Optimizing Air Cargo Handling at an International Airline Hub for AbOvo

Ilkim Canoler[*] Karthikeyan Ramasubbu[*] Luckshan Sivakumar[*] Nikhil Kulkarni [*]
Shekhar Dure[*]

8th July 2019

Abstract

In this study, we solve an operational planning problem in the air cargo industry. In an international airline hub, all shipments which arrive at the airport and have to go to another destination need to go through two processes, namely, breaking down of the unit load devices (ULDs) that arrived in and building up of ULD for their destination flight. The final goal is that all shipments have to catch their connection flight on time. Due to the complexity of the overall problem, current practices at airlines are to split up the problem into two smaller and thus simpler problems. The goal of this paper is to solve the full planning problem using a single model. Working with our industrial partner AbOvo, we solved the problem of scheduling the break down and build up process according to the departure timings of the destination flights to ensure that all shipments reaches the flight. We formalize its requirements and the objectives. Furthermore, we develop and evaluate two different solution approaches. We solve the full planning problem by using integer linear programming.

Keywords: *Time Discretization,.*

[*] {ilkim.canoler@rwth-aachen.de, karthikeyan.ramasubbu@rwth-aachen.de, luckshan.sivakumar@rwth-aachen.de, nikhil.kulkarni@rwth-aachen.de, shekhar.dure@rwth.aachen.de},M.Sc. Data Analytics and Decision Science, RWTH Aachen University, Germany

Contents

1 Introduction

An international airline hub needs to plan the air cargo movement so that all shipments catch their connecting flights or road connections. Air cargo shipments are transported in so-called unit load devices (ULDs). It allows a large quantity of cargo to be bundled into a single unit like a pallet or a container. A ULD may contain many shipments which need to go to different final destinations. Hence, at the first step of the whole process as seen in Figure 1 below, the incoming ULDs are broken down on arrival. Here in the breakdown zone (BD), ULDs have to be unpacked and seperated. The seperate shipments are then transported to the warehouse and stored there. These shipments are then moved to the packing area which is called buildup zone (BU). Finally, the outbound ULDs are constructed from these shipments, and they are taken to the outbound flight for loading. Each zone has different attributes such as capacitiy for BD zone, the number of workstations for BU zone, transportation times between zones and the warehouse and handling times for process in each zone. Shipments have different attributes namely shipment id, destination flight number, date of arrival and departure, type of shipment and weight. Using all the attributes , we formulate the problem as an Integer linear program by writing constraints and the parameters which are stated in detail in the problem description section. The goal of the optimization process is to build up all ULDs such that there is sufficient time available for them to catch their flight connections.

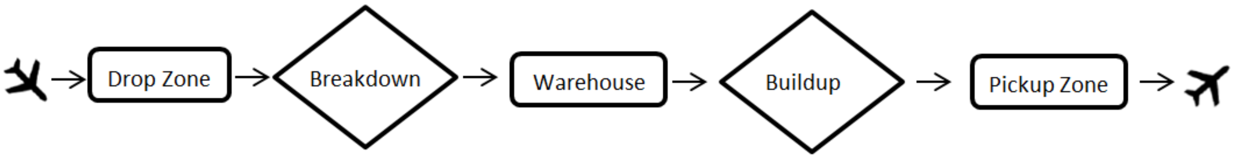

Figure 1: The Process That Takes Place at a Hub

2 Problem Description

In the given problem, shipments arrive in the form of an ULD at four types of Drop Zones (DZ) in the airport. ULDs are unloaded from an aircraft by the ground handling agent and placed in one of the drop zones. The decision to assign the DZ is not a part of our planning problem in this paper as it has already been given in the data associated with each shipment. From the drop zone, each ULD has to be transported to the breakdown zone to be broken down into shipments. The decision of allocating an ULD to a BD zone has to be taken at this point.

In BD zone, ULDs are seperated into shipments. There are 12 different BD zones in the hub that are located at different places of the airport. Attributes related to BD zones are:

- Type of shipment a BD zone handles. For examle : some of the BD zones handle only ULDs containing animals, some of them handle only cooled products and others handle regular ULDs.

- Capacity of the BD zone.
- Transportation times from DZ and to Warehouse.
- Handling time per ULD.

The inbound ULDs are classified into product categories of, 'Cooled', 'normal' and 'animals'. When the shipments are unpacked, depending on the scheduled departure time of the connection flight it is booked for, it is either stored temporarily in the warehouse or directly sent to the build up process. The warehouse (WH) is fully automated and with the assumption, that there are never capacity constraints in the WH. As soon as enough stock is available in the storage WH to build up one ULD for a specific aircraft, these can be requested to be provided to the BU zone.

In a next step, shipments are built up into ULDs depending on their connecting flight. This is done in a build up zone (BU zone). A departing aircraft type has a link to a specific BU zone. So with the provided information on which aircraft the shipment should go on, the BU zone is known. There are 8 different BU zones and each BU zone has different number of workstations in it. At each work station only one ULD can be built up at any given time. The shipments are provided from the storage WH to the chosen work station. From each BU zone, there are different transportation times for ULDs to each flight. In addition to this, there are different pre-processing times for each flight which is also given in the data. Maximum capacity by weight that each built up ULD can have is given as 400 kg.

Some other requirements are that if the building up of two ULDs for the same aircraft are planned at one work station, it is not allowed to build up a ULD for a different aircraft in between, even if there is some idle time. The reason is that after building up one ULD there might be some shipments with the same destination left. It should be avoided to move those around, so they would stay at the work station for the build up of the next ULD for the same aircraft. Listen below are attributes associated with BU zone

- Handling times to buildup a ULD
- Transport times from the WH
- Number of workstations
- Flights list associated with each BU zones and the transportation time between the flight and the BU zone.
- Pre-processing buffer times for the flights.

The goal is to build up all ULDs before the given buffer time so that all can catch the connections.

3 Mathematical Model for the Problem

Our mathematical model for this problem is a combination of two puzzles namely, the Break Down Zone and the Build Up Zone assignments. The overall view of the general process can be seen in Figure 2 below.

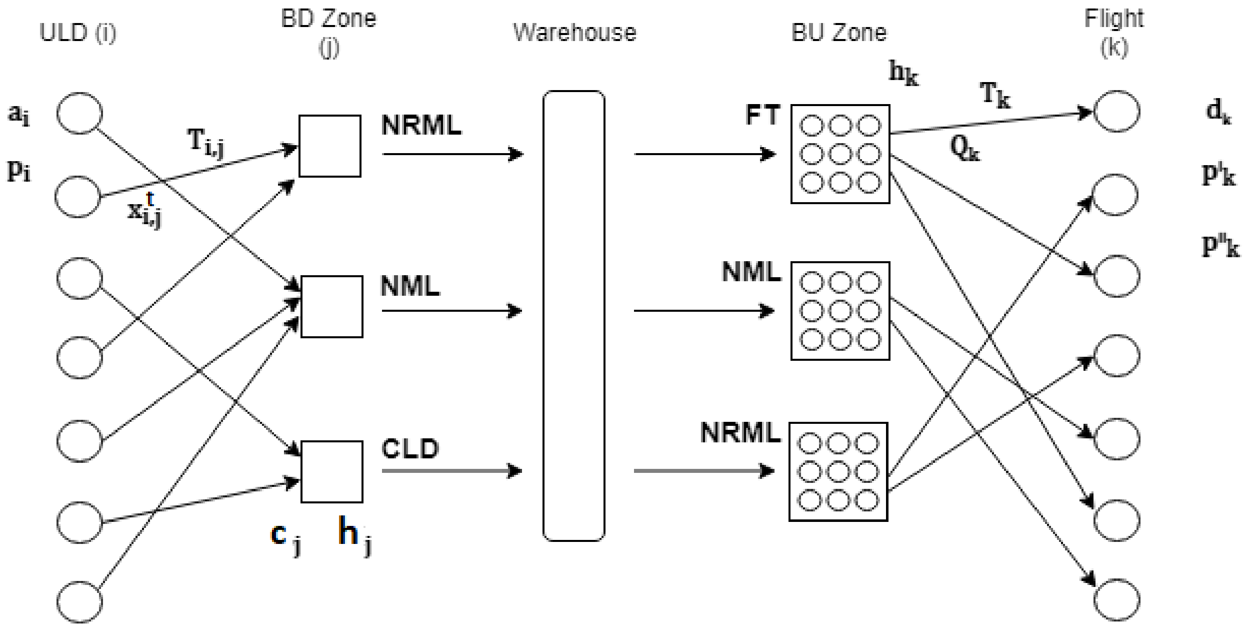

Figure 2: Overall View for The Aircargo Process

3.1 Break Down Process

Each ULD is a part of the ULD set I and each BD zone is a part of the BD zones set J. Time is discretized into minutes in set T.

ULD i ∈ I = {1,2,3,...,N}

BD zone j ∈ J = {1,2,3,...,M}

Time horizon t ∈ T = {1,2,3,...,T}

Shipments l in ULD i ∈ Li = {1,2,3....,L}

a_i : Arrival time of ULD i in drop zone
d_i : Earliest Departure time of a shipment in ULD i
c_j : Capacity of BD zone j
h_j : Handling time in BD zone j
T_{ij} : Time taken to transport an ULD i to BD zone j
T_j : Time taken to transport shipments from i to BD zone j to Warehouse

3.1.1 Decision Variables for Break Down Process

In the Break Down zone, the decision needs to be taken which ULDs should be assigned to which BD zone at which time. For this reason, the decision variable has to be binary. The decision variable takes value 1 if the ULD i is assigned to BD zone j at time t and value 0 if the ULD i is not assigned to BD zone j at time t.

$x_{ij}^t \in \{0,1\}, \forall i \in I, j \in J, t \in T$: if a ULD i is assigned to BD zone j at time t or not.

3.1.2 Constraints for Break Down Process

The first constraint (1) ensures that the ULD is assigned at a feasible time, the second constraint (2) ensures that all ULDs have to be assigned to exactly one BD zone.

The time at which Break down of ULD i starts, $t_{i,start} = \sum_{j \in J} \sum_{t=a_i}^{d_i} t \cdot x_{ij}^t$

Each ULD takes i T_{ij} time to travel to its assigned BD zone j, therefore the following constraint ensures that the start time of ULD i is greater than the sum of its arrival time and transportation time :

$$\sum_{j \in J} \sum_{t=a_i}^{d_i} (t - T_{ij}) \cdot x_{ij}^t \geq a_i, \qquad \forall i \in I \tag{1}$$

Each ULD i must be assigned to a BD zone j before the earliest departure time of the shipment it contains. We will further tighten this time limit in other constraints that follow. The following constraint ensures that each ULD is assigned :

$$\sum_{j \in J} \sum_{t=a_i}^{d_i} x_{ij}^t = 1 \qquad \forall i \in I \tag{2}$$

For the special case given in this problem, where we have multiple drop zones, the each ULD has to be assigned to multiple BD zones associated with it. In the constraints (3) and (4), we assign such multiple drop zone ULDs to their respective BD zones i.e animal BD zone and normal BD zone.

$$\sum_{j \in J_{NML}} \sum_{t=a_i}^{d_i} x_{ij}^t = 1 \qquad \forall i \in I \tag{3}$$

$$\sum_{j \in J_{NRML}} \sum_{t=a_i}^{d_i} x_{ij}^t = 1 \qquad \forall i \in I \tag{4}$$

In the fifth consraint (5), we consider the case that for multiple drop zone ULDs, animal BD zone has to be assigned before the regular shipment BD zone.

$J_{NML} \bigcup J_{NRML} = J$

$$\sum_{j \in J_{NML}} \sum_{t=a_i}^{d_i} (t + h_j) \cdot x_{ij}^t \leq \sum_{j \in J_{NRML}} \sum_{t=a_i}^{d_i} (t) \cdot x_{ij}^t \tag{5}$$

The sixth constraint (6) is the capacity constraint. In the sixth constraint (6), we check that if ULD's in subset i are started to be processed at time t, no more ULDs than the remaining capacity of the BD zone j can start before $t + h_j$, because the process of breaking down the current ULDs needs to be completed. If ULDs in subset i start at t in BD zone j, then other ULDs are in $u \in I \setminus \{i\}$. In the first part of the constraint, we are calculating the remaining capacity for the BD zone at time t . For the second part of the constraint, we are stating that whatever will be assigned in next time period $t + h_j$ should be less than or equal to remaining capacity.

$$c_j - \sum_{i \in I} x_{ij}^t \geq \sum_{u \in I \setminus \{i\}} \sum_{\tau=t+1}^{t+h_j-1} x_{ij}^\tau \qquad \forall t \in T, \forall j \in J \tag{6}$$

The effect is that no more ULDs than the capacity of BD zone are assigned at any given time. The same equation can be simplieifed if we initially discretize time in a coarse manner i.e instead of discretizing time in minutes, if we discretize time in handling time intervals of the BD zone and define the decision variable only at discretized time values, then we can rewrite the equations as follows:

$$c_j - \sum_{i \in I} x_{ij}^t \geq 0 \qquad \forall t \in T, \forall j \in J \tag{7}$$

The above simplified form of equation ensures that at any given time t in the set T, which contains time values discretized according to handling time intervals, the capacity of BD zone is never violated.

ULD i is broken down at time, $t_{i,end} = \sum_{j \in J} \sum_{t=a_i}^{d_i} (t + h_j).x_{ij}^t$

Now, let's consider the journey of each shipment as in Figure 3 shown below. Each ULD i has $l \in L_i$ shipments in it. If ULD i is assigned at time t to BD zone j, then $x_{ij}^t = 1$ and it also means that all shipments in ULD i are assigned at that time t to BD zone j.

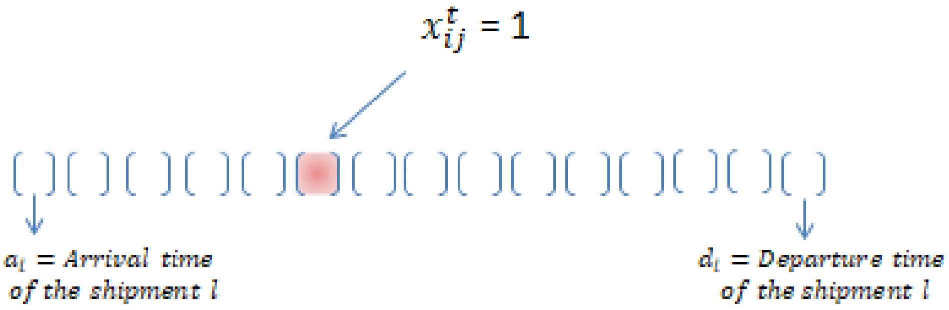
x_{ij}^t = 1
a_l = Arrival time of the shipment l
d_l = Departure time of the shipment l

Figure 3: Journey of Shipment

3.2 Build Up Process

We define parameters for flights and Build up zones :

Flight k ∈ K = {1,2,3,...,K}

BU zone for flight k has Workstations m ∈ M_k = {1,2..,M}

Capacity of ULD in Kgs (W)= 400

Weight of each shipment= w_l

Time taken to build up a ULD in BU zone for Flight k= b_k

Pre-processing time for each Flight k= p_k

3.2.1 Decision Variables for Build Up Process

In the Build Up zone model, the decision needs to be taken regarding which shipments should be assigned to which workstation at what time in a given Build Up zone. For this reason, we introduce a decision variable: z_{lm}^{t}

Shipment l having departure flight k is assigned to one of "m" workstations in its BU Zone.

$$z_{lm}^{t} \in \{0,1\}, \forall i \in I, l \in L_i, m \in M_k, t \in T$$

The second decision we need to take is regarding the number of workstations that need to be allocated to a flight to start its ULD build up process. For this purpose we introduce the second decision variable in BU zone as follows:

Q_{mk}^{t} is our binary decision variable to decide if the the Workstation m is assigned to build ULD for Flight k at time t or not.

$$Q_{mk}^{t} \in \{0,1\}, \forall k \in K, m \in M_k, t \in T$$

3.2.2 Constraints for Build Up Process

All shipments must be assigned to a workstation in its designated Build up zone.

$$\sum_{m \in M} \sum_{t=a_l}^{d_l} z_{lm}^{t} = 1 \qquad \forall k \in K, \forall l \in L_k \tag{8}$$

If shipments in subset $l \in\ L_k$ start at t in m^{th} workstation, other shipments in subset $p \in\ L_k \setminus \{l\}$ cannot start before t + b_k

$$Q_{mk}^{t} \cdot W - \sum_{l \in L_k} z_{lm}^{t} \cdot w_l \geq \sum_{p \in L_k \setminus \{l\}} \sum_{\tau=t+1}^{t+b_k-1} z_{pm}^{\tau} \cdot w_l, \qquad \forall k \in K, \forall m \in M_k, \forall t \in T \tag{9}$$

The above simplified form of equation ensures that at any given time t in the set T, which contains time values discretized according to handling time intervals, the capacity of BU zone is never violated.

The number of workstation in the Build up zone that are to be assigned to build ULDs for flight k shall not exceed the number of ULDs that go into the flight and hence we need the following constraint to take care of this.

$$\sum_{m\in M_k}\sum_{t=a_l}^{d_k} Q^t_{mk} \le \frac{\sum_{l\in L_k}\sum_{m\in M}\sum_{t=a_l}^{d_k} z^t_{lm} w_l}{W} + 1, \qquad \forall k \in K \tag{10}$$

If a workstation 'm' in a buildup zone is assigned to flight k at time t, then no other flight can start building its ULDs in that workstation before time t+bk.

$$1 - Q^t_{mk} \ge \sum_{q\in K\setminus\{k\}}\sum_{\tau=t}^{t+b_k} Q^t_{mq}, \qquad \forall k \in K, \forall m \in M_k, \forall t \in T \tag{11}$$

The above equation can also be simplified as follows, when we consider a coarse time discretization as discussed earlier.

$$1 - Q^t_{mk} \ge Q^t_{mq}, \qquad \forall k \in K, \forall q \in K \setminus \{k\}, \forall m \in M_k, \forall t \in T \tag{12}$$

In the entire timeline of a shipment from its arrival to departure as in Figure 4, we have a precedence contraint on processes a shipment goes through. The Break down process preceeds the Build up process and hence The assigment for BD zone should happen before Workstation assignment in a BU Zone.

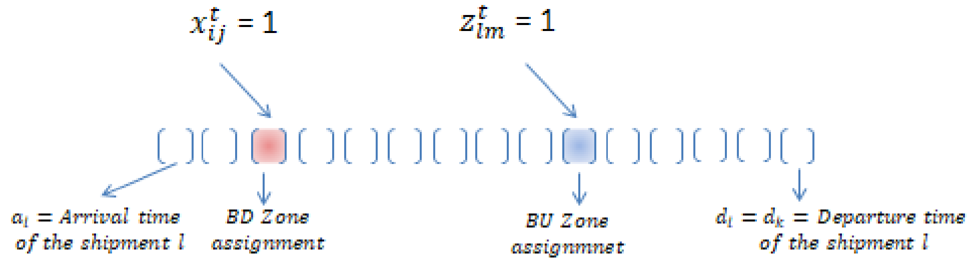
$x^t_{ij} = 1$
$z^t_{lm} = 1$
a_l = Arrival time of the shipment l
BD Zone assignment
BU Zone assignmnet
$d_l = d_k$ = Departure time of the shipment l

Figure 4: Journey of Shipment - 2

The end time of ULD i in Break down process and the summation of transportation time needed to reach Warehouse shall be less that the time at which the shipment l is assigned to a workstation 'm' in Build up zone of flight k.

$$\sum_{j\in J}\sum_{t=a_l}^{d_i} (t + h_j + T_j) \cdot x^t_{ij} \le \sum_{m\in M_k}\sum_{t=a_l}^{d_l} z^t_{lm_k} \cdot t \qquad \forall i \in I, \forall l \in L_i \tag{13}$$

For Multiple DZ ULDs we have to consider only NRML BD zone assignments for writing precedence constraint :

$$\sum_{j \in J_{NRML}} \sum_{t=a_l}^{d_i} (t + h_j + T_j) \cdot x_{ij}^t \leq \sum_{m \in M_k} \sum_{t=a_l}^{d_l} z_{lm_k}^t \cdot t \qquad \forall i \in I, \forall l \in L_i \tag{14}$$

Slack for each shipment is given by difference in departure time and the time at which shipments are built up into ULDs. Let this slack time for each shipment be greater than s_{min}

$$(d_l + p_k) - (\sum_{t=a_l}^{d_l} z_{lm_k}^t \cdot (t + b_k)) \geq s_{min} \qquad \forall i \in I, \forall l \in L_i, \forall k \in K, \forall t \in T \tag{15}$$

$$s_{min} \geq 0 \tag{16}$$

3.3 Objective Function

By maximizing the minimum slack of all shipments, we ensure that each shipment is ready well before its departure time.

$$\max s_{min}$$

3.4 Alternate Model for Build Up Process

In this alternate model, we define the following for the flights and the time horizons.

Flight k $\in$ K = {1,2,3,...,K}

Time horizon t $\in$ T = {1,2,3,...,T}

The first constraint is about available weight of shipment of flight k at any time t , which means the summation of the weights of the seperate shipments which are waiting in the warehouse and ready to be built up.

$$avail_{kt} \in \mathbb{R}$$

The second one is the due time of the build up process for a flight k.

$$due_k \in \mathbb{R}$$

The third one is the total shipment weight as a demand for a flight k.

$dem_k \in T$

The fourth one is the maximum number of ULDs building up simultaneously for a flight k. It actually means number of workstations which are working parallel. In practice we only want to build a limited number of ULDs for the same flight 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 maximum split denote the maximum desired number of parallel build-ups of a flight. Enforcing a hard limit in practice might lead to offloads for flights where the shipment 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 shipments can be packed. For each period t, we determine the shipment weight arriving in the future (weight demand of flight k) − (available weight at time t for flight k) and the weight that could be packed in the remaining time, if no parallel build-ups are allowed. Note that the number of ULDs that can be built sequentially in the time between t and due time for flight k, can be determined as (due date of flight k − t − 1) / (duration to build a ULD for a flight k). 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 maximum split, we allow maximum split to be built in parallel.

$split_k \in \mathbb{N}$

The fifth parameter is the number of ULDs scheduled for a flight k. It means that total shipment weight for a flight k is divided into capacity which is 400 kg and gives us the value of the parameter.

$uld_k \in \mathbb{N}$

The sixth one is the duration in minutes for building a ULD for flight k. We actually know which BU zone is used for flight k before, that is why we do not consider here BU zone index.

$dur_k \in \mathbb{N}$

The seventh one is the weight of an ULD which is 400 kg as stated in the data.

$cap \in \mathbb{R}$

The last one is the number of workstations handling the ULDs during time t. It is for all flights in time horizon t.
$open_t \in \mathbb{N}$

3.4.1 Decision Variables for Build Up Process

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 shipment is available. We capture the effect of each reason in one variable.

$start_{kt} \in \mathbb{N}, \forall k \in K, t \in T$: Number of ULDs started to build at time t for a flight k $t \leq due_k$ - dur_k

$offload_k \in \mathbb{N}, \forall k \in K$: Number of NOT scheduled ULDs for flight k

$dead_{kt} \in \mathbb{R} \geq 0, \forall k \in K, t \in T$: Lost weight of shipments on flight k if ULDs are built before enough shipments available at time t

3.4.2 Constraints for Build Up Process

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 $-$ dur_k $+$ 1) to period t are still work in progress. For each flight k we limit the number of simultaneously built ULDs by splitk as:

$$\sum_{t' \in T} start_{kt'} \leq split_k \qquad \forall k \in K, \forall t \in T \tag{17}$$

$t - dur_k \leq t' \leq t$

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 during period t by $open_t$. To calculate this number for a point in time, we sum up the ULDs of all flights that have been started during earlier periods and are not finished yet:

$$\sum_{k \in K} \sum_{t' \in T} start_{kt'} \leq open_t \qquad \forall t \in T \tag{18}$$

$t - dur_k \leq t' \leq t$

3.5 Objective Function for Alternate model

While satisfying all the given constraints, we want to load as much cargo as possible. o maximize the load, we minimize the amount of weight the departing flight lost and the number of ULDs building up unassociated to the departing flight.

$$\min \qquad (\sum_{k \in K} \sum_{t \in T} dead_{kt} + \sum_{k \in K} offload_k)$$

4 Solution and Discussion

We implemented all above presented variables and constraints in python using Jupyter notebook. We checked the notebook in both Linux and Windows 64-bit operating systems and it compile without any errors in both. As a MIP solver we used Gurobi in version 8.1.0. For data cleaning Pandas(0.24.0) and Numpy(1.15.4) was used. All the graphs generated using matplotlib(3.0.2).

We perform our testing in two machines. One machine had Intel Core i7-4500U processor with 2 cores clocked at 1.8 GHz and equipped with 4 MB cache running on a 64-bit Ubuntu 18.04 lts. Another machine had Intel Core i7-6500U processor with 2 cores clocked at 2.5 GHz and equipped with 4 MB cache running on a 64-bit Windows 10. Each machine had 8 GB main memory and all the experiments were performed under full load.

First model takes three hours to build the entire model and for the second model takes around two hours to build entire model, probably due to a very fine time discretization that we used (We can improve the time by using a coarse discretization of time). In the solution, we see that all shipments catch their flights on time as was our goal in this project. To choose a planning time horizon that fits our needs, we evaluated different planning time horizons with repect to the solution quality and the runtime. For the BD zones, first we determined the planning horizon as one minute but it took a lot of time for the solution. Then we decided to discretize the time using handling time intervals for each BD zone. For the BU zones, time is discretized in one minute intervals.

4.1 Break down zone results

The results for the BD zones are visualized in both Figure 5 and Figure 6. Most number of ULDs allocated to BD NRML-1 because transportation time and the handling time for this particular BD zone are less than the others. Similarly this is applied to the ULD which is containing animals. For this ULDs BD NML-3 was selected most. Hence, our model picked the best BD zone. As per the data we received it used only three normal break down zones and three animal break down zones. Other break down zones are not picked by our model for this data.

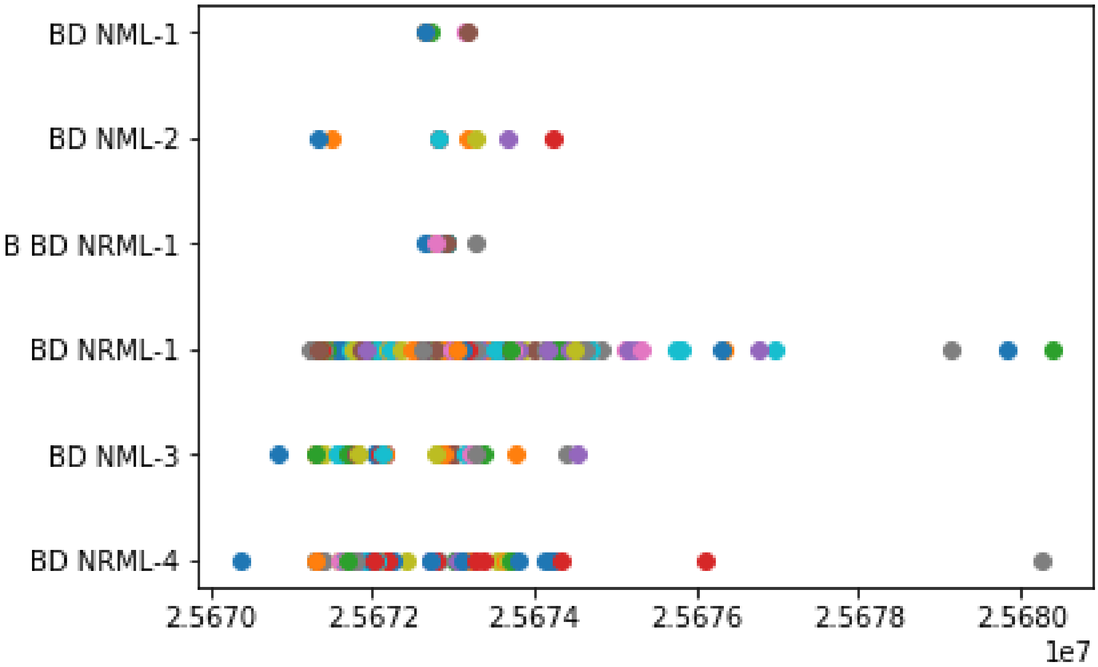

Figure 5: Shipment Arrival Times (X-axis) and BD Zones (Y-axis)

4.2 Build-up zone results - main model

The results for the BD zones are visualized in both Figure 5 and Figure 6. Most number of ULDs allocated to BD NRML-1 because transportation time and the handling time for this particular BD zone are less than the others. Similarly this is applied to the ULD which is containing animals. For this ULDs BD NML-3 was selected most. Hence, our model picked the best BD zone. As per the data we received it used only three normal break down zones and three animal break down zones. Other break down zones are not picked by our model for this data.

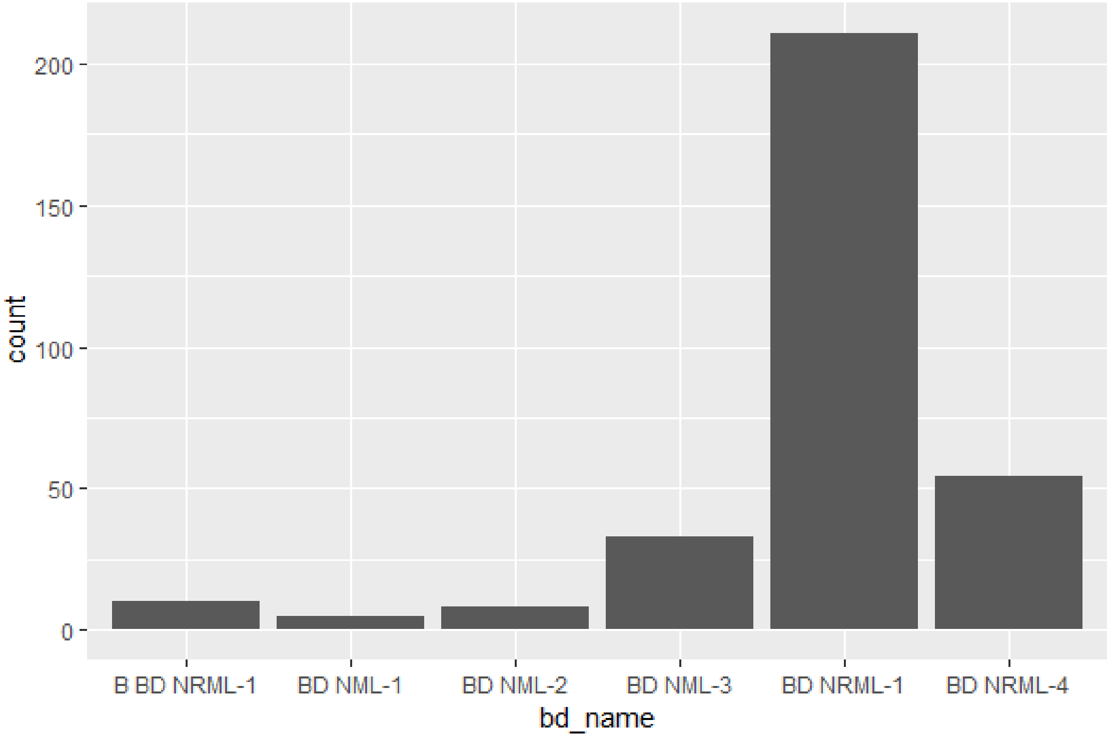

Figure 6: BD zones (X-axis) and ULD count (Y-axis)

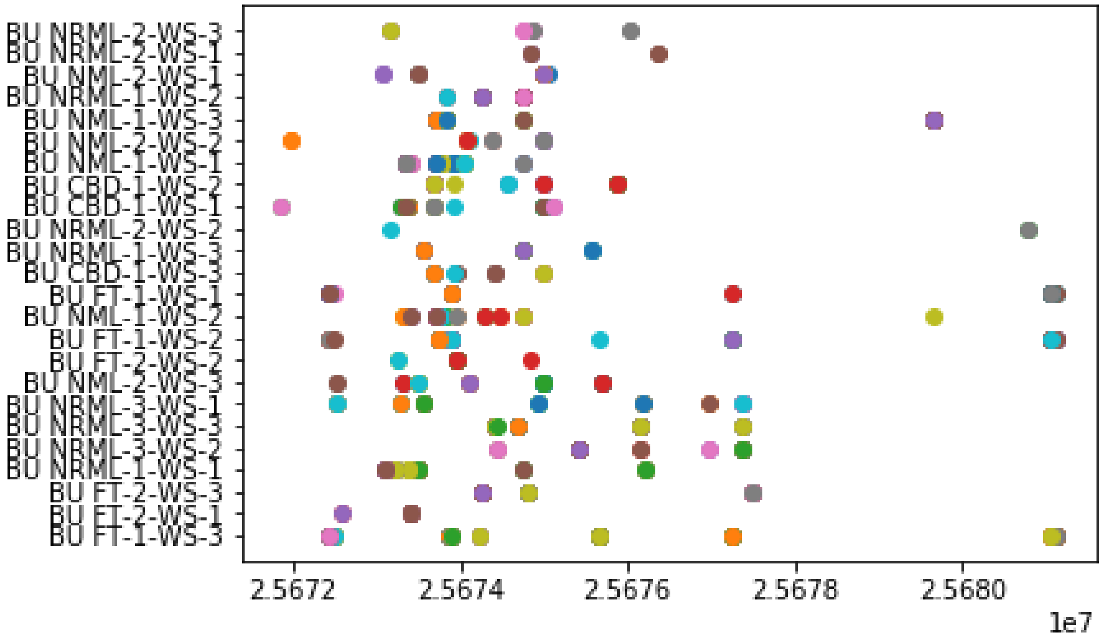

Figure 7: Shipment Arrival Times (X-axis) and BD Zones (Y-axis)

4.3 Build-up zone results - alternative model

The results for the BD zones are visualized in both Figure 5 and Figure 6. Most number of ULDs allocated to BD NRML-1 because transportation time and the handling time for this particular BD zone are less than the others. Similarly this is applied to the ULD which is containing animals. For this ULDs BD NML-3 was selected most. Hence, our model picked the best BD zone. As per the data we received it used only three normal break down zones and three animal break down zones. Other

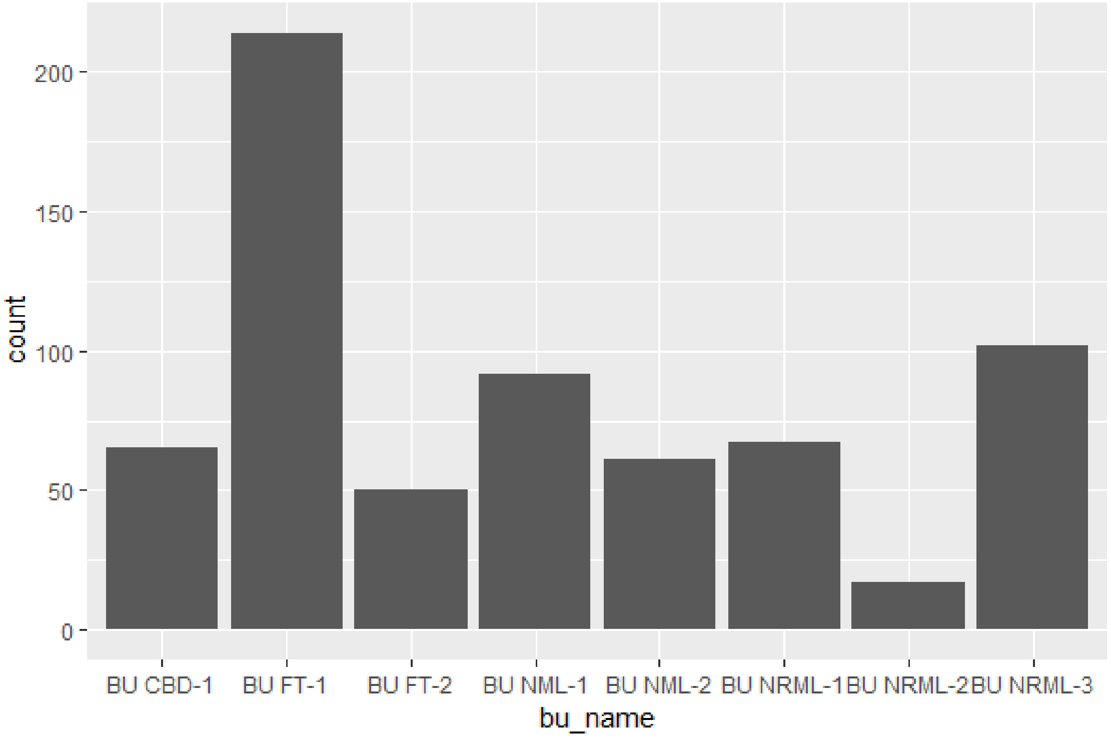

Figure 8: BD zones (X-axis) and ULD count (Y-axis)

break down zones are not picked by our model for this data.

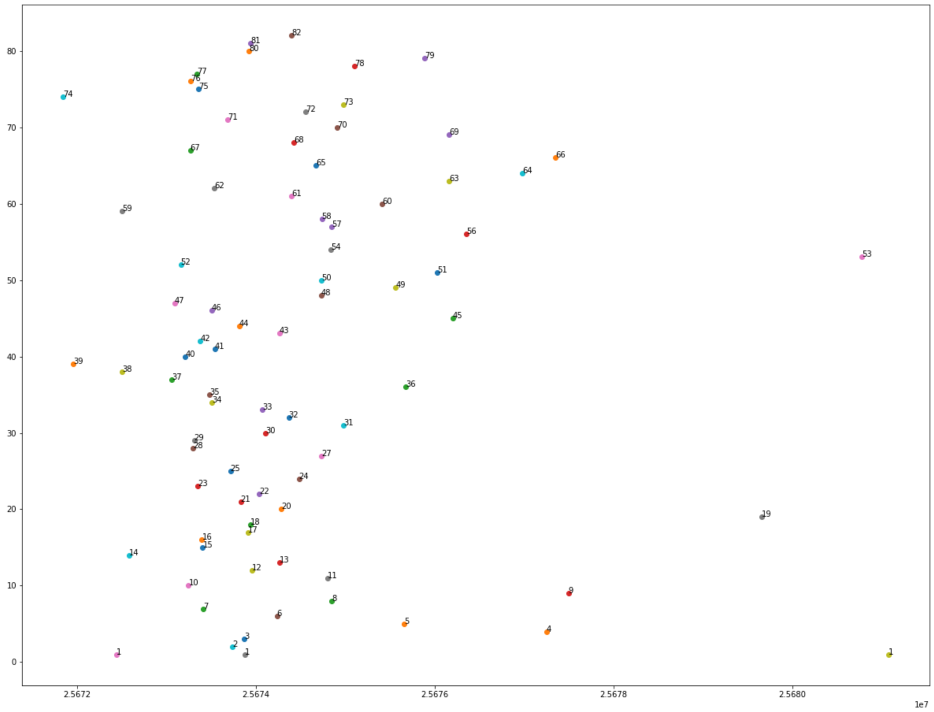

Figure 9: Shipment Arrival Times (X-axis) and BD Zones (Y-axis)