up. We note that the study only considered one-dimensional packing. The effects for three-dimensional packing will presumably be larger.

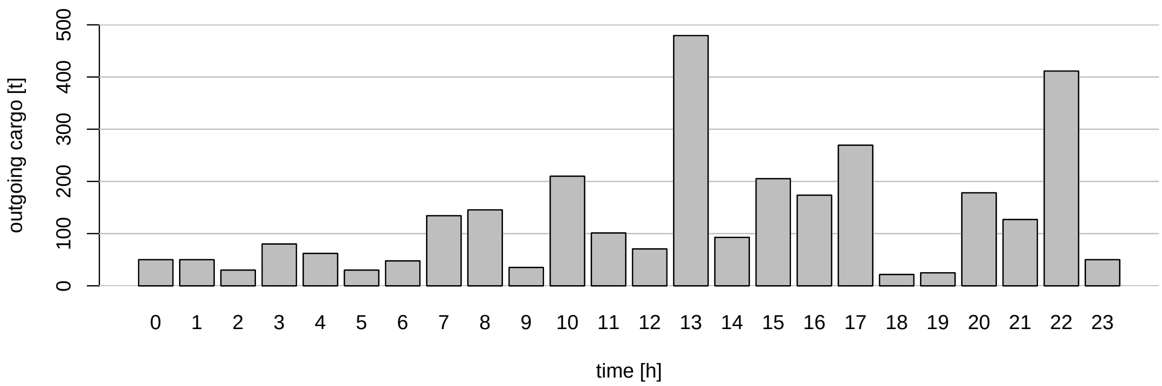


Figure 5.1: Hourly cargo capacity of Lufthansa Cargo out of Frankfurt on 21.10.2016

So, one scheduling objective is to minimize the earliness of the build-up to be able to pack as much cargo as possible. The later a ULD is planned for build-up the more shipments are available and more item combinations are possible. Further objectives are to use as few workstations as possible or to reduce the peak worker demand. We note that the exact build-up schedule is typically determined only a few hours before the flights. Therefore, the number of available workers and workstations is essentially fixed. But, using less workstations or workers can provide flexibility throughout the day to react to delays or other disruptions.

The BSP requirements comprise scheduling and organizational rules. Building a ULD requires a compatible workstation and a crew of workers. Each workstation and crew can only process one ULD at a time. Each ULD has to be build-up before its due date or must not be scheduled. We expect that a suitable number of workstations and workers was predetermined by tactical planning out of scope of this work.

From a theoretical perspective the BSP is a Multi-Resource Job Shop Scheduling Problem with the objective of minimizing the total job earliness.

In the next section, we discuss literature related to the BSP followed by a mathematical formulation of the problem in Section 5.2.

5.1 Related literature

To the best of our knowledge there is no work regarding the BSP as introduced in the last section. The closest publications cover the shift design and workforce planning for build-up operations.

Yan et al. (2006b) develop two MIP models to select shift patterns and determine the amount of workers for a weekly planning horizon. In their first model the workforce demand must always be fulfilled. The second model allows to use temporary workers at higher costs and a reward for planned excess workforce, because it provides flexibility. In a later work the same authors extend both models by stochastic aspects (Yan et al., 2008a). Both previous works are also published as Yan et al. (2006a) and Yan et al. (2008b) with negligible changes.

Nobert and Roy (1998) present a MIP model to select shift patterns and determine the amount of workers for a typical day. Their goal is to minimize the total shift cost. The total handling demand is given as a matrix of cargo arrivals and their corresponding departures for each period. Rong and Grunow (2009) present a similar model but also schedule the amount of cargo of each individual flight in the break-down and build-up for each period.

Both models do not consider individual ULDs or workstations. Instead they only limit the total handling capacity across all flights for each time slot. While this seems fine to estimate the total workforce demand, it is not sufficient to plan ULD build-ups. In our case both approaches might lead to undesired solutions where many ULDs are partially built in each period. In reality, this would lead to many started ULDs blocking the workstations for several periods before they can be finished. Furthermore, both papers do not aim at a particular early or late build-up.

For further reading, we refer to a recent review of publications concerning air cargo terminal operations which is given by Feng et al. (2015).

5.2 The Build-up Scheduling Model

In this section, we define a mathematical model of the BSP, which we denote as the Build-up Scheduling Model (BSM) in the following. The goal of the model is to find an assignment of the planned ULDs to build-up periods, while respecting workstation and worker capacity restrictions and performing build-ups as late as possible. We expect that sufficient workforce is available to handle the total cargo demand throughout the day. This can be ensured by applying the same model earlier during tactical planning with the expected upper bound of demand. The basis for our model is the model given by Rong and Grunow (2009). Instead of scheduling a certain amount of demand in each period, we schedule the ULDs and their respective workload explicitly.

We start by defining the problem parameters. In the following sections we introduce the problem's constraints and the objective. At last, we present a short overview of the model.

5.2.1 Input parameters

The BSM considers four sets of entities. Let K denote the set of transport segments for which ULDs have to be build, V the set of ULD types, and S the set of shifts of the build-up crews. Furthermore, let T denote the periods of the planning horizon. In the following we use the respective lower case indices to represent single elements: $k \in K$, $s \in S$, $t \in T$, $v \in V$.

Segment parameters

- $\text{avail}_{k,t}^{\text{vol}} \in \mathbb{R}_{\geq 0}$: cargo volume of segment k arriving up to period t
- $\text{dem}_{k}^{\text{vol}} \in \mathbb{R}_{\geq 0}$: total cargo volume on segment k

- $\text{due}_k \in T$: build-up due date of segment k
- $\text{loss}_k^{\text{cost}} \in \mathbb{R}_{\geq 0}$: expected cost if a unit of the volume on segment k cannot be loaded
- $\text{split}_k \in \mathbb{N}$: maximum number of ULDs to build simultaneously for segment k
- $\text{uld}_{k,v} \in \mathbb{N}$: number of ULDs of type v to schedule for segment k

ULD parameters

- $\text{dur}_v \in \mathbb{N}$: number of periods to build a ULD of type v
- $\text{cap}_v^{\text{vol}} \in \mathbb{R}_{\geq 0}$: volume capacity per ULD of type v

Capacity parameters

- $\text{cost}_s \in \mathbb{R}_{\geq 0}$: cost of one build-up crew in shift s
- $\text{end}_s \in T$: last period after shift s
- $\text{max}_s \in \mathbb{N}$: maximum available build-up crews in shift s
- $\text{open}_{v,t} \in \mathbb{N}$: number of workstations that can handle ULD type v during period t
- $\text{start}_s \in T$: first period of shift s

5.2.2 Decision variables

The primary decision made by the model is the number of ULDs to start building at each period. To calculate the objective value, we introduce further decision variables. One part of the objective is to load as much as possible. There are two reasons that might prevent us from achieving this goal: First, if not all ULDs can be scheduled for build-up. Second, if a ULD is scheduled before enough cargo is available. We capture the effect of each reason in one variable. Furthermore, we decide about the number of build-up crews to deploy during each shift.

- $\mathit{start}_{k,v,t} \in \mathbb{N}$: number of ULDs of type v started to build at period t for segment k ($t \leq \text{due}_k - \text{dur}_v$)
- $\mathit{offload}_{k,v} \in \mathbb{N}$: number of not scheduled ULDs of type v on segment k
- $\mathit{dead}_{k,t} \in \mathbb{R}_{\geq 0}$: lost volume capacity on segment k if ULDs are built before enough cargo volume is available
- $\mathit{crews}_s \in \mathbb{N}$: number of selected build-up crews in shift s

5.2.3 Constraints

In the following we define the constraints that define a feasible build-up schedule.

Schedule ULDs All requested ULDs have to be scheduled to a period before the due date or marked as offloaded.

$$\sum_{\substack{t \in T: \\ t \leq \text{due}_k - \text{dur}_v}} \mathit{start}_{k,v,t} = \text{uld}_{k,v} - \mathit{offload}_{k,v} \qquad k \in K; v \in V \tag{5.1}$$

Unused cargo capacity Only physically available cargo can be packed onto a ULD. If a ULD is scheduled before enough cargo is available, some space in the ULD will be lost. We add this constraint to determine the amount of lost volume $dead_{k,t}$ of scheduled ULDs for segment k during period t.

We introduce an auxiliary variable $pack_{k,t}$ to represent the packed amount of cargo of segment k until period t, and derive it from the previous period $t-1$ as:

$$pack_{k,0} = 0 \qquad k \in K \tag{5.2}$$

$$pack_{k,t} = pack_{k,t-1} + \sum_{v \in V} start_{k,v,t}\mathrm{cap}_v^{\mathrm{vol}} - dead_{k,t} \qquad k \in K; t \in T : 0 < t \tag{5.3}$$

We restrict the packed amount of cargo at a period t by the available cargo volume $\mathrm{avail}_{k,t}^{\mathrm{vol}}$ during that period:

$$pack_{k,t} \leq \mathrm{avail}_{k,t}^{\mathrm{vol}} \qquad k \in K; t \in T \tag{5.4}$$

Parallel build-ups There is no technical reason to restrict the number of ULDs that are built in parallel for a segment. However, in practice there might occur problems during a build-up when an item cannot be packed onto the planned ULD. In this case, the load planner has to find another ULD with enough remaining capacity under massive time pressure. The more ULDs that are not started yet, the more options and time he has for replanning. Furthermore, if only a few ULDs for a segment are built in parallel, they can be assigned to spatially close workstations. This might reduce transport distances and improve overview during build-up. We note that, during a period t, the ULDs that were started during period $(t - \mathrm{dur}_v + 1)$ to period t are still work in progress. For each segment k we limit the number of simultaneously built ULDs by split_k as:

$$\sum_{v \in V} \sum_{\substack{t' \in T: \\ t - \mathrm{dur}_v < t' \leq t}} start_{k,v,t'} \leq \mathrm{split}_k \qquad k \in K; t \in T \tag{5.5}$$

Workstation availability The number of usable workstations is limited. As each workstation can process one ULD at a time, we limit the number of simultaneously processed ULDs of type v during period t by $\mathrm{open}_{v,t}$. To calculate this number for a point in time, we sum up the ULDs of all segments that have been started during earlier periods and are not finished yet:

$$\sum_{k \in K} \sum_{\substack{t' \in T: \\ t - \mathrm{dur}_v < t' \leq t}} start_{k,v,t'} \leq \mathrm{open}_{v,t} \qquad v \in V; t \in T \tag{5.6}$$

Workforce capacity Similar to the workstation limit, we limit the total number of all ULDs that can simultaneously be processed by the available build-up crews during each period.

$$\sum_{k \in K} \sum_{v \in V} \sum_{\substack{t' \in T: \\ t - \mathrm{dur}_v < t' \leq t}} start_{k,v,t'} \leq \sum_{\substack{s \in S: \\ \mathrm{start}_s \leq t < \mathrm{end}_s}} crews_s \qquad t \in T \tag{5.7}$$

5.2.4 Objective

While satisfying all the given constraints, we want to load as much cargo as possible and release not required build-up crews for short-term dispatch. To maximize the load, we minimize the cost of lost capacity. We sum up the lost capacity of ULDs that are not scheduled $\mathit{offload}_{k,v}$ and where not enough cargo is available $\mathit{dead}_{k,t}$. At last, we add the cost of the selected crews crews_s. In short-term planning this is probably not the real cost but the expected profit of having spare crews.

$$\text{minimize} \sum_{k \in K} \text{loss}_k^{\text{cost}} \left(\sum_{t \in T} \mathit{dead}_{k,t} + \sum_{v \in V} \text{cap}_v^{\text{vol}} \mathit{offload}_{k,v} \right) + \sum_{s \in S} \text{cost}_s \mathit{crews}_s \tag{5.8}$$

5.2.5 Model overview

After introducing all components of the BSM in the last sections, we now give a consolidated view of the full model:

$$\text{minimize} \sum_{k \in K} \text{offload}_k^{\text{vol}} \left(\sum_{t \in T} \mathit{dead}_{k,t} + \sum_{v \in V} \text{cap}_v^{\text{vol}} \mathit{offload}_{k,v} \right) + \sum_{s \in S} \text{cost}_s \mathit{crews}_s \tag{5.9}$$

subject to

$$\text{uld}_{k,v} = \mathit{offload}_{k,v} + \sum_{\substack{t \in T: \\ t \leq \text{due}_k - \text{dur}_v}} \mathit{start}_{k,v,t} \qquad k \in K; v \in V \tag{5.10}$$

$$\mathit{pack}_{k,0} = \text{pack}_k \tag{5.11}$$

$$\mathit{pack}_{k,t} + \mathit{dead}_{k,t} = \mathit{pack}_{k,t-1} + \sum_{v \in V} \mathit{start}_{k,v,t} \text{cap}_v^{\text{vol}} \qquad k \in K; t \in T : 0 < t \tag{5.12}$$

$$\text{avail}_{k,t} \geq \mathit{pack}_{k,t} \qquad k \in K; t \in T \tag{5.13}$$

$$\text{split}_k \geq \sum_{v \in V} \sum_{\substack{t' \in T: \\ t - \text{dur}_v < t' \leq t}} \mathit{start}_{k,v,t'} \qquad k \in K; t \in T \tag{5.14}$$

$$\text{open}_{v,t} \geq \sum_{k \in K} \sum_{\substack{t' \in T: \\ t - \text{dur}_v < t' \leq t}} \mathit{start}_{k,v,t'} \qquad v \in V; t \in T \tag{5.15}$$

$$\sum_{\substack{s \in S: \\ \text{start}_s \leq t < \text{end}_s}} \mathit{crews}_s \geq \sum_{k \in K} \sum_{v \in V} \sum_{\substack{t' \in T: \\ t - \text{dur}_v < t' \leq t}} \mathit{start}_{k,v,t} \qquad t \in T \tag{5.16}$$

$$\mathit{start}_{k,v,t} \in \mathbb{N} \qquad k \in K; v \in V; t \in T \tag{5.17}$$

$$\mathit{dead}_{k,t} \in \mathbb{R}_{\geq 0} \qquad k \in K; t \in T \tag{5.18}$$

$$\mathit{offload}_{k,v} \in \mathbb{N} \qquad k \in K; v \in V \tag{5.19}$$

$$\mathit{pack}_{k,t} \in \mathbb{R}_{\geq 0} \qquad k \in K; t \in T \tag{5.20}$$

$$\mathit{crews}_s \in \mathbb{N} \qquad s \in S \tag{5.21}$$

The objective, given in Equation (5.9), is to minimize the sum of the cost of the lost profit due to early build-ups or unscheduled ULDs and the cost of the deployed crews.

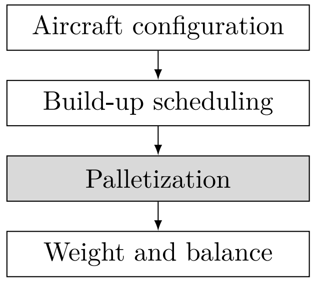
Equation (5.10) models the ULD scheduling, Equations (5.11) to (5.13) the unused cargo capacity, Equation (9.20) the parallel build-ups, Equation (9.21) the workstation availability, and Equation (9.22) the workforce capacity.

5.3 Summary

In this chapter, we introduced the Build-up Scheduling Problem (BSP) in detail, which is to find a feasible and cost-efficient schedule for the assembly of the ULDs of all outgoing flights in a given time horizon. Furthermore, we analyzed the related literature and presented a formal model, denoted as Build-up Scheduling Model (BSM), of its features.

Build-up scheduling itself has not been dealt with in the literature. The closest publications cover the shift design and workforce planning for build-up operations in air cargo terminals where no individual ULDs are considered.

The presented model comprises a typical job-shop scheduling, with the extension that up to any point in time no more cargo of a flight can be built than is physically available at the terminal. The overall objective is to maximize the amount of ULD capacity scheduled on time, i.e., before the respective flight's departure time, with as few build-up crews as possible.



6 Air cargo palletization

The most prominent part of air cargo load planning is to decide how the items are arranged inside the ULDs. Technically the task to solve is a three-dimensional container loading problem with a multitude of constraints stemming from physical, regulatory, and operational requirements.

The objective is to load as much cargo as possible while keeping the necessary physical handling effort low. In practice the palletization determines the handling efforts for both build-up and the corresponding break-down at the destination. During build-up complex three-dimensional arrangement patterns increase the effort. At the break-down additional effort results from the consolidation of shipments that have been split across multiple ULDs or the separation of shipments on one ULD but with diverging ongoing processes.

Physical requirements contain typical volume constraints: items have to be fully inside the ULD contours and are not allowed to intersect. Furthermore, items might only be placed in certain orientations. Finally, if items need to be stacked on top of each other the load-bearing strength of the lower item and a stable placement of the upper item have to be satisfied.

Regulatory requirements comprise rules for the handling of dangerous goods (DGR) or for clearing customs. DGR rules as defined by IATA (2016) typically state that certain types of substances must not be placed on the same ULD or within proximity, for example, chemicals that easily react with each other. Loading positions for items might be restricted. For instance, only low dense cargo is allowed right behind the cockpit or radio-active materials must be placed on the floor where they are farthest away from any passenger. Furthermore, the total net weight of certain substances, such as explosives or dry ice, inside all the loaded items might be limited per compartment for safety reasons. Finally, in some countries customs regulations require that all items of a shipment must arrive with the same flight, effectively creating an all or nothing rule for the airline.

Operational requirements mostly align the palletization result with the neighboring process steps. Items need to be assigned to a ULD that is scheduled for build-up after the item arrives at the terminal. Furthermore, the weight limits of the ULD type and the weight distribution limits inside each ULD must be respected, to allow easy and error free aircraft loading.

We define the APP as the subproblem of the ACLPP to decide about the assignment of items to ULDs and the positioning of the items therein. In our APP model, each flight and segment is considered independently since all interdependencies with respect to time and ULD types are considered in the previous subproblems. Afterwards, the built ULDs of all segments of a flight can be combined for upcoming planning steps. This means all items put into a ULD will join and leave the flight at the same airport. Thereby, any multi-drop restriction is postponed to the weight and balance step, which we present in Chapter 7. The input parameters of the APP are the item and ULD attributes as well as a set of palletization rules. We refrain from modeling logically similar domain specific rules, like net weights or sorting constraints, individually. Instead we present them in an abstract format that can be instantiated as needed.

In the next section, we discuss literature related to the APP followed by a mathematical formulation of the problem in Section 6.2.

6.1 Related literature

There is a vast amount of literature regarding container loading problems. Bortfeldt and Wäscher (2013) present a recent review of more than 160 papers. They categorize the works by their objective, heterogeneity of the considered objects, and constraints. Pollaris et al. (2015) update this review and focus on vehicle routing problems. Thereby, they explicitly highlight papers with an aircraft loading context.

According to these reviews, the typical objectives of container loading problems are either to minimize the used number of bins, referred to as input minimization, or maximizing the utilization of a given set of bins, referred to as output maximization. The problem considered in this work combines both objectives. If all items can be loaded, the number of bins should be small to save handling operations. But, if the aircraft capacity is exceeded, all bins will be used and as many items as possible should be loaded.

Furthermore, Bortfeldt and Wäscher (2013) characterize the problems by the level of heterogeneity of the items and bins. In this respect, the APP falls into the category of strongly heterogeneous items with heterogeneous bins, classified as Multiple Bin-Size Bin Packing Problem (MBSBPP) and Multiple Heterogeneous Knapsack Problem (MHKP). According to an earlier classification by Hodgson (1982) the APP represents a classical distributors packing problem.

In Bischoff and Ratcliff (1995) the authors describe a set of twelve practical requirements of packing problems: individual allowed item orientations, maximum load-bearing strength of each item, item positioning inside/outside a certain area, physical stability when stacking items, grouping items into the same bin, multi-drop accessibility, item separation, complete shipment, item priorities, complexity of loading arrangements, maximum bin weight, and weight distribution inside each bin. Except for multi-drop restrictions all the requirements are present in the APP. A ULD with multiple drops is usually not feasible. In general, ULDs cannot be opened or altered inside the aircraft and the effort of temporary unloading a ULD from the aircraft is quite high. Multi-drop constraints between different ULDs in the aircraft are considered during weight and balance in Chapter 7. There we assign suitable loading positions and hence decide about the loading order.

Bortfeldt and Wäscher (2013) pick up the classification presented above and correlate the recent literature to it. They state that there has been only little research that simultaneously deals with multiple types of practical constraints. However, there are a few papers on general bin packing problems that cover at least some of the constraints present in our problem. Techanitisawad and Tangwiwatwong (2004) and Lin et al. (2006) present two-step solutions: first they distribute items across the bins by solving a MIP model

First author	Year	Orientation	Grouping	Stability	Weight limits	Weight distr.	Stacking	Availability	Objective
Techanitisawad	2004	X	-	X	-	X	X	-	1. min cost, 2. max load
Chan	2006	X	X	X	X	X	-	-	1. min cost, 2. max load
Lin	2006	X	-	X	X	-	X	-	min cost - load + effort
Ceschia	2011	X	-	X	X	-	X	-	min cost - load - free space
Paquay	2014	X	-	X	X	X	X	-	min unused ULD volume
Hong Ha	2016	-	-	-	X	-	-	X	min unused ULD volume

Table 6.1: Recent literature on feature-rich container loading problems addressing multiple aspects present in the APP. The columns show the considered features and objective. Technical constraints like non-overlapping and placement inside container, which are present in all the papers, are not shown. We also omit those of the constraints mentioned by Bischoff and Ratcliff (1995) that are not considered by any of the papers above, e.g., positioning, separation, or complexity constraints.

and then use heuristics to pack each bin with respect to the covered constraints. Ceschia and Schaerf (2013) present a local search procedure and a fast wall-building heuristic to evaluate the updated bins.

The literature regarding feature-rich problems in an air cargo context is even more scarce. The main distinction here is that the approaches must be able to deal with irregular shaped containers. Chan et al. (2006) present a two-stage approach. In the first phase, they solve a MIP model to select a promising set of ULDs. In the second step, they apply a heuristic that loads the items in descending priority while keeping track of empty spaces. If no more items can be loaded, the procedure is repeated for the remaining items. The approach considers only five of the twelve aspects introduced by Bischoff and Ratcliff (1995), namely: orientation, grouping items, physical stability, weight limits, and weight distribution. The procedure is evaluated on instances with up to 671 items of around 50 different formats from a forwarding company. No runtimes are reported but the authors state that the heuristic can place between 20-100 items per minute.

The work of Paquay et al. (2016) introduces a MIP model with a detailed mathematical definition of the considered constraints: orientation, fragility (a binary version of a stacking constraint), physical stability, weight limits, and weight distribution. However, only small instances with up to 12 items can be solved in a reasonable amount of time.

Hong Ha and Nananukul (2016) also present a MIP model for ULD packing from a forwarders perspective. Their model incorporates item release dates and only items of similar dates can be placed on the same ULD. The model is evaluated on one box-shaped ULD type and solved successfully for instances with 15 items. To be able to solve larger instances the authors relax the three-dimensional constraints and instead cap the maximum volume utilization of the ULDs. The three-dimensional packing is then solved for each ULD in a separate postprocessing step. If no feasible solution is found for a ULD, the main MIP model is solved again with a reduced utilization cap.

Some more papers deal with palletization problems from a forwarders view: Wu (2008) presents a MIP model and Lau et al. (2009) a genetic algorithm to select ULDs and assign items to them. However, they do not consider practical constraints. Both papers aim to minimize the rental and transport fees of the built ULDs.

Moreover, there exist papers considering packing problems in air cargo in the military domain like Heidelberg et al. (1998), Guéret et al. (2003), and Nance et al. (2011). However, these problems are fundamentally different because the loaded cargo is homogeneous or is not placed on ULDs such as trucks or large machinery, which are loaded directly.

To sum up, there is a large number of publications regarding container loading and the majority of practical constraints is known. But, all the works are missing important aspects of our problem at hand and only cover parts of the constraints. We give an overview of the mentioned papers, their covered features and objectives in Table 6.1.

6.2 The Air Cargo Palletization Model

In this section, we define a mathematical model of the APP, which we denote as the Air Cargo Palletization Model (APM) in the following. Its goal is to find an assignment and arrangement of the items into ULDs, while respecting the physical, regulatory, and operational constraints. As the majority of items are box-shaped, we only deal with cuboid items here. Accordingly, we model irregular items, cylinders or sacks by their cuboid bounding box with a suitable load-bearing strength.

The basis for our model is the one given by Paquay et al. (2016). Our model contains some key features that have, to the best of our knowledge, not been presented in literature:

- We introduce time as a new dimension of packing problems. Each item has an individual period at which it becomes available for loading. All ULDs have an individual due date at which they will be built. Consequently, only items that are available before the due date of a ULD can be assigned to it.
- Next to the item gross weight, we introduce a set of net weights for each item. These typically represent DGR substances, e.g., dry ice, explosives, or radio-active materials, whose total weight inside certain ULD subsets must be limited.
- We present a formal definition of a load-bearing constraint that allows items to be stacked with partial support and calculates how the weight is transmitted between the items.
- In practice, stacked items do not need a perfectly even surface. It is also acceptable if the items below vary slightly in their height. Therefore we add a tolerance parameter ϵ and allow stacked items to float "in the air" within the tolerance.
- We use a single coordinate system for all ULDs, which allows for an easy integration of item separation and positioning constraints.

- To facilitate fine-grained item sorting, we add soft allocation constraints that can be seen from two perspectives. On the one hand, we want to split a group of items across as few ULDs as possible. On the other hand, we want to load only few different groups of items into a ULD.

We start by defining the input parameters and decision variables followed by the problem's constraints and objective. At last, we present a short overview of the model.

6.2.1 Parameters

The APM primarily contains two sets of entities, items I and ULDs U, and considers a time horizon t. To describe the constraints we introduce another three sets of entities:

- **Rotations** R: We consider only orthogonal item rotations. Each rotation is described by its permutation of item dimensions. Accordingly, there are six possible rotations: $R = \{\text{XYZ}, \text{XZY}, \text{YXZ}, \text{YZX}, \text{ZXY}, \text{ZYX}\}$. XYZ is the original item rotation. For example, rotation ZYX represents a turn by 90 degrees around the y-axis. In this case the item's z-axis corresponds to the bin's x-axis and vice versa.
- **Sorting criteria** Q: During palletization we want to group or separate certain items. For example items of the same shipment should be grouped together, while express items should be kept separate from standard items. Each sorting criterion describes such a constraint to group or separate by. For each criterion $q \in Q$ let Q_q denote the union of groups all the considered items belong to.
- **Weight categories** G: Besides the gross weight of each item we need to consider the net weight of certain substances inside the items. Each considered substance like dry ice or explosives represents a weight category.

In the following we use the respective lower case indices to represent single entities: $g \in G$, $i, j \in I$, $u \in U$, $q \in Q$, $r \in R$, $t \in T$.

Item parameters

- $A \subseteq I \times I$: pairs of items that are not allowed within one ULD ($\langle i, j \rangle \in A$)
- $\text{avail}_i \in T$: first period that item i is physically available for build-up
- $\text{clique}_i \in I$: item that must be loaded iff item i is loaded
- $\text{dim}_i^{\text{x}}, \text{dim}_i^{\text{y}}, \text{dim}_i^{\text{z}} \in \mathbb{R}_{\geq 0}$: dimensions of item i
- $\text{dist}_{i,j} \in \mathbb{R}_{\geq 0}$: minimum required distance between items i and j
- $\text{offload}_i \in \mathbb{R}_{\geq 0}$: penalty when item i is offloaded
- $\text{rot}_i \subseteq R$: allowed rotations of item i
- $\text{sort}_{q,i} \in Q_q$: group of item i for sorting criterion q
- $\text{supp}_i \in [0, 1]$: minimum required supported floor area of item i (ratio)
- $\text{stack}_i^{\text{x}}, \text{stack}_i^{\text{y}}, \text{stack}_i^{\text{z}} \in \mathbb{R}_{\geq 0}$: maximum load-bearing strength of item i
- $U_i \subseteq U$: ULDs that item i can be loaded into
- $\text{weight}_i^{\text{grs}} \in \mathbb{R}_{\geq 0}$: gross weight of item i
- $\text{weight}_{g,i}^{\text{net}} \in \mathbb{R}_{\geq 0}$: net weight of item i for weight category g

- $\underline{\mathrm{x}}_i^{\mathrm{ac}}, \overline{\mathrm{x}}_i^{\mathrm{ac}}, \underline{\mathrm{y}}_i^{\mathrm{ac}}, \overline{\mathrm{y}}_i^{\mathrm{ac}}, \underline{\mathrm{z}}_i^{\mathrm{ac}}, \overline{\mathrm{z}}_i^{\mathrm{ac}} \in \mathbb{R}$: bounding box of the position of item i inside the aircraft
- $\underleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \overleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \underleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \overleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \underleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}}, \overleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}} \in \mathbb{R}$: box of the start position of item i in ULD u
- $\underrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \overrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \underrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \overrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \underrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}}, \overrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}} \in \mathbb{R}$: box of the end position of item i in ULD u

ULD parameters

- C_u: set of contour planes of ULD u in Hesse normal form ($\langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u$)
- $\mathrm{limit}_u \in \mathbb{R}_{\geq 0}$: gross weight limit of ULD u
- $\mathrm{start}_u \in T$: Period the ULD u is scheduled for build-up
- $\mathrm{start}_u^{\mathrm{x}}, \mathrm{start}_u^{\mathrm{y}}, \mathrm{start}_u^{\mathrm{z}} \in \mathbb{R}$: starting coordinate of ULD u
- $\mathrm{udim}_u^{\mathrm{z}}, \mathrm{udim}_u^{\mathrm{y}}, \mathrm{udim}_u^{\mathrm{x}} \in \mathbb{R}_{\geq 0}$: dimensions of ULD u
- $\underline{\mathrm{x}}_u^{\mathrm{cg}}, \overline{\mathrm{x}}_u^{\mathrm{cg}}, \underline{\mathrm{y}}_u^{\mathrm{cg}}, \overline{\mathrm{y}}_u^{\mathrm{cg}}, \underline{\mathrm{z}}_u^{\mathrm{cg}}, \overline{\mathrm{z}}_u^{\mathrm{cg}} \in \mathbb{R}$: box of the valid CG of ULD u

Sorting parameters

- $\mathrm{penalty}_q \in \mathbb{R}_{\geq 0}$: linear penalty for sorting errors of criterion q
- $\mathrm{pow}_q \in \mathbb{N}$: exponential penalty for sorting errors of criterion q
- $Q^{\mathrm{item}} \subseteq Q$: sorting criteria to group items by
- $Q^{\mathrm{uld}} \subseteq Q$: sorting criteria to separate items by

Weight limit parameters

- D_g: set of net weight limits that apply to subsets of ULDs ($d \in D_g$)
- $\mathrm{limit}_d^{\mathrm{net}} \in \mathbb{R}_{\geq 0}$: net weight limit for all items placed in ulds_d for weight limit d
- $\mathrm{ulds}_d \subseteq U$: ULD subset considered for net weight limit d

6.2.2 Decision variables

The primary decision made by the model is where to place each item. This decision effectively also determines all other variables like if the item is loaded at all or if the sorting criteria are met.

- $\mathit{assign}_{u,i} \in \{0, 1\}$: 1 if item i is assigned to ULD u
- $\mathit{assign}_i^{\mathrm{off}} \in \{0, 1\}$: 1 if the item is offloaded
- $n_q \in \mathbb{N}$: total sorting penalty of all items regarding sorting criterion q
- o_i: total supported area of item i
- $\mathit{weight}_u \in \mathbb{R}_{\geq 0}$: total gross weight of all items loaded into ULD u
- $x_i, y_i, z_i \in \mathbb{R}$: coordinates of the front left bottom corner of item i
- $x'_i, y'_i, z'_i \in \mathbb{R}$: coordinates of the rear right top corner of item i

6.2.3 Constraints

In the following we define the constraints that describe a valid packing of items into the ULDs. We note that a solution to this subproblem alone does not guarantee that all planned ULDs can be loaded into the aircraft. Further constraints regarding the aircraft loading are presented in Chapter 7. Along the upcoming constraints further auxiliary variables are defined that are only relevant inside the defining constraint.

ULD assignment Each item i needs either be assigned to a compatible ULD from the set U_i or marked as offloaded:

$$assign_i^{\text{off}} + \sum_{u \in U_i} assign_{u,i} = 1 \qquad i \in I \tag{6.1}$$

Item orientation and size The rotation of each item determines the effective size of the item in each dimension. Therefore, we combine the calculation of the start and end coordinates of each item with the determination of its rotation. To describe the rotation of item i, we use a 3×3 rotation matrix r_i of auxiliary binary decision variables. The following examples show such a matrix representing the original item orientation XYZ and a horizontal rotation around the y-axis ZYX:

$$\text{XYZ} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \text{ZYX} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \tag{6.2}$$

By using the coordinates of the front left bottom corner (x_i, y_i, z_i) of item i, its dimensions $(\text{dim}_i^{\text{x}}, \text{dim}_i^{\text{y}}, \text{dim}_i^{\text{z}})$ and the rotation matrix of the item, we calculate the coordinates of the rear right top corner (x_i', y_i', z_i') of item i as follows:

$$\begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} + \begin{pmatrix} r_i^{11} & r_i^{12} & r_i^{13} \\ r_i^{21} & r_i^{22} & r_i^{23} \\ r_i^{31} & r_i^{32} & r_i^{33} \end{pmatrix} \begin{pmatrix} \text{dim}_i^{\text{x}} \\ \text{dim}_i^{\text{y}} \\ \text{dim}_i^{\text{z}} \end{pmatrix} = \begin{pmatrix} x_i' \\ y_i' \\ z_i' \end{pmatrix} \qquad i \in I \tag{6.3}$$

To give a complete example, let us assume item i has orientation ZYX. Then, the item dimensions along the x- and z-axis are flipped and the rear right top corner is calculated as follows:

$$\begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} + ZYX \begin{pmatrix} \text{dim}_i^{\text{x}} \\ \text{dim}_i^{\text{y}} \\ \text{dim}_i^{\text{z}} \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} + \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \text{dim}_i^{\text{x}} \\ \text{dim}_i^{\text{y}} \\ \text{dim}_i^{\text{z}} \end{pmatrix} = \begin{pmatrix} x_i + \text{dim}_i^{\text{z}} \\ y_i + \text{dim}_i^{\text{y}} \\ z_i + \text{dim}_i^{\text{x}} \end{pmatrix} = \begin{pmatrix} x_i' \\ y_i' \\ z_i' \end{pmatrix} \tag{6.4}$$

The matrix only contains a valid orthogonal rotation if each row and column of the matrix contains exactly one 1. Accordingly, we add the following constraints for each row and column:

$$\sum_{b=1}^{3} r_i^{ab} = 1 \qquad i \in I; a \in \{1, 2, 3\} \tag{6.5}$$

$$\sum_{a=1}^{3} r_i^{ab} = 1 \qquad i \in I; b \in \{1, 2, 3\} \tag{6.6}$$

Finally, we restrict the rotation matrix to rotations that are explicitly allowed for item i:

$$\text{XYZ} \notin \text{rot}_i \Rightarrow r_i^{11} + r_i^{22} + r_i^{33} \leq 1 \qquad i \in I \tag{6.7}$$

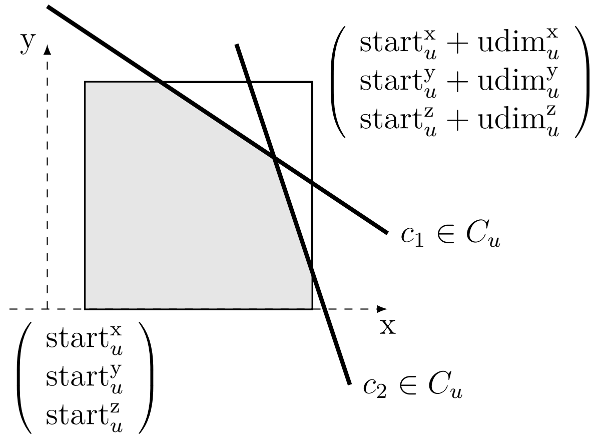


Figure 6.1: Modeling of irregular shaped bins: The bounding box of the bin is given by its start (start_u^*) and end ($\text{start}_u^* + \text{dim}_u^*$) coordinates. The two planes c_1 and c_2 describe the contour. All items must be placed inside the gray region.

$$\text{XZY} \notin \text{rot}_i \Rightarrow r_i^{11} + r_i^{23} + r_i^{32} \leq 1 \qquad i \in I \tag{6.8}$$
$$\text{YXZ} \notin \text{rot}_i \Rightarrow r_i^{12} + r_i^{21} + r_i^{33} \leq 1 \qquad i \in I \tag{6.9}$$
$$\text{ZXY} \notin \text{rot}_i \Rightarrow r_i^{13} + r_i^{21} + r_i^{32} \leq 1 \qquad i \in I \tag{6.10}$$
$$\text{YZX} \notin \text{rot}_i \Rightarrow r_i^{12} + r_i^{23} + r_i^{31} \leq 1 \qquad i \in I \tag{6.11}$$
$$\text{ZYX} \notin \text{rot}_i \Rightarrow r_i^{13} + r_i^{22} + r_i^{31} \leq 1 \qquad i \in I \tag{6.12}$$

Item position inside ULD Each loaded item i must be positioned fully inside its assigned ULD u. Note that we use a single coordinate system for all ULDs. This allows us to easily state positioning and segregation constraints within the whole aircraft:

$$\mathit{assign}_{u,i} \Rightarrow \text{start}_u^{\text{x}} \leq x_i \qquad i \in I; u \in U \tag{6.13}$$
$$\mathit{assign}_{u,i} \Rightarrow \text{start}_u^{\text{y}} \leq y_i \qquad i \in I; u \in U \tag{6.14}$$
$$\mathit{assign}_{u,i} \Rightarrow \text{start}_u^{\text{z}} \leq z_i \qquad i \in I; u \in U \tag{6.15}$$
$$\mathit{assign}_{u,i} \Rightarrow x_i' \leq \text{start}_u^{\text{x}} + \text{udim}_u^{\text{x}} \qquad i \in I; u \in U \tag{6.16}$$
$$\mathit{assign}_{u,i} \Rightarrow y_i' \leq \text{start}_u^{\text{y}} + \text{udim}_u^{\text{y}} \qquad i \in I; u \in U \tag{6.17}$$
$$\mathit{assign}_{u,i} \Rightarrow z_i' \leq \text{start}_u^{\text{z}} + \text{udim}_u^{\text{z}} \qquad i \in I; u \in U \tag{6.18}$$

Besides the orthogonal bounding box of the ULD, many ULD types have a convex contoured shape inside which the items must be placed. This shape is defined by the aircraft fuselage. Examples of ULD types were given in Figure 2.3. We describe such shapes as a set of planes as shown in Figure 6.1. Inside the bounding box of the ULD the intersection of the lower half spaces of these planes defines the feasible region in which items have to be placed. Accordingly, for each item i placed in ULD u we check that it is fully below each plane from C_u. For each plane $\langle \text{x}, \text{y}, \text{z}, \text{a} \rangle \in C_u$ we only need to check the corner of the item that is closest to the plane when the item is in the feasible region. We determine the coordinates $(\tilde{x}, \tilde{y}, \tilde{z})$ of the closest corner as follows:

$$\tilde{x} = \begin{cases} x_i & \text{if x} \leq 0 \\ x_i' & \text{else} \end{cases} \qquad \tilde{y} = \begin{cases} y_i & \text{if y} \leq 0 \\ y_i' & \text{else} \end{cases} \qquad \tilde{z} = \begin{cases} z_i & \text{if z} \leq 0 \\ z_i' & \text{else} \end{cases} \tag{6.19}$$

The following constraint ensures that the item is placed in the feasible region:

$$\mathrm{x}\tilde{x} + \mathrm{y}\tilde{y} + \mathrm{z}\tilde{z} \leq \mathrm{a} \tag{6.20}$$

Item non-overlapping If two items i, j are loaded, then their positions must not overlap. Note that this constraint is independent of the assigned ULD because all ULDs share the same coordinate system in our model. We describe the position of item i by its front left bottom corner (x_i, y_i, z_i) and its rear right top corner (x'_i, y'_i, z'_i). The two items i, j do not overlap with respect to a certain dimension if i is either in front of j $(x'_i \leq x_j)$ for dimension x or behind j $(x'_j \leq x_i)$. In three-dimensional space items do not overlap if they do not overlap in any dimension. Accordingly, we formulate the following constraint:

$$\neg(assign_i^{\text{off}} \vee assign_j^{\text{off}}) \Rightarrow \begin{pmatrix} x'_i \leq x_j \vee x'_j \leq x_i \vee \\ y'_i \leq y_j \vee y'_j \leq y_i \vee \\ z'_i \leq z_j \vee z'_j \leq z_i \end{pmatrix} \qquad i, j \in I \tag{6.21}$$

Item availability Each item i can only be assigned to a ULD u whose build-up period start_u is not before the item arrival time avail_i. For transferred items, avail_i can be determined by the inbound connection plus a suitable transfer time. For items delivered to the terminal by the customer the parameter can be derived from the agreed hand-over time.

$$\mathrm{avail}_i\, assign_{u,i} \leq \mathrm{start}_u \qquad u \in U; i \in I \tag{6.22}$$

ULD weight limit For each ULD u the sum of the item gross weights $\mathrm{weight}_i^{\mathrm{grs}}$ of all items placed on the ULD is limited. We expect that the weight limit limit_u is already adjusted to allow for the weight of the ULD itself, the weight of the covering net and straps, the weight of possibly necessary additional loading devices like wooden boards, as well as a suitable tolerance. We calculate the total weight of the ULD as $weight_u$ and will reuse it in the upcoming weight distribution constraint.

$$\sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}}\, assign_{u,i} = weight_u \qquad u \in U \tag{6.23}$$

$$weight_u \leq \mathrm{limit}_u \qquad u \in U \tag{6.24}$$

Net weight limits Besides the gross weight of each item, some items may contain substances (dry ice, radiation, chemicals, explosives) whose total sum needs to be limited within a subset of ULDs, typically all ULDs that will be put into the same compartment of the aircraft, for safety reasons.

We denote the set of restricted substances as G and the constraints related to substance $g \in G$ as D_g. For each constraint $d \in D_g$ the subset of ULDs in which the substance must be limited is given as ulds_d. The weight limit itself is given as $\mathrm{limit}_d^{\mathrm{net}}$. We denote the net weight of substance g inside item i as $\mathrm{weight}_{g,i}^{\mathrm{net}}$. The loaded net weights can now be restricted as follows:

$$\sum_{u \in \mathrm{ulds}_d} \sum_{i \in I} \mathrm{weight}_{g,i}^{\mathrm{net}}\, assign_{u,i} \leq \mathrm{limit}_d^{\mathrm{net}} \qquad g \in G; d \in D_g \tag{6.25}$$

ULD weight distribution For safety reasons, the items must be packed in a way such that the center of gravity (CG) of each ULD u is within a defined bounding box. We denote the coordinates of the front left bottom corner of this bounding box as $(\underline{\mathrm{x}}_u^{\mathrm{cg}}, \underline{\mathrm{y}}_u^{\mathrm{cg}}, \underline{\mathrm{z}}_u^{\mathrm{cg}})$ and the rear right top corner as $(\overline{\mathrm{x}}_u^{\mathrm{cg}}, \overline{\mathrm{y}}_u^{\mathrm{cg}}, \overline{\mathrm{z}}_u^{\mathrm{cg}})$. As a simplification, we assume that the CG of each item is at its geometric center. In dimension x, for example, this geometric center can be calculated as $(x_i + x'_i)/2$. To determine the CG of a set of items for a dimension we calculate the weighted sum of their geometric centers. Accordingly, for dimension x the CG can be calculated as:

$$\sum_i \left(\frac{\mathrm{weight}_i^{\mathrm{grs}}}{\text{total weight}}\right)\left(\frac{x_i + x'_i}{2}\right) \tag{6.26}$$

To limit the weight distribution inside the ULD u, we restrict the CG for each dimension to stay within the defined bounding box $(\underline{\mathrm{x}}_u^{\mathrm{cg}}, \underline{\mathrm{y}}_u^{\mathrm{cg}}, \underline{\mathrm{z}}_u^{\mathrm{cg}}) - (\overline{\mathrm{x}}_u^{\mathrm{cg}}, \overline{\mathrm{y}}_u^{\mathrm{cg}}, \overline{\mathrm{z}}_u^{\mathrm{cg}})$. Only items loaded into the ULD u need to be considered which is indicated by variable $assign_{u,i}$. From the ULD weight limit constraint we know the total weight of the loaded items $weight_u$. The CG can now be restricted for each dimension as follows:

$$\underline{\mathrm{x}}_u^{\mathrm{cg}} \leq \frac{1}{2weight_u} \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \, (x_i + x'_i) \, assign_{u,i} \leq \overline{\mathrm{x}}_u^{\mathrm{cg}} \qquad u \in U \tag{6.27}$$

$$\underline{\mathrm{y}}_u^{\mathrm{cg}} \leq \frac{1}{2weight_u} \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \, (y_i + y'_i) \, assign_{u,i} \leq \overline{\mathrm{y}}_u^{\mathrm{cg}} \qquad u \in U \tag{6.28}$$

$$\underline{\mathrm{z}}_u^{\mathrm{cg}} \leq \frac{1}{2weight_u} \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \, (z_i + z'_i) \, assign_{u,i} \leq \overline{\mathrm{z}}_u^{\mathrm{cg}} \qquad u \in U \tag{6.29}$$

Stacking When loading multiple items into the same ULD, some items might have to be stacked on top of others. As most items carry significant weight and the load-bearing strength of each item is limited, we need to properly restrict the stacking in our model.

Modeling exact stacking constraints is hard for two reasons. First, modeling the dynamics between stacked items is usually done via finite element simulations. Integrating such a simulation into any optimization model would most probably render it impossible to solve. Second, the real load-bearing strength of each item is seldom known, it depends on where the forces act, at the center or along the edges, and it cannot easily be determined without breaking the item.

Therefore, we model a simplified stacking constraint that considers only vertical forces and no torque. We assume that an item's weight is equally distributed across its volume and that the load-bearing strength is the same all across the top surface. Each item passes its weight onto the items below proportional to their overlapping area. The item below adds the received weight to its own weight and again passes the sum to the items below. This way, we can calculate the stress, i.e., the weight per area unit, transferred between two items and limit it to an acceptable level.

Figure 6.2 gives an example how the stacking restriction works. There are four items with different weights (A: 100 kg, B: 500 kg, C: 600 kg, D: 1,000 kg). Item A has no items on top, so its total weight is 100 kg. It passes its weight to item B resulting in a pressure of 50 $\mathrm{kg} \cdot \mathrm{m}^{-1}$ at the contact area. The total weight of item B is 600 kg, the weight received

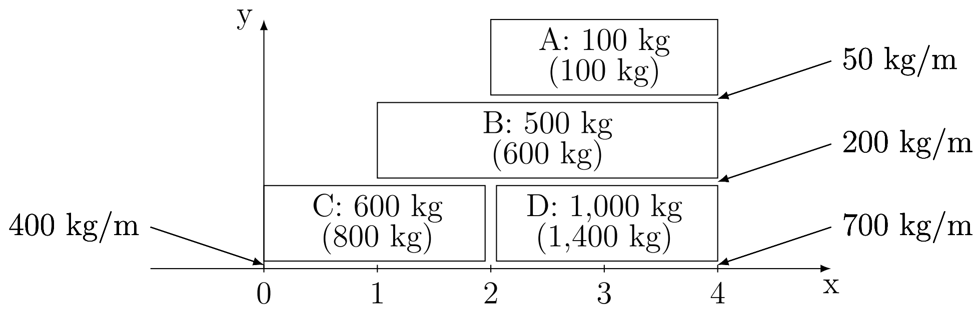


Figure 6.2: Illustration of the stacking model: The item weight is passed to the supporting items. Numbers in braces show the effective item weight including all weight stacked on top of it. Arcs show the respective stress at the contact areas.

from A plus B's own weight. Item B distributes its total weight onto items C and D proportionally to their contact areas, i.e., 200 kg onto item C and 400 kg onto item D.

There are other concepts in the literature to model stacking restrictions. A model where items are tagged as fragile or non-fragile, and non-fragile items are not allowed on top of fragile ones is given by Fuellerer et al. (2010). In a similar approach Egeblad et al. (2010) only allow lighter items to be stacked on top of heavier ones. For the strongly heterogeneous items seen in air cargo these models do not fit well, because none of them limits the total weight that is allowed to be put onto a certain item. A similar model to ours, where stress is transferred between the items is presented by Junqueira et al. (2012). However, the authors discretize the three-dimensional space, which limits their approach to small instances. Furthermore, they require that all items are fully supported. As ULDs on the lower deck of aircraft have to contain overhanging items to fill the loading space, this limitation is not acceptable in an air cargo environment.

To model the stacking restrictions, we describe a constraint to calculate the horizontal overlap between stacked items. Then, we model the transfer of weight between the overlapping items. Finally, we derive the stress at the contact areas from the overlap and transferred weights and limit the stress to the allowed level. The maximum allowed stress for item i is given for each dimension as $\mathrm{stack}_i^{\mathrm{x}}$, $\mathrm{stack}_i^{\mathrm{y}}$, and $\mathrm{stack}_i^{\mathrm{z}}$. We choose the limiting parameter depending on the item rotation.

As a new feature, we add a tolerance parameter ϵ to our stacking constraint. In practice larger items are often stacked on top of multiple other items, which are placed next to each other but are not exactly the same height. This is feasible because most packagings are a bit flexible. They will deform or compress a little, such that weight can be equally distributed between the stacked items. Optionally, a worker could fill the gap with dunnage material like a wooden board. Therefore, we accept height differences within the tolerance for this constraint. A similar idea was introduced in Egeblad et al. (2010), where the maximum height difference of the supporting items was limited. However, this was only checked a-posteriori and the size of the supporting area was not considered at all.

First, we calculate the overlapping lengths $o_{i,j}^{\mathrm{x}}$ and $o_{i,j}^{\mathrm{z}}$ between a lower item i and upper item j in both horizontal dimensions x and z. We consider i and j as overlapping if they are both loaded and less than the tolerance ϵ apart in vertical dimension. As all ULDs in our model use the same coordinate system, we do not need to care about the assigned

ULD of each item.

$$\neg\left(assign_i^{\text{off}} \vee assign_j^{\text{off}}\right) \wedge (0 \leq y_j - y_i' \leq \epsilon) \Rightarrow \\ o_{i,j}^{\text{x}} = \max(0, \min(x_i', x_j') - \max(x_i, x_j)) \qquad i,j \in I \qquad (6.30)$$

$$\neg\left(assign_i^{\text{off}} \vee assign_j^{\text{off}}\right) \wedge (0 \leq y_j - y_i' \leq \epsilon) \Rightarrow \\ o_{i,j}^{\text{z}} = \max(0, \min(z_i', z_j') - \max(z_i, z_j)) \qquad i,j \in I \qquad (6.31)$$

From the overlaps we calculate the total supported area o_j of each item j as follows. We note that this formulation is non-linear, making this part of the problem intractable for plain LP solving.

$$o_j = \sum_{i \in I} o_{i,j}^{\text{x}} o_{i,j}^{\text{z}} \qquad j \in I \qquad (6.32)$$

We now model the transfer of weight from upper items j to lower item i. Even if the upper item is not fully supported, our formulation ensures that all the weight of an upper item is transferred to the items below it. We introduce an auxiliary variable w_i representing the total weight of item i, which is the sum of the weight of the lower item i plus the weight transferred to it by items above.

$$w_i = \text{weight}_i^{\text{grs}} + \sum_{j \in I} \frac{o_{i,j}^{\text{x}} o_{i,j}^{\text{z}}}{o_j} w_j \qquad i \in I \qquad (6.33)$$

Finally, we restrict the stress of an upper item j onto lower item i by the load-bearing strength of the vertical orientation of item i. To detect the vertical orientation we reuse auxiliary variables r_i^{2*} from the rotation matrix of the item orientation constraint. We determine the stress at the contact area by the total weight and the total supported area of the item above.

$$o_{i,j}^{\text{x}} o_{i,j}^{\text{z}} > 0 \Rightarrow r_i^{21}\text{stack}_i^{\text{x}} + r_i^{22}\text{stack}_i^{\text{y}} + r_i^{23}\text{stack}_i^{\text{z}} \geq \frac{w_j}{o_j} \qquad i,j \in I \qquad (6.34)$$

Complete shipment There are different variations of complete shipment constraints. According to the classification given by Eley (2003), the problem at hand is of the type where subsets of boxes have to be either all loaded or left out completely.

Shipments may consist of multiple items that can in theory be shipped individually. But, in some countries customs regulations do not permit the arrival of partial shipments. In such countries the airline has to pay a considerable fine if not all items of a shipment arrive with the same flight. Accordingly, we introduce a constraint that either all or no items of a certain group are loaded.

Therefore, we select one item of each shipment to represent the group that belongs together. Each item i has a parameter clique_i pointing to the master item of its group. For each item we force the offload variable $assign_i^{\text{off}}$ to be equal to the same variable of the group's master item. This effectively creates a clique where either all or no items are loaded.

$$assign_i^{\text{off}} = assign_{\text{clique}_i}^{\text{off}} \qquad i \in I \qquad (6.35)$$

Item sorting Besides the requirement to load only complete shipments, airlines also want to spread a shipment with multiple items across as few ULDs as possible. This decreases the risk of disruptions, e.g., if a ULD does not catch its flight, and reduces the handling effort to consolidate the shipment again at the destination. In practice there are even more reasons to reduce the spread of a set of items across ULDs. For example items that arrive with the same inbound flight at the terminal, that have the same final destination, or items that need special care during flight, like animals. Placing these items on only a few ULDs saves a lot of handling effort.

Item sorting can also be seen from a ULD perspective. Air cargo is often offered with different service levels, say standard and premium. As the premium shipments are sold with the promise to be processed faster at the destination airport, they should not be mixed with standard cargo. Another good reason for item sorting is to fill a ULD only with items that have the same connecting flight after the current one. In this case, a so called thru-unit can be built that can transit the next station without requiring any handling effort.

While these are different requirements from a business view, they are technically equivalent. Therefore, we formulate them all in the same way in our model. We identify two vantage points on the sorting criteria. The first is item-bound: a group of items should be split across as few ULDs as possible. The other is ULD-bound: inside one ULD there should be as few different item groups as possible. We denote the corresponding criteria sets as Q^{uld} and Q^{item} ($Q = Q^{\text{uld}} \cup Q^{\text{item}}$). For each sorting criterion $q \in Q$ each item i has a parameter $\text{sort}_{q,i} \in Q_q$ indicating the group it belongs to with respect to criterion q. We add auxiliary variables $sort_{q,u,e}$ for each constraint q indicating if the item group $e \in Q_q$ is present in ULD u. This is the case if any item of this group is assigned to u:

$$assign_{u,i} \leq sort_{q,u,e} \qquad q \in Q; u \in U; e \in Q_q; i \in I : \text{sort}_{q,i} = e \tag{6.36}$$

To calculate the degree of sorting n_q for each item-bound constraint q we count the number of ULDs each group is present in. To better express the impact of bad sorting for each constraint we add an exponential penalty parameter pow_q. We note that this non-linear restriction can be reformulated into piecewise linear restrictions because the number of ULDs a group is spread across will usually be small. We calculate the degree of sorting as:

$$n_q = \sum_{e \in Q_q} \left(\sum_{u \in U} sort_{q,u,e} \right)^{\text{pow}_q} \qquad q \in Q^{\text{item}} \tag{6.37}$$

To calculate the degree of sorting for ULD-bound constraints, we sum up the same incidence matrix as above just in the opposite order of dimensions:

$$n_q = \sum_{u \in U} \left(\sum_{e \in Q_q} sort_{q,u,e} \right)^{\text{pow}_q} \qquad q \in Q^{\text{uld}} \tag{6.38}$$

Item compatibility Besides the soft item grouping constraints, there are also hard constraints on items, which must not be placed on the same ULD or in close proximity. These constraints mostly stem from official dangerous goods regulations (DGR). These define a long list of items that have to be separated by a minimum distance. Examples are food and perfumery items, animals and items containing sources of radiation, or items with chemicals that could react with each other. There is only one paper by Eley (2003) dealing with this kind of item separation constraint.

To split incompatible items onto different ULDs we introduce a constraint for all pairs $\langle i, j \rangle$ of incompatible items A:

$$assign_{u,i} + assign_{u,j} \leq 1 \qquad u \in U; \langle i, j \rangle \in A \tag{6.39}$$

Item spacing For pairs of items that require a certain minimum distance we introduce a parameter $\mathrm{dist}_{i,j}$. Remember the coordinates of item i: the front bottom left corner (x_i, y_i, z_i) end the rear top right corner (x'_i, y'_i, z'_i). For this constraint we consider horizontal distance only, because vertical distance might not be sufficient for leakages. To simplify the model we calculate the Manhattan distance instead of Euclidean distance between the items. Furthermore, multiplying a Euclidean distance by a factor of $\sqrt{2}$ yields a valid upper bound for the distance in Manhattan distance. We define the minimum distance constraint between two items i and j as:

$$\max(0, x_i - x'_j, x_j - x'_i) + \max(0, z_i - z'_j, z_j - z'_i) \geq \sqrt{2}\mathrm{dist}_{i,j} \qquad i, j \in I \tag{6.40}$$

Item positioning For some items the allowed position inside the aircraft or inside the assigned ULD needs to be restricted.

For example, if radioactive materials are loaded into a passenger aircraft they must have a minimum distance to the passengers' cabin, i.e., be loaded not above a certain absolute height. To model a global positioning constraint of item i we introduce a bounding box $(\underline{\mathrm{x}}_i^{\mathrm{ac}}, \underline{\mathrm{y}}_i^{\mathrm{ac}}, \underline{\mathrm{z}}_i^{\mathrm{ac}})$ to $(\overline{\mathrm{x}}_i^{\mathrm{ac}}, \overline{\mathrm{y}}_i^{\mathrm{ac}}, \overline{\mathrm{z}}_i^{\mathrm{ac}})$. The item's position has to stay within this box:

$$\underline{\mathrm{x}}_i^{\mathrm{ac}} \leq x_i \qquad i \in I \tag{6.41}$$

$$x'_i \leq \overline{\mathrm{x}}_i^{\mathrm{ac}} \qquad i \in I \tag{6.42}$$

$$\underline{\mathrm{y}}_i^{\mathrm{ac}} \leq y_i \qquad i \in I \tag{6.43}$$

$$y'_i \leq \overline{\mathrm{y}}_i^{\mathrm{ac}} \qquad i \in I \tag{6.44}$$

$$\underline{\mathrm{z}}_i^{\mathrm{ac}} \leq z_i \qquad i \in I \tag{6.45}$$

$$z'_i \leq \overline{\mathrm{z}}_i^{\mathrm{ac}} \qquad i \in I \tag{6.46}$$

Similarly, the relative position of an item inside the ULD might need to be constrained. This is usually the case for dangerous goods that need to be placed on the periphery of the ULD to be inspected or accessed if necessary. The same applies to living animals that need ventilation. On the other hand, items of high value should not be placed on the periphery to prevent theft. To model this constraint of item i when assigned to ULD u we introduce two more bounding boxes, one for the item start coordinates $(\underleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \underleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \underleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}})$ to $(\overleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \overleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \overleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}})$ and another for the item end coordinates $(\underrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \underrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \underrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}})$ to $(\overrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \overrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \overrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}})$. We apply the Big M method to relax this constraint if the item is offloaded at all. The relative position is then limited as:

$$x_i \geq \underleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.47}$$

$$x_i \leq \overleftarrow{\mathrm{x}}^{\mathrm{uld}}_{u,i} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.48}$$
$$y_i \geq \underleftarrow{\mathrm{y}}^{\mathrm{uld}}_{u,i} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.49}$$
$$y_i \leq \overleftarrow{\mathrm{y}}^{\mathrm{uld}}_{u,i} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.50}$$
$$z_i \geq \underleftarrow{\mathrm{z}}^{\mathrm{uld}}_{u,i} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.51}$$
$$z_i \leq \overleftarrow{\mathrm{z}}^{\mathrm{uld}}_{u,i} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.52}$$
$$x'_i \geq \underrightarrow{\mathrm{x}}^{\mathrm{uld}}_{u,i} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.53}$$
$$x'_i \leq \overrightarrow{\mathrm{x}}^{\mathrm{uld}}_{u,i} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.54}$$
$$y'_i \geq \underrightarrow{\mathrm{y}}^{\mathrm{uld}}_{u,i} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.55}$$
$$y'_i \leq \overrightarrow{\mathrm{y}}^{\mathrm{uld}}_{u,i} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.56}$$
$$z'_i \geq \underrightarrow{\mathrm{z}}^{\mathrm{uld}}_{u,i} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.57}$$
$$z'_i \leq \overrightarrow{\mathrm{z}}^{\mathrm{uld}}_{u,i} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.58}$$

Stability Load stability is required for all ULDs to prevent damage of the cargo and movements inside the aircraft, which would endanger flight safety. We distinguish between vertical and horizontal stability. Vertical stability prevents the items from falling down or tipping. To enforce vertical stability we require that a minimum ratio supp_i of item i's floor area is supported by other items or the item is placed directly on the ULD floor. In many papers load stability is assured by requiring full support of all items. In an air cargo setup this often not necessary and sometimes even not possible. For example, ULDs on the lower deck only have a small floor area and reach their full width only at around 50 cm height. Accordingly, there must be overhanging items to fully utilize the loading space.

In our formulation the minimum support ratio supp_i is item dependent, because different item geometries might require different supported areas. For a cardboard box with a flat bottom a support of $\mathrm{supp}_i = 0.75$ might be sufficient. But, a pallet standing on four feet at its corners only might require full support of $\mathrm{supp}_i = 1.0$. For items standing on the ULD floor we assume full support. Therefore, we assume that items are placed directly on the ULD floor if their vertical starting position y_i is larger the vertical ULD starting position $\mathrm{start}^{\mathrm{y}}_u$ plus the stacking tolerance ϵ. Accordingly, we restrict the minimum supported area by the following non-linear constraint:

$$\Big(y_i > assign_{u,i}\mathrm{start}^{\mathrm{y}}_u + \epsilon\Big) \Rightarrow (o_i \geq \mathrm{supp}_i(x'_i - x_i)(z'_i - z_i)) \qquad u \in U; i \in I \tag{6.59}$$

Horizontal stability prevents the shifting of items when the ULD is moved or tilted. In air cargo this is usually ensured by the usage of nets and straps after the ULD is build-up. Therefore, we do not model horizontal stability explicitly.

6.2.4 Objective

The objective of the palletization is to load as much as possible while keeping the necessary physical handling effort low. Therefore, we penalize all offloaded items, where $assign^{\mathrm{off}}_i = 1$, with their individual penalty $\mathrm{offload}_i$.

To simplify all handling operations we would also like the load plan to be as sorted as possible. For each sorting criterion $q \in Q$ the variable n_q contains a measure how well the items are sorted. We weight all measures by their individual penalty factors penalty_q.

Our objective minimizes the sum of penalties of the offloaded items and violated sorting criteria:

$$\text{minimize} \sum_{i \in I} \text{offload}_i \, \mathit{assign}_i^{\text{off}} + \sum_{q \in Q} \text{penalty}_q n_q \tag{6.60}$$

6.2.5 Model overview

After introducing all components of the APM in the last sections, we now give a consolidated view of the full model:

$$\text{minimize} \sum_{i \in I} \text{offload}_i \, \mathit{assign}_i^{\text{off}} + \sum_{q \in Q} \text{penalty}_q n_q \tag{6.61}$$

subject to

$$1 = \mathit{assign}_i^{\text{off}} + \sum_{u \in U_i} \mathit{assign}_{u,i} \qquad i \in I \tag{6.62}$$

$$x'_i = x_i + r_i^{11} \text{dim}_i^{\text{x}} + r_i^{12} \text{dim}_i^{\text{y}} + r_i^{13} \text{dim}_i^{\text{z}} \qquad i \in I \tag{6.63}$$

$$y'_i = y_i + r_i^{21} \text{dim}_i^{\text{x}} + r_i^{22} \text{dim}_i^{\text{y}} + r_i^{23} \text{dim}_i^{\text{z}} \qquad i \in I \tag{6.64}$$

$$z'_i = z_i + r_i^{31} \text{dim}_i^{\text{x}} + r_i^{32} \text{dim}_i^{\text{y}} + r_i^{33} \text{dim}_i^{\text{z}} \qquad i \in I \tag{6.65}$$

$$1 = \sum_{b=1}^{3} r_i^{ab} \qquad i \in I; a \in \{1,2,3\} \tag{6.66}$$

$$1 = \sum_{a=1}^{3} r_i^{ab} \qquad i \in I; b \in \{1,2,3\} \tag{6.67}$$

$$\text{XYZ} \notin \text{rot}_i \Rightarrow r_i^{11} + r_i^{22} + r_i^{33} \leq 1 \qquad i \in I \tag{6.68}$$

$$\text{XZY} \notin \text{rot}_i \Rightarrow r_i^{11} + r_i^{23} + r_i^{32} \leq 1 \qquad i \in I \tag{6.69}$$

$$\text{YXZ} \notin \text{rot}_i \Rightarrow r_i^{12} + r_i^{21} + r_i^{33} \leq 1 \qquad i \in I \tag{6.70}$$

$$\text{ZXY} \notin \text{rot}_i \Rightarrow r_i^{13} + r_i^{21} + r_i^{32} \leq 1 \qquad i \in I \tag{6.71}$$

$$\text{YZX} \notin \text{rot}_i \Rightarrow r_i^{12} + r_i^{23} + r_i^{31} \leq 1 \qquad i \in I \tag{6.72}$$

$$\text{ZYX} \notin \text{rot}_i \Rightarrow r_i^{13} + r_i^{22} + r_i^{31} \leq 1 \qquad i \in I \tag{6.73}$$

$$\mathit{assign}_{u,i} \Rightarrow \text{start}_u^{\text{x}} \leq x_i \qquad i \in I; u \in U \tag{6.74}$$

$$\mathit{assign}_{u,i} \Rightarrow \text{start}_u^{\text{y}} \leq y_i \qquad i \in I; u \in U \tag{6.75}$$

$$\mathit{assign}_{u,i} \Rightarrow \text{start}_u^{\text{z}} \leq z_i \qquad i \in I; u \in U \tag{6.76}$$

$$\mathit{assign}_{u,i} \Rightarrow x'_i \leq \text{start}_u^{\text{x}} + \text{udim}_u^{\text{x}} \qquad i \in I; u \in U \tag{6.77}$$

$$\mathit{assign}_{u,i} \Rightarrow y'_i \leq \text{start}_u^{\text{y}} + \text{udim}_u^{\text{y}} \qquad i \in I; u \in U \tag{6.78}$$

$$\mathit{assign}_{u,i} \Rightarrow z'_i \leq \text{start}_u^{\text{z}} + \text{udim}_u^{\text{z}} \qquad i \in I; u \in U \tag{6.79}$$

$$\text{a} \geq \text{x}x_i + \text{y}y_i + \text{z}z_i \qquad i \in I; \langle \text{x}, \text{y}, \text{z}, \text{a} \rangle \in C_u \tag{6.80}$$

$$\text{a} \geq \text{x}x_i + \text{y}y_i + \text{z}z'_i \qquad i \in I; \langle \text{x}, \text{y}, \text{z}, \text{a} \rangle \in C_u \tag{6.81}$$

$$\mathrm{a} \geq \mathrm{x}x_i + \mathrm{y}y'_i + \mathrm{z}z_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{6.82}$$

$$\mathrm{a} \geq \mathrm{x}x_i + \mathrm{y}y'_i + \mathrm{z}z'_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{6.83}$$

$$\mathrm{a} \geq \mathrm{x}x'_i + \mathrm{y}y_i + \mathrm{z}z_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{6.84}$$

$$\mathrm{a} \geq \mathrm{x}x'_i + \mathrm{y}y_i + \mathrm{z}z'_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{6.85}$$

$$\mathrm{a} \geq \mathrm{x}x'_i + \mathrm{y}y'_i + \mathrm{z}z_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{6.86}$$

$$\mathrm{a} \geq \mathrm{x}x'_i + \mathrm{y}y'_i + \mathrm{z}z'_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{6.87}$$

$$\neg \begin{pmatrix} \mathit{assign}_i^{\mathrm{off}} \vee \\ \mathit{assign}_j^{\mathrm{off}} \end{pmatrix} \Rightarrow \begin{pmatrix} x'_i \leq x_j \vee x'_j \leq x_i \vee \\ y'_i \leq y_j \vee y'_j \leq y_i \vee \\ z'_i \leq z_j \vee z'_j \leq z_i \end{pmatrix} \qquad i, j \in I \tag{6.88}$$

$$\mathrm{start}_u \geq \mathrm{avail}_i \mathit{assign}_{u,i} \qquad u \in U; i \in I \tag{6.89}$$

$$\mathit{weight}_u = \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \mathit{assign}_{u,i} \qquad u \in U \tag{6.90}$$

$$\mathit{weight}_u \leq \mathrm{limit}_u \qquad u \in U \tag{6.91}$$

$$\mathrm{limit}_d^{\mathrm{net}} \geq \sum_{u \in \mathrm{ulds}_d} \sum_{i \in I} \mathrm{weight}_{g,i}^{\mathrm{net}} \mathit{assign}_{u,i} \qquad g \in G; d \in D_g \tag{6.92}$$

$$2\mathit{weight}_u \underline{\mathrm{x}}_u^{\mathrm{cg}} \leq \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \left(x_i + x'_i\right) \mathit{assign}_{u,i} \qquad u \in U \tag{6.93}$$

$$2\mathit{weight}_u \overline{\mathrm{x}}_u^{\mathrm{cg}} \geq \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \left(x_i + x'_i\right) \mathit{assign}_{u,i} \qquad u \in U \tag{6.94}$$

$$2\mathit{weight}_u \underline{\mathrm{y}}_u^{\mathrm{cg}} \leq \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \left(y_i + y'_i\right) \mathit{assign}_{u,i} \qquad u \in U \tag{6.95}$$

$$2\mathit{weight}_u \overline{\mathrm{y}}_u^{\mathrm{cg}} \geq \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \left(y_i + y'_i\right) \mathit{assign}_{u,i} \qquad u \in U \tag{6.96}$$

$$2\mathit{weight}_u \underline{\mathrm{z}}_u^{\mathrm{cg}} \leq \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \left(z_i + z'_i\right) \mathit{assign}_{u,i} \qquad u \in U \tag{6.97}$$

$$2\mathit{weight}_u \overline{\mathrm{z}}_u^{\mathrm{cg}} \geq \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \left(z_i + z'_i\right) \mathit{assign}_{u,i} \qquad u \in U \tag{6.98}$$

$$\begin{aligned} \neg \left(\mathit{assign}_i^{\mathrm{off}} \vee \mathit{assign}_j^{\mathrm{off}}\right) \wedge (0 \leq y_j - y'_i \leq \epsilon) &\Rightarrow \\ o_{i,j}^{\mathrm{x}} = \max(0, \min(x'_i, x'_j) - \max(x_i, x_j)) \end{aligned} \qquad i, j \in I \tag{6.99}$$

$$\begin{aligned} \neg \left(\mathit{assign}_i^{\mathrm{off}} \vee \mathit{assign}_j^{\mathrm{off}}\right) \wedge (0 \leq y_j - y'_i \leq \epsilon) &\Rightarrow \\ o_{i,j}^{\mathrm{z}} = \max(0, \min(z'_i, z'_j) - \max(z_i, z_j)) \end{aligned} \qquad i, j \in I \tag{6.100}$$

$$o_j = \sum_{i \in I} o_{i,j}^{\mathrm{x}} o_{i,j}^{\mathrm{z}} \qquad j \in I \tag{6.101}$$

$$w_i = \mathrm{weight}_i^{\mathrm{grs}} + \sum_{j \in I} \frac{o_{i,j}^{\mathrm{x}} o_{i,j}^{\mathrm{z}}}{o_j} w_j \qquad i \in I \tag{6.102}$$

$$o_{i,j}^{\mathrm{x}} o_{i,j}^{\mathrm{z}} > 0 \Rightarrow r_i^{21} \mathrm{stack}_i^{\mathrm{x}} + r_i^{22} \mathrm{stack}_i^{\mathrm{y}} + r_i^{23} \mathrm{stack}_i^{\mathrm{z}} \geq \frac{w_j}{o_j} \qquad i, j \in I \tag{6.103}$$

$$\mathit{assign}_i^{\mathrm{off}} = \mathit{assign}_{\mathrm{clique}_i}^{\mathrm{off}} \qquad i \in I \tag{6.104}$$

$$\mathit{assign}_{u,i} \leq \mathit{sort}_{q,u,\mathrm{sort}_{q,i}} \qquad q \in Q; u \in U; i \in I \tag{6.105}$$

$$n_q = \sum_{e \in Q_q} \left(\sum_{u \in U} sort_{q,u,e} \right)^{\mathrm{pow}_q} \qquad q \in Q^{\mathrm{item}} \tag{6.106}$$

$$n_q = \sum_{u \in U} \left(\sum_{e \in Q_q} sort_{q,u,e} \right)^{\mathrm{pow}_q} \qquad q \in Q^{\mathrm{uld}} \tag{6.107}$$

$$1 \geq assign_{u,i} + assign_{u,j} \qquad u \in U; \langle i, j \rangle \in A \tag{6.108}$$

$$\sqrt{2}\mathrm{dist}_{i,j} \leq \begin{array}{l} \max(0, x_i - x'_j, x_j - x'_i) + \\ \max(0, z_i - z'_j, z_j - z'_i) \end{array} \qquad i, j \in I \tag{6.109}$$

$$\underline{\mathrm{x}}_i^{\mathrm{ac}} \leq x_i \qquad i \in I \tag{6.110}$$

$$x'_i \leq \overline{\mathrm{x}}_i^{\mathrm{ac}} \qquad i \in I \tag{6.111}$$

$$\underline{\mathrm{y}}_i^{\mathrm{ac}} \leq y_i \qquad i \in I \tag{6.112}$$

$$y'_i \leq \overline{\mathrm{y}}_i^{\mathrm{ac}} \qquad i \in I \tag{6.113}$$

$$\underline{\mathrm{z}}_i^{\mathrm{ac}} \leq z_i \qquad i \in I \tag{6.114}$$

$$z'_i \leq \overline{\mathrm{z}}_i^{\mathrm{ac}} \qquad i \in I \tag{6.115}$$

$$x_i \geq \underleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.116}$$

$$x_i \leq \overleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.117}$$

$$y_i \geq \underleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.118}$$

$$y_i \leq \overleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.119}$$

$$z_i \geq \underleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.120}$$

$$z_i \leq \overleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.121}$$

$$x'_i \geq \underrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.122}$$

$$x'_i \leq \overrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.123}$$

$$y'_i \geq \underrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.124}$$

$$y'_i \leq \overrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.125}$$

$$z'_i \geq \underrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}} + (assign_{u,i} - 1)M \qquad u \in U; i \in I \tag{6.126}$$

$$z'_i \leq \overrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}} + (1 - assign_{u,i})M \qquad u \in U; i \in I \tag{6.127}$$

$$\left(y_i > assign_{u,i}\mathrm{start}_u^{\mathrm{y}} + \epsilon \right) \Rightarrow \left(o_i \geq \mathrm{supp}_i(x'_i - x_i)(z'_i - z_i) \right) \qquad u \in U; i \in I \tag{6.128}$$

The objective, given in Equation (6.61), is to minimize the sum of the penalties of offloaded items and for violating the sorting constraints. Equation (6.62) models the ULD assignment, Equations (6.63) to (6.73) the item rotation, Equations (6.74) to (6.87) the item position inside the ULD, Equation (6.88) the item non-overlapping, Equation (6.89) the item availability, Equations (6.90) and (6.91) the ULD weight limit, Equation (6.92) the net weight limits, Equation (6.93) to (6.98) the ULD weight distribution, Equation (6.99) to (6.103) the stacking of items, Equation (6.104) the complete shipment of items, Equations (6.105) to (6.107) the item sorting, Equation (6.108) the item compatibility on ULDs, Equation (6.109) the item spacing, Equations (6.110) to (6.127) the item positioning constraints, and Equation (6.128) the item vertical stability.

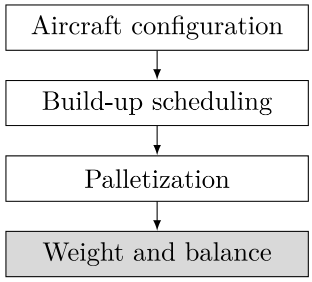
6.3 Summary

In this chapter, we introduced the Air Cargo Palletization Problem (ACP) in detail, which is to solve a three-dimensional container loading problem while respecting a multitude of physical, regulatory, and organizational constraints. Furthermore, we analyzed the related literature and presented a formal model, denoted as the Air Cargo Palletization Model (APM), of the problem features.

There is a vast amount of literature regarding container loading problems and most of the constraints present in air cargo have been mentioned in the literature. However, only few works exist that deal with several of the practically relevant constraints simultaneously. Of the twelve types of requirements described by Bischoff and Ratcliff (1995) eleven can be found in our problem at hand and are built into our model.

The presented model is basically a three-dimensional multi-container loading problem, also denoted as Multiple Heterogeneous Knapsack Problem (MHKP) in the literature. We formally defined all of its constraints, which is especially interesting for the stackability and stability constraints, because these are often only described verbally in literature.

Besides, our model introduces some new practically relevant features that have not yet been presented in the literature. Most notably, we consider time, both of item arrivals and bin due dates, and we add a small tolerance for the stacking constraint that allows the stacked item to be supported by items of slightly different heights.



7 Weight and balance

The final step of aircraft load planning is the weight and balance (FAA, 2016), often abbreviated as WAB. Here, a set of packed and weighed ULDs has to be arranged on distinct loading positions inside the aircraft's cargo compartments with respect to structural and loading constraints. The weight and balance has to be calculated for each flight leg individually. Cargo flights often contain multiple legs. After each leg some ULDs might be loaded or unloaded while others continue onto the next flight leg. Rearranging the continuing ULDs between two flight legs is possible. Depending on the situation this might save fuel or allow to load more weight. But, it also incurs cost for additional unloading and loading operations. Therefore, we solve the weight and balance for all legs of a flight within one model, to find a cost-optimal assignment for the whole flight.

The structural limitations of an aircraft include a total weight limit, weight limits for individual loading positions as well as subsets of loading positions, and a defined range for the aircraft's CG. The loading restrictions consist of compatibility and assignment constraints between the ULDs and the partially overlapping loading positions. Furthermore, loading positions must be accessible from the door during loading and unloading, i.e., not be blocked by ULDs staying on board.

With regard to the weight and balance objective, the arrangement of ULDs influences the flight in three ways. Firstly, it defines the CG of the aircraft. During flight there is an optimal CG at which the aircraft consumes the least fuel. Therefore, a good arrangement of ULDs puts the actual CG close to the optimum. Secondly, the assignment defines the moment of inertia (MI) of the aircraft around the CG. A lower MI makes the aircraft easier maneuverable as less force is necessary to change the direction of flight. Therefore, the MI should be small. Finally, on multi-leg flights where only some ULDs are unloaded or loaded at each stop, the arrangement of ULDs determines the number of ground handling operations. Besides the cost, each ground handling operation takes time, increases the wear and tear of the aircraft, and raises the risk of damaging the cargo. Therefore, the number of operations should also be minimized. If the right ULDs are placed at the doors no additional ULD movements are necessary.

We define the Weight and Balance Problem (WBP) as the subproblem of the ACLPP to decide about the assignment of ULDs to loading positions on each leg of a multi-leg flight. In the WBP we consider each flight independently, but successive legs must be handled within one problem because handling operations take place between the legs.

In the next section, we discuss literature related to the WBP followed by a mathematical formulation of the problem in Section 7.2.

First author	Year	Optional loading	Cumulative weights	Overlapping positions	Moment of inertia	ULD separation	Number of legs	Objective
Brosh	1981	X	-	-	-	-	1	max load
Mongeau	2003	X	-	X	-	-	1	max load
Limbourg	2012	-	X	X	X	-	1	min moment of inertia
Vancroonenburg	2014	X	X	X	-	X	1	min center of gravity offset
Lurkin	2015	-	X	X	-	X	2	min cost
Dahmani	2016	X	-	-	-	-	1	max load, max priority

Table 7.1: Overview of recent literature on weight an balance problems with ULDs. The columns show the considered features and objective. Technical constraints like center of gravity limits, which are present in all the papers, are not shown.

7.1 Related literature

The earliest notable work regarding the loading of ULDs into civilian aircraft was given by Larsen and Mikkelsen (1980). The authors present an interactive loading procedure that considers practical constraints like (combined) weight limits, bounds for loading imbalance, and assignment restrictions between ULDs and loading positions.

An early LP formulation was given by Brosh (1981). The model deals with bulk loads, not ULDs. The objective is to maximize the profit of the transported cargo while respecting capacity and balance limits.

Mongeau and Bes (2003) consider the loading of ULDs into aircraft compartments, but not to individual loading positions. The authors present a MIP model where each compartment's capacity is limited by a maximum total weight and a set of linear sums on the number of certain ULD types. Thus their model only calculates a rough estimation of the center of gravity (CG) as the individual loading positions are not considered and the compartments are rather large. Their single objective is to maximize the loaded weight.

Limbourg et al. (2012) present a MIP model for the assignment of ULDs to loading positions. They limit longitudinal and lateral imbalances, introduce load limits on subsets of ULDs, and incorporate mutually excluding loading positions. The objective is to minimize the aircraft's moment of inertia. They solved their model with CPLEX and report runtimes of a few seconds for large cargo aircraft.

Vancroonenburg et al. (2014) extend the work above. Their model selects the most profitable subset of ULDs and minimizes the deviation from the optimal CG. Furthermore, they add a set of balance restrictions that apply during different phases of the flight to cater for changing fuel weights.

Lurkin and Schyns (2015) also extend the work of Limbourg et al. (2012). They introduce multi-leg flights, however limiting themselves to only two legs, and model the accessibility of loading positions at the stop-over airport. All given ULDs must be loaded. The objective is to minimize the sum of additional fuel cost caused by a suboptimal CG and the cost of reloading operations at the stop-over airport.

Dahmani and Krichen (2016) integrate weight and balance and palletization aspects in a two-level problem. They assign items to bins and bins to loading positions while maximizing the total weight and priority of loaded cargo. They present a multi-objective particle swarm optimization approach and compare their results to optimal MIP solutions for small instances. However, their model lacks most of the features that are presented in other recent papers. The item-to-bin assignment is solved as a one-dimensional packing problem based on item weights. The bin-to-position assignment is unrestricted and allows only a single aircraft configuration.

We give an overview of the mentioned papers, their covered features and objectives in Table 7.1. The majority of features that are present in our problem at hand have already been covered in the literature. However, none of the present papers simultaneously covers all aspects of our problem at hand.

7.2 The Weight and Balance Model

In this section, we define a mathematical model of the WBP, which we denote as the Weight and Balance Model (WBM) in the following. The goal of the WBM is to find an assignment of ULDs to loading positions. The objective is to minimize the fuel consumption, additional handling cost and improve the maneuverability of the aircraft. The basis for our formulation are the models given by Limbourg et al. (2012), Vancroonenburg et al. (2014), and Lurkin and Schyns (2015). In addition to them our model provides the following contributions:

- We consider a superset of features presented in the mentioned models: optional loading of the ULDs, cumulative weight limits for loading positions, overlapping loading positions, and ULD separation. This set of constraints has not been presented as a unified model yet.
- We present a first formulation of the WBP for an arbitrary number of legs. ULDs can be transported between any pair of stops along the flight.
- We add a new constraint that forces certain loading positions to be occupied to use other loading positions. This requirement stems from a practical safety consideration where the loading positions next to the cockpit must be loaded with a ULD to be allowed to use the positions behind them. In case of an accident the ULDs to the front should provide a cushion and protect the cockpit.
- We standardize the cumulative load constraints. In previous works, weight limits of individual loading positions, groups of adjacent positions and sections of the aircraft were specified as different constraints.

We start by defining the problem parameters. In the following sections we introduce the problem's constraints and objective. At last, we present a short overview of the model.

7.2.1 Parameters

The WBM operates on three sets of entities: the legs of the considered flight L, the loading positions of the aircraft P, and the transported ULDs U. In the following we use the respective lower case indices to represent single entities: $l \in L$, $p \in P$, $u \in U$.

Aircraft parameters

- $B_p \subseteq P$: set of loading positions that block loading operations on position p
- $\mathrm{BA}^{\mathrm{ac}} \in \mathbb{R}$: longitudinal balance arm of the aircraft at OEW
- $\mathrm{BA}_p^{\mathrm{lat}} \in \mathbb{R}$: lateral balance arm of loading position p
- $\mathrm{BA}_p^{\mathrm{lng}} \in \mathbb{R}$: longitudinal balance arm of loading position p
- $\mathrm{dist}_p \in \mathbb{R}_{\geq 0}$: distance between loading position p and the compartment door
- $\mathrm{dist}_{p,p'} \in \mathbb{R}_{\geq 0}$: distance between loading positions p and p'
- $\mathrm{f}_{n,p} \in \mathbb{R}$: weight factor of loading position p for constraint n
- $\mathrm{limit}_n \in \mathbb{R}$: cumulative weight limit of constraint n
- N: set of cumulative weight constraints in the aircraft ($n \in N$)
- $\mathrm{O} \subseteq P \times P$: set of overlapping loading positions ($\langle p, p' \rangle \in \mathrm{O}$)
- $R_p \subseteq P$: set of loading positions that position p depends on
- $\mathrm{w}^{\mathrm{OEW}} \in \mathbb{R}_{\geq 0}$: operating empty weight of the aircraft (OEW)

Flight parameters

- $\mathrm{a}_l^{\mathrm{cg}} \in \mathbb{R}$: cost of additional fuel for deviations from the optimal CG on leg l
- $\mathrm{a}_l^{\mathrm{dist}} \in \mathbb{R}$: cost of working time to move a ULD by one position after leg l
- $\mathrm{a}_l^{\mathrm{mi}} \in \mathbb{R}$: cost of additional fuel depending on the MI on leg l
- $\mathrm{a}_l^{\mathrm{on}} \in \mathbb{R}$: cost of one ULD loading operation after leg l
- $\mathrm{a}_l^{\mathrm{un}} \in \mathbb{R}$: cost of one ULD unloading operation after leg l
- $\mathrm{BA}_l^{\mathrm{fuel}} \in \mathbb{R}$: longitudinal balance arm of the fuel tanks on leg l
- $\mathrm{BA}_l^{\mathrm{MAX}} \in \mathbb{R}_{\geq 0}$: maximum allowed lateral CG imbalance on leg l
- $\mathrm{BA}_l^{\mathrm{OPT}} \in \mathbb{R}$: optimal longitudinal CG on leg l
- $\mathrm{FW}_l \in \mathbb{R}_{\geq 0}$: total weight of loaded fuel before starting leg l
- $\mathrm{MAX}_l \in \mathbb{R}$: maximum loadable total ULD weight on leg l
- $\mathrm{MAXCG}_l \in \mathbb{R}$: aft limit of the longitudinal CG on leg l
- $\mathrm{MINCG}_l \in \mathbb{R}$: forward limit of longitudinal CG on leg l

ULD parameters

- $\mathrm{a}_u^{\mathrm{off}} \in \mathbb{R}$: penalty when offloading ULD u

- $\text{legs}_u \subseteq L$: legs the ULD u will be on board
- $\text{loadable}_u \subseteq P$: loading positions available for ULD u
- $\min_{u,u'} \in \mathbb{R}_{\geq 0}$: minimum separation distance between ULD u and u'
- $\text{S} \subseteq U \times U$: set of ULDs to separate ($\langle u, u' \rangle \in \text{S}$)
- $\text{weight}_u \in \mathbb{R}_{\geq 0}$: total weight of ULD u

7.2.2 Decision variables

- $m_l^{\text{lng-}} \in \mathbb{R}_{\geq 0}$: forward deviation of rotational moment from optimal CG on leg l
- $m_l^{\text{lng+}} \in \mathbb{R}_{\geq 0}$: aft deviation of rotational moment from optimal CGG on leg l
- $m_l^{\text{mi}} \in \mathbb{R}_{\geq 0}$: moment of inertia of the loaded ULDs on leg l
- r_l^{dist}: total ULD move distance after leg l
- r_l^{on}: total number of loaded ULDs after leg l
- r_l^{un}: total number of unloaded ULDs after leg l
- $x_{u,p,l} \in \{0, 1\}$: 1 if ULD u is loaded at position p on leg l
- $x_u^{\text{off}} \in \{0, 1\}$: 1 if ULD u is offloaded

7.2.3 Constraints

In the following, we introduce the constraints that define a valid assignment of ULDs to loading positions. Along with the upcoming constraints, we define further auxiliary variables that are only relevant inside the defining constraint. We note that in practice the $x_{u,p,l}$ variables only need to be defined for $p \in \text{loadable}_u$ and $l \in \text{legs}_u$. To simplify the notation in the following we assume that non-defined $x_{u,p,l}$ are generously handled as zero values.

ULD assignment Each ULD u must be assigned to at most one loading position on each of its flown legs legs_u. Depending on the ULD type, weight, contour, or contained dangerous goods it might not be loadable at certain loading positions. We denote the feasible loading positions as loadable_u and limit the positions u takes to this set. If u is offloaded on one of its legs, indicated by x_u^{off}, it must be offloaded on any of its legs.

$$\sum_{p \in \text{loadable}_u} x_{u,p,l} = 1 - x_u^{\text{off}} \qquad u \in U; l \in \text{legs}_u \tag{7.1}$$

Loading position assignment On each flight leg l each loading position p can load at most one ULD:

$$\sum_{u \in U} x_{u,p,l} \leq 1 \qquad p \in P; l \in L \tag{7.2}$$

Overlapping positions As shown in Section 4 on aircraft configuration, several loading positions can overlap and mutually exclude each other from being used. We denote the set of pairwise mutually excluding loading positions as O and prohibit the simultaneous use of both on each leg l as:

$$\sum_{u \in U} (x_{u,p,l} + x_{u,p',l}) \leq 1 \qquad \langle p, p' \rangle \in \mathrm{O}; l \in L \tag{7.3}$$

Total weight On each flight leg l the aircraft has a certain maximum payload weight MAX_l. The sum of weights weight_u of all loaded ULDs must not exceed this limit. The payload limit depends on the flight leg, because of the respective fuel weight and airport parameters such as runway length or outside air temperature. Usually, an aircraft's maximum allowed takeoff weight is far lower then the sum of its individual empty weight, maximum possible payload weight, and maximum possible fuel weight. Therefore, there is a tradeoff between fuel and payload on each flight.

$$\mathrm{MAX}_l \geq \sum_{u \in U} \mathrm{weight}_u \sum_{p \in P} x_{u,p,l} \qquad l \in L \tag{7.4}$$

Cumulative weights Besides the total weight limits, each aircraft has structural weight limits for each loading position. Additional structural weight limits may apply for adjacent loading positions, which are often strictly less than the sum of the individual limits. For example, the two adjacent main deck positions DL and DR from Figure 2.8 might each be able to carry a ULD with 4.1 tonnes weight but the combined weight on both positions has to be less than 6.8 tonnes. Similar limits apply for groups of loading positions, for all loading positions in the full upper and lower deck, as well as for subsets of loading positions along the fuselage.

As a generalization, we introduce as set of cumulative weight constraints N on arbitrary subsets of loading positions. For each constraint $n \in N$ and its affected loading position p we introduce a parameter $\mathrm{f}_{n,p}$ representing the influence factor of the loading position. We limit the weighted sum of ULD weights on these loading positions by parameter limit_n.

$$\sum_{u \in U} \mathrm{weight}_u \sum_{p \in P} \mathrm{f}_{n,p} x_{u,p,l} \leq \mathrm{limit}_n \qquad n \in N; l \in L \tag{7.5}$$

Center of gravity The distribution of weight of the aircraft itself, its payload, and fuel influence the aircraft CG. To ensure the safety and maneuverability of the aircraft, its CG always has to stay within a defined range. In practice this range varies with the different phases of the flight. In our model we assume that the parameters are set in a way that the resulting range is feasible for all phases of the flight.

According to basic engineering mechanics, see FAA (2016), the CG can be calculated as the average rotational moment of all point masses weighted by their distances from a reference datum. In Figure 7.1 we present a sketch of the main aircraft parameters that influence the CG. The aircraft manufacturer usually defines a reference datum as well as the longitudinal and lateral distances between this datum and the CG of each relevant aircraft component. The distances are often referred to as balance arms. Relevant components include the empty aircraft itself, the ULD loading positions, and the fuel tanks.

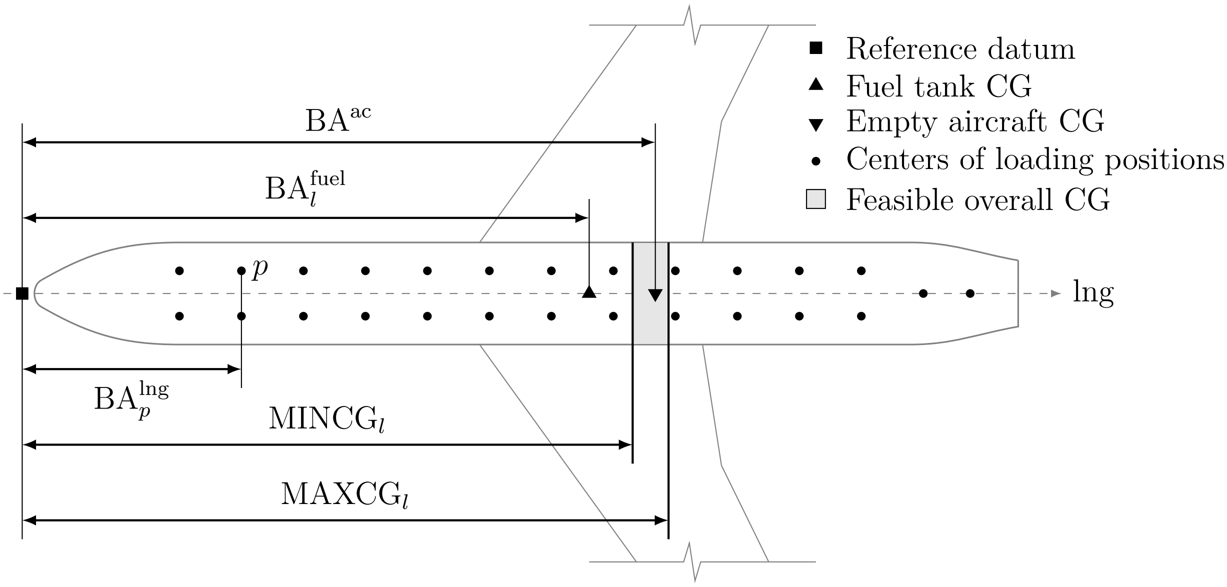


Figure 7.1: Illustration of the components influencing the longitudinal aircraft CG

The CG has to be considered in longitudinal and lateral dimension. The longitudinal CG influences the pitch of the aircraft, while the lateral CG influences the roll. As the longitudinal CG has considerably larger balance arms, it is usually of far greater importance than the lateral CG. Therefore, we will always refer to the longitudinal CG in the following, if not stated otherwise.

We introduce parameters for the longitudinal balance arms of the empty aircraft $\mathrm{BA}^{\mathrm{ac}}$, each loading position p as $\mathrm{BA}_p^{\mathrm{lng}}$, and the fuel as $\mathrm{BA}_l^{\mathrm{fuel}}$ on each leg l. The corresponding weight parameters are the operating empty weight of the aircraft $\mathrm{w}^{\mathrm{OEW}}$, the weight of fuel FW_l on leg l, and the total weight weight_u of ULD u. For each leg l we calculate the rotational moment of the loaded aircraft m_l as:

$$m_l = \mathrm{BA}^{\mathrm{ac}}\mathrm{w}^{\mathrm{OEW}} + \mathrm{BA}_l^{\mathrm{fuel}}\mathrm{FW}_l + \sum_{u \in U} \mathrm{weight}_u \sum_{p \in P} \mathrm{BA}_p^{\mathrm{lng}} x_{u,p,l} \qquad l \in L \qquad (7.6)$$

To determine the aircraft CG, we need to set the rotational moment in relation to the total weight of the aircraft. We calculate the total weight at the beginning of each leg l as:

$$w_l = \mathrm{w}^{\mathrm{OEW}} + \mathrm{FW}_l + \sum_{u \in U} \mathrm{weight}_u \sum_{p \in P} x_{u,p,l} \qquad l \in L \qquad (7.7)$$

The CG on leg l can now be calculated as m_l/w_l. On each leg the CG has a forward limit MINCG_l and aft limit MAXCG_l. To avoid the division of variables we limit the rotational moments and not the CG itself:

$$\mathrm{MINCG}_l w_l \leq m_l \leq \mathrm{MAXCG}_l w_l \qquad l \in L \qquad (7.8)$$

The lateral CG of the empty aircraft, the fuel weights, and the optimal lateral CG are usually all at the center of the aircraft. Therefore, we only look at the payload for calculating the lateral CG, or rather the payload imbalance. Let $\mathrm{BA}_p^{\mathrm{lat}}$ denote the lateral balance arm of loading position p. We introduce the auxiliary variables $m_l^{\text{lat-}}$ and $m_l^{\text{lat+}}$ indicating a lateral moment to the left or right of the aircraft. We calculate the moment of the lateral imbalance as:

$$\sum_{p \in P} \mathrm{weight}_u \sum_{u \in U} \mathrm{BA}_p^{\mathrm{lat}} x_{u,p,l} = m_l^{\text{lat+}} - m_l^{\text{lat-}} \qquad l \in L \qquad (7.9)$$

The pilots may be able to adjust small lateral imbalances by pumping fuel between the wing tanks. However, the imbalance should be small to maintain flexibility. Therefore, we limit the imbalance on each leg l to $\mathrm{BA}_l^{\mathrm{MAX}}$ as:

$$m_l^{\text{lat+}} + m_l^{\text{lat-}} \leq \mathrm{BA}_l^{\mathrm{MAX}} w_l \qquad l \in L \qquad (7.10)$$

Moment of inertia Weight limits and an ideal CG are not sufficient to describe a good loading pattern. As an example, a good CG might be reached by placing one ULD on the front and one at the back of the aircraft and nothing in between. While this might be feasible with respect to weight and CG considerations, it would result in a higher than necessary MI of the aircraft. With a higher MI the aircraft becomes less maneuverable as more power is needed to raise or lower the aircraft's pitch. Consequently, we want to minimize the MI by placing heavier ULDs closer to the aircraft CG.

To simplify the formulation we use the optimal CG of the aircraft $\mathrm{BA}_l^{\mathrm{OPT}}$ instead of the actual CG, because the former is a constant while the latter is a decision variable. This is reasonable because we expect the actual CG to be close to the optimum as this is part of our objective function.

We introduce an auxiliary variable m_l^{mi} to approximate the MI added by the loaded ULDs on each leg l as:

$$m_l^{\mathrm{mi}} = \sum_{u \in U} \mathrm{weight}_u \sum_{p \in P} (\mathrm{BA}_l^{\mathrm{OPT}} - \mathrm{BA}_p^{\mathrm{lng}})^2 x_{u,p,l} \qquad l \in L \qquad (7.11)$$

Loading dependencies In practice a loading position might only be usable, if other loading positions are also in use. The most common use case is that the loading positions right after the cockpit should be occupied on flights where ULDs are loaded on the loading positions behind them. In case of an accident, the ULDs to the front provide a cushion and reduce the impact of broken loose ULDs from behind, to protect the crew and reduce damage to the aircraft.

For each loading position p we introduce a set of loading positions R_p the loadability of p depends on. On each leg l at least one position of R_p must be occupied to be allowed to use p.

$$\sum_{u \in U} x_{u,p,l} \leq \sum_{p' \in R_p} \sum_{u \in U} x_{u,p',l} \qquad p \in P; l \in L \qquad (7.12)$$

ULD separation The same item incompatibility constraints that apply during palletization also apply during flight. Therefore, two ULDs that contain incompatible items must be loaded at positions that are a certain minimum distance apart.

We introduce the set S of conflicting ULDs and a parameter $\min_{u,u'}$ representing the minimum required distance between the pairs of ULDs. Furthermore, let $\mathrm{dist}_{p,p'}$ denote the distance in the aircraft between loading positions p and p'. On each leg l we restrict the applicable loading positions as:

$$\sum_{\substack{p' \in P: \\ \mathrm{dist}_{p,p'} < \min_{u,u'}}} x_{u,p,l} + x_{u',p',l} \leq 1 \qquad \langle u, u' \rangle \in \mathrm{S}; p \in P; l \in L \tag{7.13}$$

Loading operations Cargo flights often contain multiple legs, at which only a part of the ULDs are unloaded and others are loaded. As the loading positions are generally only accessible in a first-in-last-out style, the assignment of ULDs to loading positions influences the handling effort at each stop. ULDs can be reloaded between the legs, if this is beneficial, i.e., it reduces the fuel consumption, or allows to load more cargo.

We consider two types of handling operations in the following: unloading a ULD and loading a ULD. We introduce the auxiliary variable $q_{l,p}$ to indicate if the loading position p needs to be cleared at the stop after leg l. A position needs to be cleared if a ULD is removed from this position, a ULD is loaded at the position, or if the position is on the path B_p of another position that needs to be cleared. For all legs but the last, denoted as L', we add:

$$q_{l,p} \geq x_{u,p,l} - x_{u,p,l+1} \qquad u \in U; p \in P; l \in L' \tag{7.14}$$
$$q_{l,p} \geq x_{u,p,l+1} - x_{u,p,l} \qquad u \in U; p \in P; l \in L' \tag{7.15}$$
$$q_{l,p} \leq q_{l,p'} \qquad p \in P; p' \in B_p; l \in L' \tag{7.16}$$
$$q_{l,p} \leq q_{l,p'} \qquad p \in P; p' \in B_p; l \in L' \tag{7.17}$$

Furthermore, we add auxiliary variables $r^{\mathrm{un}}_{l,p}$ and $r^{\mathrm{on}}_{l,p}$ to indicate if a real handling operation takes place. These are set to one if a ULD at this position p is loaded or unloaded at the stop after leg l.

$$r^{\mathrm{un}}_{l,p} \geq q_{l,p} + \sum_{u \in U} x_{u,p,l} - 1 \qquad p \in P; l \in L \tag{7.18}$$
$$r^{\mathrm{on}}_{l,p} \geq q_{l,p} + \sum_{u \in U} x_{u,p,l+1} - 1 \qquad p \in P; l \in L \tag{7.19}$$

To sum up the overall loading effort after leg l we use three variables representing the total number of offloaded ULDs r^{un}_{l}, the total number of loaded ULDs r^{on}_{l}, and the distance of moving ULDs out of and into the aircraft r^{dist}_{l}.

$$r^{\mathrm{un}}_{l} = \sum_{p \in P} r^{\mathrm{un}}_{l,p} \qquad l \in L \tag{7.20}$$
$$r^{\mathrm{on}}_{l} = \sum_{p \in P} r^{\mathrm{on}}_{l,p} \qquad l \in L \tag{7.21}$$
$$r^{\mathrm{dist}}_{l} = \sum_{p \in P} \mathrm{dist}_p (r^{\mathrm{un}}_{l,p} + r^{\mathrm{on}}_{l,p}) \qquad l \in L \tag{7.22}$$

7.2.4 Objective

There are four factors that make up a good weight and balance solution. First, all ULDs should be loaded. Second, the longitudinal deviation from the optimal CG should be minimized. Third, the MI of the payload should be small. And last, there should be as few handling operations as possible at each stop-over airport. In our objective we weight these factors by their induced cost. First, we describe how to determine the cost parameters.

In some setups it might not be possible to load all ULDs. In these cases the least important ULDs should be left behind. Let $\mathrm{a}_u^{\mathrm{off}}$ denote the penalty cost of offloading ULD u. In practice this can be derived from the items loaded into the ULD and the respective contracts with the shippers.

As mentioned before, the CG also influences the fuel consumption of the aircraft during flight. Therefore, we estimate the additional fuel consumption attributed to the deviation from the optional CG on each flight leg. Let $\mathrm{BA}_l^{\mathrm{OPT}}$ denote the optimal point at which the aircraft consumes the least fuel on leg l. The variable w_l denotes the total weight of the aircraft and m_l its total rotational moment at the beginning of leg l. Let the variables $m_l^{\text{lng-}}$ and $m_l^{\text{lng+}}$ denote the absolute deviation from the moment of the optimal CG to the front and aft. We calculate them as:

$$\mathrm{BA}_l^{\mathrm{OPT}} w_l - m_l = m_l^{\text{lng+}} - m_l^{\text{lng-}} \qquad \forall l \in L \tag{7.23}$$

To transform this into cost, we add parameter $\mathrm{a}_l^{\mathrm{cg}}$ for each leg l representing the additional fuel cost for each unit the moment deviates from the optimum. This cost factor mostly depends on the aircraft type, the expected flight duration and the fuel price.

Nearly the same applies to the MI. A higher MI increases the necessary force to change the aircraft's direction of flight. This leads to higher drag and additional fuel consumption. To transform the MI into cost, we weight it by a parameter $\mathrm{a}_l^{\mathrm{mi}}$ for each leg l.

To determine the cost of loading operations, we introduce parameters that represent the cost at the destination airport of leg l for one unloading operation $\mathrm{a}_l^{\mathrm{un}}$, one loading operation $\mathrm{a}_l^{\mathrm{on}}$, and the cost of working time when moving a ULD by one loading position $\mathrm{a}_l^{\mathrm{dist}}$. These costs are usually well known to the airline and defined in the contracts with the respective ground handling agents or airports.

To sum up, we define the total objective function for the WBM to minimize the sum of costs stemming from offload penalties, additional fuel consumption on all flight legs, a penalty for the MI, as well as the handling efforts for ULD unloading, loading, and moving efforts at all visited airports along the flight:

$$\begin{aligned} \text{minimize} \quad & \sum_{u \in U} \mathrm{a}_u^{\mathrm{off}} x_u^{\mathrm{off}} + \sum_{l \in L} \mathrm{a}_l^{\mathrm{cg}} (m_l^{\text{lng-}} + m_l^{\text{lng+}}) \\ & + \mathrm{a}_l^{\mathrm{mi}} m_l^{\mathrm{mi}} + \mathrm{a}_l^{\mathrm{un}} r_l^{\mathrm{un}} + \mathrm{a}_l^{\mathrm{on}} r_l^{\mathrm{on}} + \mathrm{a}_l^{\mathrm{dist}} r_l^{\mathrm{dist}} \end{aligned} \tag{7.24}$$

7.2.5 Model overview

After introducing all components of the WBM in the last sections, we now give a consolidated view of the full model:

$$\text{minimize} \quad \sum_{u\in U} \mathrm{a}^{\mathrm{off}}_u x^{\mathrm{off}}_u + \sum_{l\in L} \mathrm{a}^{\mathrm{cg}}_l (m^{\text{lng-}}_l + m^{\text{lng+}}_l) \\ +\mathrm{a}^{\mathrm{mi}}_l m^{\mathrm{mi}}_l + \mathrm{a}^{\mathrm{un}}_l r^{\mathrm{un}}_l + \mathrm{a}^{\mathrm{on}}_l r^{\mathrm{on}}_l + \mathrm{a}^{\mathrm{dist}}_l r^{\mathrm{dist}}_l \tag{7.25}$$

subject to

$$x^{\mathrm{off}}_u = 1 - \sum_{p\in \mathrm{loadable}_u} x_{u,p,l} \qquad u\in U; l\in \mathrm{legs}_u \tag{7.26}$$

$$1 \geq \sum_{u\in U} x_{u,p,l} \qquad p\in P; l\in L \tag{7.27}$$

$$1 \geq \sum_{u\in U} (x_{u,p,l} + x_{u,p',l}) \qquad \langle p,p'\rangle \in \mathrm{O}; l\in L \tag{7.28}$$

$$\mathrm{MAX}_l \geq \sum_{u\in U} \mathrm{weight}_u \sum_{p\in P} x_{u,p,l} \qquad l\in L \tag{7.29}$$

$$\mathrm{limit}_n \geq \sum_{u\in U} \mathrm{weight}_u \sum_{p\in P} \mathrm{f}_{n,p} x_{u,p,l} \qquad n\in N; l\in L \tag{7.30}$$

$$m_l = \begin{array}{l} \mathrm{BA}^{\mathrm{ac}}\mathrm{w}^{\mathrm{OEW}} + \mathrm{BA}^{\mathrm{fuel}}_l \mathrm{FW}_l \\ + \sum_{u\in U} \mathrm{weight}_u \sum_{p\in P} \mathrm{BA}^{\mathrm{lng}}_p x_{u,p,l} \end{array} \qquad l\in L \tag{7.31}$$

$$w_l = \mathrm{w}^{\mathrm{OEW}} + \mathrm{FW}_l + \sum_{u\in U} \mathrm{weight}_u \sum_{p\in P} x_{u,p,l} \qquad l\in L \tag{7.32}$$

$$\mathrm{BA}^{\mathrm{OPT}}_l w_l = m_l + m^{\text{lng+}}_l - m^{\text{lng-}}_l \qquad l\in L \tag{7.33}$$

$$m_l \leq \mathrm{MAXCG}_l w_l \qquad l\in L \tag{7.34}$$

$$m_l \geq \mathrm{MINCG}_l w_l \qquad l\in L \tag{7.35}$$

$$m^{\text{lat+}}_l = \sum_{p\in P} \mathrm{weight}_u \sum_{u\in U} \mathrm{BA}^{\mathrm{lat}}_p x_{u,p,l} + m^{\text{lat-}}_l \qquad l\in L \tag{7.36}$$

$$\mathrm{BA}^{\mathrm{MAX}}_l w_l \geq m^{\text{lat+}}_l + m^{\text{lat-}}_l \qquad l\in L \tag{7.37}$$

$$m^{\mathrm{mi}}_l = \sum_{u\in U} \mathrm{weight}_u \sum_{p\in P} (\mathrm{BA}^{\mathrm{OPT}}_l - \mathrm{BA}^{\mathrm{lng}}_p)^2 x_{u,p,l} \qquad l\in L \tag{7.38}$$

$$\sum_{u\in U} x_{u,p,l} \leq \sum_{p'\in R_p} \sum_{u\in U} x_{u,p',l} \qquad p\in P; l\in L \tag{7.39}$$

$$1 \geq \sum_{\substack{p'\in P: \\ \mathrm{dist}_{p,p'} < \mathrm{min}_{u,u'}}} x_{u,p,l} + x_{u',p',l} \qquad \langle u,u'\rangle \in \mathrm{S}; p\in P; l\in L \tag{7.40}$$

$$q_{l,p} \geq x_{u,p,l} - x_{u,p,l+1} \qquad u\in U; p\in P; l\in L \tag{7.41}$$

$$q_{l,p} \geq x_{u,p,l+1} - x_{u,p,l} \qquad u\in U; p\in P; l\in L \tag{7.42}$$

$$q_{l,p} \leq q_{l,p'} \qquad p\in P; p'\in B_p; l\in L \tag{7.43}$$

$$q_{l,p} \leq q_{l,p'} \qquad p\in P; p'\in B_p; l\in L \tag{7.44}$$

$$r^{\mathrm{un}}_{l,p} \geq q_{l,p} + \sum_{u\in U} x_{u,p,l} - 1 \qquad p\in P; l\in L \tag{7.45}$$

$$r^{\mathrm{on}}_{l,p} \geq q_{l,p} + \sum_{u\in U} x_{u,p,l+1} - 1 \qquad p\in P; l\in L \tag{7.46}$$

$$r^{\mathrm{dist}}_l = \sum_{p\in P} \mathrm{dist}_p (r^{\mathrm{un}}_{l,p} + r^{\mathrm{on}}_{l,p}) \qquad l\in L \tag{7.47}$$

$$m_l, m_l^{\text{lat-}}, m_l^{\text{lat+}}, m_l^{\text{lng-}}, m_l^{\text{lng+}}, m_l^{\text{mi}}, w_l \in \mathbb{R}_{\geq 0} \qquad l \in L \quad (7.48)$$

$$r_l^{\text{dist}} \in \mathbb{R}_{\geq 0} \qquad l \in L \quad (7.49)$$

$$q_{l,p}, r_{l,p}^{\text{un}}, r_{l,p}^{\text{on}} \in \{0,1\} \qquad l \in L; p \in P \quad (7.50)$$

$$x_{u,p,l} \in \{0,1\} \qquad u \in U; p \in P; l \in L \quad (7.51)$$

$$x_u^{\text{off}} \in \{0,1\} \qquad u \in U \quad (7.52)$$

The objective, given in Equation (7.25), is to minimize the sum of additional costs of the flight. Equation (7.26) models the ULD assignment, Equation (7.27) the loading position assignment, Equation (7.28) the overlapping positions, Equation (7.29) the total weight limit of the aircraft, Equation (7.30) the cumulative weight limits of loading positions, Equations (7.31) to (7.37) the center of gravity, Equation (7.38) the moment of inertia, Equation (7.39) the loading dependencies, Equation (7.40) the ULD separation, and Equations (7.41) to (7.47) the loading operations between the flight legs.

7.3 Summary

In this chapter, we introduced the Weight and Balance Problem (WBP) in detail, which is to assign ULDs to loading positions in the aircraft for every leg of a flight while respecting the aircraft's structural limitations and reducing the costs of required fuel and reloading operations. Furthermore, we analyzed the related literature and presented a formal model, denoted as Weight and Balance Model (WBM), of the problem features.

There are a few works dealing with aircraft weight and balance in the literature. Most of them focus solely on maximizing the load and consider flights with only a single leg. The closest model we found, given by Lurkin and Schyns (2015), describes flights with two legs and covers five of the eight requirements we identified in practice.

We combined all the features found in the literature into our model and extended it for an arbitrary number of flight legs. Furthermore, we added a new practical constraint that describes the conditional usage of loading positions, depending on the usage of other loading positions. Furthermore, we standardized the way of specifying cumulative load constraints.

8 Benchmark instances

To evaluate and compare solution approaches to the ACLPP one needs a set of benchmark instances. Unfortunately, there are no suitable instances available yet, neither synthetic ones nor instances stemming from practice. We mainly attribute this to the fact that the ACLPP has not been considered as a whole in the past. But even in papers dealing with parts of the problem, like Mongeau and Bes (2003), Rong and Grunow (2009), or Lurkin and Schyns (2015), the authors did not publish any of the problem instances they worked with. From the airline's point of view, this is comprehensible as booking data are seen as a trade secret. They would allow insights into the airline's state of business in a highly competitive market. Furthermore, the concrete shipment data could even allow to identify the shippers and draw conclusions about their state of business. The same issues apply in our case. For the development and evaluation of our approach, we work with original datasets provided by our industrial partner, but we cannot publish them.

To make our results reproducible, we also create a set of publishable synthetic instances. Simply creating random instances would not be of much use, as the ACLPP exhibits a lot of structure that can be exploited by a solution approach. Hence, the instances need to be carefully designed, to keep a maximum of the typical problem structure of the ACLPP in practice. We do this by analyzing the provided real world data, we extract their key characteristics and derive new synthetic instances from the given booking data in a suitable way.

This chapter is structured as follows. First, we introduce the ACLPP data model in Section 8.1. It combines the models and parameters described in Chapters 4 to 7. Afterwards, we present an analysis of the typical properties and problem structure of air cargo bookings found at our industrial partner in Section 8.2. We present the actual datasets used for our evaluation in Section 8.3. Finally, we describe a set of performance indicators to compare solutions to the ACLPP and to relate our results to practice in Section 8.4.

8.1 The ACLPP data model

To describe our data model of the ACLPP, we differentiate three types of relevant information: master data, booking data, and model decisions. In Figure 8.1 we give an overview of the data model, presenting all entities and their relations.

The master data describe all the properties of the aircraft, ULD types, airline objectives, regulatory requirements, terminal workstations, and available workers. We collected and validated these data with great effort to evaluate our solution approaches in a realistic setup. Most of the parameters are similar for many airlines in the industry.

The booking data are specific to each flight. All flights together determine the workload of the air cargo terminal. Each flight's booking data consists of a list of shipments. A

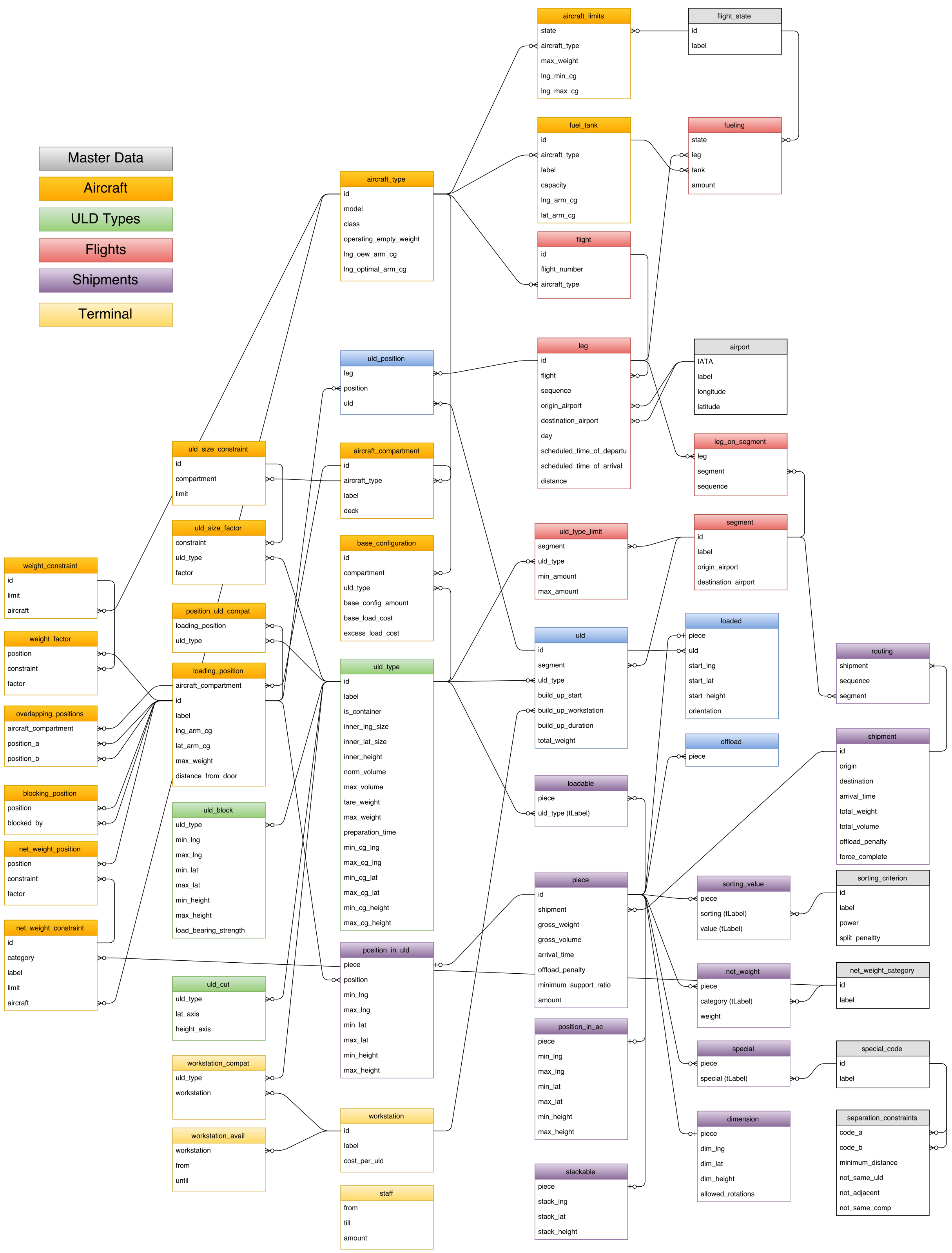


Figure 8.1: Data model representing all master data (yellow, green), flight data (red), booking data (violet) and decisions (blue) present in the ACLPP.

shipment represents a single booking of a shipper with a certain origin, destination, product, and the expected arrival time at the export terminal. A product defines the service level of the shipment, typically standard or express. Shipments can contain multiple different items. Each item type is specified by its dimensions, weight, offload penalty, and handling requirements. The latter contain information about the allowed item rotations, load-bearing strength, and dangerous goods with their respective net weights. Often, a booking contains multiple items of an item type. Until the airline gets hold of the items, some parameters are still uncertain. The item dimensions and weight might slightly change and it might turn out that items are not stackable or not rotatable although stated differently by the shipper.

The model decisions correspond to the subproblems of the ACLPP. They contain the set of used ULDs with their assigned build-up workstation, build-up time, and loading position inside the aircraft on each flight leg. For each item of each shipment, the decisions contain the information if the item is loaded or offloaded and, if loaded, the assigned ULD as well as the placement position and rotation inside the ULD.

8.2 Analysis and insights into real world instances

Before we introduce our benchmark instances in the next section, we first analyze a collection of real booking data provided by our industrial partner. The main purpose of our analysis is to get an impression of the typical problem structure and the order of magnitude of the main parameters.

The basis for this analysis is a representative set of booking data of more than 400 flights with over 35,000 real shipments from the year 2014. The concrete characteristics of the data will most likely be different for other companies and network structures. An obvious reason is the varying composition of transported commodities between different countries and continents. The Boeing World Air Cargo Forecast (Crabtree et al., 2014) for example states that more than 50 percent of the shipments from Europe to the Americas consist of chemicals and machinery. On the other hand, shipments from Asia to Europe contain more than 50 percent computers and apparel. Shipments containing the latter commodities are typically lighter, less bulky, and more fragile. Therefore, loading problems between Asia and Europe will most likely be of a different nature than between Europe and the Americas. For further insights into the structure of shipments between different regions of the world, we refer to the Boeing report mentioned above.

We start our analysis by looking at the typical utilization of flights, followed by the structure of shipments. Afterwards, we investigate the item dimensions, weights, and present transfer times in the terminal.

Flight legs A flight might consist of multiple successive legs. The larger the number of legs the more complex the aircraft configuration and weight and balance decisions, because ULDs with different origins or destinations must be arranged in the aircraft. Figure 8.2 presents the distribution of legs per flight in the typical flight plan of our partner. Less than one third of the flights had only one leg. The majority needed to deal with multiple legs and transport segments.

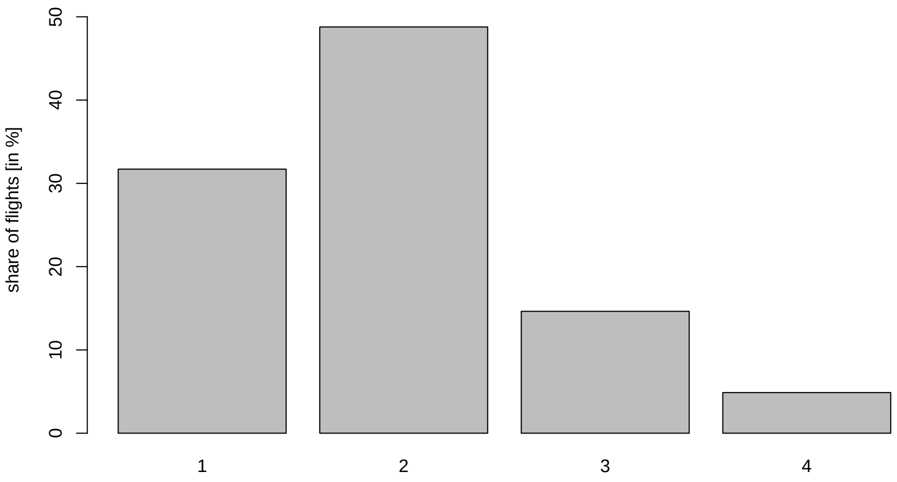


Figure 8.2: Distribution of the number of legs per flight

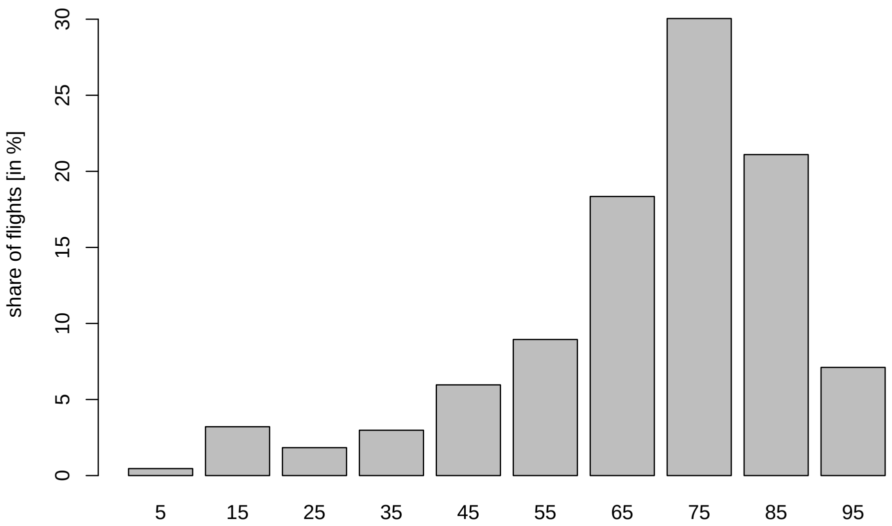


Figure 8.3: Distribution of weight utilization among flights (in %)

Weight utilization According to practitioners, most flights are restricted by volume and are not operated near the aircraft weight limit. The analyzed data match this statement. Figure 8.3 presents the distribution of the weight utilization on flights in the analyzed data. Consequently, weight limits need to be considered in practice but the total weight of all items will most probably not make the problems hard to solve.

Volume utilization Although most cargo flights are characterized as restricted by volume the typical volume utilization is far away from 100 percent. Figure 8.4 presents the respective distribution found in the analyzed data. On most flights the physical volume capacity could only be filled up to 60 or 70 percent. We attribute this to the heterogeneity of the cargo, stacking, stability, and other handling requirements. However, in retrospect, it is hard to identify a single reason for unused capacity. There might not have been

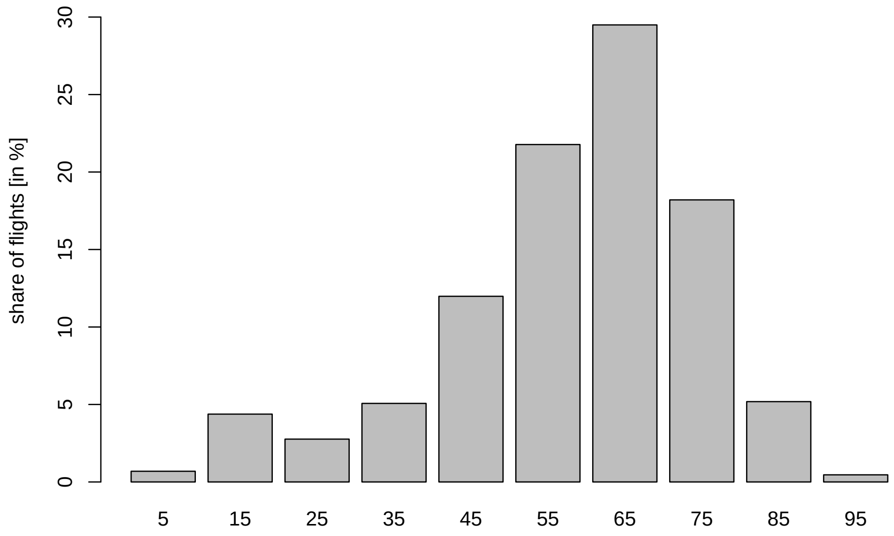


Figure 8.4: Distribution of volume utilization on flights (in %)

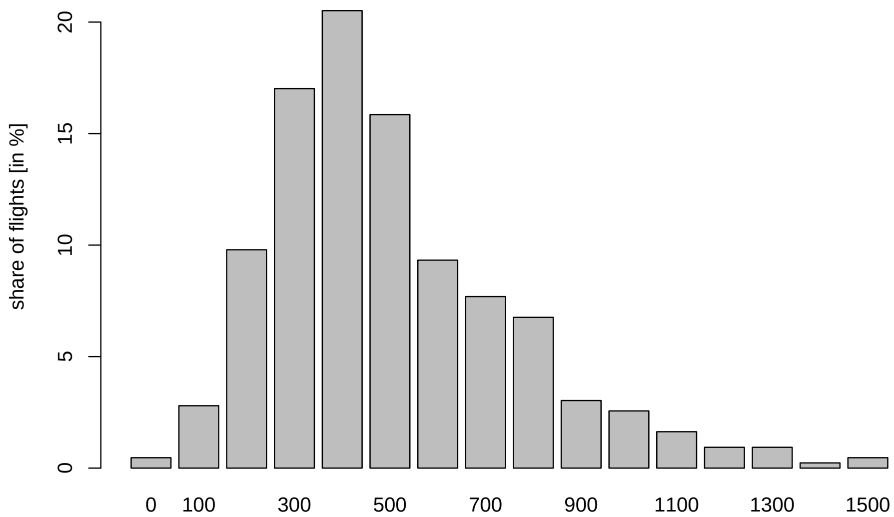


Figure 8.5: Distribution of the number of items per segment

enough bookings, bookings might have been canceled at short notice, some shipments might have arrived too late, the cargo might have had adverse dimensions or stacking properties, or it might have been packed in an inefficient way.

Items per segment As packing problems are known to be NP-complete, the number of items to pack strongly influences the solution complexity. In Figure 8.4 we analyze the distribution of items per transport segment found in the analyzed data. On average, each segment contained 349 items. Around 3.8 percent of the segments had more than 1,000 individual items. The maximum number we found on any segment was close to 2,000 items.

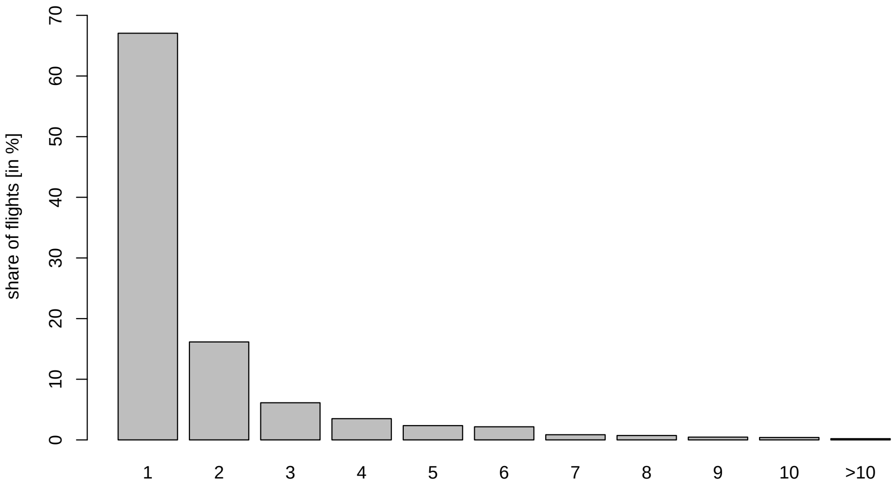


Figure 8.6: Distribution of the number of item groups per shipment

Length ± 5 *cm*	120	40	30	60	80
Share of items	19.3%	12.0%	11.9%	8.9%	8.0%

Table 8.1: Most frequent item lengths

Item groups Air cargo shipments might contain different items. We denote all items in a shipment with the same dimensions, weight, and handling requirements as an item group. The empirical distribution of the number of distinct groups per shipment is given in Figure 8.6. The majority of shipments, around 67 percent, contained one item group only and shipments with more item types occurred less frequently.

Item frequency Shipments might also contain multiple identical items. In the analyzed data only 11 percent of the items were unique in a shipment and around 90 percent occurred less than 20 times. But, there have also been significant outliers with more than 100 equal items. The distribution of item group sizes up to 20 items is given in Figure 8.7. Generally, individual items appeared most often and higher multiples less frequently. Interestingly, item groups with an even multiple occurred a little more often than their odd neighbors. In total even multiples account for 52 percent of the items.

Item length and width In the following, we denote the longer horizontal edge as length and the shorter as width of an item. Figures 8.8 and 8.9 show the empirical distribution of item lengths and widths. In terms of the length, there was a peak at 120 cm, which is the length of a standard EUR-pallet (see Section 2.2). This length alone accounted for 13.4 percent of all items. Further typical items lengths and their frequencies are shown in Table 8.1. Multiples of these lengths also fit well on a standard EUR-pallet. A similar effect shows with respect to the item width. Around 11.6 percent of all items are 80 cm wide, which is the width of a standard EUR-pallet.

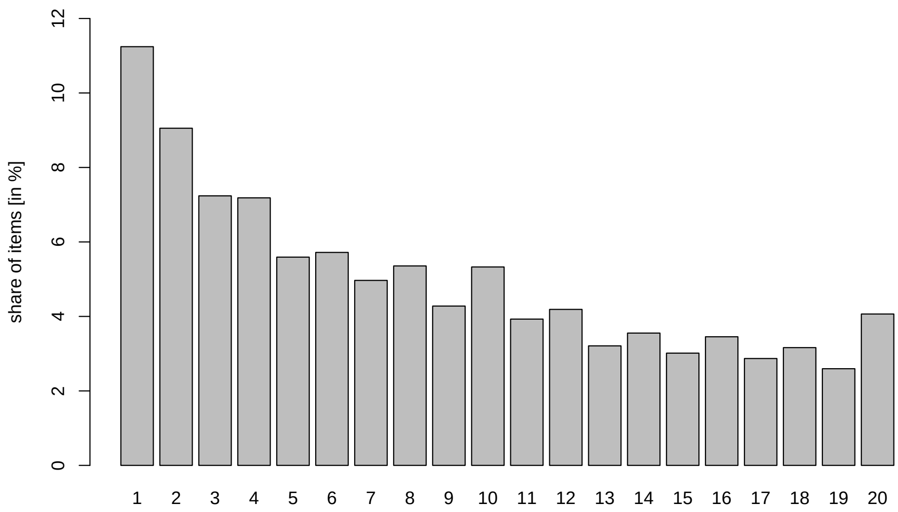


Figure 8.7: Distribution of item frequencies in all shipments (to be read as: around 9 percent of all items belong to a group of 2 items)

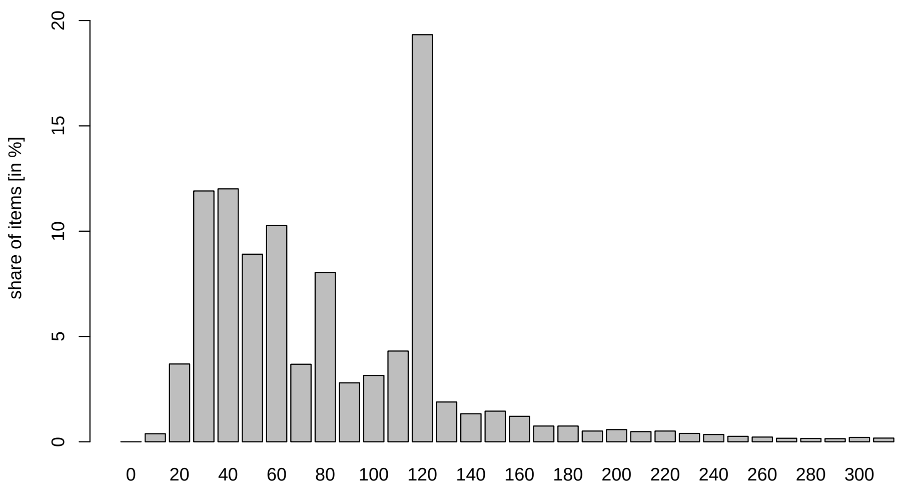


Figure 8.8: Distribution of item lengths (in cm)

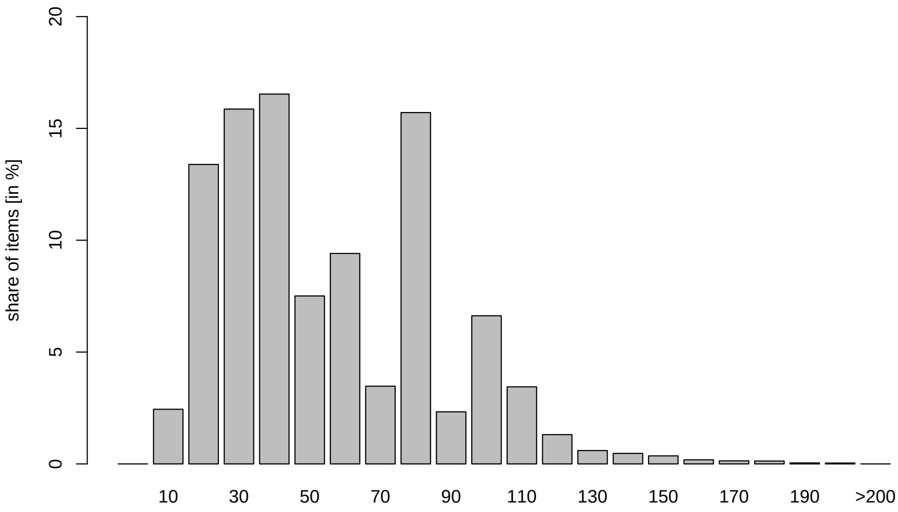


Figure 8.9: Distribution of item widths (in cm)

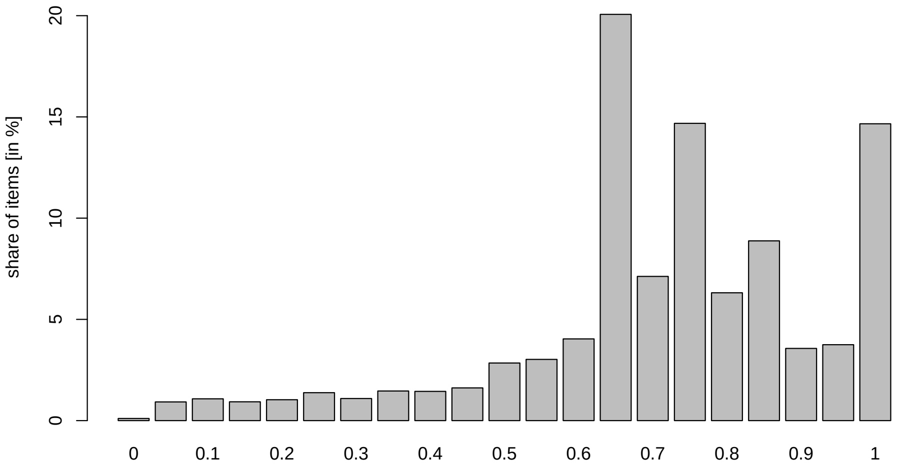


Figure 8.10: Distribution of width-to-length aspect ratio

Aspect ratio We define the aspect ratio by the ratio between the horizontal edges of the items, i.e., the item width divided by the item length. Figure 8.10 shows the empirical distribution of the aspect ratio. The most frequent ratio was 0.667, accounting for more than 20 percent of all items. Further typical aspect ratios are shown in Table 8.2.

Not surprisingly the most frequent item length (120 cm) and width (80 cm) resemble the most frequent aspect ratio (2:3). Around 10 percent of the items had this base size.

Item height Figure 8.11 shows the empirical distribution of the item heights. The most frequent heights were 20 cm to 40 cm and the mean height was 57 cm. Comparing this to a maximum height of up to 300 cm for main deck ULDs, some stacking seems to be necessary, but there will hardly be more than 10 items on top of each other. In the analyzed dataset, item heights tend to be higher for larger items. Therefore, we also look at the length-to-height ratio of the items. Figure 8.12 shows the corresponding

Ratio	0.667	0.750	1.000	0.833
Alias	2:3	3:4	1:1	5:6
Share of items	20.1%	14.7%	14.7%	8.9%

Table 8.2: Popular width-to-length aspect ratios of items

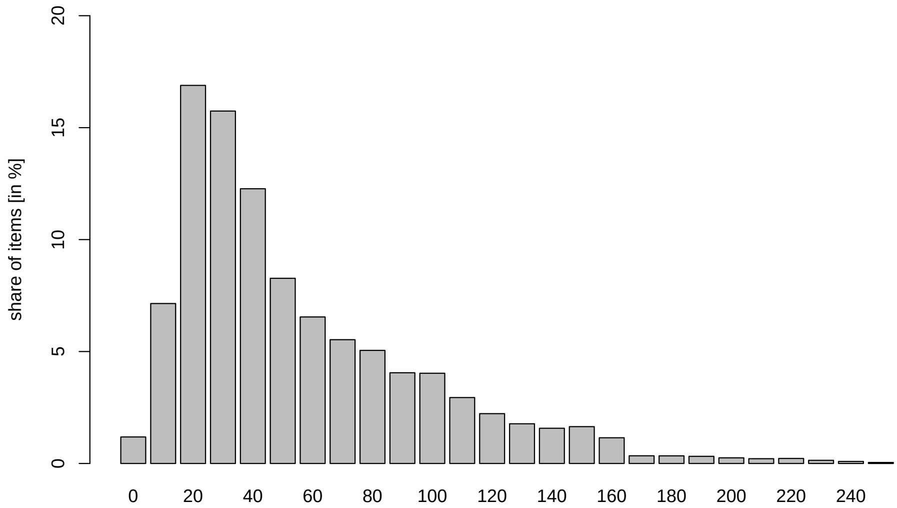


Figure 8.11: Distribution of item heights (in cm)

distribution. The most frequent ratio was 1.5, accounting for 8.6 percent of the items. Furthermore, integer multiples occurred a bit more often than the fractional values close-by. However, it is not clear if this is a real effect or just caused by shippers rounding item dimensions.

Item weight In the analyzed booking data item weight correlates with item volume. Therefore, we consider the item density in the following. The corresponding distribution is shown in Figure 8.13. The most frequent item densities were between 100 and 150 $\mathrm{kg} \cdot \mathrm{m}^{-3}$. From Table 2.4 we know that typical cargo aircraft have a capacity ratio of around 180 $\mathrm{kg} \cdot \mathrm{m}^{-3}$. Therefore, we expect the loading problems to be more constrained by volume than by weight and thus the related three-dimensional packing problems are non-trivial to solve. This observation is in line with our findings regarding the weight utilization above.

Transfer times Each shipment must be handled in the time between its arrival at the airport and the scheduled departure time of its outgoing flight. We denote this interval as transfer time. In general, a shorter transfer time provides less options to combine items onto ULDs and thus makes it harder to find good packing solutions. In the following, we analyze the transfer time of shipments that arrived via an inbound flight or RFS. In our analysis, we omit shipments that started their journey at the terminal due to missing arrival data. According to practitioners, these shipments typically arrive rather late, most of them less than 12 hours before the flight departure.

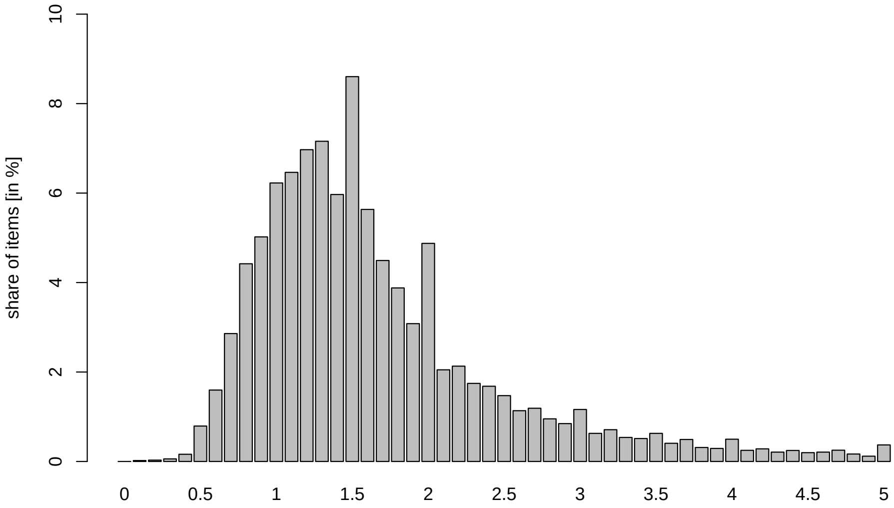


Figure 8.12: Distribution of height-to-length aspect ratio

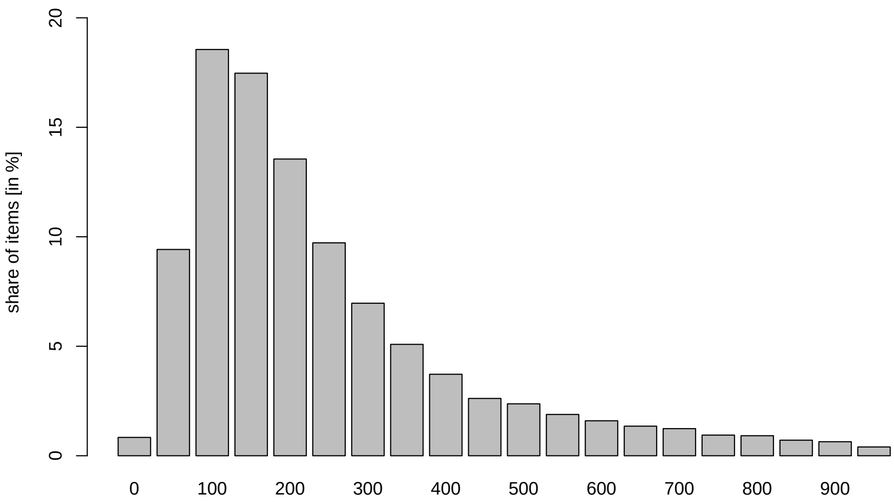


Figure 8.13: Distribution of item density (in $\mathrm{kg} \cdot \mathrm{m}^{-3}$)

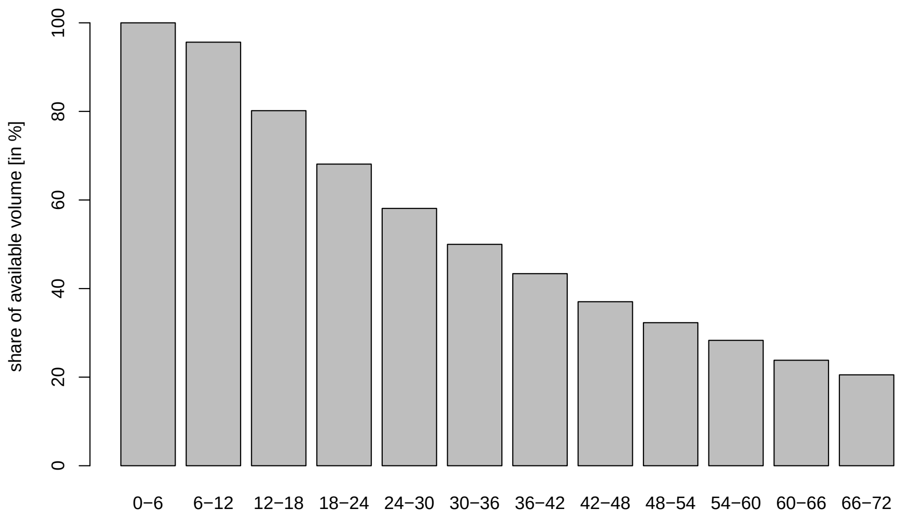


Figure 8.14: Item availability of standard shipments in hours before departure

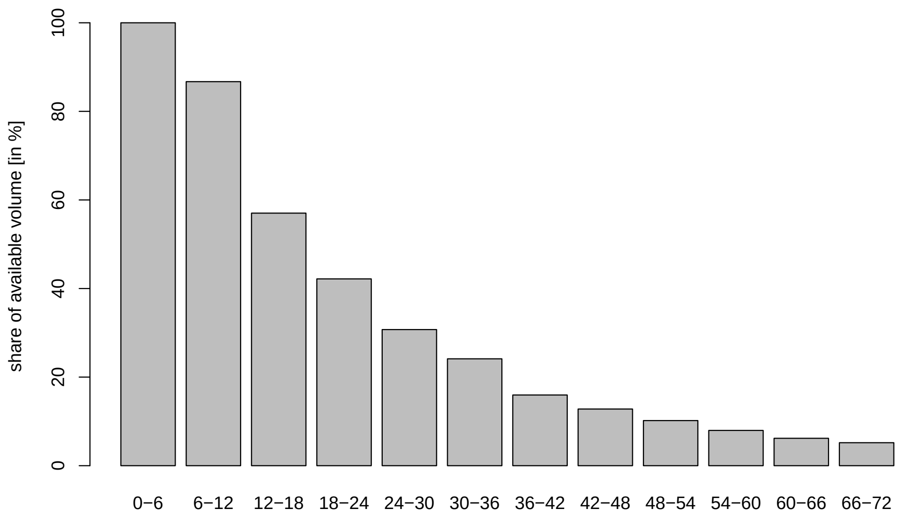


Figure 8.15: Item availability of express shipments in hours before departure

We consider standard and express shipments separately. Express shipments are sold at a premium and allow the shipper to book connections with a shorter transfer time. For both product types, the available cargo volume within a certain transfer time is shown in Figures 8.14 and 8.15. As one would expect, express shipments generally exhibit a shorter transfer time than standard shipments. Accordingly, around 42 percent of the standard volume and 77 percent of the express volume arrives during the last 24 hours before departure.

We note that the available time for the actual build-up of the cargo is significantly shorter than the transfer time. First, the inbound ULDs must be unloaded from the aircraft, transported to the terminal and broken down. The same applies to the aircraft loading after the build-up. Accordingly, the handling time window especially for express shipments can become really narrow at times.

8.3 Datasets

We present two datasets to evaluate our approach. The first is a set of 89 real historic all-cargo flights (dataset A) provided by our industrial partner. The second set contains 82 synthetic flights (dataset B) that we created based on real-world booking data. Dataset B is publicly available under `https://github.com/fbrandt/ACLPP`. In both datasets, the flights add up to a one-week flight schedule. All flights are performed by an MD11F aircraft, whose characteristics we introduced in Section 2.4. Large all-cargo aircraft like the MD11F are the most challenging to plan, because they can load much more cargo than smaller or passenger aircraft and a lot more loading constraints must be adhered. Accordingly, we expect results to these datasets to be easily transferable to other aircraft types. For evaluation we use the same set of master data for both datasets.

8.3.1 Master data

We created a simplified model of the MD11F aircraft depicted in Figure 2.8. It has 53 loading positions and 24 pairs of positions are overlapping. We consider 16 partial weight limits and one net weight constraint. The aircraft can load four different types of ULDs. On the main deck, standard pallets (PMC) and double-sized pallets (PGE) can be loaded. On the lower deck, standard pallets (PMC) and half-size containers (AKE) can be loaded. Note that the PMC pallets built for the main and lower deck have different contours and cannot be loaded into the other deck. All four ULD types have been introduced in Section 2.3. Furthermore, we consider the 115 item incompatibility constraints stemming from the original IATA dangerous goods regulations. We apply two sorting criteria. The first is to split items of the same shipment across as few ULDs as possible. The second is to avoid mixing standard and express shipments on the same ULD.

8.3.2 Generation of synthetic flights

The basis for dataset B is the pool of real booking data analyzed in Section 8.2. Dataset A is a one week slice of this pool.

The flights of the dataset B are taken from the public flight schedule of Lufthansa Cargo AG during calendar week 48 of the year 2015. We selected all flights departing from Frankfurt Airport that were performed by freighter aircraft.

For each flight b in B, we drew a random flight x from the pool that has the same number of legs. From the flight x we took the total payload and the distribution of payloads among its transport segments. Then, we iteratively drew random shipments from the flights in the pool until their sum reached the total payload of flight x. Afterwards, we assigned the drawn shipments to the transport segments of flight b with the same probability as seen at flight x. Finally, we adjusted the arrival time of the drawn shipments to represent the same transfer time as in their original flight.

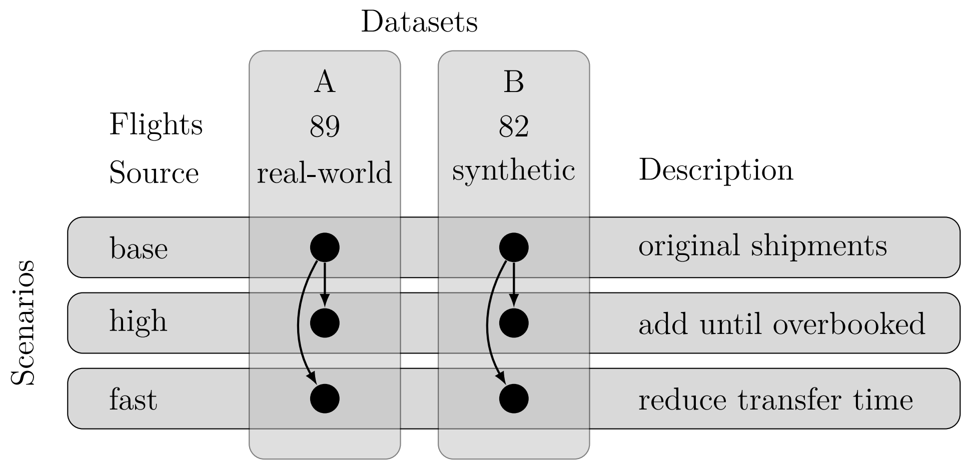


Figure 8.16: Derivation of benchmark datasets and scenarios

8.3.3 Scenarios

Because our datasets originate from booking lists of real historic flights, we do not expect them to contain many extreme features. During the booking period, sales staff is continuously monitoring each flight and managing its capacity conservatively. Confirmed bookings are extended by an expected stowage loss and buffers are added for special handling requirements, like those of dangerous goods or living animals. Future bookings must comply with the set safety margins or will only be accepted after a manual check for compliance by the sales staff.

Nevertheless, a decision support system for the ACLPP should be able to find acceptable solutions also for extreme cases. Furthermore, such a system could be used to automate compliance checks and reduce the conservative safety margins of today to the individually required level.

To evaluate solution approaches to the ACLPP also under difficult conditions, we introduce two additional scenarios: one with a high load and one with shorter transfer times. From a commercial view, these scenarios are very interesting. Higher loads mean more shipments and thus more revenues. Shorter transfer times allow faster connections and thus higher premiums can be charged.

We denote the original real-world and synthetic instances as *base* scenario in the following. According to an IATA report (Tyler, 2015), the average weight load factor in the market varies between 40 to 50 percent, which is also the case in our instances. Accordingly, load maximization is not the prime challenge of many flights. Instead, the minimization of handling efforts and at the same time streamlined aircraft operations are focused in the base scenario.

The *high* load scenario on the other hand puts special focus on the load maximization. In this case we extended the list of shipments for each flight until its volume capacity is overbooked. We added shipments until the booking list contained either more than 100 tonnes of shipment weight or 600 m^3 of volume. The respective norm capacities of the considered aircraft are 93 tonnes and 535 m^3. The exact overbooking limits were chosen arbitrary, but our intention is to put some stress on the palletization step without

Dataset-Scenario	A-base	A-high	A-fast	B-base	B-high	B-fast
Flights	89	89	89	82	82	82
∅ **weight**	45.7 t	102.6 t	45.7 t	45.5 t	101.9 t	45.5 t
∅ **volume**	200 m^3	455 m^3	200 m^3	212 m^3	469 m^3	212 m^3
∅ **shipments**	56	130	56	58	127	58
∅ **items**	478	1114	478	514	1150	514

Table 8.3: Scenario characteristics of the benchmark instances

leading to plain cherry picking. In dataset A, we randomly added shipments of other flights with the same flight number, i.e., flights from another day or week. In dataset B, we added shipments drawn from the pool.

The purpose of our third *fast* scenario is to evaluate the impact of shorter transfer times on all objectives. We reused the instances from the base scenario and reduced the transfer times of all shipments by 50 percent. However, we set a minimum transfer time for standard shipments to 2 hours and express to 1 hour.

Figure 8.16 provides an overview of the introduced datasets and scenarios as well as a description of the relations between them. In total, our datasets contain 513 flights, used for the evaluation of our approach.

8.4 Performance indicators

To measure the quality of planning results to the ACLPP, we define a set of performance indicators that are interesting from a practitioner's point of view. We categorize them based on the stakeholder objectives identified in Section 3.2. There are three stakeholders: sales, ground handling, and aircraft operations. For each group we present the respective indicators and define their calculation and objective. An overview of the indicators is also given in Table 8.4.

Sales indicators The sales department aims at maximizing the utilization and revenue of all flights. To measure the weight and volume utilizations, we use the average ratio of loaded items to total capacity. In terms of weight we refer to the ratio as weight load factor (WLF). Depending on the capacity definition, we distinguish two different load factors with respect to volume. We denote them as gross load factor (GLF) and net load factor (NLF). The GLF relates the loaded item volume to the full volume capacity of the aircraft, while the NLF relates the loaded item volume to the volume capacity of the used ULDs. If all loading positions in the aircraft are used, both GLF and NLF are equal. But, on flights where the demand is lower than the capacity of the aircraft, the NLF is a more meaningful measure of packing quality. The NLF is also interesting from a revenue maximization perspective. For two solutions of a flight with the same GLF, a higher NLF indicates that more loading positions in the aircraft remain available for highly profitable last minute sales. As another measure for revenue maximization, we use the sum of offload penalties (PEN). Each item that is offloaded incurs a certain cost, representing a customer

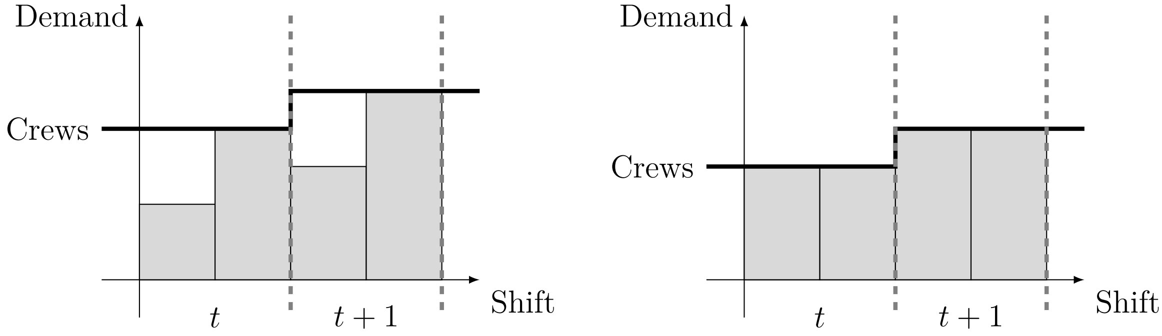


Figure 8.17: Sketch of an undesired schedule (left) with peaks and low periods compared to a desired schedule with even workload (right). Both have the same total workload, but the right requires less crews.

refund for late delivery or lost revenue, if an item cannot be transported due to capacity restrictions. Solutions with a lower PEN are generally preferred by the sales department.

Handling indicators The primary objective of the handling department is to minimize loading effort, to keep things simple, and to operate with an even workload. To measure handling effort we introduce four indicators. The first is the total number of used ULDs (UNITS). In a good solution, this should be minimized because each ULD comes with a fixed cost as well as the handling effort for preparation and postprocessing. The other three indicators relate to item sorting as described in Section 6.2.3. Splitting items that belong to the same shipment across multiple ULDs increases the handling effort. For the build-up, the shipment needs to be separated and multiple transports to the respective workstations are necessary. This increases the risk that the shipment is partially offloaded. Furthermore, at the next break-down, the items need to be consolidated again. Therefore, we measure the share of splitted shipments (SPLIT) and the average number of different ULDs that the splitted shipments are scattered across (DISP). The last indicator reflects the goal that standard and express shipments should not be mixed on the same ULDs when possible because they will be treated differently at the destination. Therefore, we determine the share of mixed ULDs (MIX) that contain both standard and express shipments. All sorting indicators should be low in a good solution. To measure the workload distribution, we introduce a measure for the total number of build-up crews needed during all shifts (CREWS). As sketched in Figure 8.17, schedules with an even workload during a shift require less crews than schedules containing peak loads and idle times.

Aircraft operations indicators The aircraft operations department aims at minimizing the additional fuel consumption, which is induced by aircraft imbalances, and the number of reloading operations at the stop-over airports. For each type of cost we introduce one indicator. The extra fuel (FUEL) measures the average cost of kerosene burned, because of a suboptimal center of gravity of the aircraft on a flight. The extra loading operations (OPS) measure the cost of unnecessary ULD unloading and loading operations at stop-over airports. We consider a loading operation as unnecessary if a ULD that continues

Name	Description	Unit	Objective
WLF	avg. weight load factor	percent	↗
GLF	avg. gross load factor	percent	↗
NLF	avg. net load factor	percent	↗
PEN	sum of offload penalty	cost	↘
UNITS	sum of used ULDs	#ULDs	↘
SPLIT	avg. share of split shipments	percent	↘
DISP	avg. split dispersion	#ULDs	↘
MIX	avg. share of mixed ULDs	percent	↘
CREWS	sum of crews in all shifts	#crews	↘
FUEL	sum of extra fuel due to imbalance	tonnes	↘
OPS	sum of unnecessary loading operations	#reloads	↘
TOTAL	sum of total cost of the load plan	cost	↘

Table 8.4: Overview of the performance indicators used for evaluation

on the next flight leg needs to be temporarily unloaded or relocated to another loading position at a stop-over airport. This might occur because the ULD is blocking another ULD that needs to be unloaded or to reduce the aircraft's imbalance on the next flight leg.

To combine all the tangible costs of a flight, we further introduce a last indicator of the total cost (TOTAL) that combines the penalties of offloaded shipments (PEN), the cost of all used ULDs (UNITS), the cost of extra fuel (FUEL), and the cost of unnecessary loading operations (OPS).

8.5 Summary

In this chapter, we took a look on the data of load planning problems in practice and derived benchmark instances to evaluate our solution approaches.

We presented an extensive data model describing the required planning parameters. On the one hand, there are a lot of required master data including the aircraft specifications, considered ULD types, workstation setup, regulatory requirements like item incompatibilities and separation constraints, and commercial objectives. On the other hand, there are the flight data describing the legs and booked shipments on the respective transport segments.

To get an understanding of the planning challenges, we analyzed a large dataset of real booking data. We found that flights are often not fully booked. Accordingly, dense packing is not always the prime objective and planning results that lead to efficient handling as well as aircraft operations are also highly important. Nevertheless, a significant part of the items to load is quite large, often exceeding the size of a standard wooden pallet, such that finding good three-dimensional packings is non-trivial. Overall, the items to load are strongly heterogeneous by their dimensions, weight, and density. However, many items appear multiple times in a shipment. Furthermore, the planning problems are significantly larger than popular benchmark instances for container loading problems. This motivates the creation of new benchmark instances of realistic size.

From the analyzed booking data, we derived two sets of benchmark instances. One contains data of real flights, while the other contains synthesized flights mimicking the real flights' properties. The latter dataset is publicly available under `https://github.com/fbrandt/ACLPP`. For each dataset we created three scenarios to examine interesting variations from a practical point of view. The first scenario contains just the base data, in the second all flights are overbooked, and in the third scenario the items have shorter transfer times at the terminal. Altogether, we created 513 flights to evaluate our solution approaches.

In preparation of the evaluation, we discussed suitable performance indicators for load planning results from a practical point of view. Fitting to the identified stakeholders and general objectives, we introduced twelve measures for aircraft utilization, physical handling effort, and aircraft operations efficiency.

9 A sequential planning approach

In this chapter, we present our first approach to solve the Air Cargo Load Planning Problem (ACLPP). To integrate well with the current manual planning processes of our industrial partner introduced in Section 3.3, i.e., the aircraft configuration, build-up scheduling, palletization, and weight and balance, we present a sequential approach that solves these steps one by one. The whole process is denoted as SeqACLPP in the following. The main goal of the SeqACLPP is to automate the manual planning steps of today. Our objective is to find a good overall load plan, with respect to all constraints and performance indicators presented in Section 8.4. Furthermore, this approach should provide a benchmark for process changes.

Since the ACLPP cannot be solved independently for each flight and the degree of interconnection depends on the planning step, we consider three scopes in our solution: the *flight plan*, *flights*, and *segments*. The flight plan usually contains multiple flights, which compete for the same resources in the air cargo terminal. A flight potentially consists of multiple transport segments, which compete for the same resources in the aircraft. In Table 9.1 we give an overview of the planing steps and which scopes are considered in each individual run of the respective step. The aircraft configuration is run for each flight. The build-up scheduling is run for the full flight plan at once and considering all the transport segments. The palletization is run individually for each transport segment. Finally, the weight and balance is solved for each flight.

This chapter is organized as follows. In Section 9.1 we present a volume and weight based approach (VWAC) to solve the Aircraft Configuration Problem (ACP). It decides what ULDs should be built for each transport segment of a flight. In Section 9.2 we present a rolling horizon planning approach (RHBS) to solve the Build-up Scheduling Problem (BSP). It decides when to build the selected ULDs. After that, we present our approach to assign and pack items onto the selected and scheduled ULDs (BDAP) by a Logic-based Benders Decomposition (LBBD) solving the Air Cargo Palletization

Problem	**Aircraft Configuration**	**Build-up Scheduling**	**Air Cargo Palletization**	**Weight and Balance**
Section	9.1	9.2	9.3	9.4
Step name	VWAC	RHBS	LBBD	UWAB
Flight plan		X		
Flight	X			X
Segment	X	X	X	X

Table 9.1: Entities considered in individual runs of the planning steps

Problem (APP). Finally, in Section 9.4 we present our solution step (UWAB) to solve the Weight and Balance Problem (WBP). In this step, the built ULDs are assigned to loading positions in the aircraft.

9.1 Volume-and-weight-based aircraft configuration

The first step of our solution approach is to solve the ACP for each flight. In practice, this step is performed multiple times between 24 and 72 hours before a flight. In the early runs, not all bookings have arrived and some bookings might still contain placeholders or unconfirmed item data. But, the state of the data continually improves until departure. Accordingly, in the beginning the primary purpose of aircraft configuration is to estimate future build-up demands for scheduling build-up workers and ULD provisioning. Only at later iterations the selected ULDs become more important for palletization preparations.

To keep things simple and the amount of required data low, we operate only on sums of the booked cargo weight and volume of each transport segment during this step. Given the total cargo volume and weight of each transport segment, the aircraft characteristics, and the available ULD types we decide how many ULDs of each type should be built for each transport segment. We denote this step as VWAC in the following. In the remainder of this section, we introduce some necessary preprocessing steps followed by a description of the employed MIP model.

9.1.1 Preprocessing

The booking data introduced in Section 8.1 contain all the information about the individual shipments. But, as our model operates on aggregated data of each transport segment, we need to derive some model parameters from the input booking data.

Offload penalties Each item to load has a specific penalty offload_i that applies if it is offloaded. This penalty indicates the lost profit or the refund to the customer. During aircraft configuration we only consider the total demand with respect to volume and weight for each transport segment, not the individual items. To do that, we need to derive suitable offload penalties for each segment. In our preprocessing we determine the penalties as the average penalty across all items I_k on the segment k, weighted by the item weight and volume respectively:

$$\text{offload}_k^{\text{vol}} = \frac{\sum\limits_{i \in I_k} \text{offload}_i}{\sum\limits_{i \in I_k} \text{volume}_i} \qquad \text{offload}_k^{\text{wgt}} = \frac{\sum\limits_{i \in I_k} \text{offload}_i}{\sum\limits_{i \in I_k} \text{weight}_i^{\text{grs}}} \qquad k \in K \tag{9.1}$$

Estimated bin capacity In practice it is unlikely that we are able to fill a ULD up to its physical volume capacity. Often the combinatorics between the items and the curved shape of the ULD will result in some unused space. Therefore, in the following we use the average achievable volume utilization of each ULD type instead of the real geometric volume capacity. In practice this utilization can be derived from the built ULDs seen on historic flights or earlier iterations of this solution process. We will discuss the impact of the estimated capacity on the constructed solutions and the choice of suitable parameters in Section 10.2.1.

Minimum number of required bins The third preprocessing step is to determine the minimum number of ULDs required for certain ULD types. Here, we check for each item if it would fit into each ULD type. For example, very long items might only fit into PGE ULDs and high items that are not rotatable must be placed into the upper deck, thus requiring a main deck PMC ULD. See Figure 2.3 for an overview of the used ULD types. The geometric volumes of items requiring a special ULD type are summed up and divided by the average achievable volume utilization for this ULD type. Rounding the result up to the next integer gives us the minimum required number of ULDs of each type.

9.1.2 Mixed-integer model

We solve the aircraft configuration by applying the full Aircraft Configuration Model (ACM) as presented in Section 4.2.5. Table 9.2 recaps the sets and parameters of the model and Table 9.3 presents its decision variables. The model primarily decides about the number $x_{k,v}$ of ULDs to build for each transport segment k and ULD type v. For completeness, we repeat the ACM with minor adjustments for this step:

$$\text{minimize} \sum_{k\in K}\left(\sum_{v\in V} x_{k,v}\text{cost}_v + \frac{1}{2}y_k^{\text{wgt}}\text{offload}_k^{\text{wgt}} + \frac{1}{2}y_k^{\text{vol}}\text{offload}_k^{\text{vol}}\right) \tag{9.2}$$

subject to

$$\text{dem}_k^{\text{wgt}} \le \sum_{v\in V} x_{k,v}\text{cap}_v^{\text{wgt}} + y_k^{\text{wgt}} \qquad k \in K \tag{9.3}$$

$$\text{dem}_k^{\text{vol}} \le \sum_{v\in V} x_{k,v}\text{cap}_v^{\text{vol}} + y_k^{\text{vol}} \qquad k \in K \tag{9.4}$$

$$\text{limit}_f \ge \sum_{k\in K: l\in \text{on}_k}\sum_{v\in V} \text{m}_{f,v}x_{k,v} + (b_{l,f}-1)M \qquad l \in L; f \in F \tag{9.5}$$

$$1 \le \sum_{f\in G} b_{l,f} \qquad l \in L; G \in F \tag{9.6}$$

$$\text{min}_{k,v} \le x_{k,v} \qquad k \in K; v \in V \tag{9.7}$$

$$\text{avail}_{l,v} \ge \sum_{k\in K: k\in \text{start}_l} x_{k,v} \qquad l \in L; v \in V \tag{9.8}$$

$$b_{l,f} \in \{0,1\} \qquad l \in L; G \in F \tag{9.9}$$

$$y_k^{\text{wgt}}, y_k^{\text{vol}} \in \mathbb{R}_{\ge 0} \qquad k \in K \tag{9.10}$$

$$x_{k,v} \in \mathbb{N} \qquad k \in K; v \in V \tag{9.11}$$

The objective, given in Equation (9.2), is to minimize the total cost from the used ULDs and offloaded cargo. Because offloaded items will appear twice, as offloaded volume and

Set/Parameter	Description
F	set of ULD combination restrictions in the aircraft
K	set of transport segments on the considered flight
L	set of legs of the considered flight
V	set of available ULD types
$\mathrm{avail}_{l,v} \in \mathbb{N}$	available units at departure airport of leg l
$\mathrm{cap}_v^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$	volume capacity per ULD of type v
$\mathrm{cap}_v^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$	weight capacity per ULD of type v
$\mathrm{cost}_v \in \mathbb{R}$	handling cost (preparation, loading) per ULD of type v
$\mathrm{dem}_k^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$	volume of the shipments on segment k
$\mathrm{dem}_k^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$	weight of all shipments on segment k
$\mathrm{limit}_f \in \mathbb{R}$	total ULD limit in restriction f
$\mathrm{m}_{f,v} \in \mathbb{R}$	factor of ULD type v in restriction f
$\mathrm{min}_{k,v} \in \mathbb{N}$	minimum required number of ULDs of type v
$\mathrm{start}_l \subseteq K$	set of transport segments starting with flight leg l
$\mathrm{offload}_k^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$	penalty for offloaded shipment volume
$\mathrm{offload}_k^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$	penalty for offloaded shipment weight
$\mathrm{on}_k \subseteq L$	set of flight legs of transport segment k

Table 9.2: Parameters of the aircraft configuration model

Variable	Description
$b_{l,f} \in \{0,1\}$	1 if on leg l combination restriction f holds
$x_{k,v} \in \mathbb{N}$	number of ULDs to built on segment k of type v
$y_k^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$	volume offloaded on segment k
$y_k^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$	weight offloaded on segment k

Table 9.3: Variables of the aircraft configuration model

weight, we multiply the penalties by $\frac{1}{2}$. In the following we briefly describe the restrictions. A detailed explanation of the constraints is given in Section 4.2.3.

Equations (9.3) and (9.4) make sure that enough weight and volume capacity is reserved or that the cargo is marked as offloaded. Equations (9.5) and (9.6) limit the selected set of ULDs to a valid aircraft configuration. The minimum number of required ULDs per type on each transport segment are modeled with Equation (9.7). Finally, Equation (9.8) ensures that no more than the available number of ULDs are used at each airport.

For simplicity we assume that the handling cost and available ULD types are the same for all visited airports. We note that the model can be easily extended to support different parameters for each airport.

9.2 Rolling horizon build-up scheduling

All flights departing during the same period compete for the same build-up resources in the air cargo terminal, namely workstations and workers. With a given aircraft configuration for each flight, we know the number of each ULD type that must be built before each flight's departure time. Because only physically available cargo can be loaded into a ULD and each flight has its individual departure time, we need to schedule the ULD build-ups. Furthermore, the schedule must be feasible with respect to the available workstations and workers.

In practice build-up scheduling is a continuous planning process. New jobs to build ULDs for upcoming flights appear constantly, while the previous flights are already partially built and ULDs currently in production cannot be changed anymore. To manage this ongoing process, we present a planning model with a rolling horizon. We cut the continuous time into discrete planning horizons, the build-up crew shifts, and plan each shift after another. We denote this step as RHBS in the following.

The remainder of this section is structured as follows. In Section 9.2.1 we introduce some necessary preprocessing steps to determine suitable parameters for the planning models from the given flights. Afterwards, we present our MIP model to schedule the build-up demand for a limited time horizon in Section 9.2.2. How this model is used for continuous planning with a rolling horizon is described in Section 9.2.3.

9.2.1 Preprocessing

Most parameters of the BSP are defined in the booking lists of the flights or are determined during the aircraft configuration, see Table 9.3. In this section, we discuss how the remaining parameters, see Section 5.2.1, can be derived from these input data.

Offload penalties Each item to load has a specific offload penalty offload_i. This penalty indicates the lost profit or the refund to the customer. Due to workstation or worker capacity restrictions we might not be able to schedule all ULDs or we have to schedule them too early, at a period where not enough cargo is available. In both cases we loose volume capacity and will have to offload items later. Analog to the offload penalties in the aircraft configuration, we estimate the loss by the volume-weighted average offload penalty across all items I_k on segment k as:

$$\text{offload}_k^{\text{vol}} := \frac{\sum\limits_{i\in I_k} \text{offload}_i}{\sum\limits_{i\in I_k} \text{volume}_i} \qquad k \in K \tag{9.12}$$

We only consider volume, and not weight, here because flights are usually volume-bound and thus we expect that the cargo weight can be redistributed onto other ULDs.

Cargo availability In our model we use the available cargo volume during each period. The booking lists contain the arrival period of each item. Therefore, we calculate the available cargo volume per segment k and period t as:

$$\text{avail}_{k,t}^{\text{vol}} := \sum_{\substack{i\in I_k:\\ \text{avail}_i \le t}} \text{volume}_i \qquad k \in K; t \in T \tag{9.13}$$

Parallel build-ups In practice we only want to build a limited number of ULDs for the same segment at the same time. This leaves more options and time to react to operational problems during build-up and allows to build the ULDs on close-by workstations. Let $\text{split}^{\text{max}}$ denote the maximum desired number of parallel build-ups of a segment. Enforcing a hard limit in practice might lead to offloads for segments where the cargo arrives rather late such that the limit is too low to build all ULDs on time. In these cases we want to set a less restrictive limit.

As a suitable limit we choose the maximum required parallel build-ups, between any build-up period and the scheduled departure period of the flight, for which all cargo can be packed. For each period t, we determine the cargo volume arriving in the future ($\text{dem}_k^{\text{vol}} - \text{avail}_{k,t}^{\text{vol}}$) and the volume that could be packed in the remaining time, if no parallel build-ups are allowed. Note that the number of ULDs of type v that can be built sequentially in the time between t and due_k can be determined as $\left\lfloor \frac{\text{due}_k - t - 1}{\text{dur}_v} \right\rfloor$. The ratio of both gives us the required parallelism and its maximum over the periods the sought limit. If the required parallelism is less than the desired $\text{split}^{\text{max}}$, we allow $\text{split}^{\text{max}}$ to be built in parallel.

$$\text{split}_k := \max\left(\text{split}^{\text{max}}; \max_{\substack{t\in T:\\ t<\text{due}_k-\text{dur}_v}} \left(\frac{\text{dem}_k^{\text{vol}} - \text{avail}_{k,t}^{\text{vol}}}{\left\lfloor \frac{\text{due}_k - t - 1}{\text{dur}_v} \right\rfloor \text{cap}_v^{\text{vol}}}\right)\right) \qquad k \in K; v \in V \tag{9.14}$$

9.2.2 Mixed-integer model

We solve the build-up scheduling by applying the full Build-up Scheduling Model (BSM) as presented in Section 5.2.5. The given set of transport segments corresponds to the jobs and the crew shifts represent the machines. Table 9.4 recaps the sets and parameters of the model and Table 9.5 presents its decision variables. The model primarily decides on the number of ULDs to build $start_{k,v,t}$ for each ULD type, segment, and period as well as

Set/Parameter	Description
K	set of transport segments to build ULDs for
S	set of crew shifts
T	set of time periods
V	set of available ULD types
$\text{avail}_{k,t}^{\text{vol}} \in \mathbb{R}_{\geq 0}$	demand volume for segment k available at period t
$\text{cap}_v^{\text{vol}} \in \mathbb{R}_{\geq 0}$	volume capacity per ULD of type v
$\text{cost}_s \in \mathbb{R}_{\geq 0}$	cost of one build-up crew in shift s
$\text{dem}_k^{\text{vol}} \in \mathbb{R}_{\geq 0}$	total volume of the shipments on segment k
$\text{due}_k \in T$	build-up due date of segment k
$\text{dur}_v \in \mathbb{N}$	number of periods to build-up a ULD of type v
$\text{end}_s \in T$	last period after shift s
$\text{max}_s \in \mathbb{N}$	max build-up crews in shift s
$\text{offload}_k^{\text{vol}} \in \mathbb{R}_{\geq 0}$	offload penalty per volume unit for segment k
$\text{open}_{v,t} \in \mathbb{N}$	open workstations for ULD type v at period t
$\text{pack}_k \in \mathbb{R}_{\geq 0}$	already packed demand volume for segment k at period 0
$\text{split}_k \in \mathbb{N}$	max simultaneous build-ups for segment k
$\text{start}_s \in T$	first period of shift s
$\text{uld}_{k,v} \in \mathbb{N}$	number of ULDs of type v to schedule for segment k
$\delta(v,t) \subseteq T$	periods when build-ups in progress during period t pot. started $\delta(v,t) = \{t' \mid t' \in T : t - \text{dur}_v < t' \leq t\}$
$\phi(k) \subseteq T$	periods when ULD build-ups for segment k can be started $\phi(k) = \{t \mid t \in T : 0 < t \leq \text{due}_k\}$
$\theta(s) \subseteq T$	periods when workers of shift s can perform build-ups $\theta(s) = \{t \mid t \in T : \text{start}_s \leq t < \text{end}_s\}$

Table 9.4: Parameters of the build-up scheduling model

the number of required crews $crews_s$ during each shift. Furthermore, it tracks the already built volume $pack_{k,t}$ for each segment and period. By comparing this with the available cargo volume in each period, we can derive the lost space inside planned ULDs $dead_{k,t}$ and if necessary the not scheduled ULDs $offload_{k,v}$. For completeness, we repeat the BSM with minor adjustments for this step:

$$\text{minimize} \sum_{k \in K} \text{offload}_k^{\text{vol}} \left(\sum_{t \in T} dead_{k,t} + \sum_{v \in V} \text{cap}_v^{\text{vol}} offload_{k,v} \right) + \sum_{s \in S} \text{cost}_s crews_s \tag{9.15}$$

Variable	Description
$\mathit{crews}_s \in \mathbb{N}$	number of crews required during shift s
$\mathit{dead}_{k,t} \in \mathbb{R}_{\geq 0}$	unusable volume capacity during build-ups for segment k in period t
$\mathit{offload}_{k,v} \in \mathbb{N}$	number of selected but not scheduled ULDs for segment k of type v
$\mathit{pack}_{k,t} \in \mathbb{R}_{\geq 0}$	already packed demand volume for segment k prior to period t
$\mathit{start}_{k,v,t} \in \mathbb{N}$	number of ULDs build-ups started for segment k of type v at period t

Table 9.5: Variables of the build-up scheduling model

subject to

$$\mathrm{uld}_{k,v} = \mathit{offload}_{k,v} + \sum_{\substack{t \in T: \\ t \leq \mathrm{due}_k - \mathrm{dur}_v}} \mathit{start}_{k,v,t} \qquad k \in K; v \in V \tag{9.16}$$

$$\mathit{pack}_{k,0} = \mathrm{pack}_k \tag{9.17}$$

$$\mathit{pack}_{k,t} + \mathit{dead}_{k,t} = \mathit{pack}_{k,t-1} + \sum_{v \in V} \mathit{start}_{k,v,t} \mathrm{cap}_v^{\mathrm{vol}} \qquad k \in K; t \in \phi(k) \tag{9.18}$$

$$\mathrm{avail}_{k,t} \geq \mathit{pack}_{k,t} \qquad k \in K; t \in T \tag{9.19}$$

$$\mathrm{split}_k \geq \sum_{v \in V} \sum_{t' \in \delta(v,t)} \mathit{start}_{k,v,t'} \qquad k \in K; t \in T \tag{9.20}$$

$$\mathrm{open}_{v,t} \geq \sum_{k \in K} \sum_{t' \in \delta(v,t)} \mathit{start}_{k,v,t'} \qquad v \in V; t \in T \tag{9.21}$$

$$\sum_{\substack{s \in S: \\ t \in \theta(s)}} \mathit{crews}_s \geq \sum_{k \in K} \sum_{v \in V} \sum_{t' \in \delta(v,t)} \mathit{start}_{k,v,t} \qquad t \in T \tag{9.22}$$

$$\mathit{crews}_s \in \mathbb{N} \qquad s \in S \tag{9.23}$$

$$\mathit{dead}_{k,t} \in \mathbb{R}_{\geq 0} \qquad k \in K; t \in \phi(k) \tag{9.24}$$

$$\mathit{offload}_{k,v} \in \mathbb{N} \qquad k \in K; v \in V \tag{9.25}$$

$$\mathit{pack}_{k,t} \in \mathbb{R}_{\geq 0} \qquad k \in K; t \in \phi(k) \tag{9.26}$$

$$\mathit{start}_{k,v,t} \in \mathbb{N} \qquad k \in K; v \in V; t \in \phi(k) \tag{9.27}$$

The objective, given in Equation (9.15), is to minimize the sum of the cost of the lost profit due to early build-ups or unscheduled ULDs and the cost of the deployed crews. In the following we briefly describe the restrictions. A detailed explanation of the constraints is given in Section 5.2.3.

Equation (9.16) makes sure that all requested ULDs are built on time or are marked as offload. Equations (9.17) and (9.18) calculate the already built cargo volume at each period and Equation (9.19) enforces that during each period not more than the already available cargo is packed. The number of parallel build-ups for a segment is limited with Equation (9.20). Similar checks make sure that the number of available workstations is respected in Equation (9.21) and that during each period enough crews are on duty in Equation (9.22).

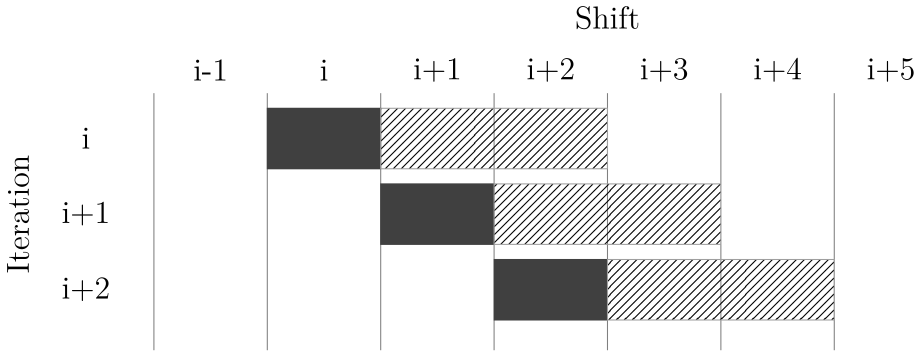


Figure 9.1: Rolling horizon planning of build-up schedules. During the i-th iteration, build-ups during shifts i to $i + k$ ($k = 2$ is this example) are planned. Of this result only ULDs built during shift i are fixed and the procedure is repeated for the next shift.

9.2.3 Rolling horizon planning

The model presented in the last section works for a given period of time. In practice the BSP is continuous and no clear cut after a certain period is possible. Transport segments and the ULDs to build continuously enter and leave the system. Therefore, we use a rolling horizon to plan the build-up schedule. The overall process is depicted in Figure 9.1. In each iteration we consider the next n shifts and the flights, respectively the transport segments, leaving during these shifts. We plan all n shifts with the model presented in the previous section. The jobs to plan are the ULDs of those flights that have not been built yet. From the planning result, we only implement the first shift and replan the remaining demand starting with the next shift. The overall procedure is shown in Algorithm 1.

Data: transport segments K, crew shifts S, planning horizon h
Result: crews per shift $crews_s$, scheduled ULDs $start_{k,v,t}$

$packed \leftarrow \emptyset$;
for $s \in S$ **do**
  $segments \leftarrow \{k \mid k \in K : \text{start}_s < \text{due}_k \leq \text{end}_{s+h}\}$;
  run MIP model for *segments* (with *packed*) starting at period start_s;
  $crews_s \leftarrow$ selected number of crews in shift s;
  $start_{k,v,t} \leftarrow$ ULDs built during shift s;
  $packed \leftarrow$ add ULDs built during shift$s$;
end

Algorithm 1: Build-up scheduling with rolling horizon

The planning horizon n is a tuning parameter. It allows us to distribute peak demands over the previous shifts. Furthermore, the parameter effectively determines how early before its departure time a transport segment can be built. Larger values can distribute the workload over a longer period at the cost of a large model and thus probably longer computation times.

9.3 Air cargo palletization

In this section, we present our solution approach to the Air Cargo Palletization Problem (APP). The APP decides which items are placed into which ULD and how they have to be arranged in the three-dimensional space to fulfill all physical, regulatory, and organizational constraints. In the earlier steps we already selected a suitable superset of ULDs for each transport segment and assigned a build-up period for each ULD. It is possible to solve the APP for each transport segment independently, because items for different segments must be placed in different ULDs. Therefore, we only consider problems with a single transport segment in the following.

As the three-dimensional packing problem is NP-complete and large real world instances are typically computationally very hard to solve, we decompose the problem and solve it heuristically. Our approach is known as Logic-based Benders Decomposition (LBBD), which was introduced by Hooker (2007). Its basic idea is to split an optimization problem into a linear master problem and an arbitrary subproblem. For a given solution to the master, the subproblem is solved heuristically. If the subproblem turns out to be infeasible, problem-specific heuristic cuts are derived from the subproblem and added to the master. Then the master is solved again and the process is repeated until the subproblem can be solved.

In our case, the master problem is a vector container loading problem, denoted as *BinAssign* in the following. It decides, which ULDs to use and how to assign items to them. In the master problem all non-geometric constraints are handled. Our subproblem is a three-dimensional knapsack problem, denoted as *CPPack* in the following. It needs to be solved for each individual ULD and deals with all the geometric and physical aspects of the placement of items inside the ULD. For cases, in which not all boxes can be packed in the subproblem, we derive valid inequalities from the packable subset of items and add them to the master problem before resolving it. We repeat this process until all ULDs can be successfully packed in the subproblem. On top of the LBBD, we reduce the search space heuristically to make the problem solvable within an acceptable time limit. We denote this process as BDAP in the following.

Our approach implements all but two features of the APP introduced in Section 6.2. We only omit item spacing restrictions between ULDs and item positioning restrictions within the aircraft, because the ULDs do not have an assigned loading position yet and thus no absolute coordinates within the aircraft are given. Nevertheless, both restrictions will be modeled and thus be respected as ULD spacing constraints in the final weight and balance step. We give an overview which constraints are handled by which part of our solution in Table 9.6.

The remainder of this section is structured as follows. In Section 9.3.1 we introduce the MIP formulation of the BinAssign master problem. In Section 9.3.2 we present the constraint program of the CPPack subproblem. In Section 9.3.3 we describe the cut generation and LBBD. Afterwards, we present two methods to reduce the search space and to speed up our approach. In Sections 9.3.4 we introduce a decomposition by item volume and in Section 9.3.5 the estimation of stowage loss for the items. At last, we recap the overall procedure in Section 9.3.6.

Constraint	Master problem BinAssign	Subproblem CPPack
ULD assignment of items	$\surd$	$\surd$
item rotation		$\surd$
item inside ULD		$\surd$
items non-overlapping		$\surd$
item available before ULD build-up	$\surd$	
ULD weight limit	$\surd$	
net weight limits across ULDs	$\surd$	
ULD weight distribution		$\surd$
item stacking		$\surd$
complete shipment	$\surd$	
item grouping in ULDs	$\surd$	
item compatibility in ULD	$\surd$	
item spacing within ULD		$\surd$
item positioning in ULD		$\surd$
item placement stability		$\surd$

Table 9.6: Overview which constraints are handled in the master and subproblem

9.3.1 Mixed-integer model for bin assignment (BinAssign)

The master problem of our palletization step is the selection of ULDs and the assignment of items to them. We use a MIP model derived from the mathematical problem formulation of the APM in Section 6.2. Table 9.7 recaps the sets and parameters of the model and Table 9.8 its decision variables.

The model primarily decides which ULDs are used $load_u$ and how many items of a type are assigned to which ULD $assign_{u,i}$ or offloaded $assign_i^{\text{off}}$. Auxiliary variables exist to indicate which item types are present in a ULD $in_{u,i}$, the sorting values that are present in a ULD $sort_{q,u,e}$, how many sorting errors occurred n_q, and if an item type was offloaded completely $none_i$.

We define the BinAssign model as follows:

$$\text{minimize} \sum_{u \in U} \text{cost}_u load_u + \sum_{i \in I} \text{offload}_i assign_i^{\text{off}} + \sum_{q \in Q} \text{penalty}_q n_q \tag{9.28}$$

subject to

$$assign_{u,i} \leq \text{amount}_i in_{u,i} \qquad u \in U; i \in I \tag{9.29}$$

$$load_u \geq in_{u,i} \qquad u \in U; i \in I \tag{9.30}$$

$$\text{amount}_i = assign_i^{\text{off}} + \sum_{u \in U_i} assign_{u,i} \qquad i \in I \tag{9.31}$$

$$\text{limit}_u \geq \sum_{i \in I} \text{weight}_i^{\text{grs}} assign_{u,i} \qquad u \in U \tag{9.32}$$

$$\text{volume}_u \geq \sum_{i \in I} \text{volume}_i assign_{u,i} \qquad u \in U \tag{9.33}$$

Set/Parameter	Description
D_g	set of net weight limits applying to subsets of ULDs ($d \in D_g$)
G	set of considered net weight categories ($g \in G$)
I	set of item types to load ($i \in I$)
Q	set of sorting criteria ($q \in Q$)
Q_q	value domain of sorting criterion q
U	set of available ULDs ($u \in U$)
$A \subseteq I \times I$	pairs of item types not allowed within one ULD ($\langle i, j \rangle \in A$)
amount_i	number of items of type i
$\text{clique}_i \in I$	item type that must be loaded completely to also load type i
$\text{cost}_u \in \mathbb{R}_{\geq 0}$	cost when using ULD u
$\text{limit}^{\text{net}}_d \in \mathbb{R}_{\geq 0}$	net weight limit for all items placed in ulds_d for weight limit d
$\text{limit}_u \in \mathbb{R}_{\geq 0}$	gross weight limit of ULD u
$\text{offload}_i \in \mathbb{R}_{\geq 0}$	penalty for each item of type i that is offloaded
$\text{penalty}_q \in \mathbb{R}_{\geq 0}$	linear penalty for sorting errors of criterion q
$\text{pow}_q \in \mathbb{N}$	exponential penalty for sorting errors of criterion q
$\text{sort}_{q,i} \in Q_q$	group of item type i w.r.t. sorting criterion q
$Q^{\text{item}} \subseteq Q$	set of sorting criteria to group items by
$Q^{\text{uld}} \subseteq Q$	set of sorting criteria to separate items by
$U_i \subseteq U$	ULDs that item type i can be packed into
$\text{ulds}_d \subseteq U$	ULDs considered for net weight limit d
$\text{volume}_i \in \mathbb{R}_{\geq 0}$	volume of each item of type i
$\text{volume}_u \in \mathbb{R}_{\geq 0}$	maximum usable volume of ULD u
$\text{weight}^{\text{grs}}_i \in \mathbb{R}_{\geq 0}$	gross weight of each item of type i
$\text{weight}^{\text{net}}_{g,i} \in \mathbb{R}_{\geq 0}$	weight of each item of type i w.r.t. weight category g

Table 9.7: Parameters of the BinAssign model

Variable	Description
$assign_{u,i} \in \mathbb{N}$	number of items of type i loaded into ULD u
$assign_i^{\text{off}} \in \mathbb{N}$	number of items of type i that are offloaded
$in_{u,i} \in \{0,1\}$	is item i present in ULD u
$load_u \in \{0,1\}$	1 if ULD u is loaded
$n_q \in \mathbb{N}$	number of penalties w.r.t. sorting criterion q
$none_i \in \{0,1\}$	1 if item i is completely loaded
$sort_{q,u,e} \in \{0,1\}$	1 if sorting value n_q is present in ULD u

Table 9.8: Decision variables of the BinAssign model

$$\text{limit}_d^{\text{net}} \geq \sum_{u \in \text{ulds}_d} \sum_{i \in I} \text{weight}_{g,i}^{\text{net}} assign_{u,i} \qquad g \in G; d \in D_g \tag{9.34}$$

$$assign_i^{\text{off}} = \text{amount}_i none_i \qquad i \in I \tag{9.35}$$

$$none_i = none_{\text{clique}_i} \qquad i \in I \tag{9.36}$$

$$in_{u,i} \leq sort_{q,u,\text{sort}_{q,i}} \qquad q \in Q; u \in U; i \in I \tag{9.37}$$

$$n_q = \sum_{e \in Q_q} \left(\sum_{u \in U} sort_{q,u,e} \right)^{\text{pow}_q} \qquad q \in Q^{\text{item}} \tag{9.38}$$

$$n_q = \sum_{u \in U} \left(\sum_{e \in Q_q} sort_{q,u,e} \right)^{\text{pow}_q} \qquad q \in Q^{\text{uld}} \tag{9.39}$$

$$1 \geq in_{u,i} + in_{u,j} \qquad u \in U; \langle i,j \rangle \in A \tag{9.40}$$

$$assign_{u,i} \in \mathbb{N} \qquad i \in I; u \in U_i \tag{9.41}$$

$$assign_i^{\text{off}} \in \mathbb{N} \qquad i \in I \tag{9.42}$$

$$none_i \in \{0,1\} \qquad i \in I \tag{9.43}$$

$$n_q \in \mathbb{N} \qquad q \in Q \tag{9.44}$$

$$in_{u,i} \in \{0,1\} \qquad i \in I; u \in U_i \tag{9.45}$$

$$load_u \in \{0,1\} \qquad u \in U \tag{9.46}$$

$$sort_{q,u,e} \in \{0,1\} \qquad q \in Q; u \in U; e \in Q_q \tag{9.47}$$

The objective, given in Equation (9.28), is to minimize the sum of cost of the used ULDs, the penalties of offloaded items, and the penalties for violating the sorting constraints. In the following we briefly describe the restrictions. A detailed explanation of the constraints is given in Section 6.2.3.

Equation (9.29) tracks what item types are present in a ULD. According to Equation (9.30) items can only be assigned to a loaded ULD. Equation (9.31) models the assignment of items to ULDs. Equations (9.32) and (9.33) ensure that the ULD's weight and volume capacity is respected. Complete shipment constraints are enforced by Equations (9.35) and (9.36). Equations (9.37) to (9.39) calculate the penalties with respect to the different item and bin sorting criteria. Finally, Equation (9.40) models incompatibility constraints between items that must not be placed in the same ULD.

Set/Parameter	**Description**
C_u	set of contour planes of ULD u ($\langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u$)
I	the items to pack ($i \in I$)
$\mathrm{dim}_i^{\mathrm{x}}, \mathrm{dim}_i^{\mathrm{y}}, \mathrm{dim}_i^{\mathrm{z}} \in \mathbb{R}_{\geq 0}$	dimensions of item i
$\mathrm{dist}_{i,j} \in \mathbb{R}_{\geq 0}$	minimum required distance between items i and j
$\mathrm{offload}_i \in \mathbb{R}_{\geq 0}$	penalty when item i is offloaded
$\mathrm{rot}_i \subseteq R$	allowed rotations of item i
$\mathrm{rotations}_i \subseteq \mathbb{R}_{\geq 0}^4$	rotated dimensions and load-bearing strength of item i
$\mathrm{stack}_i^{\mathrm{x}}, \mathrm{stack}_i^{\mathrm{y}}, \mathrm{stack}_i^{\mathrm{z}} \in \mathbb{R}_{\geq 0}$	maximum load-bearing strength of item i
$\mathrm{start}_u^{\mathrm{x}}, \mathrm{start}_u^{\mathrm{y}}, \mathrm{start}_u^{\mathrm{z}} \in \mathbb{R}$	start coordinate of ULD u
$\mathrm{supp}_i \in \mathbb{R}_{\geq 0}$	minimum supported area of item i (ratio)
$\mathrm{udim}_u^{\mathrm{z}}, \mathrm{udim}_u^{\mathrm{y}}, \mathrm{udim}_u^{\mathrm{x}} \in \mathbb{R}_{\geq 0}$	dimensions of ULD u
$\mathrm{weight}_i^{\mathrm{grs}} \in \mathbb{R}_{\geq 0}$	gross weight of item i
$(\underline{\mathrm{x}}_u^{\mathrm{cg}}, \underline{\mathrm{y}}_u^{\mathrm{cg}}, \underline{\mathrm{z}}_u^{\mathrm{cg}}) \in \mathbb{R}^3$	lower bound of the valid CG of ULD u
$(\overline{\mathrm{x}}_u^{\mathrm{cg}}, \overline{\mathrm{y}}_u^{\mathrm{cg}}, \overline{\mathrm{z}}_u^{\mathrm{cg}}) \in \mathbb{R}^3$	upper bound of the valid CG of ULD u
$(\underleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \underleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \underleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}}) \in \mathbb{R}^3$	lower bound of item i start position inside ULD u
$(\overleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \overleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \overleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}}) \in \mathbb{R}^3$	upper bound of item i start position inside ULD u
$(\underrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \underrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \underrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}}) \in \mathbb{R}^3$	lower bound of item i end position in ULD u
$(\overrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}, \overrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}, \overrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}}) \in \mathbb{R}^3$	upper bound of item i end position in ULD u
ϵ	packing height tolerance

Table 9.9: Parameters of the CPPack model for ULD u

9.3.2 Constraint program for the 3D knapsack problem (CPPack)

In the master problem we assigned items to bins with respect to the total volume, weight, and further domain specific constraints. In this section, we present a constraint programming model to determine the three-dimensional positions of the items inside the bin with respect to the geometric and physical constraints. The model considers only one bin, i.e., it must be run for each bin from the solution of the master problem. We derive our model from the mathematical formulation of the APM in Section 6.2.

Table 9.9 introduces the sets and parameters of the model and Table 9.10 presents its decision variables. The model decides which of the items assigned to the bin are actually packed $pack_i$, at which position (x_i, y_i, z_i), and orientation $(size_i^{\mathrm{x}}, size_i^{\mathrm{y}}, size_i^{\mathrm{z}})$. Auxiliary variables track the mechanical relations, like the actual load-bearing strength of the item $stack_i$, how much an item weighs including items stacked on top of it w_i, if an item is placed on the floor $floor_i$, or if it supports another item $base_{i,j}$. For the calculation of the contact areas between two items we track the overlap along the x-axis $o_{i,j}^{\mathrm{x}}$ and z-axis $o_{i,j}^{\mathrm{z}}$ as well as the total supported area of an item o_i. Finally, to limit the bin's center of

Name	Min	Max	Description
$base_{i,j}$	0	1	1 if item i supports item j
$floor_i$	0	1	1 if item i is placed on the ULD floor
m_{x}	$-\infty$	∞	moment of packed items on x-axis ($2\times$)
m_{y}	$-\infty$	∞	moment of packed items on y-axis ($2\times$)
m_{z}	$-\infty$	∞	moment of packed items on z-axis ($2\times$)
o_i	0	$\max(\mathrm{dim}_i^*)^2$	supported floor area of item i
$o_{i,j}^{\mathrm{x}}$	0	$\max(\mathrm{dim}_i^*)$	overlap of items i and j along x-axis
$o_{i,j}^{\mathrm{z}}$	0	$\max(\mathrm{dim}_i^*)$	overlap of items i and j along z-axis
$pack_i$	0	1	1 if the item is packed
$size_i^{\mathrm{x}}$	$\min(\mathrm{dim}_i^*)$	$\max(\mathrm{dim}_i^*)$	size of item i along x-axis
$size_i^{\mathrm{y}}$	$\min(\mathrm{dim}_i^*)$	$\max(\mathrm{dim}_i^*)$	size of item i along y-axis
$size_i^{\mathrm{z}}$	$\min(\mathrm{dim}_i^*)$	$\max(\mathrm{dim}_i^*)$	size of item i along z-axis
$stack_i$	$\min(\mathrm{stack}_i^*)$	$\max(\mathrm{stack}_i^*)$	vertical load-bearing strength of item i
w_u	0	$\sum \mathrm{weight}_i^{\mathrm{grs}}$	total item weight packed on ULD u
w_i	$\mathrm{weight}_i^{\mathrm{grs}}$	$\sum \mathrm{weight}_i^{\mathrm{grs}}$	weight of item i including stacked items
x_i	$\mathrm{start}_u^{\mathrm{x}}$	$\mathrm{start}_u^{\mathrm{x}} + \mathrm{udim}_u^{\mathrm{x}}$	start coordinate of item i on x-axis
x_i'	$\mathrm{start}_u^{\mathrm{x}}$	$\mathrm{start}_u^{\mathrm{x}} + \mathrm{udim}_u^{\mathrm{x}}$	end coordinate of item i on x-axis
y_i	$\mathrm{start}_u^{\mathrm{y}}$	$\mathrm{start}_u^{\mathrm{y}} + \mathrm{udim}_u^{\mathrm{y}}$	start coordinate of item i on y-axis
y_i'	$\mathrm{start}_u^{\mathrm{y}}$	$\mathrm{start}_u^{\mathrm{y}} + \mathrm{udim}_u^{\mathrm{y}}$	end coordinate of item i on y-axis
z_i	$\mathrm{start}_u^{\mathrm{z}}$	$\mathrm{start}_u^{\mathrm{z}} + \mathrm{udim}_u^{\mathrm{z}}$	start coordinate of item i on z-axis
z_i'	$\mathrm{start}_u^{\mathrm{z}}$	$\mathrm{start}_u^{\mathrm{z}} + \mathrm{udim}_u^{\mathrm{z}}$	end coordinate of item i on z-axis

Table 9.10: Decision variables of the CPPack model for ULD u

gravity (CG), we keep the total bin weight w_u as well as the moment of force of all packed items $(m_\mathrm{x}, m_\mathrm{y}, m_\mathrm{z})$.

Data: item dimensions dim_i^*, load-bearing strength stack_i^*, allowed rotations rot_i
Result: set of tuples of possible item dimensions and load-bearing strength $\mathrm{rotations}_i$

$\mathrm{rotations}_i \leftarrow \emptyset$;
if $XYZ \in rot_i$ **then**
    add $(\mathrm{dim}_i^\mathrm{x}, \mathrm{dim}_i^\mathrm{y}, \mathrm{dim}_i^\mathrm{z}, \mathrm{stack}_i^\mathrm{y})$ to $\mathrm{rotations}_i$;
end
if $XZY \in rot_i$ **then**
    add $(\mathrm{dim}_i^\mathrm{x}, \mathrm{dim}_i^\mathrm{z}, \mathrm{dim}_i^\mathrm{y}, \mathrm{stack}_i^\mathrm{z})$ to $\mathrm{rotations}_i$;
end
if $YXZ \in rot_i$ **then**
    add $(\mathrm{dim}_i^\mathrm{y}, \mathrm{dim}_i^\mathrm{x}, \mathrm{dim}_i^\mathrm{z}, \mathrm{stack}_i^\mathrm{x})$ to $\mathrm{rotations}_i$;
end
if $YZX \in rot_i$ **then**
    add $(\mathrm{dim}_i^\mathrm{y}, \mathrm{dim}_i^\mathrm{z}, \mathrm{dim}_i^\mathrm{x}, \mathrm{stack}_i^\mathrm{z})$ to $\mathrm{rotations}_i$;
end
if $ZYX \in rot_i$ **then**
    add $(\mathrm{dim}_i^\mathrm{z}, \mathrm{dim}_i^\mathrm{y}, \mathrm{dim}_i^\mathrm{x}, \mathrm{stack}_i^\mathrm{y})$ to $\mathrm{rotations}_i$;
end
if $ZXY \in rot_i$ **then**
    add $(\mathrm{dim}_i^\mathrm{z}, \mathrm{dim}_i^\mathrm{x}, \mathrm{dim}_i^\mathrm{y}, \mathrm{stack}_i^\mathrm{x})$ to $\mathrm{rotations}_i$;
end

Algorithm 2: Preprocessing for generating all allowed dimensions and the corresponding load-bearing strength of the vertical axis for an item

To simplify the modeling of item rotations, we use a preprocessing step to create a set of tuples (x, y, z, s) containing all valid rotations of the item with its respective dimensions (x, y, z) and the corresponding load-bearing strength (s) of the vertical dimension. To assemble the set we iterate over all possible rotations. See Algorithm 2 for a pseudo-code description of the preprocessing.

We define the CPPack constraint optimization model as follows:

$$\text{minimize} \sum_{i \in I} (1 - pack_i)\mathrm{offload}_i \tag{9.48}$$

subject to

$$(size_i^\mathrm{x}, size_i^\mathrm{y}, size_i^\mathrm{z}, stack_i) \in \mathrm{rotations}_i \qquad i \in I \tag{9.49}$$

$$x_i' = x_i + size_i^\mathrm{x} \qquad i \in I \tag{9.50}$$

$$y_i' = y_i + size_i^\mathrm{y} \qquad i \in I \tag{9.51}$$

$$z_i' = z_i + size_i^\mathrm{z} \qquad i \in I \tag{9.52}$$

$$\mathrm{a} \geq \mathrm{x}x_i + \mathrm{y}y_i + \mathrm{z}z_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{9.53}$$

$$\mathrm{a} \geq \mathrm{x}x_i + \mathrm{y}y_i + \mathrm{z}z'_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{9.54}$$

$$\mathrm{a} \geq \mathrm{x}x_i + \mathrm{y}y'_i + \mathrm{z}z_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{9.55}$$

$$\mathrm{a} \geq \mathrm{x}x_i + \mathrm{y}y'_i + \mathrm{z}z'_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{9.56}$$

$$\mathrm{a} \geq \mathrm{x}x'_i + \mathrm{y}y_i + \mathrm{z}z_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{9.57}$$

$$\mathrm{a} \geq \mathrm{x}x'_i + \mathrm{y}y_i + \mathrm{z}z'_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{9.58}$$

$$\mathrm{a} \geq \mathrm{x}x'_i + \mathrm{y}y'_i + \mathrm{z}z_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{9.59}$$

$$\mathrm{a} \geq \mathrm{x}x'_i + \mathrm{y}y'_i + \mathrm{z}z'_i \qquad i \in I; \langle \mathrm{x}, \mathrm{y}, \mathrm{z}, \mathrm{a} \rangle \in C_u \tag{9.60}$$

$$(pack_i \wedge pack_j) \Rightarrow \begin{array}{l} (x'_i \leq x_j) \vee (x'_j \leq x_i) \vee \\ (y'_i \leq y_j) \vee (y'_j \leq y_i) \vee \\ (z'_i \leq z_j) \vee (z'_j \leq z_i) \end{array} \qquad i, j \in I \tag{9.61}$$

$$m_{\mathrm{x}} = \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \, (x_i + x'_i) \, pack_i \tag{9.62}$$

$$m_{\mathrm{y}} = \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \, (y_i + y'_i) \, pack_i \tag{9.63}$$

$$m_{\mathrm{z}} = \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} \, (z_i + z'_i) \, pack_i \tag{9.64}$$

$$w_u = \sum_{i \in I} \mathrm{weight}_i^{\mathrm{grs}} pack_i \tag{9.65}$$

$$2w_u \underline{\mathrm{x}}_u^{\mathrm{cg}} \leq m_{\mathrm{x}} \leq 2w_u \overline{\mathrm{x}}_u^{\mathrm{cg}} \tag{9.66}$$

$$2w_u \underline{\mathrm{y}}_u^{\mathrm{cg}} \leq m_{\mathrm{y}} \leq 2w_u \overline{\mathrm{y}}_u^{\mathrm{cg}} \tag{9.67}$$

$$2w_u \underline{\mathrm{z}}_u^{\mathrm{cg}} \leq m_{\mathrm{z}} \leq 2w_u \overline{\mathrm{z}}_u^{\mathrm{cg}} \tag{9.68}$$

$$base_{i,j} \Leftrightarrow pack_i \wedge pack_j \wedge (0 \leq y_j - y'_i \leq \epsilon) \qquad i, j \in I \tag{9.69}$$

$$o_{i,j}^{\mathrm{x}} = base_{i,j} \max(0, \min(x'_i, x'_j) - \max(x_i, x_j)) \qquad i, j \in I \tag{9.70}$$

$$o_{i,j}^{\mathrm{z}} = base_{i,j} \max(0, \min(z'_i, z'_j) - \max(z_i, z_j)) \qquad i, j \in I \tag{9.71}$$

$$o_j = \sum_{i \in I} o_{i,j}^{\mathrm{x}} o_{i,j}^{\mathrm{z}} \qquad j \in I \tag{9.72}$$

$$w_i = \mathrm{weight}_i^{\mathrm{grs}} + \sum_{j \in I} \frac{o_{i,j}^{\mathrm{x}} o_{i,j}^{\mathrm{z}}}{o_j} w_j \qquad i \in I \tag{9.73}$$

$$o_{i,j}^{\mathrm{x}} o_{i,j}^{\mathrm{z}} > 0 \Rightarrow stack_i o_j \geq w_j \qquad i, j \in I \tag{9.74}$$

$$(pack_i \wedge pack_j) \Rightarrow \mathrm{dist}_{i,j}^2 \leq \begin{array}{l} \max(0, x_i - x'_j, x_j - x'_i)^2 \\ + \max(0, z_i - z'_j, z_j - z'_i)^2 \end{array} \qquad i, j \in I \tag{9.75}$$

$$pack_i \Rightarrow \begin{array}{l} (\underleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}} \leq x_i \leq \overleftarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}) \wedge \\ (\underleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}} \leq y_i \leq \overleftarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}) \wedge \\ (\underleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}} \leq z_i \leq \overleftarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}}) \end{array} \qquad u \in U; i \in I \tag{9.76}$$

$$pack_i \Rightarrow \begin{array}{l} (\underrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}} \leq x'_i \leq \overrightarrow{\mathrm{x}}_{u,i}^{\mathrm{uld}}) \wedge \\ (\underrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}} \leq y'_i \leq \overrightarrow{\mathrm{y}}_{u,i}^{\mathrm{uld}}) \wedge \\ (\underrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}} \leq z'_i \leq \overrightarrow{\mathrm{z}}_{u,i}^{\mathrm{uld}}) \end{array} \qquad u \in U; i \in I \tag{9.77}$$

$$floor_i \Leftrightarrow pack_i \wedge (y_i \geq \mathrm{start}_u^{\mathrm{y}} + \epsilon) \qquad i \in I \tag{9.78}$$

$$\neg floor_i \Rightarrow o_i \geq \mathrm{supp}_i size_i^{\mathrm{x}} size_i^{\mathrm{z}} \qquad i \in I \tag{9.79}$$

$$base_{i,j} \in \{0, 1\} \qquad i, j \in I \tag{9.80}$$

$$\mathit{floor}_i, \mathit{pack}_i \in \{0,1\} \qquad i \in I \quad (9.81)$$

$$m_{\mathrm{x}}, m_{\mathrm{y}}, m_{\mathrm{z}} \in \mathbb{R} \qquad (9.82)$$

$$o_i, \mathit{size}_i^{\mathrm{x}}, \mathit{size}_i^{\mathrm{y}}, \mathit{size}_i^{\mathrm{z}}, \mathit{stack}_i, w_i \in \mathbb{R}_{\geq 0} \qquad i \in I \quad (9.83)$$

$$o_{i,j}^{\mathrm{x}}, o_{i,j}^{\mathrm{z}} \in \mathbb{R}_{\geq 0} \qquad i, j \in I \quad (9.84)$$

$$w_u \in \mathbb{R}_{\geq 0} \qquad (9.85)$$

$$x_i, x_i', y_i, y_i', z_i, z_i' \in \mathbb{R} \qquad i \in I \quad (9.86)$$

The objective, given in Equation (9.48), is to minimize the penalty from offloaded items. As soon as a solution without offloaded items is found the subproblem is solved successfully and the search can be stopped. In the following we briefly describe the constraint groups. A detailed explanation of the constraints is given in Section 6.2.3.

Equation (9.49) selects a valid rotation and the corresponding load-bearing strength for each item. It is implemented as a table constraint. Equations (9.50) to (9.52) set the item start and end coordinates in relation to the item size. Equations (9.53) to (9.60) ensure that the item is fully inside the bins contour shape, see the explanation along Equation (6.20) and Figure 6.1. Furthermore, Equation (9.61) checks that packed items do not overlap.

Equations (9.62) to (9.68) take care of a valid CG of the packed bin. Therefore, Equations (9.62) to (9.64) calculate the combined moment of force of all packed items for each axis, Equation (9.65) calculates the weight of the packed bin, and Equations (9.66) to (9.68) check that the CG, which is defined as the moment of force divided by the weight, is within the allowed limits for each axis.

Equations (9.69) to (9.74) model the load-bearing of the stacked items. Equation (9.69) detects if two items have suitable vertical positions to form a stack and Equations (9.70) and (9.71) calculate the horizontal overlap along the x and z-axis. Equation (9.72) sums up the overlaps to calculate the total supported area of an item. Equation (9.73) determines the effective weight of an item, i.e., including the weight stacked on top of it. Finally, Equation (9.74) ensures that the load-bearing strengths of all items are respected.

Beyond that, Equation (9.75) enforces the minimum spacing between two packed items, if the items must not be placed in close proximity. Equations (9.76) to (9.77) check that the items are placed within their positioning limits. Finally, Equation (9.78) and (9.79) assure a minimum supported area for each item that is not placed on the floor of the ULD.

Solution search To find valid packings we apply a depth first search on the constraint model. Our branching strategy fixes the items in descending order of their volume. For each item we first fix the variable pack_i followed by the coordinates y, x, z and finally select a rotation. In the left (right) branch pack_i is set to 1 (0). For each axis the branching decision is chosen in the same way. We create a set of *points of interest* (POI), which contains promising coordinates for placing the considered item. The set contains the end coordinates of the already placed items along the given axis plus the lower and upper bounds to place the current item. See Figure 9.2 for an illustration. Crainic et al. (2008) introduced a similar concept called Extreme Points. They show that placing items at these points provides an efficient way to utilize the bin volume. However, in our

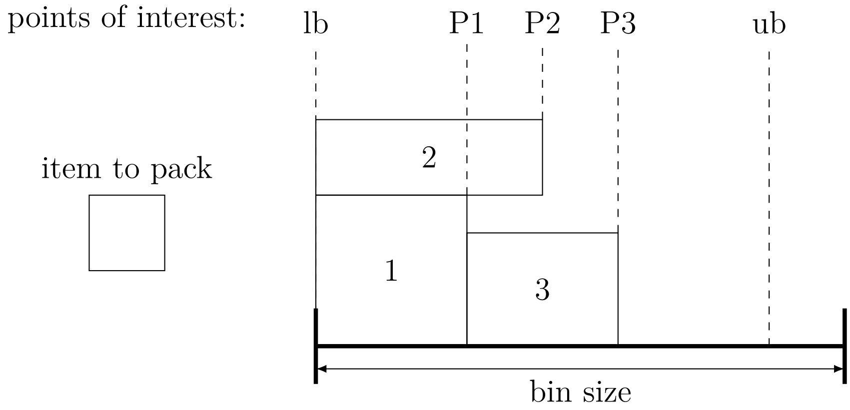


Figure 9.2: Points of interest (POI) along a given axis. Positions considered for branching are the end coordinates of all placed items (P1, P2, P3) as well as the lower (lb) and upper bound (lb) to place the item into the bin.

case we refrain from just placing items at the chosen points because the irregular shaped ULDs and the weight distribution constraints might allow or require different coordinates. Instead, we will use the POI for branching.

Let p denote the integer domain variable of the starting coordinate the considered item should be placed on an axis and P the set of POI along the same axis. In the left branch we set p to its domain minimum:

$$p = \min(p) \tag{9.87}$$

and in the right branch we constrain p to be at least at its next POI:

$$p \geq \min(\{p' | p' \in P : p' > p\}) \tag{9.88}$$

If a solution is found that packs all items, the search is finished. For each found solution that does not pack all items, we restrict the remaining search to solutions that strictly pack more volume than the current solution.

If the search did not finish after a given time limit, we stop it and return the best solution found so far. Furthermore, to save time during future iterations we cache all successfully packed ULDs. Preliminary experiments have shown that a significant part of the BinAssign solution does not change between the iterations of the master problem and thus a lot of time in the subproblem can be saved by caching its results.

9.3.3 Logic-based Benders Decomposition

If the CPPack search presented in the last section can pack all the items assigned to each bin by the master problem, we found a solution to the APP. But, as our master problem only considers the item volume and weight it is likely that the subproblem will not be able to pack all items of a BinAssign solution without further restrictions.

In this section, we introduce a heuristic to generate such restrictions, also known as cuts. Adding those to the BinAssign model will prune infeasible packings from the BinAssign search space. The assignment and the packing phase are then repeated. It can be easily shown that this process will lead to a feasible packing after a finite number of iterations. We note that this procedure has no optimality guarantee, because the constraint program generating the cuts is a heuristic. The overall procedure is known as logic-based Benders Decomposition (LBBD) and was first described by Hooker (2007).

In a setup similar to ours Pisinger and Sigurd (2007) applied LBBD to a two-dimensional knapsack problem. They introduced a valid inequality to their one-dimensional knapsack master problem if a set of items I' assigned to a bin could not be fully loaded in the subproblem, which was a constraint satisfaction problem for two-dimensional packing. Denoting x_i as indicator that item type i is loaded, they add the inequality

$$|I'| - 1 \geq \sum_{i \in I'} x_i \tag{9.89}$$

to the master problem and solve it again. In their setup each item type represents a single item. Therefore, the inequality forbids the current unsuccessful solution in the next run and forces the master problem to assign at least one item less into the bin. Repeating this procedure leads to a feasible solution of the master problem and subproblem in the end. However, this approach would be highly ineffective in our case, because our problems at hand typically contain multiple items of the same type and the inequality in Equation (9.89) does not cut off symmetric solutions. Furthermore, we deal with heterogeneous bins, i.e., each inequality is only valid for one ULD type.

Therefore, we propose an adapted inequality. The underlying idea is the same: if only a subset of the assigned items I' can be packed in a CPPack solution, we add cuts to the BinAssign model, which forbid solutions dominating the packable subset. Let $assign_{u,i}$ denote the number of items of type i assigned to a bin u in a BinAssign solution and $actual_{u,i}$ denote the number of items that could be packed in the CPPack solution.

Technically, we forbid solutions dominating the set of $actual_{u,i}$ for item types where $actual_{u,i} < assign_{u,i}$ in the next LBBD iteration. Therefore, we add one cut for each item type i where $actual_{u,i} < assign_{u,i}$. This cut allows to assign a number of $actual_{u,i}$ items of type i to the bin. More than $actual_{u,i}$ items can only be packed, if at least one of the other item types in I' is assigned strictly less than in the current solution. An illustration of the generated cuts and how they interact is given in Figure 9.3. For item types that could be packed completely, i.e., where $actual_{u,i} = assign_{u,i}$, no cut is added, because packing more items of this type might be possible.

Next, we describe how to model the cut for an item type i that could not be packed completely. Let cut_j denote the allowed number of items of type $j \in I'$ in the cut. For the item type i we set $\text{cut}_i = actual_{u,i} + 1$ and for all other item types $j \in I'$ we set $\text{cut}_j = actual_{u,j}$. Let U' be the set of ULDs of the type that I' could not be packed into. For each ULD $u \in U'$ and item type $j \in I'$ we introduce a new variable $facet_{u,j}$ indicating that the item type j is packed with a number complying with the cut. Finally, we need to make sure that at least one $facet_{u,j}$ holds for each ULD u. Formally, we add the following heuristic cut to the BinAssign model:

$$assign_{u,j} \leq \text{cut}_j - 1 + (1 - facet_{u,j})\text{amount}_j \qquad j \in I'; u \in U' \tag{9.90}$$

$$1 \leq \sum_{j \in I'} facet_{u,j} \qquad u \in U' \tag{9.91}$$

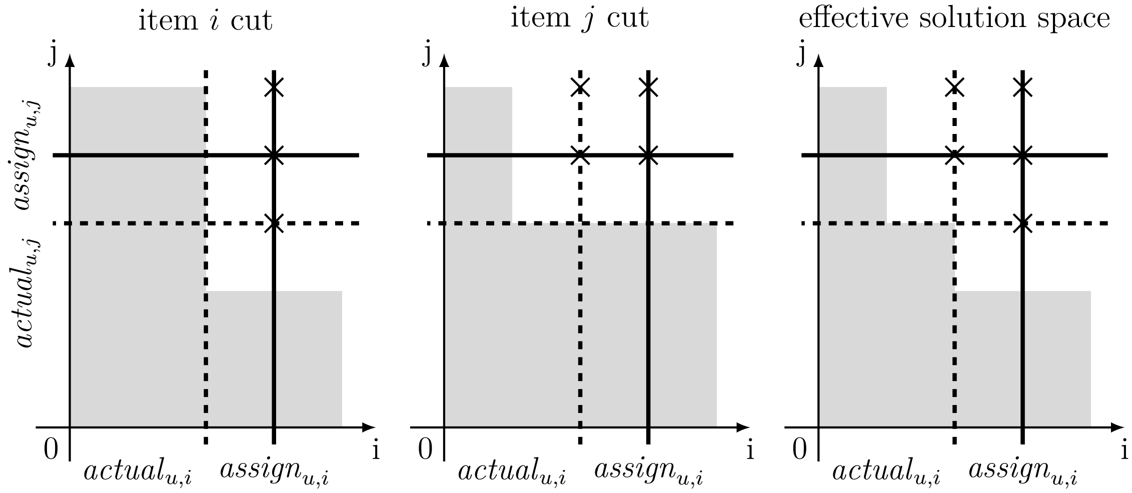


Figure 9.3: Illustration of the created cuts: Two item types i and j are assigned to a bin $assign_{u,*}$ times by the master problem, but can only be packed $actual_{u,*}$ times by the subproblem. For each item we add a cut (left and center sketch) with two facets (one for i and another for j). Thereby, the crossed out solutions are cut off. The solution space for the BinAssign (right sketch) is reduced to the current packable solution or to solutions where at least one item is strictly less often packed into the bin.

9.3.4 Sliding window of item sizes

On average our instances contain more than 700 items compared to a maximum of 100 items used by Pisinger and Sigurd (2007). Due to the large problem sizes, the runs of the LBBD with the cut generation presented in the last section did often not terminate in our preliminary experiments. To reach acceptable runtimes, we therefore further decompose the problem to reduce the search space of each LBBD run.

The idea is to run multiple successive LBBD, which consider only a subset of the items. In the beginning, we consider only large items and move to smaller items after valid packings for the larger items have been found. Therefore, we introduce a sliding window of the item volume starting with the largest present item volume. Items below the window volume are considered in the master problem but not in the subproblems. As soon as a valid packing of the larger items is found the window is moved towards smaller items and all items larger than the window are fixed to their current bins. This way, we care for the hardest to pack items first, which is also the usual approach in practice.

For the width of the window we introduce a parameter *wsize*. Let *ub* denote the upper bound of the volume window, which is initialized to the largest volume of any item type in I. The lower bound of the window is set to $ub/wsize$, which also becomes the upper bound after moving the window. The general procedure of the sliding window is depicted in Figure 9.4. We analyze suitable values for the *wsize* in our evaluation in Section 10.2.3.

One technical remark: For items larger than *ub*, we set the variable $pack_i$ in the CPPack model to 1. Otherwise, the subproblem might offload an item already fixed to the ULD in the BinAssign model.

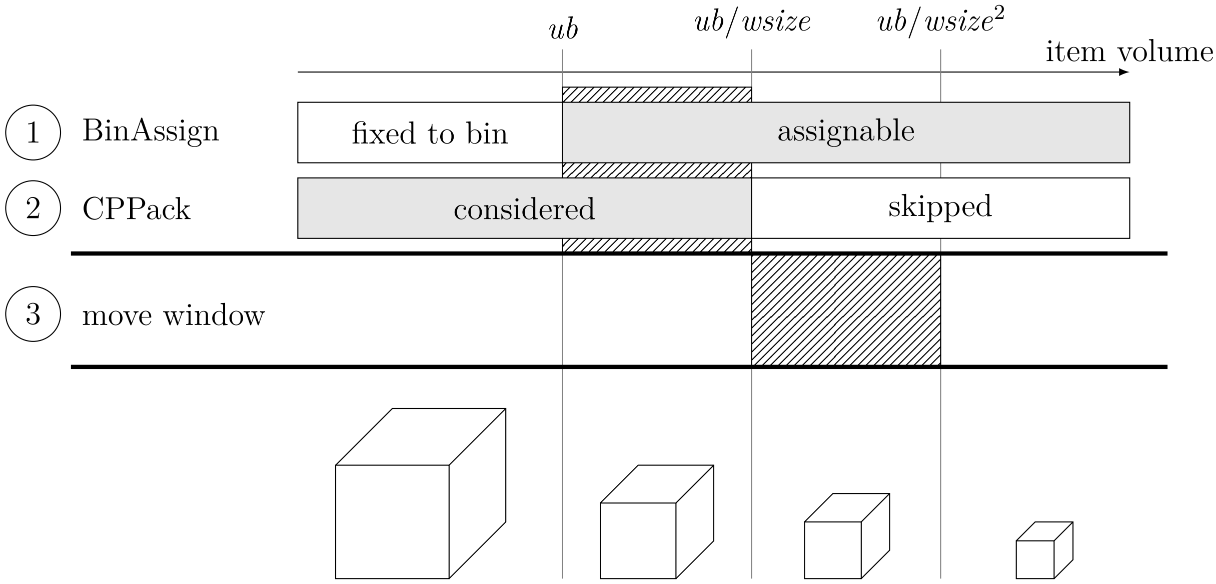


Figure 9.4: Search space reduction by sliding window. In the BinAssign (1) model only items below the volume *ub* are free to be assigned. During the CPPack (2) search only items larger than $ub/wsize$ are considered. If the packing succeeds for all ULDs, the window is moved (3) to the next smaller items.

9.3.5 Estimation of stowage loss

To further speed up the solution process, we add another extension to the master problem. If the subproblem is unable to pack certain items during several LBBD iterations, we assume that these items are probably harder to pack. Because the BinAssign adds items to bins until the bin volume is reached we expect that, to pack the "hard" items, some space in the bin has to be left free. Or, in other words, that these items incur some stowage loss. The stowage loss might vary for different ULD types, because each type has a specific form factor.

As an example, let us take a two-dimensional bin of size $(4, 4)$ and three items of size $(3, 1.5)$. As the bin capacity is 16 and the item volume is 4.5, the BinAssign will assign all three items to the bin. However, only two of the items can be packed into the bin. But, if we would expect a 20 percent stowage loss for the given item type, we could set the volume of the item to $4.5 \cdot 1.2 = 5.4$ in the BinAssign, which would then only assign two items to the bin. However, we note that the real stowage loss of an individual item cannot be determined upfront and depends on the combination of items.

To respect the stowage loss in the master problem, we add a second set of volume limiting constraints to the BinAssign model. In these constraints each item i is weighted by its estimated volume consumption $\text{stowage}_{v,i}$, which is the item's physical volume plus the estimated stowage loss in a ULD of type v. The limit is the bin capacity volume_u. Let U_v denote the set of ULDs of type v. We add the following constraints to the BinAssign model:

$$\text{volume}_u \geq \sum_{i \in I} \text{stowage}_{v,i}\, assign_{u,i} \qquad u \in U_v \qquad (9.92)$$

The question remains how to choose a suitable stowage loss for a given item. For items outside the sliding window we assume no stowage loss. The larger items have already been successfully packed to their assigned bin and smaller items are not considered for packing yet, i.e., we have no information about their packing behavior. We estimate the stowage loss of an item within the sliding window based on the frequency the item appears in the cuts for a specific ULD type. Starting with the original item volume, we multiply the parameter $\text{stowage}_{v,i}$ by a constant factor each time the item is part of a generated cut. Hence, the more often an item occurs in an unpackable bin, the higher its stowage loss gets. We reset the stowage loss of each item when the sliding window is moved forward, i.e., when a packable assignment for all currently considered items is found. We analyze suitable values for the stowage loss factor in our evaluation in Section 10.2.3.

9.3.6 Overall procedure

Finally, we put all the introduced components together and describe our overall procedure for solving the air cargo palletization. We give a pseudocode description in Algorithm 3. The outer loop (Lines 3 to 21) realizes the sliding window of item sizes. For each entity of the sliding window we start with a reset stowage loss and perform an LBBD run (Lines 7 to 18) until all items inside the window can be packed. In each LBBD iteration (Lines 10 to 17) we solve the BinAssign model and try to pack all selected bins with one CPPack search per bin. If the CPPack cannot pack all items, we derive suitable cuts and increase the items' stowage loss. If no feasible packing solution is found for a bin, we add the full set of items assigned to the bin as a cut. The full procedure ends, when the sliding window reached the smallest items, i.e., when all items have been packed in the subproblem.

9.4 Weight and balance

The final step of our sequential load planning approach is the weight and balance for each flight. At this stage the ULDs for the different transport segments have been assembled and weighed. Now, for each ULD we need to select a loading position in the aircraft on each leg of of the flight, such that all technical and regulatory restrictions are fulfilled. We denote this step as UWAB in the following.

The remainder of this section is structured as follows. In Section 9.4.1 we introduce some necessary preprocessing steps to determine suitable model parameters. After that, we present the employed MIP model to assign the ULDs to loading positions in Section 9.4.2.

9.4.1 Preprocessing

Before introducing the model itself, we present two preprocessing steps. These steps derive required model parameters from the solution of the previous air cargo palletization step.

```
Data: problem instance for one transport segment k, items I_k
Result: air cargo palletization solution
ub ← max({volume_i | i ∈ I_k});
lb ← min({volume_i | i ∈ I_k});
while ub ≥ lb do
    stowage ← volume;
    set sliding window of item sizes (Section 9.3.4);
    items ← {i | i ∈ I_k : ub/wsize < volume_i ≤ ub};
    do
        solve BinAssign model (Section 9.3.1);
        offloads ← false;
        for each bin in BinAssign solution do
            solve CPPack model (Section 9.3.2) for items in bin;
            if not all items could be packed then
                generate cuts from bin and add to BinAssign (Section 9.3.3);
                increment stowage loss of items in bin (Section 9.3.5);
                offloads ← true;
            end
        end
    while offloads = true;
    fix items to their currently assigned bin;
    ub ← ub/wstep;
end
```

Algorithm 3: Overall procedure to solve the air cargo palletization

Offload penalties Each item to load has a specific penalty that applies if it is offloaded. As we only deal with ULDs during weight and balance, we sum up the penalties of the items packed into each ULD to determine the offload penalty of the ULD.

Minimum spacing of ULDs During air cargo palletization we already dealt with incompatible items that have to be placed into separate ULDs. Some goods also require not to be loaded into adjacent ULDs or to have a minimum spacing within the aircraft. Accordingly, we calculate a minimum spacing between two ULDs as the maximum of all pairwise minimum distances of items in the considered ULDs.

9.4.2 Mixed-integer model

We solve the weight and balance by applying the full Weight and Balance Model (WBM) as presented in Section 7.2.5. Table 9.11 recaps the sets and parameters of the model and Table 9.12 presents its decision variables. The model primarily decides about the loading position of each ULD on each leg $x_{u,p,l}$ or if the ULD is offloaded x_u^{off}. Auxiliary variables limiting the aircraft center of gravity (CG) capture the rotational moment of the loaded aircraft m_l, the forward (aft) deviation from the optimal CG $m_l^{\text{lng-}}$ ($m_l^{\text{lng+}}$), the moment of inertia m_l^{mi}, and the total weight of the aircraft w_l. Further variables track the necessary reloading operations after each leg ($q_{l,p}$, $r_{l,p}^{\text{un}}$, $r_{l,p}^{\text{on}}$, r_l^{dist}), the related loading costs ops_l, and the required extra fuel cost due to imbalances $fuel_l^+$.

For completeness, we repeat the WBM with minor adjustments for this step:

$$\text{minimize} \sum_{u \in U} \mathrm{a}_u^{\text{off}} x_u^{\text{off}} + \sum_{l \in L} fuel_l^+ + \sum_{l \in L} ops_l + \sum_{l \in L} \mathrm{a}_l^{\text{mi}} m_l^{\text{mi}} \tag{9.93}$$

subject to

$$x_u^{\text{off}} = 1 - \sum_{p \in \text{loadable}_u} x_{u,p,l} \qquad u \in U; l \in \text{legs}_u \tag{9.94}$$

$$1 \geq \sum_{u \in U} x_{u,p,l} \qquad p \in P; l \in L \tag{9.95}$$

$$1 \geq \sum_{u \in U} (x_{u,p,l} + x_{u,p',l}) \qquad \langle p, p' \rangle \in \mathrm{O}; l \in L \tag{9.96}$$

$$\mathrm{MAX}_l \geq \sum_{u \in U} \text{weight}_u \sum_{p \in P} x_{u,p,l} \qquad l \in L \tag{9.97}$$

$$\text{limit}_n \geq \sum_{u \in U} \text{weight}_u \sum_{p \in P} \mathrm{f}_{n,p} x_{u,p,l} \qquad n \in N; l \in L \tag{9.98}$$

$$m_l = \begin{array}{l} \mathrm{BA}^{\text{ac}} \mathrm{w}^{\text{OEW}} + \mathrm{BA}_l^{\text{fuel}} \mathrm{FW}_l \\ + \sum_{u \in U} \text{weight}_u \sum_{p \in P} \mathrm{BA}_p^{\text{lng}} x_{u,p,l} \end{array} \qquad l \in L \tag{9.99}$$

$$w_l = \mathrm{w}^{\text{OEW}} + \mathrm{FW}_l + \sum_{u \in U} \text{weight}_u \sum_{p \in P} x_{u,p,l} \qquad l \in L \tag{9.100}$$

$$\mathrm{BA}_l^{\text{OPT}} w_l = m_l + m_l^{\text{lng+}} - m_l^{\text{lng-}} \qquad l \in L \tag{9.101}$$

$$fuel_l^+ = \text{cost}_l^{fuel-} m_l^{\text{lng-}} + \text{cost}_l^{fuel+} m_l^{\text{lng+}} \qquad l \in L \tag{9.102}$$

Set/Parameter	Description
L	flight legs
N	set of cumulative weight constraints in the aircraft ($n \in N$)
P	loading positions in aircraft
U	ULDs to load
$\mathrm{a}_l^{\mathrm{dist}} \in \mathbb{R}$	cost of working time to move a ULD by one position after leg l
$\mathrm{a}_l^{\mathrm{mi}} \in \mathbb{R}$	cost of additional fuel depending on the MI on leg l
$\mathrm{a}_u^{\mathrm{off}} \in \mathbb{R}$	penalty when offloading ULD u
$\mathrm{a}_l^{\mathrm{on}} \in \mathbb{R}$	cost of one ULD loading operation after leg l
$\mathrm{a}_l^{\mathrm{un}} \in \mathbb{R}$	cost of one ULD unloading operation after leg l
$B_p \subseteq P$	blocking positions when (un)loading a ULD at position p
$\mathrm{BA}^{\mathrm{ac}} \in \mathbb{R}$	longitudinal balance arm of the empty aircraft (OEW)
$\mathrm{BA}_l^{\mathrm{fuel}} \in \mathbb{R}_{\geq 0}$	longitudinal balance arm of the aircraft's fuel tanks
$\mathrm{BA}_p^{\mathrm{lat}} \in \mathbb{R}$	lateral balance arm of loading position p
$\mathrm{BA}_p^{\mathrm{lng}} \in \mathbb{R}$	longitudinal balance arm of loading position p
$\mathrm{BA}_l^{\mathrm{MAX}} \in \mathbb{R}_{\geq 0}$	maximum allowed lateral CG imbalance on leg l
$\mathrm{BA}_l^{\mathrm{OPT}} \in \mathbb{R}$	optimal longitudinal CG on leg l
$\mathrm{cost}_l^{fuel-} \in \mathbb{R}_{\geq 0}$	cost of extra fuel for forward CG deviations on leg l
$\mathrm{cost}_l^{fuel+} \in \mathbb{R}_{\geq 0}$	cost of extra fuel for aft CG deviations on leg l
$\mathrm{dist}_p \in \mathbb{R}_{\geq 0}$	distance between position and door
$\mathrm{dist}_{p,p'} \in \mathbb{R}_{\geq 0}$	distance between loading positions
$\mathrm{f}_{n,p} \in \mathbb{R}$	factor of loading position p for constraint n
$\mathrm{FW}_l \in \mathbb{R}_{\geq 0}$	fuel weight loaded on leg l
$\mathrm{legs}_u \subseteq L$	legs the ULD will be on board
$\mathrm{limit}_n \in \mathbb{R}$	cumulative weight limit of constraint n
$\mathrm{loadable}_u \subseteq P$	loading positions available for ULD u
$\mathrm{MAX}_l \in \mathbb{R}$	maximum loadable ULD weight on leg l
$\mathrm{MAXCG}_l \in \mathbb{R}$	aft limit of the longitudinal CG on leg l
$\mathrm{min}_{u,u'} \in \mathbb{R}_{\geq 0}$	minimum distance of separation constraint
$\mathrm{MINCG}_l \in \mathbb{R}$	forward limit of longitudinal CG on leg l
$\mathrm{O} \subseteq P \times P$	set of overlapping loading positions ($\langle p, p' \rangle \in \mathrm{O}$)
$R_p \subseteq P$	set of loading positions that position p depends on
$\mathrm{S} \subseteq U \times U$	set of ULDs to separate ($\langle u, u' \rangle \in \mathrm{S}$)
$\mathrm{w}^{\mathrm{OEW}} \in \mathbb{R}_{\geq 0}$	operating empty weight (OEW) of the aircraft
$\mathrm{weight}_u \in \mathbb{R}_{\geq 0}$	total weight of ULD u

Table 9.11: Parameters of the weight and balance model

Variable	**Description**
$fuel_l^+ \in \mathbb{R}_{\geq 0}$	cost of required extra fuel on leg l
$m_l \in \mathbb{R}_{\geq 0}$	longitudinal rotational moment of loaded aircraft on leg l
$m_l^{\text{lat-}} \in \mathbb{R}_{\geq 0}$	left lateral imbalance from aircraft center on leg l
$m_l^{\text{lat+}} \in \mathbb{R}_{\geq 0}$	right lateral imbalance from aircraft center on leg l
$m_l^{\text{lng-}} \in \mathbb{R}_{\geq 0}$	forward deviation of rotational moment from optimal CG on leg l
$m_l^{\text{lng+}} \in \mathbb{R}_{\geq 0}$	aft deviation of rotational moment from optimal CG on leg l
$m_l^{\text{mi}} \in \mathbb{R}_{\geq 0}$	moment of inertia of the loaded ULDs on leg l
$ops_l \in \mathbb{R}_{\geq 0}$	extra handling cost after leg l
$q_{l,p} \in \{0,1\}$	1 if loading position p needs to be cleared after leg l
$r_{l,p}^{\text{un}} \in \{0,1\}$	1 if a ULD is unloaded from position p after leg l
$r_{l,p}^{\text{on}} \in \{0,1\}$	1 if a ULD is loaded on position p after leg l
$r_l^{\text{dist}} \in \mathbb{R}_{\geq 0}$	total ULD move distance after leg l
$w_l \in \mathbb{R}_{\geq 0}$	total weight of loaded aircraft of leg l
$x_{u,p,l} \in \{0,1\}$	1 if ULD u is loaded at position p on leg l
$x_u^{\text{off}} \in \{0,1\}$	1 if ULD u is not transported

Table 9.12: Decision variables of the weight and balance model

$$m_l \leq \text{MAXCG}_l w_l \qquad l \in L \qquad (9.103)$$

$$m_l \geq \text{MINCG}_l w_l \qquad l \in L \qquad (9.104)$$

$$m_l^{\text{lat+}} = \sum_{p \in P} \text{weight}_u \sum_{u \in U} \text{BA}_p^{\text{lat}} x_{u,p,l} + m_l^{\text{lat-}} \qquad l \in L \qquad (9.105)$$

$$\text{BA}_l^{\text{MAX}} w_l \geq m_l^{\text{lat+}} + m_l^{\text{lat-}} \qquad l \in L \qquad (9.106)$$

$$m_l^{\text{mi}} = \sum_{u \in U} \text{weight}_u \sum_{p \in P} (\text{BA}_l^{\text{OPT}} - \text{BA}_p^{\text{lng}})^2 x_{u,p,l} \qquad l \in L \qquad (9.107)$$

$$\sum_{u \in U} x_{u,p,l} \leq \sum_{p' \in R_p} \sum_{u \in U} x_{u,p',l} \qquad p \in P; l \in L \qquad (9.108)$$

$$1 \geq \sum_{\substack{p' \in P: \\ \text{dist}_{p,p'} < \text{min}_{u,u'}}} x_{u,p,l} + x_{u',p',l} \qquad \langle u, u' \rangle \in \text{S}; p \in P; l \in L \qquad (9.109)$$

$$q_{l,p} \geq x_{u,p,l} - x_{u,p,l+1} \qquad u \in U; p \in P; l \in L \qquad (9.110)$$

$$q_{l,p} \geq x_{u,p,l+1} - x_{u,p,l} \qquad u \in U; p \in P; l \in L \qquad (9.111)$$

$$q_{l,p} \leq q_{l,p'} \qquad p \in P; p' \in B_p; l \in L \qquad (9.112)$$

$$q_{l,p} \leq q_{l,p'} \qquad p \in P; p' \in B_p; l \in L \qquad (9.113)$$

$$r_{l,p}^{\text{un}} \geq q_{l,p} + \sum_{u \in U} x_{u,p,l} - 1 \qquad p \in P; l \in L \qquad (9.114)$$

$$r_{l,p}^{\text{on}} \geq q_{l,p} + \sum_{u \in U} x_{u,p,l+1} - 1 \qquad p \in P; l \in L \qquad (9.115)$$

$$r_l^{\text{dist}} = \sum_{p \in P} \text{dist}_p(r_{l,p}^{\text{un}} + r_{l,p}^{\text{on}}) \qquad l \in L \quad (9.116)$$

$$ops_l = \text{a}_l^{\text{un}} \sum_{p \in P} r_{l,p}^{\text{un}} + \text{a}_l^{\text{on}} \sum_{p \in P} r_{l,p}^{\text{on}} + \text{a}_l^{\text{dist}} r_l^{\text{dist}} \qquad l \in L \quad (9.117)$$

$$fuel_l^+, ops_l, r_l^{\text{dist}} \in \mathbb{R}_{\geq 0} \qquad l \in L \quad (9.118)$$

$$m_l, m_l^{\text{lat-}}, m_l^{\text{lat+}}, m_l^{\text{lng-}}, m_l^{\text{lng+}}, m_l^{\text{mi}}, w_l \in \mathbb{R}_{\geq 0} \qquad l \in L \quad (9.119)$$

$$q_{l,p}, r_{l,p}^{\text{un}}, r_{l,p}^{\text{on}} \in \{0,1\} \qquad l \in L; p \in P \quad (9.120)$$

$$x_{u,p,l} \in \{0,1\} \qquad u \in U; p \in P; l \in L \quad (9.121)$$

$$x_u^{\text{off}} \in \{0,1\} \qquad u \in U \quad (9.122)$$

The objective, given in Equation (9.93), is to minimize the sum of additional costs of the flight. These costs may arise from offloaded ULDs, the extra fuel consumed due to imbalance, extra handling operations to reload ULDs inside the aircraft between legs, and a penalty proportional to the moment of inertia of the aircraft. In the following we briefly describe the restrictions. For an in depth explanation and motivation of the constraints we refer to Section 7.2.3.

The unique assignment of each ULD is modeled in Equations (9.94) and (9.95). The mutual exclusion of overlapping loading positions is enforced by Equation (9.96). Equations (9.97) and (9.98) track the total and cumulative weights of loaded ULDs. Equations (9.99) to (9.104) model the extra fuel required due to the position of the longitudinal CG of the aircraft. The lateral imbalance of the aircraft is restricted with Equations (9.105) and (9.106). Equation (9.107) models the moment of inertia of the aircraft. Interdependent loading positions are enforced by Equation (9.108). Minimum required distances of ULDs are handled within Equation (9.109). Finally, Equations (9.110) to (9.117) model the required handling operations to load and unload the ULDs between the respective legs of the flight.

9.5 Summary

In this chapter, we introduced a first solution approach for the Air Cargo Load Planning Problem (ACLPP) denoted as SeqACLPP. It consists of a sequential four-step planning pipeline that follows the current manual processes in operational practice.

For the first step, dealing with the aircraft configuration, denoted as VWAC, we presented a MIP model that selects the number of ULDs of each type to build for each transport segment of a flight based on aggregated booking data. The selected ULDs together with the arrival times of the cargo and departure time of the flights build the input for the next step.

To solve the second step, dealing with the build-up scheduling, denoted as RHBS, we introduced a rolling horizon planning approach that determines the ULD build-up start times for all flights departing during the planning horizon. In each round we apply a MIP model solving one shift while also taking the demand of the next n shifts into account.

For the third step, dealing with the air cargo palletization, denoted as BDAP, we realized a Logic-based Benders Decomposition approach. The LBBD master problem is a MIP containing a vector container loading problem that assigns items to ULDs. In the master problem all non-geometric aspects like weights, time, sorting and compatibilities are handled. The LBBD subproblem, which is solved for each ULD individually, is implemented as a constraint program and solves the three-dimensional knapsack problem. In it we deal with all the geometric aspects as well as stacking and stability issues. If some items cannot be packed into the ULD in a subproblem, we heuristically derive cuts, add them to the master problem, and solve it again. To make the solution process come to an end within a reasonable amount of time we introduced two extensions. The first reduces the search space by considering only a part of the items during each iteration. The second increases the virtual volume of items that repeatedly cause problems during the packing.

For the forth step, dealing with the weight and balance, denoted as UWAB, we presented a MIP model that assigns the weighed ULDs to loading positions of the aircraft. The model can handle flights with an arbitrary number of legs at once and minimizes the total cost caused by additional fuel required due to aircraft imbalances and necessary reloading operations at stop-over airports to access ULDs.

10 Evaluation

In this chapter, we describe our experimental setup and the achieved results. We implemented the SeqACLPP approach introduced in Chapter 9 and evaluated it on the benchmark instances described in Chapter 8. To evaluate its applicability for operational practice, we compare the achieved solution quality to real-world solutions. Furthermore, we survey the achieved solution quality for the introduced scenarios and discuss the required runtimes. Before we ran our experiments, we performed an extensive parameter tuning to choose suitable configuration settings for the SeqACLPP approach.

This chapter is organized as follows: First, we describe our implementation and testing environment in Section 10.1. Afterwards, we introduce our preliminary tests to determine suitable parameter settings values for the SeqACLPP approach in Section 10.2. Finally, we describe the results achieved with the SeqACLPP approach in Section 10.3.

10.1 Implementation and testing environment

We implemented all presented models in C++ using the Visual Studio 2012 compiler with standard Release mode settings. All executables were built in 32-bit mode. As MIP solver we employed Gurobi in version 7.0.1. Gurobi was set to use only a single thread, to stop at a 1 percent MIP gap, and to spend 95 percent of the time with heuristics. All other parameters remained at their default values. For the constraint program introduced in Section 9.3.2 we used Gecode, an open-source constraint programming framework, in version 4.4.0 with its standard branch and bound search.

We performed our experiments on two types of machines. The smaller machines had one Intel Core i5-760 processor with 4 cores clocked at 2.8 GHz and equipped with 8 MB cache. Each machine had 8 GB main memory and was running on a 64-bit Windows 7 Professional. The bigger machine had two Intel Xeon E5-2650 v2 processors each with 8 cores clocked at 2.6 GHz and equipped with 20 MB cache. The machine had 128 GB main memory and was running on a 64-bit Windows Server 2012. In total we employed six machines of the smaller and one of the bigger type. All machines were simultaneously executing as many solution processes as they had cores, i.e., all experiments were performed under full load. Accordingly, the presented runtimes can be seen as a conservative benchmark originating from adverse conditions. The differences of the average runtimes seen across all the used machines were less than 15 percent. Therefore, we omit a breakdown of the runtimes by machines in the following.

Option	Value
Aircraft configuration (VWAC)	
estimated achievable volume utilization in percent	66
Build-up scheduling (RHBS)	
planning horizon in shifts	3
MIP time limit in seconds	900
Palletization (BDAP)	
size of the sliding window of item sizes	1.1
stowage loss increment factor	1.02
BinAssign initial time limit in seconds	300
BinAssign warm start time limit in seconds	60
CPPack time limit in seconds	1
Weight and balance (UWAB)	
MIP time limit in seconds	300

Table 10.1: Parameters of the SeqACLPP and their values chosen for evaluation

10.2 Preliminary parameter tuning

As shown in Table 10.1 the SeqACLPP approach is configurable in many ways. Before running our main evaluation, we performed preliminary experiments to identify and select suitable values for those parameters. In the following we describe those experiments and introduce the chosen parameter values for the aircraft configuration, build-up scheduling, palletization, and weight and balance steps.

10.2.1 Aircraft configuration

The volume-and-weight-based aircraft configuration (VWAC) step, presented in Section 9.1, operates on the estimated achievable volume utilization of the ULDs. In this section, we discuss how to find a suitable utilization parameter value. Our hypothesis is that the net load factor (NLF) of previous flights provides a good estimate.

In a first test, we verified this hypothesis by performing tests for different parameter settings on the real world flights of our dataset A base scenario (A-base). In the solutions seen in practice for the A-base flights, the NLF was around 55 percent. Therefore, we varied the utilization parameter around this value in our experiments. The resulting numbers of selected ULDs are shown in Figure 10.1. As one would expect, lower achievable utilizations lead to more selected ULDs. Furthermore, the aircraft configurations selected by the model with 55 percent utilization come closest to the real-world results. Therefore, we argue that the VWAC produces realistic configurations if it is parameterized with a realistic NLF.

Looking into the details at 55 percent utilization we see that the VWAC selects about 24 percent more of the large PGE ULDs (141 instead of 114) and about 34 percent less of the small AKE ULDs (52 instead of 79) than were built in reality. We attribute this to

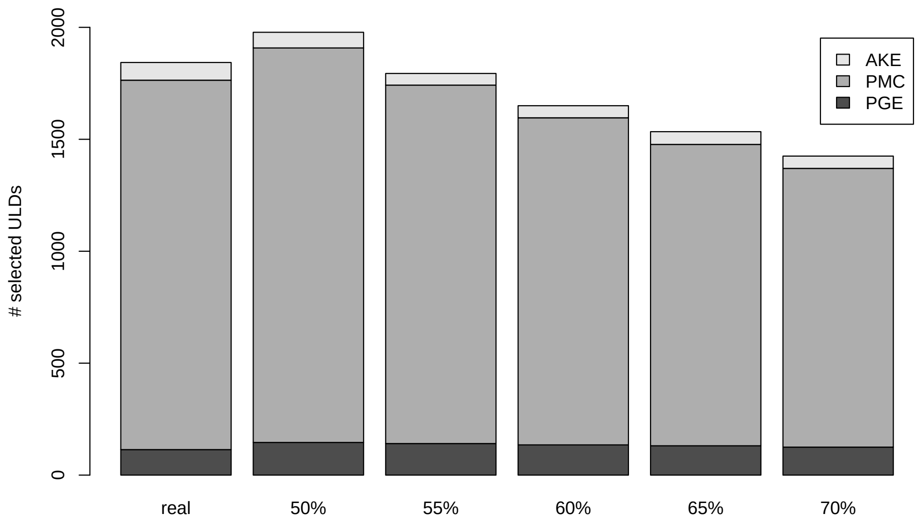


Figure 10.1: Number of selected ULDs in practice (real) and for VWAC runs with different expected volume utilizations. Shown are all flights of the A-base scenario.

the observation that larger containers are easier to pack with a high utilization, because they provide more options to arrange the items. To keep things simple, we assume the same utilization for all ULD types in the following. Thereby, we slightly overestimate the capacity of small ULD types and underestimate it for large ULD types.

Setting the utilization parameter to 55 percent in the SeqACLPP would not be a good choice, because the utilization achieved by our palletization step (BDAP) introduced in Section 9.3 is probably different from the one seen in practice. Therefore, we ran a second test to survey the average achievable volume utilization of the BDAP. We ran this test on the individual segments of the A-base flights without limiting the number of usable ULDs. During these experiments an NLF of around 66 percent could be achieved. Therefore, in the remainder of this work, we set the volume capacities of all ULD types to 66 percent of their geometric volume, when running the VWAC.

The model solves for almost all instances in well under a second. Therefore, we set no time limit for the MIP solver.

10.2.2 Build-up scheduling

One question regarding the rolling horizon build-up scheduling (RHBS) step presented in Section 9.2 is how to choose the planning horizon. We do not expect that there is a general answer to this question. More likely a suitable planning horizon depends on the structure of the shifts and the flight plan. To choose a planning horizon that fits our needs, we evaluated different planning horizons with respect to their solution quality and runtime.

Based on the aircraft configuration solutions of the dataset A base scenario (A-base) from Section 10.2.1 we ran the RHBS with a planning horizon ranging between one shift (8 hours) to six shifts (48 hours). A horizon of n shifts means that ULDs of flights departing during shift m can be built during shifts $m-n$ to m. We note that a planning

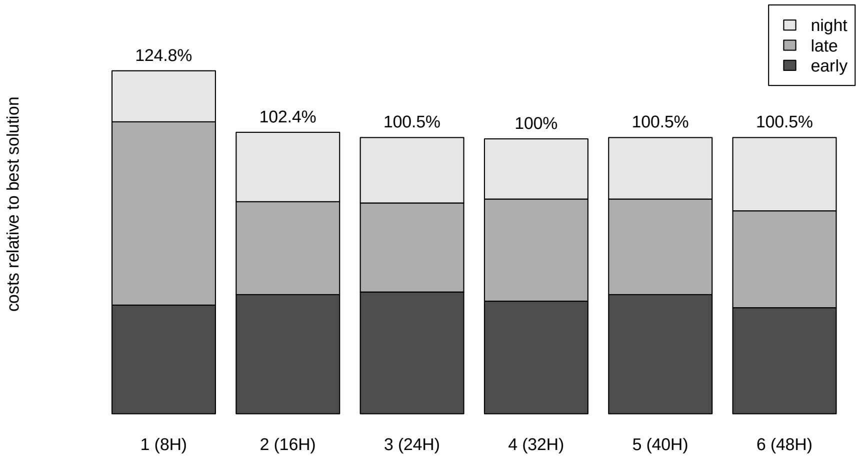


Figure 10.2: Costs of crews required for different planning horizons

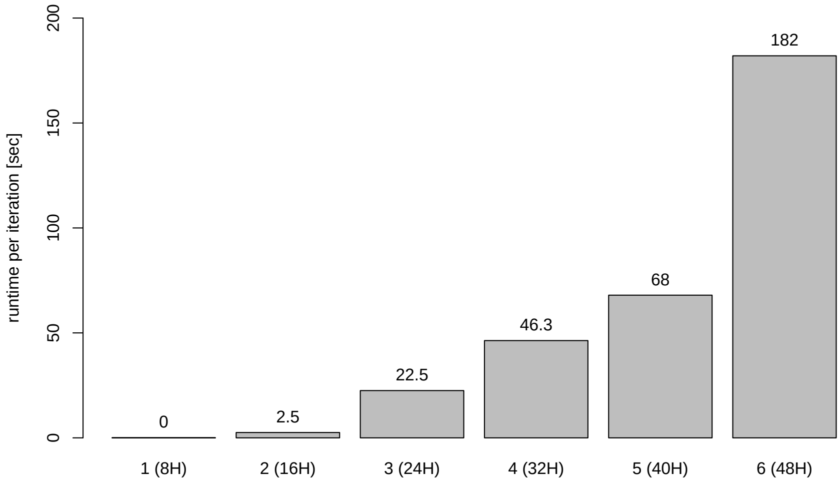


Figure 10.3: Runtimes per shift for different planning horizons

horizon of 0 is not useful. In this case it would not be possible to build ULDs of flights departing early in a shift during the previous shift, which is necessary in practice.

As it is common practice we use three shifts per day: an early (06:00 to 14:00), late (14:00 to 22:00), and night (22:00 to 06:00) shift. For our experiments, we expect the cost of a night shift to be 20 percent above the cost of the other shifts. An overview of the costs of the selected crews per shift for different planning horizons is given in Figure 10.2. All numbers are given relative to the best found solution, which had a planning horizon of four shifts (32 hours).

As expected, shorter planning horizons give worse results, but from three shifts on the quality is nearly constant. Interestingly, the number of late shifts for a planning horizon of one shift is almost twice as high as for all others. We attribute this to two effects. First, instead of the more expensive night shifts the model selects more late shifts. Accordingly, a planning horizon of 1 shift shows the lowest number of night shifts. The second effect is that, with a planning horizon of one shift, flights departing during the late shifts cannot be built during the night shift before, although a significant amount of their cargo has already arrived. But, during the night there are only a few departing flights in the schedule of the A-base scenario and thus the selected crews often cannot be fully utilized.

Besides the solution quality, we are interested in the runtime of the RHBS to decide about a suitable planning horizon. Remember from Figure 9.1, that we solve the Build-up Scheduling Model (BSM) in a rolling horizon approach, i.e., we execute it successively for each shift with a lookahead of a planning horizon of n shifts. Figure 10.3 shows the average runtime required for one BSM run for different planning horizons. We stopped the search after 900 seconds or when the MIP gap became smaller than 1 percent. As one would expect the runtime grows exponentially for larger planning horizons. We note that for planning the shifts of one week we must run 21 iterations, i.e., planning with a 48 hours horizon takes more than 1 hour.

Given the above insights into solution quality and runtime we select a planning horizon of 3 shifts, or 24 hours respectively, for the remainder of this work.

10.2.3 Palletization

The Logic-based Benders Decomposition (LBBD) solving the palletization (BDAP), which we presented in Section 9.3.3, has five parameters for which we need to identify suitable values. First, there is the size of the sliding window of item volume. During each iteration, only items within the window are considered for assignment and packing. Larger items have a fixed assignment while smaller items are not considered during packing.

The second parameter is the stowage loss factor. After each failed ULD packing, i.e., where some items could not be placed into the assigned ULD, we multiply the virtual volume of all items assigned to this ULD by an expected stowage loss factor. Accordingly, in upcoming BinAssign iterations, less of these items will be assigned to the ULD and thus it should become easier to find a feasible packing.

The other parameters limit the runtime: a cold start and warm start time limit for the BinAssign MIP model and a time limit for the CPPack search.

To find suitable parameter settings with tolerable effort we randomly selected 30 flights from our datasets, 5 of each scenario, and solved those flights for different parameter values. We repeated the experiments three times and show the average values in the following. In the remainder of this section we present the achieved results and justify our choice of parameters.

Sliding window size

First, we analyze the effects of different sizes of the sliding window of item volume. As introduced in Algorithm 3, the window size determines the number of successive runs of the LBBD to process all items, starting with the biggest items in the first run and then moving to smaller items. The window during an LBBD run is defined as $(ub/wsize, ub]$ for a window size $wsize$ and upper bound ub. As stated above, only items with a volume within the window are considered for the bin assignment and packing. Thus, the window size determines the number of items handled during each iteration. A larger window size leads to larger subproblems in each iteration, but also fewer shifts of the window are

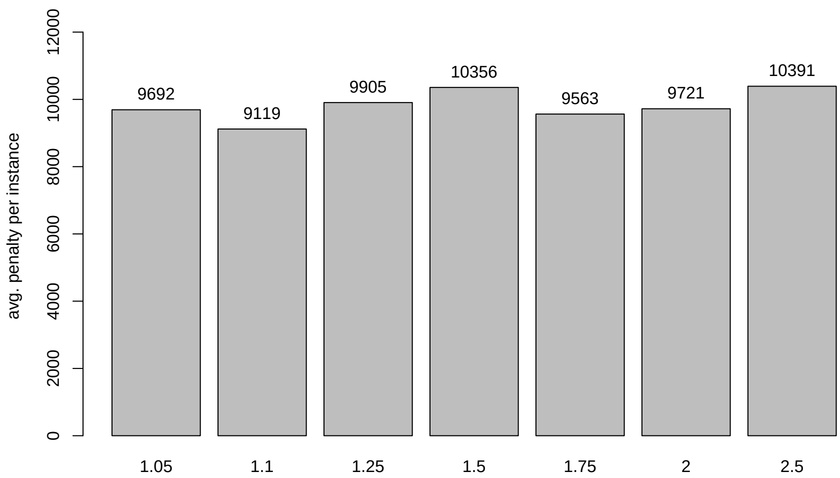


Figure 10.4: Penalties for different window sizes

necessary until all items have been covered. On our sample instances we evaluated the SeqACLPP approach for varying values of *wsize* ranging between 1.05 and 2.5. In the following, we discuss the impact of the different values on solution quality and required effort.

The first thing we note is that there is no clear relationship between the window size and solution quality. Figure 10.4 shows the average offload penalty of our sample instances for the different sizes. All values are within a range of 12 percent. One explanation of this effect might be the large number of items in the instances. Even if only a part of the items is considered at each sliding window position, the remaining solution space seems to contain many alternative solutions of similar quality.

As one would expect, the required number of total LBBD iterations, i.e., the sum of iterations of all LBBD runs, of Algorithm 3 decreases with a larger window size. Figure 10.5 shows the average total LBBD iterations it takes to solve the sample instances for different window sizes. However, this decrease is not linear. Larger window sizes require less LBBD runs with different volume windows, but lead to more infeasible packings and thus more LBBD iterations in each run for adding cuts before a feasible packing is found. Figure 10.6 presents the average number of iterations the LBBD takes for a run before the packings can be solved successfully.

Furthermore, there is a tradeoff between window size and solution runtime, with a window size of 1.25 showing the shortest runtimes. Figure 10.7 shows the respective average runtimes. At small window sizes, the high number of total LBBD iterations lead to high runtimes. Larger window sizes also increase runtimes, because more items have to be considered in the BinAssign model and thus each iteration takes longer. Given the results above, we choose a value of 1.1 as the size of the sliding window in the remainder of this work. This setting showed the lowest penalty and the second best runtimes.

Stowage loss factor

Next, we analyze the impact of different settings for the stowage loss factor on solution quality and effort. As introduced in Section 9.3.1, in the LBBD master problem, the

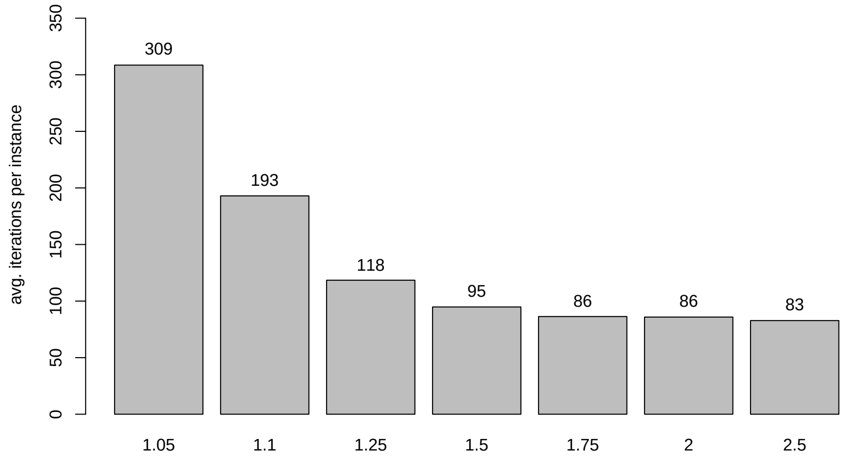


Figure 10.5: Total LBBD iterations for different window sizes

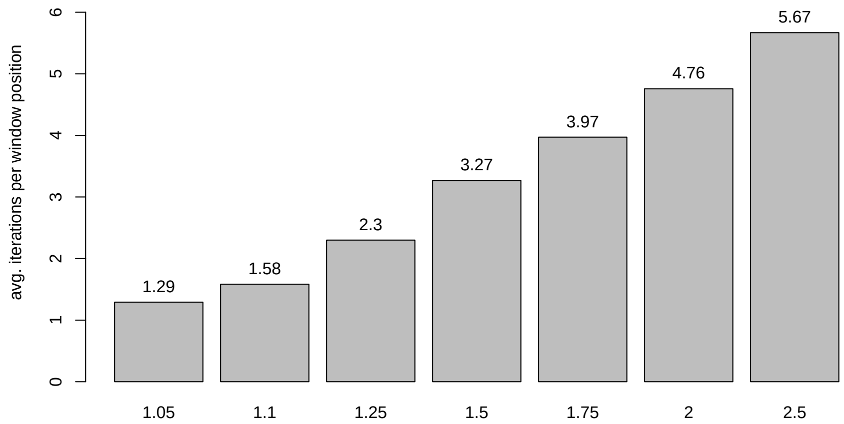


Figure 10.6: Average LBBD iterations per run for different window sizes

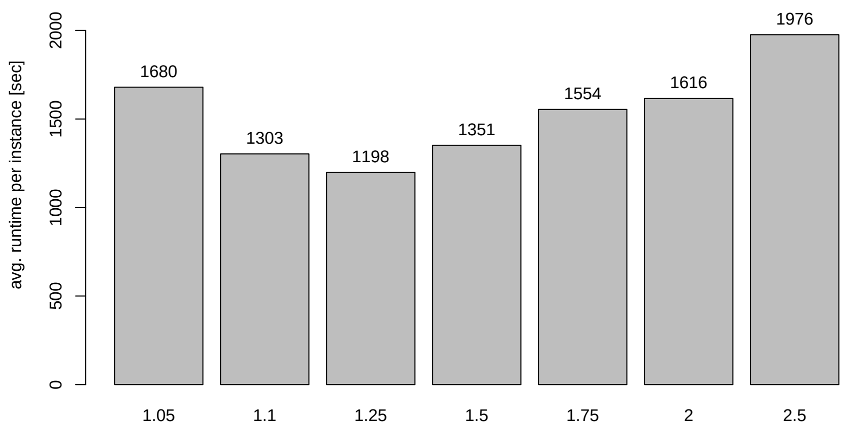


Figure 10.7: Runtimes for different window sizes

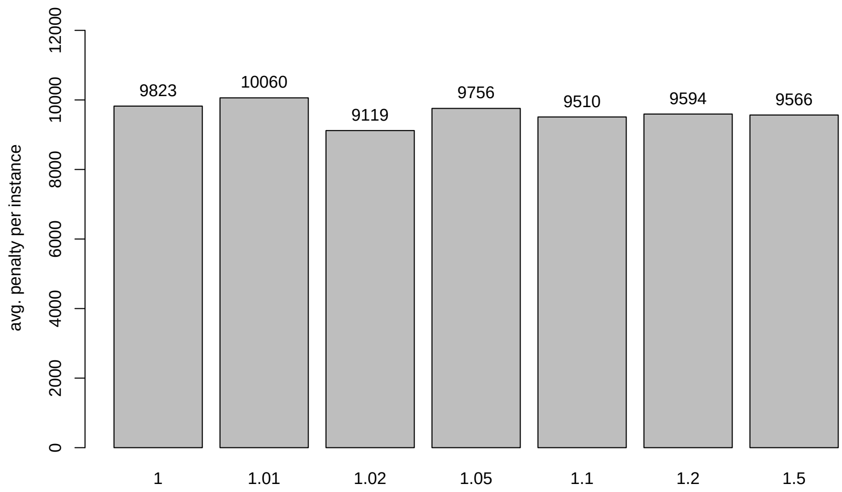


Figure 10.8: Penalties for different stowage loss factors

BinAssign model, we solve a vector container loading problem with respect to item volume and weight. If the assignment result cannot be packed by the CPPack subproblem, we assume that the assigned items incur some stowage loss. To overcome this in the next LBBD iteration, we increase the item volume considered in the master problem by a factor *stow*. The more often an item occurs in unpackable bins, the higher its stowage loss gets. Accordingly, larger factors should result in less required iterations before a feasible three-dimensional packing can be found. On our sample instances we evaluated the SeqACLPP approach for varying values of *stow* between 1.0 and 1.5, representing a volume increment between 0 and 50 percent for each failed packing attempt.

As the first result we note, similar to the window size, that there is no clear relationship between the stowage loss factor and solution quality. Figure 10.8 shows the average offload penalty of our sample instances for different factors. As for the window size, we assume that this effect stems from the large number of items in the instances. Even if an increased stowage loss cuts of promising solutions, the remaining solution space seems to contain many alternative solutions of similar quality.

The required number of total LBBD iterations, i.e., the sum of iterations of all LBBD runs, of Algorithm 3 decreases slightly for higher stowage loss factors. Figure 10.9 shows the average number of total iterations it takes to solve the sample instances. For small values between 1.01 and 1.05 the number of required iterations does not change at all. The same picture shows for the average number of iterations the LBBD takes for a run, i.e., before the packings can be solved successfully, which is shown in Figure 10.10. Again, higher values lead to less iterations, because the stowage loss rises faster.

Regarding the total solution runtime, we see a clear drop from more than 1,600 seconds to around 1,300 seconds when the stowage loss factor is used. For larger factors the runtime decreases slowly in the same way as the number of required iterations. Figure 10.11 shows the respective average runtimes. According to the found results, we choose a value of 1.02 for the stowage loss factor in the remainder of this work.

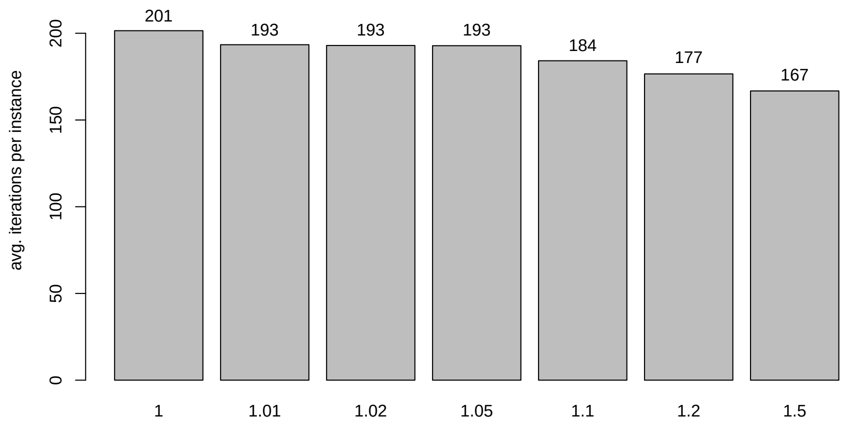


Figure 10.9: Total LBBD iterations for different stowage loss factors

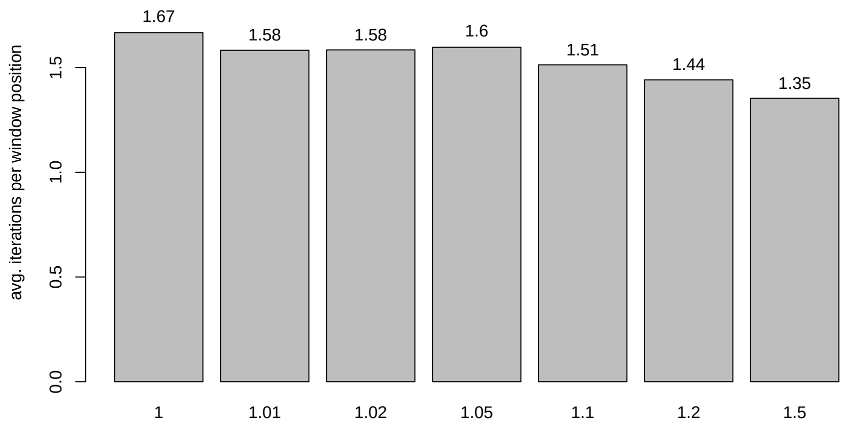


Figure 10.10: Average LBBD iterations per run for different stowage loss factors

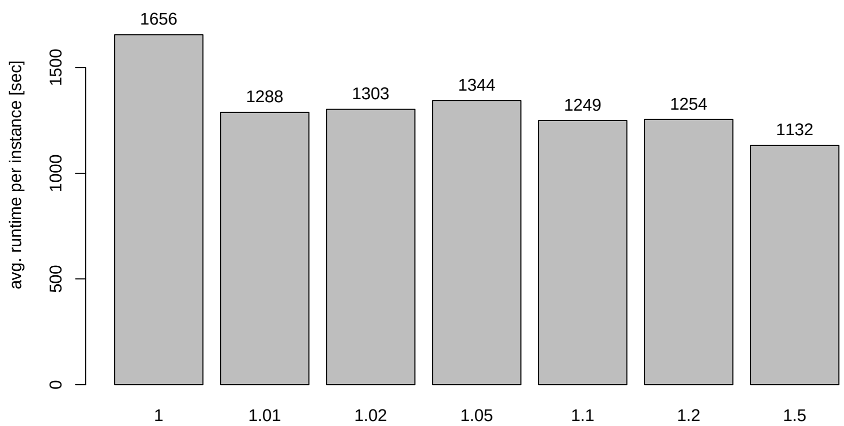


Figure 10.11: Runtimes for different stowage loss factors

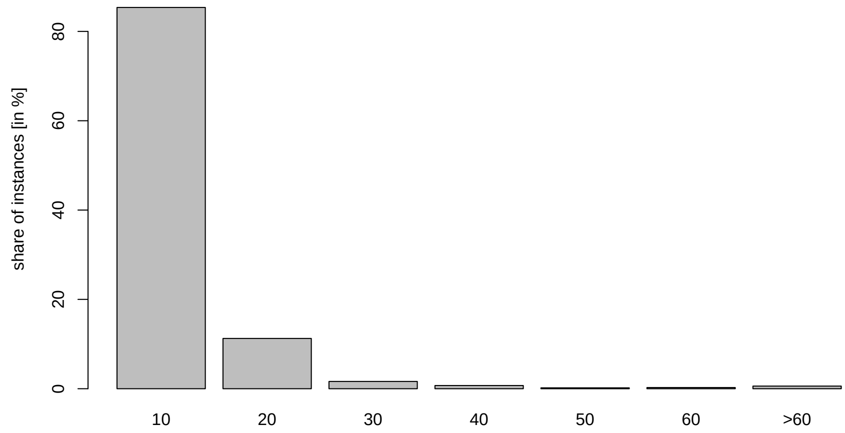


Figure 10.12: Distribution of runtimes of the BinAssign model (warm start)

Time limits

We introduce three time limits to restrict the runtime of the BDAP step. The first is the runtime of the initial cold start run of the BinAssign model for a transport segment. As the MIP is started without an initial solution, it typically takes longer to reach a small MIP gap. Therefore, we set a time limit of 300 seconds for this first run. The following warm start iterations use the previous solution and reach small MIP gaps much faster. The distribution of runtimes seen in the experiments above is shown in Figure 10.12. We set the time limit of successive runs of the BinAssign model to 60 seconds.

Furthermore, we need to limit the runtime of the CPPack search. In our experiments above, the CPPack model showed to be extremely fast for most bins. However, sometimes the search gets stuck and does not return any solution. As the model is run very often, we set a short time limit of 1 second for the branch and bound search. Figure 10.13 shows the distribution of unsolvable bins within a 1 second time limit across the instances. In almost all instances less than 10 percent of the bins were unsolvable within the time limit. From our point of view this is acceptable, especially because the unsolvable bins will add a cut to the BinAssign and might be tested again in the next iteration with a similar but smaller set of items. Thus, we expect to lose not much solution quality by a short time limit.

10.2.4 Weight and balance

For the proposed weight and balance step (UWAB), we need to set a MIP time limit. To get an insight into the required runtimes, we ran the weight and balance step on the results of the experiments performed in Section 10.2.3. To perform the experiments with tolerable effort, we set an initial time limit of 300 seconds for the weight and balance step.

Although 59 percent of the instances can be solved within the first 30 seconds to a MIP gap of less than 1 percent, see Figure 10.14, around 14 percent of the instances reach the time limit of 300 seconds. However, those long running instances mostly exhibit

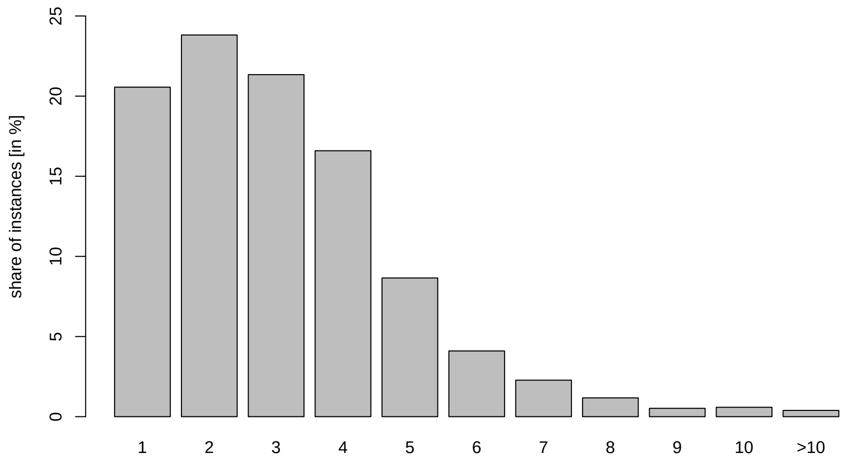


Figure 10.13: Distribution of the share of unsolvable bins per instance for CPPack with a time limit of 1 second (in %)). To be read as: in 24 percent of all instances between 1 and 2 percent of the bins could not be solved within 1 second.

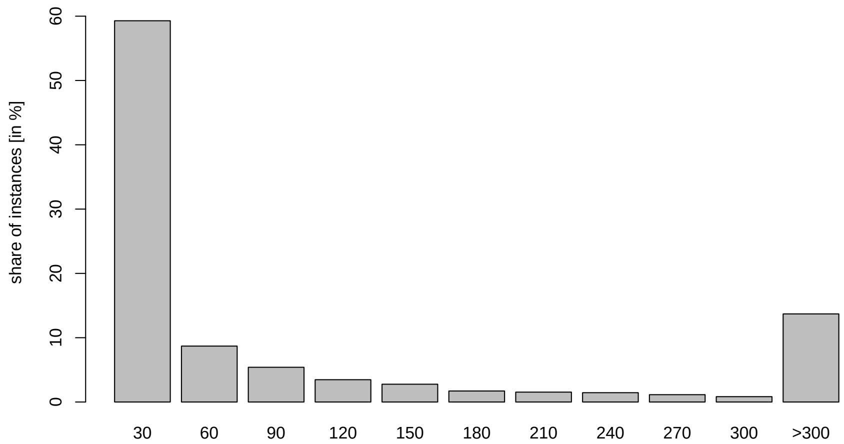


Figure 10.14: Distribution of runtimes of the weight and balance step (in sec)

small MIP gaps, see Figure 10.15. Around 70 percent of them have a gap smaller than 10 percent. Therefore, in our evaluation we keep the initial time limit of 300 seconds.

10.3 Results of the sequential approach

We evaluated our approach on the two datasets introduced in Section 8.3. First, we relate our results to actual load plans seen in practice to check the plausibility of our solutions. Afterwards, we discuss the results achieved for the different scenarios.

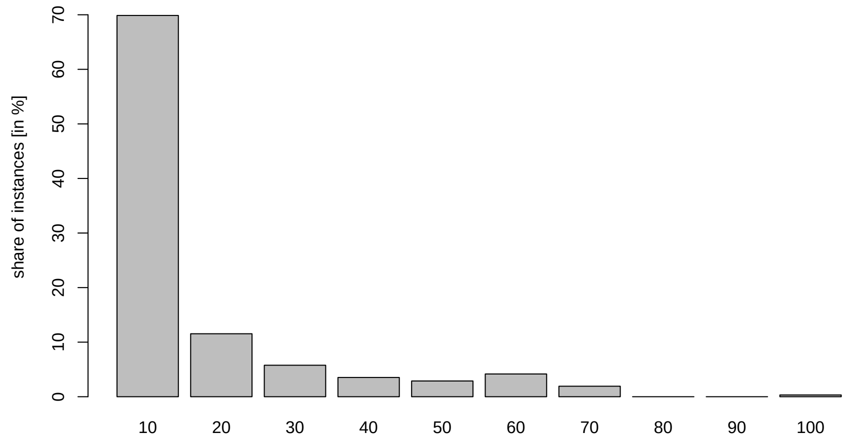


Figure 10.15: Distribution of the remaining MIP gap of flights reaching a time limit of 300 seconds for the weight and balance step (in %)

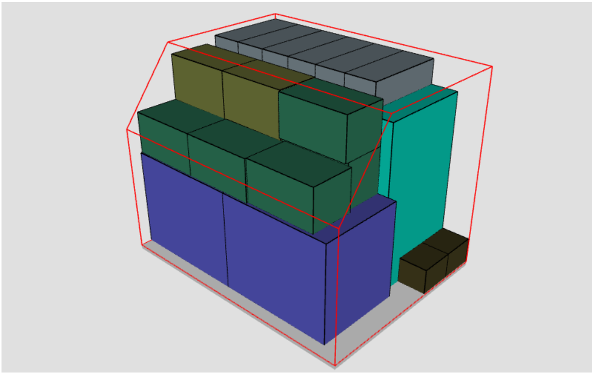
	Real-world solutions	**SeqACLPP approach**	**Δ**
WLF	0.495	0.493	-0.4%
GLF	0.327	0.322	-1.5%
NLF	0.546	0.662	+21.2%
UNITS	23.000	15.685	-31.8%
SPLIT	0.283	0.201	-29.0%
DISP	3.100	2.585	-16.6%
MIX	0.173	0.116	-32.9%
CREWS	n/a	211	n/a

Table 10.2: Performance of the SeqACLPP approach compared to results seen in practice

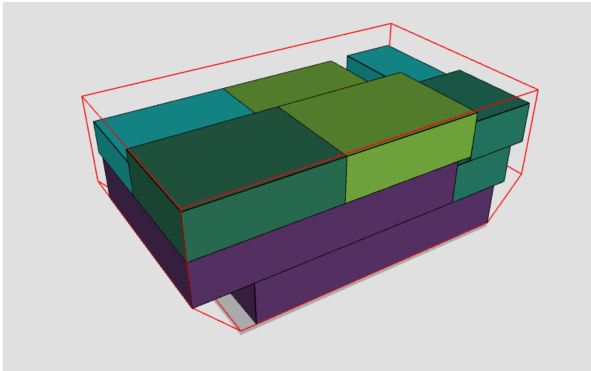
Comparison to practice

We compare our results of the real world dataset base scenario (A-base) with the actual solutions created by load planners and palletization workers in practice. Table 10.2 gives an overview of the respective performance indicators.

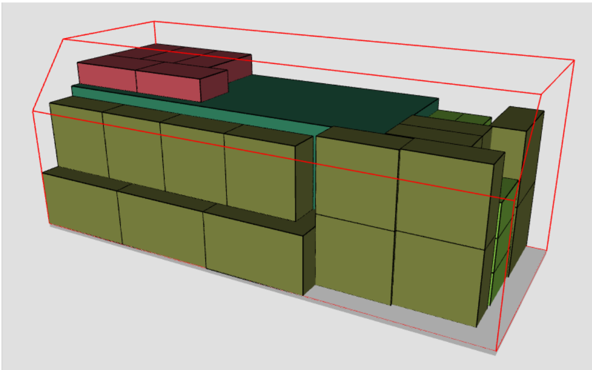
With regard to the aircraft utilization indicators, the weight load factor (WLF) and volume gross load factor (GLF) cannot be further improved, because the given flight instances contained only the transported items of the real flights. Some items, representing around 0.4 percent of the weight and 1.5 percent of the volume, that were loaded in practice were offloaded by our approach. For these cases we identify three reasons. First, the workers in practice are more flexible than the SeqACLPP approach. For example, they can postpone a ULD to wait for a late item or add dunnage material to provide an even surface for a large and light item. Second, there might be errors in the booking data that made the real packing easier. Maybe, some items in reality were smaller than in the booking data or one large booked item turned out to be a set of smaller items. Third,



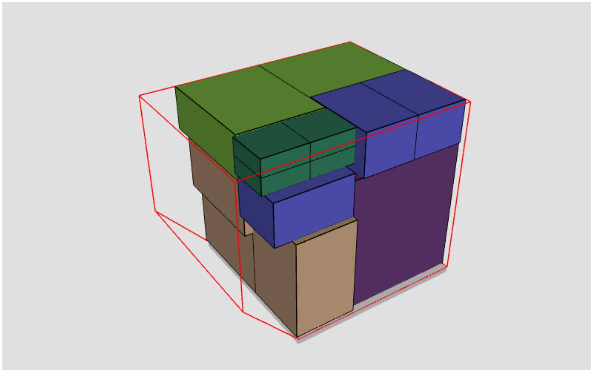
(a) Main deck PMC pallet (NLF: 77%)



(b) Lower deck PMC pallet (NLF: 72%)



(c) Main deck PGE pallet (NLF: 68%)



(d) Lower deck AKE container (NLF: 80%)

Figure 10.16: Examples of highly utilized ULDs

some items loaded in practice were offloaded by our approach in favor of saving another ULD position.

However, in our solutions the average volume net load factor (NLF) of the ULDs could be improved from 54.6 percent seen in practice to 66.2 percent. This represents an increase in the achieved packing density of more than 21 percent. Example packing patterns for the considered types of ULDs and the related volume utilization are shown in Figure 10.16. In the same manner the average number of built ULDs for each flight (UNITS) decreased by more than 31 percent. Part of this can be attributed to the higher packing density. The remainder results from choosing larger ULDs, especially the use of standard PMC pallets instead of half-sized AKE containers on the lower deck. The latter shows that the SeqACLPP approach is able to tailor its solutions to the assembly cost of the ULDs. For the benchmark instances these cost were only roughly estimated and thus the results differ from practice.

The indicators representing handling effort could all be improved considerably. The ratio of shipments that are split onto multiple ULDs (SPLIT) could be reduced from 28.3 percent to 20.1 percent. Furthermore, the average number of ULDs a splitted shipment was distributed across (DISP) could be reduced from 3.1 ULDs to 2.6 ULDs. To grasp the total effect of the SPLIT and DISP changes on the transport effort within the air cargo terminal, we calculate the total number of transports between the warehouse and the

	A-base	**A-high**	**A-fast**	**B-base**	**B-high**	**B-fast**
WLF	0.493	0.969	0.492	0.501	0.966	0.506
GLF	0.322	0.680	0.322	0.346	0.672	0.346
NLF	0.662	0.709	0.668	0.664	0.695	0.661
UNITS	15.685	31.854	15.607	16.585	32.549	16.793
SPLIT	0.201	0.259	0.210	0.206	0.256	0.198
DISP	2.585	2.966	2.554	2.552	2.849	2.577
MIX	0.116	0.119	0.119	0.117	0.141	0.124
CREWS	211	410	227	215	391	276

Table 10.3: Performance of the SeqACLPP approach across all scenarios

workstations, which can be defined as:

$$\text{SPLIT} \cdot \text{DISP} + (1 - \text{SPLIT}) \cdot 1 \tag{10.1}$$

It represents the average number of warehouse to ULD workstation transports for splitted shipments and a single transport for non-splitted shipments. Compared to the current practice, our results realize a reduction of transports of more than 17 percent.

The ratio of ULDs that contain both standard and premium shipments (MIX) could be reduced from 17.3 percent to 11.6 percent. Together with the reduction of handled UNITS, this results in around 2 ULDs less per flight that would need preferred express handling and thus incur higher cost.

With respect to the required crews, we cannot make any statement. In practice, the build-up crews are also used for other tasks, for example, in the break-down of incoming ULDs, and thus no specific number of build-up crews was available to us.

Besides the numbers, we also discussed individual load plans with the respective experts of our industrial partner who found the results to be reasonable. In general, we can say that the SeqACLPP approach works and provides solutions showing better performance values than the current practice.

Scenario evaluation

As the second step of our evaluation, we analyze the results of the SeqACLPP approach for the respective scenarios defined in Section 8.3.3. Table 10.3 shows the achieved performance values.

In the base and fast scenarios of both datasets the achieved utilization with respect to the aircraft weight (WLF) and volume (GLF and NLF) capacity is pretty stable. Almost all items are loaded and an NLF between 66.1 percent and 66.8 percent is achieved. In the high-load scenarios much more cargo has to be offloaded, because all the flights are overbooked. Nevertheless, our approach utilizes more than 96 percent of the aircraft weight capacity (WLF) and fills on average 69.5 percent and 70.9 percent of the physical volume of the ULDs (NLF), which is slightly better than in the base and fast scenarios. According to practitioners, utilizing 70 percent of the volume of a single ULD is already a very good result.

	A-base	A-high	A-fast	B-base	B-high	B-fast
PEN	2807.000	20407.000	2774.000	3006.000	40035.000	2200.000
UNITS	3680.899	7238.202	3633.708	4070.732	7531.707	3950.000
FUEL	20.438	417.213	38.169	56.988	465.890	44.500
OPS	83.258	166.506	68.652	91.939	247.293	99.854
TOTAL	6591.595	27310.921	6513.529	7225.659	48279.890	6294.354

Table 10.4: Cost of the SeqACLPP approach across all scenarios

The aircraft type used in our experiments had a standard configuration of 3 PGE, 20 main deck PMC, and 10 lower deck PMC ULDs, totaling 33 units. Accordingly, almost all loading positions are used in the high-load flights. Therefore, we conclude that the SeqACLPP approach is able to find good solutions, which are also well-balanced with respect to the weight and volume capacities of the aircraft.

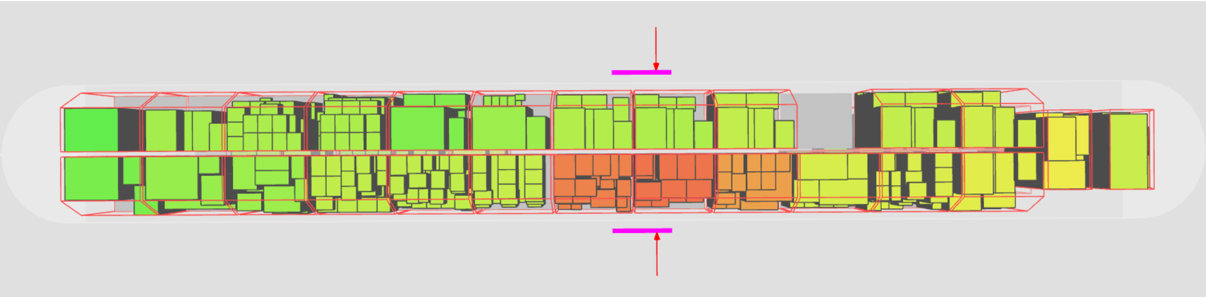
The indicators representing handling effort, namely SPLIT, DISP, and MIX, are similar across all scenarios, indicating that these objectives can also be robustly improved compared to the current manual practice.

The number of crews employed in the scenarios correlates with the built ULDs. Crew productivity for all scenarios ranges between 5.0 and 6.9 ULDs per crew and shift. The fast scenarios show the lowest productivity with 6.1 and 5.0 ULDs per crew and shift. The reason here clearly is the unleveled workload stemming from the short transfer times of the cargo. During rush hours a high number of crews is needed, but before and after the crews are idling.

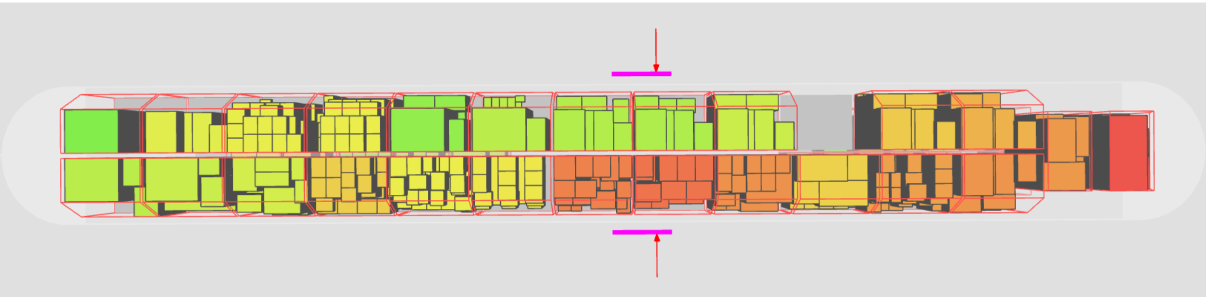
Next, we look at the total costs achieved by the SeqACLPP approach. Table 10.4 shows the respective average values for each scenario. A significant part of the costs stems from the penalties of offloaded items (PEN). While this is reasonable for the overbooked high-load scenarios, this result is not satisfactory for the base and fast scenarios. Analyzing the results in detail, we identified two primary reasons for offloaded items.

First, during aircraft configuration the capacity to accommodate large items is overestimated. We only consider the item volume and weight here, neglecting the real item dimensions. Thereby, the model is too optimistic when selecting ULDs for large items. Later, during palletization these items cannot be combined on ULDs and some of them must be offloaded. In the fast scenarios, we see less offloads, because the ULDs are generally built later and therefore more combinations of items are possible.

The second reason causing significant offloads in our solutions are too heavy ULDs created by the palletization step for which no loading position in the aircraft can be found during the weight and balance step. While the individual weight limit of a ULD type is quite high, for example, a main deck PMC pallet can weight up to 6,800 kg, the possible loading positions for heavy ULDs are very limited. Only four of the 26 positions on the main deck of our sample MD11F aircraft can handle this weight. Furthermore, the cumulative load limits of adjacent positions reduce the solution space significantly. For example, only two of the four positions mentioned above can be used simultaneously for such heavy ULDs without violating the maximum running load along the aircraft fuselage. On some positions the combined weight limit of two adjacent ULDs is less than 125 percent



(a) Weight relative to maximum weight of ULD type (6,800 kg)



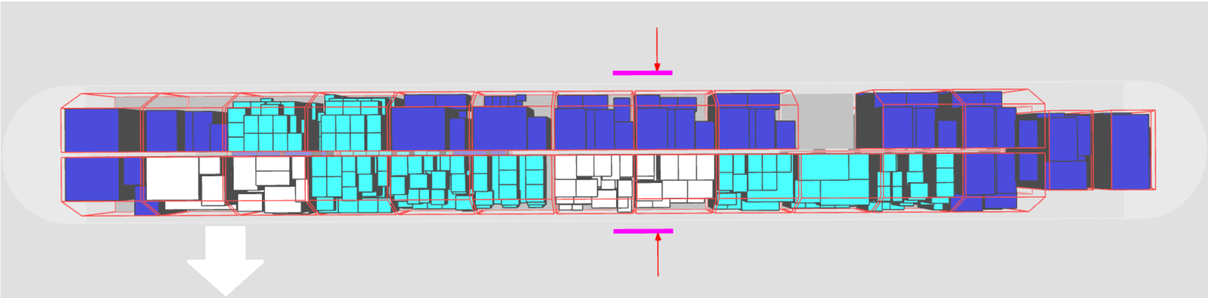
(b) Weight relative to maximum allowed weight at the loading position

Figure 10.17: Example weight distribution on a freighter main deck. The pink bars at both sides indicate the admissible range and the actual the center of gravity. Lighter ULDs are colored green, heavier ones red. For example, the ULD to the right only weights 2,485 kg but is close to the weight limit at its position of 2,700 kg.

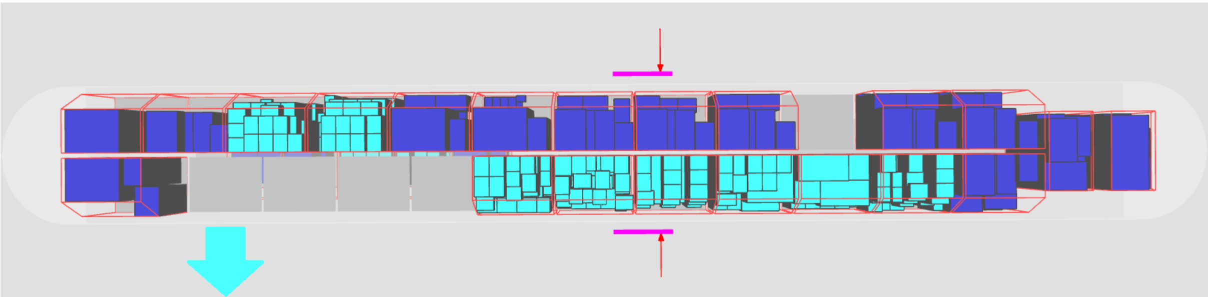
of the individual weight limit of each position. Therefore, it is hard to come up with general weight limits for individual ULDs. This observation coincides with the current planning practice. Experienced load planners have a loading position in mind for each planned ULD and partly take cumulative weight limits into account to reduce the risk of ULD offloads later. However, this is a mostly manual approach, heavily depending on the individual experience of the load planner, and other aspects adding additional complexity like weight distribution or the relocation of ULDs between the flight legs are not systematically considered.

Besides the high offload penalties, the achieved load plans of the SeqACLPP approach are already very good. The additional fuel cost due to aircraft imbalances (FUEL) for the base and fast scenarios is almost negligible for practical setups. In the high-load scenarios some extra fuel burn seems to be inevitable. As the optimal center of gravity (CG) of our sample MD11F aircraft is closer to the aft of the aircraft, there are 14 loading positions in front of and only 10 behind the point of the optimal CG. As we operate close to the aircraft weight limit in the high-load scenarios, most loading positions carry significant weight. Consequently, this leads to a forward CG increasing the fuel burn. An example of the weight distribution and aircraft CG on a highly booked flight is given in Figure 10.17.

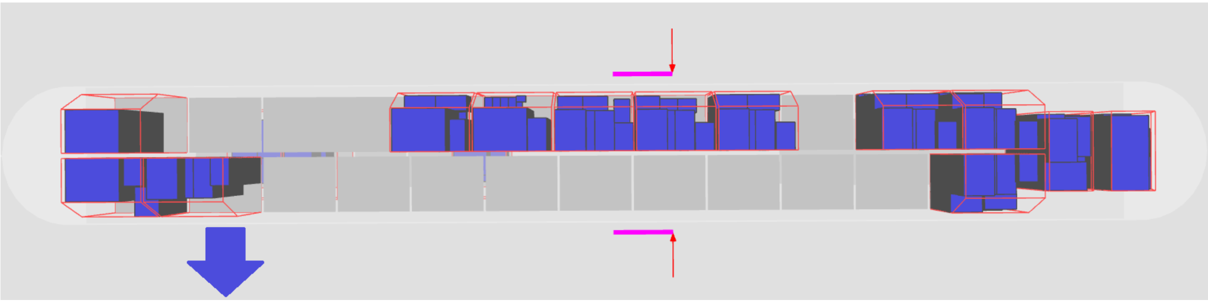
Similarly, the amount of conducted reloading operations (OPS) in the presented results is significantly lower than in current practice. We give an overview of the distribution of reloading operations in Table 10.5 and describe the actual reloading operations of a sample



(a) After leg 1/3: Four ULDs (white) need to be unloaded. Therefore, three additional ULDs must be temporarily removed.



(b) After leg 2/3: Eight ULDs (light blue) need to be unloaded. One ULD must be temporarily removed and is reloaded at another loading position to reduce lateral imbalance.



(c) After leg 3/3: All remaining ULDs are unloaded

Figure 10.18: Loading patterns on a freighter main deck over a three-leg flight. Colors indicate the port of unloading of the ULD: after the first leg (white), the second (light blue), or third (dark blue). The large arrows show the position of the cargo door. The pink bars at both sides indicate the admissible range and the actual the center of gravity.

Handling operations	**Legs** **1**	**2**	**3**	**4**
0	100%	67%	31%	0%
1	0%	19%	15%	0%
2	0%	8%	33%	0%
3	0%	4%	18%	0%
4	0%	1%	3%	33%
≥5	0%	1%	0%	67%

Table 10.5: Distribution of reloading operations per flight

Step name	**A-base**	**A-high**	**A-fast**	**B-base**	**B-high**	**B-fast**	∅
VWAC [sec]	0	0	0	0	0	0	0
BDAP [sec]	746	2,080	776	748	2,186	769	1,218
UWAB [sec]	40	65	35	58	113	53	61
Total [sec]	786	2,145	811	806	2,299	822	1,278
LBBD iterations	158	241	158	162	276	165	193

Table 10.6: Computational effort per flight of the SeqACLPP approach

flight in Figure 10.18. On most flights with two or three legs, up to two reloaded ULDs were sufficient. Instances with four legs, always exhibited reloading operations. During these flights, it is hardly possible to place ULDs destined for the last stop without blocking ULDs that need to be unloaded at earlier stops. Even if it would be possible, the caused shifts of aircraft CG at each stop might already justify the cost of the reloading operations to save fuel. The highest number of reloading operations we found on any resulting flight was 13. Therefore, we argue that the model finds a good tradeoff between the extra fuel burned due to aircraft imbalances and the effort to establish a lower imbalance between two legs. This poses a significant improvement to the current practice, where sometimes the complete aircraft is unloaded and reloaded again at a stop-over airport.

At last we give an overview of the computational effort required to solve the instances in the different scenarios, see Table 10.6. Solving a flight of the base or fast scenario takes on average 14 minutes (806 seconds). Assuming that load planning starts around 24 hours before a flight, this amount of time is acceptable in practice to create good initial load plans. However, the shown runtimes are too long to be used in an interactive process with the load planner. Therefore, our solution is not yet suitable for handling frequent updates due to changed booking data or manual interventions by the load planner. Runtimes for the high-load scenarios are even longer and take on average 37 minutes (2,222 seconds) per flight. The main cause is that the generated cuts cannot effectively reduce the search space of the overbooked flights as there are a lot of alternative items and thus many LBBD iterations are necessary until a feasible packing is found. An easy way to start speeding up the solution approach is to introduce parallelization. All our experiments were performed with a single thread. Nevertheless, our BDAP approach can easily be parallelized at multiple points. For example, the branch-and-bound part of the BinAssign

MIP can be multi-threaded with current MIP solvers. Also, the CPPack search can be run independently for each ULD.

10.4 Summary

In this chapter, we evaluated the SeqACLPP approach with respect to solution quality and runtime by solving the benchmark instances from Chapter 8.

To find suitable parameter settings for the planning steps, we described a set of preliminary experiments. We ran those experiments on a subset of the benchmark instances, showed how different settings impact solution performance, and chose suitable values for our general evaluation. Nevertheless, we expect that in other setups, i.e., with respect to the structure of the shipments, aircraft, or flight plans, different parameters might produce better results and the presented experiments should be rerun to identify them.

In our main evaluation we compared the benchmark results to actual load plans seen in practice. We found that the SeqACLPP approach performs well and is able to improve most performance indicators. The average packing density of the ULDs (NLF) can be improved by more than 21 percent compared to current practice. In parallel, the item sorting (SPLIT and MIX) can be improved by around 30 percent, which results in less and cheaper handling operations. In our results, the absolute cost for extra fuel due to aircraft imbalance and reloading operations is very low for most flights.

On many flights of the high-load scenarios the aircraft capacity can be fully utilized. However, in the base and fast scenarios sometimes items are offloaded, although there is enough aircraft capacity. We identified two causes. In the first case not enough ULDs were selected during the aircraft configuration. At this step the items are not individually considered and therefore the capacity is overestimated. In the second case a set of heavy ULDs, which were created at the palletization step, cannot be placed together in the aircraft during the weight and balance step. Although the individual weights of the ULDs are fine, no combination can be found that meets all the cumulative structural load limits of the aircraft. We address these issues in the next chapter.

The average solution runtimes range between 13 and 38 minutes per flight. In practice this is acceptable to provide non-interactive initial solutions for the load planners. However, it is too slow for interactive usage and a further speed up, for example by parallelization would be necessary.

11 Coordinated planning approaches

In the evaluation of the SeqACLPP approach, see Section 10.3, we discovered an unsatisfactory high amount of offloaded cargo although loading positions in the aircraft remained free. During the SeqACLPP approach, we executed each of the four planning steps aircraft configuration (VWAC), build-up scheduling (RHBS), palletization (BDAP), and weight and balance (UWAB) individually. In each step we solely considered the constraints and objective of the respective step. Thereby, we aimed for a local optimum of each step at the risk of producing results that do not fit well with the successive planning steps. We identified two typical cases.

First, there were not enough ULDs selected during the VWAC step to accommodate large items. Second, the ULDs built during the BDAP step had a weight distribution that often did not fit into the aircraft in the UWAB step. In effect some of the ULDs and items could not be transported.

In this chapter, we introduce and evaluate two methods to avoid such local optimization by introducing coordination between the planning steps. The basic idea is to already respect constraints of later planning steps during the solution of an earlier step, here the aircraft configuration.

At first, we examine the integration of palletization aspects into the aircraft configuration in Section 11.1. After that, we incorporate weight and balance aspects into the aircraft configuration in Section 11.2.

11.1 Shipment-based aircraft configuration

One drawback of the SeqACLPP approach is the significant number of offloaded items, although there is enough aircraft capacity and loading positions remain unused. As explained in Section 10.3, we attribute this to our aircraft configuration step being too optimistic. There might be several reasons why items cannot be packed during the three-dimensional packing step. First, the aircraft configuration only considers the total volume and weight. Therefore, it overestimates how items, especially large ones, can be geometrically combined on the ULDs. Another reason for offloads is that item incompatibilities might require the usage of multiple separate ULDs although the total volume and weight of the shipments on a segment would fit into less, sometimes just one, ULDs. In these cases, the aircraft configuration selects the smaller number which leads to offloading some of the incompatible items at the palletization step.

To overcome these problems, we improve the aircraft configuration by a more detailed planning approach, which we denote as Shipment-based Aircraft Configuration (SAC). The basic idea is to incorporate aspects of the palletization step already within the aircraft configuration step. We do this by combining the VWAC model presented in Section 9.1.2

	A-base	A-high	A-fast	B-base	B-high	B-fast
WLF	0.495	0.971	0.499	0.505	0.961	0.507
GLF	0.322	0.683	0.322	0.346	0.688	0.346
NLF	0.665	0.708	0.666	0.665	0.703	0.662
UNITS	16.090	32.157	16.056	16.915	34.439	16.890
SPLIT	0.208	0.263	0.212	0.204	0.276	0.195
DISP	2.569	2.940	2.578	2.578	2.941	2.584
MIX	0.086	0.112	0.106	0.095	0.155	0.099
CREWS	216	415	227	225	409	283

Table 11.1: Performance of SeqACLPP-SAC across all scenarios

	A-base	A-high	A-fast	B-base	B-high	B-fast	∅
WLF	0.406	0.206	1.423	0.798	-0.518	0.198	0.419
GLF	0.000	0.441	0.000	0.000	2.381	0.000	0.470
NLF	0.453	-0.141	-0.299	0.151	1.151	0.151	0.244
UNITS	2.582	0.951	2.877	1.990	5.807	0.578	2.464
SPLIT	3.483	1.544	0.952	-0.971	7.812	-1.515	1.884
DISP	-0.619	-0.877	0.940	1.019	3.229	0.272	0.661
MIX	-25.862	-5.882	-10.924	-18.803	9.929	-20.161	-11.950
CREWS	2.370	1.220	0.000	4.651	4.604	2.536	2.564

Table 11.2: Performance improvement of SeqACLPP-SAC compared to SeqACLPP

and the BinAssign model presented in Section 9.3.1. The objective stays the same as in the original aircraft configuration step, but the added constraints of the bin assignment must be fulfilled. By doing this we still do not create three-dimensional load plans during the SAC. The items are only assigned to a ULD considering the ULD capacity, required ULD types, and item incompatibilities. Thereby, we expect to get a more realistic estimate of the number of required ULDs.

Besides the aircraft configuration step, all other steps remain the same as in the SeqA-CLPP approach, presented in Chapter 9. In particular, during palletization items must not be packed into the same ULD to which they were assigned by the SAC model. In the following we denote the SeqACLPP approach with the new SAC aircraft configuration step as SeqACLPP-SAC.

Evaluation

To evaluate the SeqACLPP-SAC approach, we repeated the experiments introduced in Section 10.3. We give an overview of the absolute performance indicators achieved in Table 11.1 and their relative deviation compared to the SeqACLPP approach in Table 11.2.

There are only minor variations in the utilization indicators. Most notably, the number of built ULDs (UNITS) increased by 2.5 percent across all scenarios. However, this does not mean that the ULD capacity increased. More often, PMC ULDs were replaced by multiple

	A-base	A-high	A-fast	B-base	B-high	B-fast
PEN	2229.000	20290.000	1756.000	2428.000	41844.000	2019.000
UNITS	3520.225	7326.966	3489.888	3865.854	8203.659	4132.927
FUEL	23.562	463.157	31.011	48.549	580.171	49.695
OPS	71.562	158.629	64.247	109.390	259.378	101.451
TOTAL	5844.349	27320.752	5341.146	6451.793	50887.208	6303.073

Table 11.3: Cost of SeqACLPP-SAC across all scenarios

	A-base	A-high	A-fast	B-base	B-high	B-fast	∅
PEN	-20.591	-0.573	-36.698	-19.228	4.519	-8.227	-13.466
UNITS	-4.365	1.226	-3.958	-5.033	8.922	4.631	0.237
FUEL	15.285	11.012	-18.753	-14.808	24.530	11.674	4.823
OPS	-14.048	-4.731	-6.416	18.981	4.887	1.599	0.045
TOTAL	-11.336	0.036	-17.999	-10.710	5.400	0.139	-5.745

Table 11.4: Cost improvement of SeqACLPP-SAC compared to the SeqACLPP

smaller AKE ULDs to better meet the demand for item sorting and incompatibilities. By using these additional ULDs, around 0.4 percent more of the shipment weight (WLF) and 0.5 percent more of the shipment volume (GLF) can be loaded. The ratio of ULDs containing both standard and express items (MIX) can be reduced significantly in most scenarios. We attribute this to the item sorting constraints, which are now respected during the aircraft configuration. Ideally, they affect any solution to contain enough selected ULDs for both sets of standard and express shipments.

With respect to the cost, the SeqACLPP-SAC approach performs better than the SeqACLPP approach and can reduce the total cost by 5.7 percent. We give an overview of the absolute cost values achieved by the SeqACLPP-SAC approach in Table 11.3 and their relative deviation compared to the SeqACLPP approach in Table 11.4.

The increase of the load factors seen above leads to 13.5 percent less offload penalties (PEN). Interestingly, for the scenarios with the highest improvements (A-base, A-fast, B-base) the cost of the chosen ULDs (UNITS) could also be reduced. Here, the SAC model produces a better fit of ULDs for the items than the VWAC model. The changes of costs for extra fuel (FUEL) and handling operations (OPS) fluctuate from scenario to scenario. As the absolute values of these costs are rather small, the impact of the relative changes appears larger than it is. Most of the changes were caused by a single flight in the respective scenario. Altogether, the SeqACLPP-SAC approach could significantly improve the cost of 3 of the 6 scenarios. However, the overbooked scenarios (A-high, B-high) could not be improved as the utilization here is already very high.

At last, we analyze the computational effort of the SeqACLPP-SAC approach. We distinguish two separate parts: the runtime of the SAC model itself and the change of runtime of all the planning steps. We give an overview of the runtimes for each step and scenario, as well as the change compared to the SeqACLPP approach in Table 11.5.

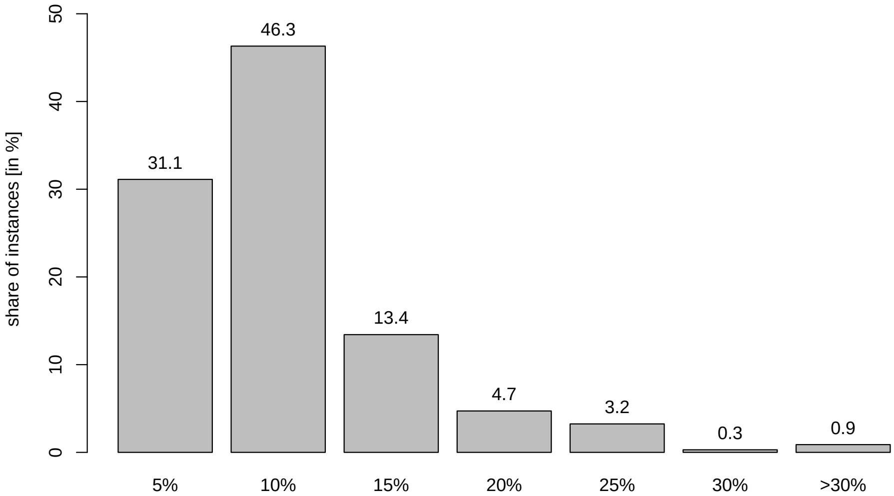


Figure 11.1: SAC MIP gap distribution after 30 seconds runtime (rounded up)

	A-base	**A-high**	**A-fast**	**B-base**	**B-high**	**B-fast**	∅	Δ
SAC [sec]	26	30	26	28	30	28	28	$+\infty$
BDAP [sec]	825	1,991	805	740	2,029	745	1,189	-2.4%
UWAB [sec]	47	72	56	66	117	65	71	+16.4%
Total [sec]	898	2,093	887	834	2,176	838	1,288	+0.8%

Table 11.5: Computational effort of SeqACLPP-SAC across all scenarios

The runtime of the VWAC model, which we presented in Section 9.1, was negligible. For almost all flights, the model solved in well under a second. The SAC model on the other hand takes much longer to be solved to optimality. Therefore, we stop the MIP search after 30 seconds. Around 10 percent of the instances can be solved down to a 1 percent MIP gap until the time limit is reached. The distribution of the resulting MIP gaps of the remaining flights is shown in Figure 11.1. Of those flights about 90 percent can be solved with a MIP gap of less than 15 percent.

The computational effort required to solve the palletization (BDAP) and weight and balance (UWAB) steps does not change substantially. The number of LBBD iterations during palletization slightly changes for each scenario but on average it stays the same. The average runtime across all scenarios increases by 0.8 percent, a deviation well within measuring tolerance.

Accordingly, we can say that the incorporation of palletization aspects into the aircraft configuration step works and can be done with acceptable computational effort, albeit its effects on solution quality are limited.

11.2 Weight-and-balance-based aircraft configuration

The second reason for offloaded items in the SeqACLPP approach, which we identified in Section 10.3, is the offloading of ULDs in the weight and balance step. In the palletization step highly utilized and heavy ULDs are created, but afterwards they cannot be loaded into the aircraft during the weight and balance step. The reason are the cumulative weight limits of adjacent loading positions, which are not taken into consideration in the first three planning steps as no loading positions for the ULDs are assigned yet. During palletization we calculate solutions for each transport segment individually. Therefore, the number and weight distribution of the ULDs of other transport segments on the same flight is not known during the palletization step for a segment. Consequently, some heavy ULDs might be built for each segment.

To be able to build ULDs of different segments with a more suitable weight distribution, we present another extension of the aircraft configuration, which we denote as Weight-and-balance-based Aircraft Configuration (WAC). The basic idea is to allot a loading position and choose a target weight for each ULD already during the aircraft configuration step. Then, in the palletization step exceeding the selected target weight will be penalized.

Technically, we create the aircraft configuration by solving a simplification of the weight and balance model (WBM), which we presented in Section 9.4.2. To keep the model simple, we only consider the first leg of the flight. This is reasonable, because we expect that for all ULDs, which are loadable on the first leg, it is possible to find a feasible loading position on subsequent legs. Furthermore, we omit all aspects modeling reloading operations. The model decides for each loading position of the aircraft, for which transport segment it is used and how much weight should be placed on it. Thereby, we expect to receive a realistic set of selected ULDs with target weights that can be loaded in the weight and balance step.

Table 11.6 introduces the parameters specific to the WAC model. All other parameters have been introduced in Tables 9.2 and 9.11. The model decides about the weight $w_{p,k}$

Set/Parameter	Description
$\mathrm{cap}_p^{\mathrm{wgt}} \in \mathbb{R}_{\geq 0}$	weight capacity at loading position p
$\mathrm{cap}_p^{\mathrm{vol}} \in \mathbb{R}_{\geq 0}$	volume capacity at loading position p
$\mathrm{cost}_p \in \mathbb{R}$	handling cost for a ULD on loading position p

Table 11.6: New parameters of the WAC model

placed on loading position p by transport segment k. Besides, auxiliary variables exist capturing the transport segment that is placed on a loading position ($x_{p,k}$), if a loading position is used at all (x_p), and what amount of weight is placed on a loading position irrespective of the transport segment (w_p).

We formally define the WAC model as follows:

$$\text{minimize} \sum_{k \in K} \left(\frac{1}{2} y_k^{\mathrm{wgt}} \mathrm{offload}_k^{\mathrm{wgt}} + \frac{1}{2} y_k^{\mathrm{vol}} \mathrm{offload}_k^{\mathrm{vol}} + \sum_{p \in P} x_p \mathrm{cost}_p \right) + fuel_l^+ \tag{11.1}$$

subject to

$$\mathrm{dem}_k^{\mathrm{wgt}} = \sum_{p \in P} w_{p,k} + y_k^{\mathrm{wgt}} \qquad k \in K \tag{11.2}$$

$$\mathrm{dem}_k^{\mathrm{vol}} \leq \sum_{p \in P} x_{p,k} \mathrm{cap}_p^{\mathrm{vol}} + y_k^{\mathrm{vol}} \qquad k \in K \tag{11.3}$$

$$x_p = \sum_{k \in K} x_{p,k} \qquad p \in P \tag{11.4}$$

$$1 \geq x_p + x_{p'} \qquad \langle p, p' \rangle \in \mathrm{O} \tag{11.5}$$

$$x_p \leq \sum_{p' \in R_p} x_{p'} \qquad p \in P \tag{11.6}$$

$$w_{p,k} \leq x_{p,k} \mathrm{cap}_p^{\mathrm{wgt}} \qquad p \in P \tag{11.7}$$

$$w_p = \sum_{k \in K} w_{p,k} \qquad p \in P \tag{11.8}$$

$$\mathrm{MAX}_l \geq \sum_{p \in P} w_p \tag{11.9}$$

$$\mathrm{limit}_n \geq \sum_{p \in P} \mathrm{f}_{n,p} w_p \qquad n \in N \tag{11.10}$$

$$m_l = \mathrm{BA}^{\mathrm{ac}} \mathrm{w}^{\mathrm{OEW}} + \mathrm{BA}_l^{\mathrm{fuel}} \mathrm{FW}_l + \sum_{p \in P} \mathrm{BA}_p^{\mathrm{lng}} w_p \tag{11.11}$$

$$w_l = \mathrm{w}^{\mathrm{OEW}} + \mathrm{FW}_l + \sum_{p \in P} w_p \tag{11.12}$$

$$\text{Equations (9.101) to (9.107) to calculate } fuel_l^+ \tag{11.13}$$

$$x_p; x_{p,k} \in \{0, 1\} \tag{11.14}$$

$$y_k^{\mathrm{wgt}}; y_k^{\mathrm{vol}}; w_p; w_{p,k} \in \mathbb{R}_{\geq 0} \tag{11.15}$$

The objective, given in Equation (11.1), minimizes the total cost from the penalties of offloaded cargo weight and volume, the used ULDs, and the extra fuel cost caused by a suboptimal center of gravity (CG) of the aircraft. Equations (11.2) and (11.3) make sure that the segment demand weight and volume is distributed onto the loading positions or is marked as offloaded. Equation (11.4) marks a loading position if it is used for any transport segment. Mutual exclusions of overlapping loading positions are enforced by Equation (11.5). The usage order of interdependent loading positions is respected by Equation (11.6). Equation (11.7) enforces that weight is only placed on used loading positions and that the individual weight limit of each loading position is respected. Equation (11.8) collects the weight on each loading position irrespective of the allotted transport segment. Equation (11.9) limits the total weight loaded into the aircraft. Equation (11.10) ensures that the cumulative weight limits of the loading positions hold. Equations (11.11) and (11.12) calculate the longitudinal balance arm and the total weight of the loaded aircraft. Both values are input variables for Equations (9.101) to (9.107) to calculate the extra fuel required due to a suboptimal CG of the loaded aircraft.

After the aircraft configuration step, we insert the chosen ULD weights as target weights for the ULDs to build during the palletization step. Therefore, we add a target weight parameter target_u for each selected ULD u. Given this parameter, we add the following constraint to the BinAssign model from Section 9.3.1 to calculate the excess weight dev_u for each ULD u:

$$\text{target}_u \geq \sum_{i \in I} \text{weight}_i^{\text{grs}} assign_{u,i} - dev_u \qquad u \in U \tag{11.16}$$

We do not consider negative deviations, i.e., when the ULD is too light. First, because it would generally not limit the loadability of the ULD. Second, these cases already lead to penalties at other parts of the model. If a ULD weight is lower than the target weight but the same amount of total weight is transported on all ULDs, then another ULD will be heavier than its target weight and incur penalty cost. Or, if a ULD weight is lower than the target weight because some items are offloaded, this also leads to offload penalty cost.

At last, we add the deviation from the target weight as cost to the objective of the bin assignment model:

$$\text{Equation (9.28)} + \alpha \sum_{u \in U} dev_u^2 \tag{11.17}$$

In our actual implementation we approximated the penalty by a piecewise linear function in steps of 100 kg. Furthermore, we add a factor α to weight the total penalty of target weight deviations because it can not be directly translated into cost. Weighting the deviations too high would lead to offloads during the palletization step although more weight could be loaded into the aircraft. On the other hand, if the deviations are weighted too low, it becomes easy for the model to disregard the target weights. In our experiments we set α to 0.01, which produced reasonable results.

Let BDAP+ denote the updated palletization step, i.e., where the BinAssign model is extended by Equations (11.16) and (11.17). The build-up scheduling (RHBS) and weight and balance (UWAB) steps remain the same as in the SeqACLPP approach, presented in Chapter 9. In the following we denote the SeqACLPP approach with the new WAC aircraft configuration step as SeqACLPP-WAC.

	A-base	A-high	A-fast	B-base	B-high	B-fast
WLF	0.509	0.919	0.513	0.514	0.934	0.517
GLF	0.322	0.639	0.322	0.346	0.650	0.346
NLF	0.655	0.676	0.658	0.661	0.689	0.656
UNITS	16.090	34.787	16.067	17.110	34.354	17.354
SPLIT	0.274	0.287	0.266	0.277	0.297	0.285
DISP	2.647	3.035	2.658	2.640	3.026	2.633
MIX	0.123	0.095	0.130	0.139	0.176	0.140
CREWS	215	428	231	216	403	272

Table 11.7: Performance of SeqACLPP-WAC across all scenarios

	A-base	A-high	A-fast	B-base	B-high	B-fast	∅
WLF	3.245	-5.160	4.268	2.595	-3.313	2.174	0.635
GLF	0.000	-6.029	0.000	0.000	-3.274	0.000	-1.550
NLF	-1.057	-4.654	-1.497	-0.452	-0.863	-0.756	-1.546
UNITS	2.582	9.208	2.947	3.166	5.545	3.341	4.465
SPLIT	36.318	10.811	26.667	34.466	16.016	43.939	28.036
DISP	2.398	2.326	4.072	3.448	6.213	2.173	3.438
MIX	6.034	-20.168	9.244	18.803	24.823	12.903	8.607
CREWS	1.896	4.390	1.762	0.465	3.069	-1.449	1.689

Table 11.8: Performance improvement of SeqACLPP-WAC compared to SeqACLPP

Evaluation

To evaluate the SeqACLPP-WAC approach, we repeated the experiments introduced in Section 10.3. We follow the same evaluation scheme as used in Section 11.1 for the Shipment-based Aircraft Configuration. We give an overview of the absolute performance indicators achieved in Table 11.7 and their relative deviation compared to the SeqACLPP approach in Table 11.8.

With respect to the solution quality, we get a heavily diverging picture for the base and fast scenarios on one side and the high scenarios on the other. The weight load factor (WLF) can be improved in all base and fast scenarios, on average by more than 3 percent. However, the WLF decreases in the high scenarios by 4.2 percent. For these overbooked flights, the WAC seems to be too conservative, when assigning weight to the ULDs. The aim of the WAC is the creation of loadable ULD target weights that result in a well balanced aircraft, but in these cases balancing should not the prime concern. Also the volume net load factor (NLF) reduces across all scenarios by around 1.5 percent while 4.5 percent more ULDs are built (UNITS). However, this is reasonable, as the chosen target weights are often significantly below the weight limit of a ULD type and thus the load needs to be split across more ULDs. This also has a negative impact on the handling indicators. The share of shipments that are split (SPLIT) increases significantly in all scenarios, on average by 28 percent. The increase of dispersion of splitted shipments

	A-base	A-high	A-fast	B-base	B-high	B-fast
PEN	958.000	27786.000	598.000	586.000	47863.000	405.000
UNITS	3844.944	7575.281	3851.685	4278.049	7991.463	4014.634
FUEL	13.933	390.225	11.449	18.927	429.220	19.341
OPS	64.258	146.067	62.809	76.098	236.220	64.976
TOTAL	4881.135	34979.573	4523.943	4960.074	56519.903	4503.951

Table 11.9: Cost of SeqACLPP-WAC across all scenarios

	A-base	A-high	A-fast	B-base	B-high	B-fast	∅
PEN	-65.871	36.159	-78.443	-80.506	19.553	-81.591	-41.783
UNITS	4.457	4.657	5.999	5.093	6.104	1.636	4.658
FUEL	-31.828	-6.469	-70.004	-66.788	-7.871	-56.537	-39.916
OPS	-22.821	-12.275	-8.511	-17.230	-4.478	-34.929	-16.707
TOTAL	-25.949	28.079	-30.545	-31.355	17.067	-28.445	-11.858

Table 11.10: Cost improvement of SeqACLPP-WAC compared to SeqACLPP

(DISP) by 3.4 percent and share of ULDs with both standard and express shipments (MIX) by 8.6 percent is also noticeable albeit less drastic.

In the base and fast scenarios the disadvantage of a higher handling effort is compensated by a significant reduction of most cost components. On average the penalties for offloaded items can be reduced by more than 76 percent in those scenarios. The bad performance of the WAC for the high scenarios also shows in the penalty cost, which are 27.9 percent higher than in the SeqACLPP approach. The cost for built ULDs increases across all scenarios. This effect is in line with the rise in the number of used ULDs. The cost for extra fuel due to imbalances (FUEL) can be reduced by 39.9 percent and handling operations (OPS) by 16.7 percent from their already low absolute levels. Thus, in the base and fast scenario an almost ideal balance can be achieved for most of the flights, while the number of performed reloading operations is less than one ULD per flight on average. Altogether, the total cost for flights in the base and fast scenarios could be reduced by more than 29 percent compared to the SeqACLPP approach. But, for highly booked flights the total cost is around 22.6 percent higher than in the SeqACLPP approach, due to the increased offloads. Therefore, it seems beneficial to choose the concrete solution approach on a flight-per-flight basis, depending on the respective utilization.

At last, we analyze the runtime of the WAC approach. As in the last section, we first look at the runtime of the WAC model itself. After that, we discuss the impact on the runtime of the other planning steps.

Similar to the SAC model, the WAC model takes a noticeable amount of time to be solved to optimality for most flights. Therefore, we also set a time limit of 30 seconds for the aircraft configuration step. Around 50 percent of the flights can be solved down to a 1 percent MIP gap until this time limit is reached. The distribution of the resulting MIP gaps of the remaining flights is shown in Figure 11.2. Almost all of those flights show a MIP gap of less than 10 percent.

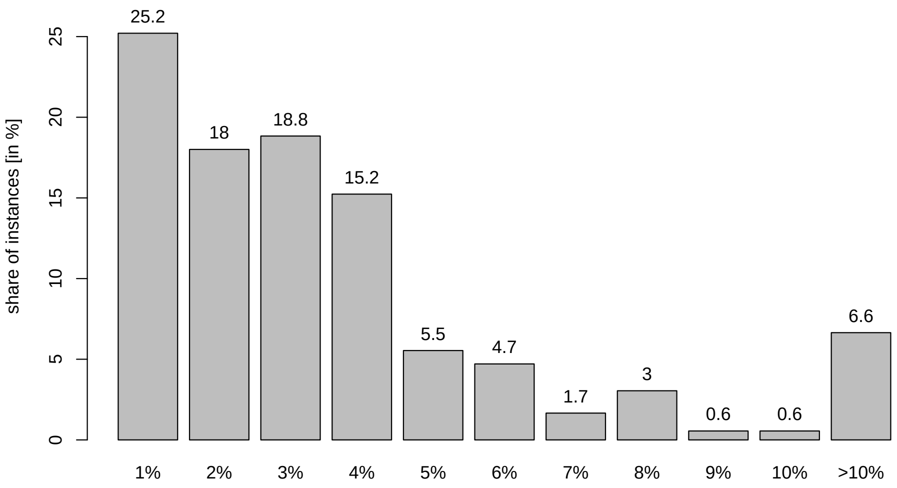


Figure 11.2: WAC MIP gap distribution after 30 seconds runtime (rounded to 5%)

Runtime	A-base	A-high	A-fast	B-base	B-high	B-fast	∅	Δ
WAC [sec]	20	11	20	17	11	15	16	$+\infty$
BDAP [sec]	1,067	2,152	1,145	1,083	2,126	1,109	1,447	+18.8%
UWAB [sec]	45	114	51	78	221	78	98	+60.6%
Total [sec]	1,132	2,277	1,216	1,178	2,358	1,202	1,561	+22.1%

Table 11.11: Computational effort of SeqACLPP-WAC across all scenarios

An overview of the computational effort required for the planning steps is given in Table 11.11. The runtime of the palletization step (BDAP+) increases by 18.8 percent compared to the original BDAP. We attribute this to the additional target weight constraint and the calculation of the related deviation penalty. Also, the weight and balance step (UWAB) takes longer, about 60.6 percent. Part of this can be explained by the larger number of ULDs to load, as was the case at the SeqACLPP-SAC approach. Furthermore, the better weight distribution across the ULDs creates a larger search space for the model, because lighter ULDs can be placed at a higher number of loading positions.

Altogether, we can say that the incorporation of weight and balance aspects into the aircraft configuration step works very well for flights that are not fully booked. However, on overbooked flights the solutions are worse than in the SeqACLPP approach. Fortunately, as the booked weight and volume of a flight is known at the time of load planning, the better suiting approach can be chosen for each flight individually.

11.3 Summary

In this chapter, we presented two extensions of the SeqACLPP approach with the aim to reduce the penalty of offloaded items as seen in Chapter 10.

In our first extension, denoted as Shipment-based Aircraft Configuration (SeqACLPP-SAC), we integrated the non-geometric aspects of the palletization step into the aircraft configuration step. Besides selecting the ULDs, the model also assigns items to ULDs. Thereby, it obtains a more detailed view of the problem that results in a more suitable set of selected ULDs. Across all scenarios, this approach could reduce the penalty of offloaded items (PEN) by 13.5 percent. This resulted in a reduction of the total cost by 5.7 percent. The increased load resulted in about 2.5 percent more ULDs to be built (UNITS) and the intended separation of items (MIX) could be improved by 12 percent. Besides, the solution quality and runtime showed no significant changes to the SeqACLPP approach.

The second extension, denoted as Weight-and-balance-based Aircraft Configuration (SeqACLPP-WAC), integrates weight and balance aspects like the weight distribution and structural aircraft limits into the aircraft configuration step. Besides the ULD selection, each ULD is now assigned to a loading position and given a target weight. If the target weights are met during the palletization step, all ULDs are loadable and lead to a good aircraft balance. Therefore, we also extended the palletization step, to respect the target weights as a soft constraint. This approach worked very well for the base and fast scenarios. Over all scenarios the penalties of offloaded items (PEN) could be reduced by 41.8 percent and resulted in 4.7 percent more ULDs (UNITS) to be built. The cost for additional fuel due to imbalance (FUEL) could be significantly reduced by 39.9 percent. Furthermore, the cost of reloading operations (OPS) could be reduced by 16.7 percent. This led to a reduction of total costs by 29 percent across the base and fast scenarios. However, over all scenarios the total cost improvement was only 11.9 percent, as the approach did not produce good results for the high-load scenarios. Here, on average 4.2 percent less weight could be loaded per flight. In this case, the new model focuses too much on cheap aircraft operations, which is not the prime concern on the highly utilized flights.

Altogether, we find that a coordination of the planning steps seems to be a promising way to improve solution quality with reasonable implementation and computational effort. While the proposed extensions work well for ordinary utilized flights, for highly booked flights the SeqACLPP approach should be preferred.

12 Conclusion

In this chapter, we summarize the main contributions and results of this thesis. We discuss our findings regarding the stated research questions and conclude with an outlook on promising future research.

12.1 Summary

The goal of this work was to understand the load planning problem faced by cargo airlines and develop suitable solution approaches by using operations research methods to create a decision support system. Along this way, we generated several contributions to the state of research, which we introduce in the following in the order of their appearance in this thesis. Besides, we give a graphical overview of the developed models and solution approaches in Figure 12.1.

In Chapter 2, we presented a brief introduction into the air cargo business as well as the operational handling processes of airlines that transport cargo. After that, we gave a comprehensive overview of the Air Cargo Load Planning Problem (ACLPP) and its four subproblems: the Aircraft Configuration Problem (ACP), the Build-up Scheduling Problem (BSP), the Air Cargo Palletization Problem (APP), and the Weight and Balance Problem (WBP) in Chapter 3. We identified the three main stakeholders of load planning: sales, handling, and aircraft operations. Their respective objectives on the problem are maximizing revenue, minimizing physical handling effort, and minimizing operational cost of the flight. Furthermore, we discussed the challenges of current manual planning approaches.

In the next four Chapters 4 to 7 we introduced the ACLPP subproblems one by one in detail. For each subproblem, we reviewed the existing literature and formally specified an optimization model (ACM, BSM, APM, WBM). The ACM and BSM are the first models tackling the respective problems. For the APP and WBP only models partially covering the aspects of our problem at hand could be found. The APM especially introduces the following features that have not yet been dealt with in the literature:

- It considers time as a new dimension of the packing problem. Only items that are available before the assembly time of a ULD can be assigned to it.
- The stacking and stability constraints track the statics between the items and allow for partial support as well as placement on uneven surfaces.
- To cover item allocation restrictions, a soft item sorting constraint is added. It allows for groups of items to be packed preferably together or onto separate ULDs.

In the WBM we consider the union of constraints found in the literature. Besides, we added the following practical aspects:

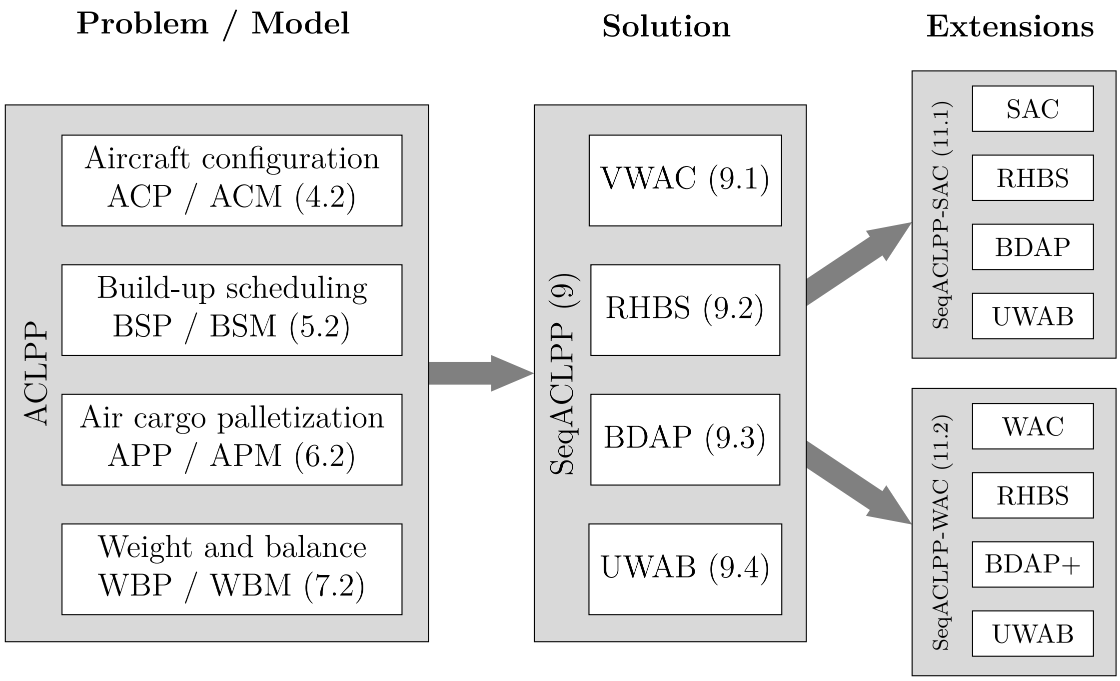


Figure 12.1: Components presented in this work, along with the sections they are introduced in.

- The formulation accommodates an arbitrary number of flight legs.
- A new dependency constraint is added that forces certain loading positions to be occupied to be allowed to use other dependent positions.
- A variety of formulations of cumulative load constraints have been presented in the literature. In our formulation, one standardized constraint is present that handles all cumulative weight limits with maximum flexibility.

To evaluate solution approaches to the ACLPP, we introduced a set of benchmark instances in Chapter 8. Therefore, we created a data model covering all the considered aircraft and ULD characteristics, regulatory requirements, and booking data. Furthermore, we presented an analysis of real world air cargo booking data to grasp the problem structure and planning difficulties. We found that the flights are often not fully booked. Accordingly, dense packing is not always the prime objective and planning results that lead to efficient handling as well as aircraft operations are also highly important. The items to be loaded are strongly heterogeneous in their dimensions, weight, and density and a significant part of the items is quite large, often exceeding the size of a standard wooden pallet. We introduced two benchmark datasets, one non-public set based on real flight data and a public set created by us that contains sampled booking data. From both sets we generated two additional scenarios of use cases that are interesting from a practical point of view. The first contains only overbooked flights and in the other the bookings have shorter transfer times at the terminal. From the public dataset, we provide a dataset of 246 realistic flights for future research.

To solve the ACLPP in practice, we designed and implemented a decision support system. Its details are presented in Chapter 9. Our approach, denoted as SeqACLPP, consists of a sequential four-step planning pipeline that follows the current manual processes in practice. The ACP is solved for each flight by a MIP model implementing a Weight-and-Volume-based Aircraft Configuration (VWAC). For the BSP we presented a rolling horizon approach (RHBS) considering all flights in the planning horizon. It solves one shift per iteration by also employing a MIP model. We realized the APP with a Logic-based Benders Decomposition approach, denoted as BDAP, which is solved for each transport segment of a flight. It consists of a MIP master problem and subproblems solved via a constraint program. The master problem, denoted as BinAssign, distributes the items among the ULDs and covers all non-geometric aspects of the problem. The subproblem, denoted as CPPack, captures all the three-dimensional packing aspects. Finally, the WBP is solved for each flight by a MIP model, denoted as ULD-based Weight and Balance (UWAB).

We successfully evaluated the SeqACLPP approach on the introduced benchmark instances and reported the results in Chapter 10. As the approach contains a number of parameters, we introduced a set of methods to determine suitable settings. Furthermore, we discussed the achieved results with respect to solution quality and effort and identified areas for improvement.

Based on identified weaknesses of the SeqACLPP approach, we introduced and evaluated two coordinated planning approaches in Chapter 11. In the SeqACLPP-SAC approach we incorporate aspects of the APP into the ACP by replacing the VWAC with a new Shipment-based Aircraft Configuration (SAC). In the same way, in the SeqACLPP-WAC approach we replace the VWAC with a Weight-and-balance-based Aircraft Configuration (WAC) to consider WBP aspects within the ACP step. Finally, we analyzed the effects of both approaches on the solution quality and runtimes. Both approaches can further improve the solutions.

12.2 Research questions

In this work, we followed three research questions introduced in Section 1.2. As indicated in the previous summary section we found satisfactory answers to all of them. In the following, we give a brief statement of the results and point out the parts of this work where detailed information can be found.

1) What are the requirements and drivers of air cargo load planning in practice? The ACLPP affects three airline departments: sales, handling, and aircraft operations. All three have partially conflicting objectives on the subject. We identified four parts of the problem: aircraft configuration, build-up scheduling, palletization, and weight and balance. After introducing the general planning process in Chapter 3, we described the four parts in detail and formally in Chapters 4 to 7.

2) Can valid and competitive load plans be generated with reasonable effort? From the identified parts of the problem, we derived the four-step planning approach SeqA-CLPP. We implemented it and evaluated its performance on real and synthetic benchmark instances. Compared to real load plans seen in practice, the average packing density could be improved by more than 20 percent and the performance with respect to the required handling effort and aircraft operations could be significantly improved. The required runtimes range between 13 and 38 minutes per flight, which is acceptable for non-interactive planning. We described our solution approach in Chapter 9 and its evaluation in Chapter 10.

3) Does splitting the problem lead to adverse effects and what can we do about it? During the evaluation of the SeqACLPP approach we found that considering each solution step independently leads to a higher than necessary amount of offloaded items. The reason typically is an overestimation of the capabilities of later steps. For example, the volume capacity of a single ULD is easily overestimated if only the volume of the items to be loaded is considered. Furthermore, the aircraft capacity for heavy ULDs is overestimated if only a subset of the ULDs to load is considered. To overcome those effects, we incorporated constraints of later planning steps into the aircraft configuration step. Thereby, we could improve the solution quality for four out of six scenarios. These coordinated planning approaches are presented in detail in Chapter 11.

12.3 Future work

The presented approaches provide numerous opportunities for improvement and future research. Our first set of suggestions targets technical aspects of the presented approaches. During the Logic-based Benders Decomposition of the palletization step (BDAP), the heuristic cuts added in each iteration are not always effective and therefore many iterations are necessary before the process comes to an end. Here, improved cuts, ideally representing minimal infeasible sets of items, could be advantageous. Especially in the high-load scenarios, the offloaded items provided many alternatives to bypass the cuts and still produce unpackable results. For these cases, a dual perspective for cuts that limit the usable bin capacity might be promising.

Furthermore, the runtimes need to be improved to make the solution process interactively usable. Besides better cuts, this could be achieved through parallelization or further decomposition of the problems. In the BDAP we decomposed the problem by the volume of the items. Besides that, the separate planning of express items or dangerous goods might be conceivable.

The second area for enhancements we identified is to improve the solution quality and reduce the search space by a further integration of the planning steps. Most obvious, the presented coordinated solution approaches, i.e., the shipment-based and weight-and-balance-based aircraft configuration, could be put into one model. Besides that, scheduling aspects, like item availability or shortages of the build-up capacity, could be added to the aircraft configuration. Furthermore, the combined weight limits of the weight and balance step could be integrated into the palletization step, to provide a more detailed picture of loadable ULD weights.

A third field of future research is the integration of additional practical aspects into the model. In the palletization step, we introduced a stacking constraint with a certain height tolerance. In practice the emerging vertical gaps can be filled by using dunnage material if necessary. However, an explicit planning of dunnage material could also be interesting. Especially on lower deck ULDs, where only overhanging items can fill the full contour, a scaffold made of wooden pallets or boards is sometimes used in practice to be able to pack more items onto the ULD.

Next, during the build-up scheduling we assumed a constant assembly time for each ULD type to keep things simple. However, discussions with practitioners suggest that the structure of the items assigned to a ULD also has an impact on the assembly time. Furthermore, it is also occasional practice to assign multiple crews to a ULD build-up to speed up the process to a certain extent. Incorporating these aspects adds more degrees of freedom to the designed load plans and could therefore improve solution quality.

Finally, our model assumed full knowledge of all item properties. Although accurate data is increasingly available, items might always differ from the shippers statement or stated properties like the load-bearing strength might turn out to be wrong. To cope with such uncertain data, an idea would be to introduce some kind of Monte Carlo sampling of item properties, solve the three-dimensional packing problem for multiple scenarios and only accept solutions with a certain feasibility probability.

12.4 Outlook

Our evaluation showed that the presented approaches are suitable to automatically generate load plans for cargo flights. The generated load plans are significantly better than plans manually created by load planning experts. Compared to the current operational practice, we could, for example, achieve a 20 percent higher packing density and reduce the required warehouse to workstation transports in the air cargo terminal by more than 17 percent due to better item sorting. The costs of additional fuel burn due to aircraft imbalances or ULD reloading operations at stop-over airports in our solutions are almost negligible.

This shows that cargo airlines can significantly profit from employing the presented approaches in their operational practice. More and especially the profitable last-minute cargo can be transported. Furthermore, the costs of load planning, handling effort, and aircraft operations can be significantly reduced. In the highly competitive air cargo business this can provide airlines with a competitive edge to operate profitably.

Besides the operational load planning, further use cases of the presented approaches are imaginable. They could be used throughout the booking process to estimate the actual aircraft utilization of a flight or to calculate the effective space required by a new booking. On a tactical level new products or routes could be evaluated by exemplary load planning. Even the aircraft can be optimized by simulating the effects of added or removed equipment, like floor locks or additional fuel tanks, on the load plans in practice. These changes modify the aircraft configuration space and hence its loading capabilities, but they also alter the weight of the unloaded aircraft and thus its fuel consumption on any flight.

Bibliography

Amadeus (2015). Amadeus Flight Management. Technical report, Amadeus IT Group SA. Available online from: www.amadeus.com, last accessed: 18.05.2017.

Bischoff, E. and M. Ratcliff (1995). Issues in the development of approaches to container loading. *Omega 23*(4), p. 377 – 390.

Boeing (2011). MD-11 airplane characteristics for airport planning. Technical report, Boeing Commercial Airplanes. Available online from: www.boeing.com, last accessed: 18.05.2017.

Bortfeldt, A. and G. Wäscher (2013). Constraints in container loading: A state-of-the-art review. *European Journal of Operational Research 229*(1), p. 1 – 20.

Brandt, F., J.-N. Popratnjak and E. Schulz (2014, 3). Packing with Item Availability and Bin Due Dates. Presentation at the 11th Meeting of the EURO Special Interest Group on Cutting and Packing.

Brosh, I. (1981). Optimal cargo allocation on board a plane: a sequential linear programming approach. *European Journal of Operational Research 8*(1), p. 40 – 46.

Burnson, P. (2013). Top 25 Freight Forwarders: Top 25 prevail despite market divergence. Logistics Management. Available online from: www.logisticsmgmt.com, last accessed: 18.05.2017.

Ceschia, S. and A. Schaerf (2013). Local search for a multi-drop multi-container loading problem. *Journal of Heuristics 19*(2), p. 275–294.

Chan, F. T., R. Bhagwat, N. Kumar, M. Tiwari and P. Lam (2006). Development of a decision support system for air-cargo pallets loading problem: A case study. *Expert Systems with Applications 31*(3), p. 472 – 485.

Crabtree, T., T. Hoang, J. Edgar and R. Tom (2014). Boeing World Air Cargo Forecast 2014-2015. Technical report, Boeing Commercial Airplanes. Available online from: www.boeing.com, last accessed: 18.05.2017.

Crabtree, T., T. Hoang, R. Tom and G. Gildemann (2016). Boeing World Air Cargo Forecast 2016-2017. Technical report, Boeing Commercial Airplanes. Available online from: www.boeing.com, last accessed: 18.05.2017.

Crainic, T. G., G. Perboli and R. Tadei (2008). Extreme Point-Based Heuristics for Three-Dimensional Bin Packing. *INFORMS Journal on Computing 20*(3), p. 368–384.

Dahmani, N. and S. Krichen (2016). Solving a load balancing problem with a multi-objective particle swarm optimisation approach: application to aircraft cargo transportation. *International Journal of Operational Research 27*(1-2), p. 62–84.

de Cleyn, S., K. de Vos, T. Kallstenius, E. van Bruystegem, W. van Daele and S. Vermeulen (2014). Optimization takes flight with weight and balance software for airlines. Technical report, iMinds insights. Available online from: www.iminds.be, last accessed: 24.04.2017.

Egeblad, J., C. Garavelli, S. Lisi and D. Pisinger (2010). Heuristics for container loading of furniture. *European Journal of Operational Research 200*(3), p. 881 – 892.

Eley, M. (2003). A bottleneck assignment approach to the multiple container loading problem. *OR Spectrum 25*(1), p. 45–60.

FAA (2016). Aircraft Weight and Balance Handbook. Technical report, Federal Aviation Administration. Available online from: www.faa.gov, last accessed: 18.05.17.

Feng, B., Y. Li and Z.-J. M. Shen (2015). Air cargo operations: Literature review and comparison with practices. *Transportation Research Part C: Emerging Technologies 56*, p. 263 – 280.

Fuellerer, G., K. F. Doerner, R. F. Hartl and M. Iori (2010). Metaheuristics for vehicle routing problems with three-dimensional loading constraints. *European Journal of Operational Research 201*(3), p. 751 – 759.

Grandjot, H., I. Roessler and A. Roland (2007). *Air Cargo Guide: An Introduction to the Air Cargo Industry*. Huss.

Guéret, C., N. Jussien, O. Lhomme, C. Pavageau and C. Prins (2003). Loading Aircraft for Military Operations. *The Journal of the Operational Research Society 54*(5), p. 458–465.

Heidelberg, K. R., G. S. Parnell and J. E. Ames (1998). Automated air load planning. *Naval Research Logistics (NRL) 45*(8), p. 751–768.

Hodgson, T. J. (1982). A Combined Approach to the Pallet Loading Problem. *AIIE Transactions 14*(3), p. 175–182.

Hong Ha, H. T. and N. Nananukul (2016). Air Cargo Loading Management System for Logistics Forwarders. In: *Proceedings of 2016 International Conference on Urban Planning, Transport and Construction Engineering*, Pattaya, p. 51–58.

Hooker, J. N. (2007). Planning and Scheduling by Logic-Based Benders Decomposition. *Operations Research 55*(3), p. 588–602.

IATA (2016). Dangerous Goods Regulations. International Air Transport Association. Available online from: www.iata.org, last accessed: 18.05.2017.

ISO (2003). ISO 6780:2003: Flat pallets for intercontinental materials handling – Principal dimensions and tolerances. Technical report, International Organization for Standardization.

Junqueira, L., R. Morabito and D. S. Yamashita (2012). Three-dimensional container loading models with cargo stability and load bearing constraints. *Computers and Operations Research 39*(1), p. 74 – 85.

Larsen, O. and G. Mikkelsen (1980). An interactive system for the loading of cargo aircraft. *European Journal of Operational Research 4*(6), p. 367 – 373.

Lau, H. C. W., T. M. Chan, W. T. Tsui, G. T. S. Ho and K. L. Choy (2009). An AI approach for optimizing multi-pallet loading operations. *Expert Systems with Applications 36*(3), p. 4296–4312.

Limbourg, S., M. Schyns and G. Laporte (2012). Automatic aircraft cargo load planning. *Journal of the Operational Research Society 63*(9), p. 1271–1283.

Lin, J. L., C. H. Chang and J. Y. Yang (2006). A study of optimal system for multiple-constraint multiple-container packing problems. *Lecture Notes in Computer Science 4031 LNAI*, p. 1200–1210.

Lurkin, V. and M. Schyns (2015). The Airline Container Loading Problem with pickup and delivery. *European Journal of Operational Research 244*(3), p. 955 – 965.

Mongeau, M. and C. Bes (2003). Optimization of aircraft container loading. *Aerospace and Electronic Systems 39*(1), p. 140–150.

Murray, G. (2016). Load factors at historic lows, says IATA. Air Cargo News. Available online from: www.aircargonews.net, last accessed: 18.05.2017.

Nance, R., A. Roesener and J. Moore (2011). An advanced tabu search for solving the mixed payload airlift loading problem. *The Journal of the Operational Research Society 62*(2), p. 337–347.

Nobert, Y. and J. Roy (1998). Freight Handling Personnel Scheduling at Air Cargo Terminals. *Transportation Science 32*(3), p. 295–301.

Paquay, C., M. Schyns and S. Limbourg (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*(2), p. 187–213.

Pisinger, D. and M. Sigurd (2007). Using decomposition techniques and constraint programming for solving the two-dimensional bin-packing problem. *INFORMS Journal on Computing 19*(1), p. 36–51.

Pollaris, H., K. Braekers, A. Caris, G. K. Janssens and S. Limbourg (2015). Vehicle routing problems with loading constraints: state-of-the-art and future directions. *OR Spectrum 37*(2), p. 297–330.

Rong, A. and M. Grunow (2009). Shift designs for freight handling personnel at air cargo terminals. *Transportation Research Part E: Logistics and Transportation Review 45*(5), p. 725 – 739.

Techanitisawad, A. and P. Tangwiwatwong (2004). A GA-based Heuristic for the Interrelated Container Selection Loading Problems. *IEMS 3*(1), p. 22–37.

The Economist (2016). Too little freight, too much space. The Economist. Available online from: www.economist.com, last accessed: 24.04.2017.

Tyler, T. (2015). IATA Annual Review 2015. Technical report, International Air Transport Association. Available online from: www.iata.org, last accessed: 18.05.2017.

Vancroonenburg, W., J. Verstichel, K. Tavernier and G. V. Berghe (2014). Automatic air cargo selection and weight balancing: A mixed integer programming approach. *Transportation Research Part E: Logistics and Transportation Review 65*, p. 70 – 83.

Wikipedia (2016). Pallet — Wikipedia, The Free Encyclopedia. Available online at: http://en.wikipedia.org/wiki/Pallet, last accessed 31.10.2016.

Wu, Y. (2008). Modelling containerisation of air cargo forwarding problems. *Production Planning & Control 19*(1), p. 2–11.

Yan, S., C.-H. Chen and C.-K. Chen (2006a). Long-term manpower supply planning for air cargo terminals. *Journal of Air Transport Management 12*(4), p. 175–181.

Yan, S., C.-H. Chen and C.-K. Chen (2008a). Short-term shift setting and manpower supplying under stochastic demands for air cargo terminals. *Transportation 35*(3), p. 425–444.

Yan, S., C.-H. Chen and M. Chen (2008b). Stochastic models for air cargo terminal manpower supply planning in long-term operations. *Applied Stochastic Models in Business and Industry 24*, p. 261–275.

Yan, S., C. K. Chen and C. H. Chen (2006b). Cargo terminal shift setting and manpower supplying in short-term operations. *Journal of Marine Science and Technology 14*(2), p. 109–118.

Yan, S., C.-T. Lo and Y.-L. Shih (2006). Cargo Container Loading Plan Model and Solution Method for International Air Express Carriers. *Transportation Planning and Technology 29*(6), p. 445–470.

List of Figures

List of Tables

Glossary

ACLPP Air Cargo Load Planning Problem
ACM Aircraft Configuration Model (see Section 4.2.5)
ACP Aircraft Configuration Problem
AKE standard half-width ULD container with dimensions 200.7x153.4 cm (79x60.4 in)
APM Air Cargo Palletization Model (see Section 6.2.5)
APP Air Cargo Palletization Problem

BDAP LBBD for air cargo palletization step (see Section 9.3)
BinAssign Bin Assignment Model (see Section 9.3.1)
BSM Build-up Scheduling Model (see Section 5.2.5)
BSP Build-up Scheduling Problem

CG center of gravity
CPPack Constraint Program 3D Packing Model (see Section 9.3.2)
CREWS sum of required build-up crews (perf. indicator, see Section 8.4)

DGR Dangerous Goods Regulations
DISP dispersion of splitted shipments (perf. indicator, see Section 8.4)

EUR-pallet standard wooden pallet 80x120 cm (ISO 6780)

Flight a sequence of legs performed under the same flight number
FUEL cost of extra fuel due to aircraft imbalance (perf. indicator, see Section 8.4)

GLF volume gross load factor (perf. indicator, see Section 8.4)

IATA International Air Transport Association

LBBD Logic-based Benders Decomposition (see Section 9.3.3)
Leg non-stop aircraft movement between two airports
LTC long-term contract

MD11F McDonnell Douglas MD-11 freighter aircraft
MHKP Multiple Heterogeneous Knapsack Problem
MI moment of inertia
MIP mixed-integer linear program

MIX share of ULDs with multiple products (perf. indicator, see Section 8.4)

NLF volume net load factor (perf. indicator, see Section 8.4)

OEW operating empty weight of an aircraft
OPS sum of unnecessary ULD loading operations (perf. indicator, see Section 8.4)

PEN penalty of offloaded cargo (perf. indicator, see Section 8.4)
PGE large ULD pallet with dimensions 243.8x605.8 cm (96x238.5 in)
PMC standard ULD pallet with dimensions 243.8x317.5 cm (96x125 in)
POI Point of interest

RFS road feeder service (trucking services between airports)
RHBS Rolling horizon build-up scheduling step (see Section 9.2)

SAC Shipment-based Aircraft Configuration (see Section 11.1)
Segment sequence of legs on which cargo can be transported
SeqACLPP Sequential planning approach to the ACLPP
SPLIT share of shipments split across multiple ULDs (perf. indicator, see Section 8.4)

ULD unit load device
UNITS sum of used ULDs (perf. indicator, see Section 8.4)
UWAB ULD weight and balance step (see Section 9.4)

VWAC Volume-and-weight-based aircraft configuration step (see Section 9.1)

WAB weight and balance
WAC Weight-and-balance-based Aircraft Configuration (see Section 11.2)
WBM Weight and Balance Model (see Section 7.2.5)
WBP Weight and Balance Problem
WLF weight load factor (perf. indicator, see Section 8.4)