UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Mathématique et Informatique**

Arrêté ministériel : 7 août 2006

Présentée par

Christophe-Marie Duquesne

Thèse dirigée par **Denis Naddef**
et codirigée par **Olivier Briant**

préparée au sein **du laboratoire G-SCOP**
et de **l'école doctorale MSTII**

Intégration du déploiement de flotte et du service aux passagers dans la gestion de la planification pour compagnie aérienne

Integration of fleet deployment and of passenger service in airline schedule management.

Thèse soutenue publiquement le **14 janvier 2013**,
devant le jury composé de :

Nadia Brauner Vettier
Professeur, Université Joseph Fourier, Présidente
Cynthia Barnhart
Professeur, Massachusetts Institute of Technology, Rapporteur
Amy Cohn
Professeur, University of Michigan, Rapporteur
Diego Klabjan
Professeur, Northwestern University, Rapporteur
Sémi Gabteni
Docteur, Manager Recherche Opérationnelle, Amadeus, Examinateur
Ekbel Bouzgarrou
Docteur, Directeur de Projet, Air France KLM, Examinateur
Denis Naddef
Professeur Émérite, Institut Polytechnique de Grenoble, Directeur de thèse
Olivier Briant
Maitre de conférence, Institut Polytechnique de Grenoble, Co-Directeur de thèse

Abstract

Given an airline schedule and demand forecasts, the Fleet Assignment Problem consists in determining how to assign aircraft types to flight legs in the best possible way. Decades of research on this problem have improved the formulations to be more and more realistic. The goal of thesis is to enhance the existing fleet assignment models using improved revenue evaluation techniques, in order to identify more accurately fleet assignments that can lead to better airline profitability.

We first propose a study involving the two models of the literature that are the most widely used by the industry, FAM and IFAM. We show that FAM can be seen as a Lagrangian Relaxation of IFAM, with particular Lagrangian multipliers. We implement this relaxation, and we apply known results to extend it in a column generation based on a Dantzig-Wolfe decomposition of IFAM.

We then present a new approach for modeling the Fleet Assignment Problem, called Market Driven Fleet Assignment Model (MDFAM). In this model, we consider the itinerary demands as decision variables, and we propose to constrain these demands rather than considering them as a fixed input of the problem. We call the resulting constraints Market Constraints. We illustrate the flexibility of this approach through various examples, and we provide a series of experiments in order to determine which types of Market Constraints give the best results. We compare the different models, and we show that the Market Constraints of MDFAM can be formulated in such a way that the profit of the model is similar to IFAM's, while allowing better expressiveness.

Acknowledgements

This section is dedicated to the persons who supported me in the process of writing this thesis.

Foremost, my gratitude goes to the persons who provided with academic guidance: my advisors Olivier Briant and Denis Naddef. I am extremely thankful to Olivier for his availability, his amazing advices, and his support. I could not have imagined a better advisor. I am equally indebted to Denis, who was absolutely great in mentoring the organisation of this work, helping us to focus on the important tasks and proof-reading several times every single chapter of this thesis.

The subject of this thesis originates from Amadeus, our industry partner. Amadeus provided funding, data, scientific support, and their collaboration was everything any academic could wish for. In particular, I would like to thank Semi Gabteni, who supervised this work from the beginning, and who made sure this project was provided with all the resources it needed. The whole optimisation team from Amadeus should be also thanked: Rodrigo Acuna Agost and Mourad Boudia were deeply involved in designing the experiment and proof-reading the thesis, and Baptiste Chatrain provided an amazing help when extracting data for building the instances. More generally a handful of people from other teams provided their help: Credits should be given to Denis Arnaud, who offered enlightening criticisms and expert opinions in the domain of Revenue Management, but also to Emmanuel Bastien, whose technical advices enabled me to learn a lot about programming, and to Daniel Perez, who shared with me some excellent programming advices. At last, I would like to thank the whole DEV-ORI team for their excellent working ambiance.

This thesis was written at G-SCOP laboratory, which both provided an excellent scientific environment, and the opportunity to gather the opinion from talented researchers. I must thank Jean-Philippe Gayon for his kind advices in the domain of statistics. And I warmly thank my wonderful officemates Samuel Vercraene, Belgacem Bettayeb and Laura Picheral, who made the laboratory such an enjoyable place. More generally, I am grateful to all the colleagues of G-SCOP for the great atmosphere they contributed to create in this place.

At last, I would like to thank the persons who provided me with moral

Contents

List of Figures

Chapter 1

Introduction and Concepts

1.1 Airline Planning Process

Airline Scheduling consists in deciding where to fly, when, and how to assign aircraft and crew. This process is traditionally divided in 5 sequential steps:

Schedule Design decides the flights to be operated. This includes origin, destination, flight times and frequencies.

Fleet Assignment decides, for each flight, what type of aircraft should operate.

Aircraft Routing decides formally what actual aircraft should operate each flight.

Crew Pairing assigns unnamed crew members to the flights.

Crew Rostering specifies who should actually be on board for each flight.

Fleet Assignment, as the second step, has a significant impact on the subsequent steps and on the overall process. The reasons of this division of the Airline Planning Process include, but are not limited to:

- Solving all of these problems simultaneously would require intractable computations.

- Even if the computation time needed for simultaneously solving these problems could be reduced, doing such a thing would still not be desirable, because the slightest change in the problem data (for example, staff members modifying the date of their holidays) would then induce the need to re-solve the whole process again. Splitting it in several layers allows to only re-solve the appropriate problems, and reduce the computation time when the schedule needs to be modified.

- Using a standard process usually improve costs. For example, in the case of airlines, it enables them to use standardized tools, such as optimization softwares. Moreover, it also enables them to share a common base of knowledge with other airlines, and to better profit from research results in the domain.

There is obviously some feedback between every steps, and airlines will often go back and forth between the problems until they find a satisfying solution.

After the schedule has been established, the tickets sales start. Two concurrent stages are involved: The first one, **Pricing**, consists in establishing the prices of all offered journeys at the various restriction levels, which are referred to as **Fare classes**. The second one, **Revenue Management**, dynamically optimizes the availability for each journey-fare class combination primarily based on actual bookings.

Of course, at any step, the airline can reconsider its decisions. For example, if the number of passengers does not meet the expectations on a given flight, the assigned aircraft can be swapped with a smaller one. Such a substitution is of course delicate to operate, and would require a number of properties to be satisfied. For example, it has to be checked that the crew assigned to the flight will be able to operate the new aircraft. The old aircraft will also need to be reassigned elsewhere, and should be large enough to provide seats for all the bookings already performed by the passengers. The aircraft routing has to be repaired. This kind of changes in the schedule can thus be risky, for possibly unplanned consequences they might induce. However, nowadays,flexibility is something airlines are after, because it is key for reacting to competition, and thus tools must be conceived in a way that allows this flexibility.

1.2 Fleet Assignment

1.2.1 Concepts

The Fleet Assignment Problem consists in assigning aircraft types to flight legs in the best possible way. A flight leg is a journey consisting of one take off and one landing. It is defined by its board point, off point, departure time and arrival time. Legs constitute the smallest elementary schedule operation, and the set of all the flight legs defines the Flight Schedule. An aircraft type is defined by the name of the family it represents, such as A320 or B747.

Most of the time, the schedule is assumed to cycle daily or weekly, and the fleet assignment is built in a way it can be repeated. However, it is also possible to consider non-cyclic schedules. In such situations, initial and terminal positions may be imposed for aircraft types: in each airport, at the beginning and at the end of the schedule, the number of aircraft of

each type is then specified as an input of the problem. A fleet assignment is characterized by its complete list of assignments (leg l, aircraft type t), meaning that the type t should be assigned to the leg l.

Formally, the objective is to find the best feasible fleet assignment, maximizing the profit:

$$\text{profit} = \text{revenue} - \text{operating cost}$$

The revenue depends on the demands of the passengers, on the fares applied by the airline, on the capacity of the assigned aircraft and on the revenue management policy applied by the airline. The operating cost depends on the taxes applied by the airports and on the fuel costs. The operating cost can safely be approximated to a linear function that exclusively depends on the fleet assignment. However, the same statement does not hold for the revenue. Because of the dynamic nature of the sales process, it is very difficult, if not impossible, to model it accurately at the planning stage: as passengers book their flights, the airline is modifying the availability of seats in order to preserve room for high-fare passengers. Sophisticated, time-dependent systems that are involved in order to react to the bookings and modify this availability. Any revenue forecast at the stage of the Fleet Assignment is thus a rough evaluation and does not accurately predict the revenue of the company.

To be feasible, a fleet assignment must respect some constraints: each leg must be served, the aircraft flow must be conserved, and the assignment must respect aircraft types availability. Aircraft flow conservation means that to be used on a flight leg, an aircraft must be previously available at the board point of the leg. Aircraft type availability means that at any time of the schedule, the number of aircraft of a given type used by the assignment must not exceed the amount of aircraft of this type owned by the airline.

Passengers, when they are considered by the Fleet Assignment model, are assumed to follow itineraries. Itineraries are sequences of (leg, cabin) that do not overlap in time, meaning that the passenger is going to fly each leg in the associated cabin. A cabin is a part of the plane used to host the seats of the fare class the passenger has booked for. Each cabin has a given seating capacity which depends on the aircraft type flying the leg. Some companies do not split their aircraft in cabins: it can then be assumed that the whole aircraft consists in one single cabin.

The action of the passengers to reach their next flight between each leg is called connection. Whenever the airline does not satisfy the demand of the passengers to fly on their preferred itinerary, it is often said that these passengers are spilled by the airline. This results in a loss of revenue also qualified as spilled revenue.

Some models also have a concept of recapture: passengers who are not able to buy their itinerary of primary choice might still want to travel on

4

the airline network: this phenomenon, when it happens, is called recapture. We respectively talk about recaptured passengers and recaptured revenue when referring to the passengers traveling on an itinerary which is not their primary choice, and to the revenue generated by these travels.

1.2.2 Instances Size

Instances for major North American airlines are reported to be typically built out of 24-hour schedules and usually have about 2000 legs, 10 fleets serving 75000 itineraries Barnhart et al. (2009). Depending on the size of the airline, these values can vary a lot. Implementations built for this work were tested on network from 3 different companies (one of the sets is public and can be found online [1]). Two of these companies yield small daily networks of typically 200 to 300 legs, and some of the public instances include networks of over 2000 legs.

1.3 Outline of this Thesis

Through the years, research on fleet assignment has brought more and more accuracy in Fleet Assignment models. With the growing interest for Revenue Management, the trend is to focus on better modeling the passengers behavior. Nevertheless, results show that the main problems in current Fleet Assignment models are related to the accuracy of demand forecasts. Because the Fleet Assignment occurs early in the Airline Planning Process - before the sales have started - these forecasts, even when very well performed, come necessarily with a high volatility. Besides, the amount of parameters needed by the state of the art models demands a tremendous effort to the airlines, with most parameters values being unknown or hard to estimate.

1.3.1 Goal

The general goal of this thesis is to improve the existing models and techniques for doing Fleet Assignment. To do so, several directions of development are possible. One of them could be to address the high variability of some input parameters: because Fleet Assignment is typically solved one year prior to the operations, the uncertainty of the demand forecast used to evaluate the revenue can be important. Finding techniques to address this uncertainty would thus help to produce more reliable fleet assignments. In this context, on could try to take advantage of various sources of data available to airlines when it comes to demand forecast.

---

[1]see https://github.com/chmduquesne/fleetassignment_instances

1.3.2 Contributions

This thesis features two contributions:

In a first part, we propose a study of the two fleet assignment models that are the most widely used in the airline industry: FAM and IFAM. We show that FAM can be seen as a Lagrangian Relaxation of IFAM. We implement this Lagrangian Relaxation, and then use known results to turn it into a column generation based on a Dantzig-Wolfe formulation of IFAM. We include graphical representations of how our implementation runs over small instances, and suggestions for improvements.

The second contribution of this thesis is a new Fleet Assignment model called Market Driven Fleet Assignment Model (MDFAM). In MDFAM, the itinerary demands, which would be a static input in IFAM, are made decision variables. These decision variables are subject to linear constraints, named Market Constraints. The Market Constraints are a form of knowledge the airline may have over the demand, that is more general than static inputs: since one is also able to consider groups of demands, but also to enforce correlations between demands components. We illustrate the flexibility of this approach through various examples, and we provide a series of experiments in order to determine which Market Constraints are the most efficient. We compare the different models, showing that MDFAM can reach a performance which is similar to IFAM's, while allowing better expressiveness.

Chapter 2

Literature Review

We previously presented the Airline Planning Process: for a more detailed overview about how the whole airline schedule is established, the reader is referred to Gopalan and Talluri (1998) and Barnhart et al. (2003). The following chapter focuses on the evolution of the formulations and of the optimization techniques for the Fleet Assignment Problem.

2.1   General Considerations

2.1.1   From FAM to ODFAM

The first Linear Programming model for Fleet Assignment is called FAM (Fleet Assignment Model) and can be found in Abara (1989). The decision to assign or not an aircraft type to a leg is modeled by a binary variable, the assignment costs are leg-based and depend on the aircraft type. American Airlines successfully uses this model in several of their departments to support Schedule Planning. At the time this thesis is being written, FAM remains a standard for the airline industry.

The complexity of FAM is studied by Gu et al. (1994). It is shown that FAM is NP-Complete when there are more than 3 aircraft types. The complexity for 2 aircraft types remains unknown.

Part of the assignment costs of FAM is computed from demand forecasts, and thus suffers from the volatility of these forecasts. To circumvent this flaw, Berge and Hopperstad (1993) present a method called Demand Driven Dispatch, also known as D^3. Its purpose is to dynamically adapt a fleet assignment as the demand forecasts become more and more accurate, adapting the FAM leg costs. Using actual airline data, the authors show that their method can improve the operating profits by 1-5%. This method is known to have had a significant impact on airline practice and is still the subject of active research: a comprehensive review process can be found in Shebalov (2009).

A first attempt to solve large scale Fleet Assignment instances is exposed by Hane et al. (1995). Various optimization techniques are applied, some of which are generic and now part of modern Linear Programming solvers. They reach CPU times that greatly outperform LP-Solvers with the default options, yielding solutions with an optimality gap of 0.02%.

As progress is being made in Fleet Assignment optimization techniques, more and more attention is dedicated to modelling the behavior of the passengers more accurately. Belobaba and Farkas (1996) give a first insight, performing an extensive study of the influence of yield management on fleet assignment decisions. They show how traditional ways of dealing with fleet assignment and revenue management can lead to overestimates of both expected revenues and spill costs. More realistic assumptions such as stochastic demands are considered, and an iterative algorithm based on Monte Carlo simulation is proposed. However, they do not manage to propose a tractable solution method for FAM.

In order to provide more opportunities for passenger connections, Rexing et al. (2000) design a model where the departure times are not fixed inputs of the problem, but are provided as time windows. An approach to solve the model is given, and it is shown that the schedules produced have significantly lower costs. This model is also used to tighten the schedules, possibly saving aircraft.

Ahuja et al. (2001, 2003) consider the problem of adding Through Connections constraints in the fleet assignment model. These constraints force the same aircraft to be assigned to two consecutive legs. The objective here is to ensure that these legs will always be connected, for the passengers not to miss their connection. Their model achieves the maximum combined benefit of assigning fleet along with through connections. Neighborhood search algorithms are used to solve the formulation.

A demonstration of the issues of FAM regarding spill modelling is given by Kniker and Banhart (1998). A new spill model, the Passenger Mix Model (PMM), is introduced to describe passenger behavior on a fleeted network: spill variables are introduced for every itinerary, with the objective to minimize the spill cost while maintaining a feasible passenger flow. The authors show that the spill estimation embedded in FAM leg costs is flawed, leading to bad decisions.

This work is followed by a clear breakthrough: Barnhart et al. (2002) present the Itinerary-Based Fleet Assignment Model (IFAM). The authors explain how, because FAM does not account for the passenger itineraries, it can lead to suboptimal assignment decisions. As a solution, the alternative they propose is to embed a Passenger Mix Model in a FAM formulation. Assuming the Passenger Mix Model accurately describes the passengers' behavior, IFAM yields the optimal decision. An extensive study of the model is made, and optimization techniques are given. This work receives community approval and is quoted in more than 100 articles.

A similar approach, named ODFAM, is presented in Smith and Johnson (2006). They incorporate airline revenue management effects into the fleet assignment model, proposing a technique called station purity: in order to reduce aircraft dispersion in the network, the number of aircraft that can fly a given leg is limited. The model is solved using a combination of column and cut generation.

2.1.2 Integrated Models

Though Fleet Assignment can be studied alone, it can also be seen as a way to solve other problems. Several models integrate Fleet Assignment Models into Larger schemes, in order to solve different classes of problems:

Clarke et al. (1996) show how to build fleet assignments in order to provide opportunities for maintenance and crew scheduling, through additional constraints in FAM.

Barnhart et al. (1998a) provide an integrated model to simultaneously solve the Fleet Assignment Problem and the Aircraft Routing problem. Various constraints are added to enforce the formulation, such as Maintenance requirement or equal aircraft utilization.

The problem of integrating Schedule Design and Fleet Assignment is discussed by Lohatepanont and Barnhart (2004), where algorithms that simultaneously choose the flight legs constituting the schedule and the fleet assignment are presented. Legs that are candidates for deletion or addition are regrouped in a list of optional flights. Two models are developed. ISD-FAM, applies demand correction terms for including or not a leg in the network, and ASD-FAM, a lighter model, which bases its decisions on recapture rates. The solutions are compared using the proprietary tool Sabre Airline Profitability Model, and it is shown that both models bring improvement, ASD-FAM being the most efficient.

Rosenberger et al. (2004) tackle the problem of generating a robust schedule with regards to disruptions. A common method for an airline to deal with disruption is to cancel a cycle. A cycle is a sequence of flights to be performed by the same aircraft that begins and ends at the same airport. The shorter this cycle, the least the number of legs to be impacted if it is cancelled. The authors expose a method that generates fleet assignments featuring more short cycles. Simulation shows that their solution performs better in operations than the traditional ones.

2.1.3 Recent Advances

All recent advances in fleet assignment focus on better modelling the passenger behavior.

Dumas and Soumis (2008) provide a numerical model for estimating the passenger flow on an airline network, given forecast concerning the distribu-

tion of the demand for each itinerary, the time distribution of the booking requests and the proportion of spill attracted from one itinerary to an other. The model also covers airline inventory management and how that affects passenger flow. It is compared to a simulation and results differ from the expected passenger flow by about 0.1% for load factors below 80%. The results of this work are then used by Dumas et al. (2009) to produce a fleet assignment model where the revenue function is altered according to the planned passenger flow. The process iterates by alternating between fleet assignment and passenger flow, improving the objective function at each step. The model is tested on a large-scale network from Air Canada and converges in a few iterations.

Jiang and Barnhart (2009) present an integrated model similar to D^3: using the improvements on demand forecasts that happen when the day of operations approaches, various part of the schedule are re-optimized in order to leverage this information. This method is called dynamic scheduling and it integrates flight re-timing and re-fleeting. Based on data from a major US airline, the authors demonstrate that their method can bring significant profitability improvements. This work is reused by Jiang and Barnhart (2013) where de-banked hub-and-spoke operations are focused. The authors propose a robust schedule design model, where the number of potentially connecting itineraries weighted by their respective revenues is maximized and a decomposition-based solution approach involving a variable reduction technique and a variant of column generation is developed.

Jacobs et al. (2008) propose a new methodology for incorporating passenger flow in the fleet assignment. The problem is decomposed to isolate a non-linear revenue management problem. Linear approximations are used to incorporate the passenger revenue in the total network revenue function, and the methodology is applied to an example of 10 cities, 48 flight legs, 534 itineraries. They show that their O&D FAM can outperform traditional methods by 2.8% in profit for their test case.

Barnhart et al. (2009), present an improvement to IFAM called subnetworks fleet assignment model. The idea here is similar to Dumas et al. (2009): spill is a local phenomenon and does not spread throughout the network, thus local approximations can be used. The solution method is designed to balance revenue approximation and model tractability. Computational results suggest that the approach yields profit improvements over comparable models and that it is computationally tractable for problems of practical size.

In the following section, 3 of the contributions cited above are reviewed into more details, giving us the opportunity to introduce our own notation. This section first digs into FAM, from Abara (1989), then describes methodologies deployed by Hane et al. (1995) and Barnhart et al. (1998b) to solve it. Then the possibilities for computing better leg costs are reviewed with the study of Kniker and Banhart (1998), showing how the kind of evaluation

used by FAM can lead to bad spill cost modeling. Eventually the Itinerary-Based Fleet Assignment of Barnhart et al. (2002) is explored along with the solution approach.

2.2 FAM

The first LP model for Fleet Assignment can be found in Abara (1989). The model is simply named FAM, which stands for Fleet Assignment Model.

2.2.1 Preliminary Notions

Timeline Networks

The schedule is modeled as a directed graph where all the vertices represent airports at given times. Three types of arcs may exist in this network:

The Flight arcs are the arcs linking nodes from different airports at different times. They represent the flight legs on the network.

The Ground arcs are the arcs linking nodes from the same airport in increasing time order. They represent time spent in the same airport for the aircraft.

The Overnight arcs are the arcs linking the last node from a given airport to the first one, based on the assumption of a daily cyclic schedule: these arcs are here to ensure the repeatability of the schedule.

Barnhart et al. (2002) refer to this kind of network as Timeline networks. Others use the term Time-Space networks.

Differences with the Literature

The reader who is familiar with Fleet Assignment literature might notice subtle differences between what is presented here and previous assumptions. We list them here:

- The literature often studies the daily fleet assignment problem. Because the networks studied allow it, this thesis extends the concept to longer durations. Instead of representing one day of the schedule, the timeline network may thus represent a week, or a month. In this context, the term "overnight" arcs can then be misleading, so we propose to talk about **repeatability arcs**.

- Most articles assume a cyclic schedule. The work presented here does not have such requirements, and the repeatability arcs might be skipped. In our actual implementation, they are replaced with initial and terminal conditions over the number of aircraft at airports. For
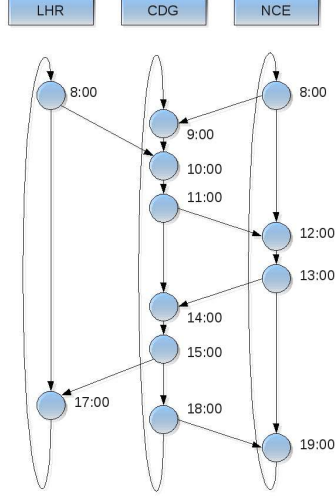
Figure 2.1: A timeline network

example, in networks where, at the same airport, the offset between inbound and outbound legs is not null, these constraints are for the number of aircraft of any given type not to exceed this offset at the end of the period. Though this implementation aspect is worth mentioning, we don't include it in the actual description of the models for the sake of simplicity.

• In the original FAM, the timeline network uses distinct flight arcs - one for each aircraft type - to represent the same flight leg. In such representations, flight arcs exist with different arrival times for each family of aircraft (e.g. they depend of the aircraft type and are labeled k, o, t, where k is an aircraft type, t a time and o an airport). We use another possibility, which is to model flight legs with single arcs, and we allow any type of aircraft on the arc. The arrival time we choose is the one of the slowest family. We assume that connections we want to guarantee will be explicitly enforced by setting an earlier arrival time and forbidding on the arc aircraft types that are too slow. In other terms, we assume that the list of passenger itineraries implicitly forbids some legs to be assigned slow aircraft types.

2.2.2 Sets, Data

FAM uses the following notations:

- L: the set of flight legs in the flight schedule (indexed by l).

- T: the set of aircraft types (indexed by t).

- $|t|$: the number of aircraft of type t.

- N: the set of nodes in the timeline network (indexed by n).

- G: the set of ground arcs in the timeline network (indexed by g).

- CT: an arbitrary time in the schedule range, used as a count time.

For any node n of the timeline network:

- $I(n)$: the set of inbound legs to n.

- $O(n)$: the set of outbound legs from n.

- $g_i(n)$ the inbound ground arc to n.

- $g_o(n)$ the outbound ground arc from n.

As a mathematical notation abuse, for any ground arc g, we note $CT \in g$ if the count time CT occurs during g time lapse. This allows us to define the set of ground arcs that cross the count time, as $\{g \in G | CT \in g\} = \{g \in G | g \ni CT\}$.

Additionally, for each leg of the schedule, a coefficient $c_{t,l}$ is computed, representing the cost of assigning an aircraft of type t to the leg l.

2.2.3 Model

Decision Variables

- $x_{t,l} = \begin{cases} 1 & \text{if aircraft type } t \text{ is assigned to leg } l \\ 0 & \text{otherwise.} \end{cases}$

- $y_{t,g}$: the number of aircraft of type t waiting on the ground arc g.

MIP Formulation

Here is FAM formulation from Abara (1989):

$$\min \sum_{l \in L} \sum_{t \in T} c_{t,l} x_{t,l} \tag{2.1}$$

$$s.t. \sum_{t \in T} x_{t,l} = 1 \qquad\qquad \forall l \in L \tag{2.2}$$

$$\sum_{l \in I(n)} x_{t,l} + y_{t,g_i(n)} = \sum_{l \in O(n)} x_{t,l} + y_{t,g_o(n)} \qquad \forall t \in T, \forall n \in N \tag{2.3}$$

$$\sum_{g \ni CT} y_{t,g} + \sum_{l \ni CT} x_{t,l} \leq |t| \qquad\qquad \forall t \in T \tag{2.4}$$

$$x_{t,l} \in \{0,1\}, y_{t,g} \geq 0$$

This translates to:

- (2.1) The objective is to minimize the assignment cost.

- (2.2) Exactly one aircraft type must be assigned to each leg.

- (2.3) The flow of aircraft is conserved at each node of the network.

- (2.4) At the count time, the number of aircraft of every type is conserved.

2.2.4 Solving the MIP

Hane et al. (1995) comes with a set of methods that can solve FAM faster.

Node aggregation: Because the only important information is how flight legs interconnect, the following rules can be applied in a preprocessing step:

- At the same airport, two successive arrivals are aggregated into the same node.

- At the same airport, two successive departures are aggregated into the same node.

- At the same airport, one (list of) arrival(s) immediately followed by one (list of) departure(s) are aggregated into the same node.

This process effectively reduces the number of ground arcs and makes the problem faster to solve.

Forcing variables: In order to reach feasibility faster after having solved the LP relaxation, before starting the branch and bound, some rules can be applied (with no guarantee on optimality):

- Decision variables that are greater than a certain threshold, say 0.99, may be fixed to 1

- For a given leg, if the fractional solution is shared by a subset of the aircraft types, the choice may be restricted to aircraft types within the capacity range of this subset.

Branching rules: Hane et al. (1995) propose a branching rule related to the cover constraints (2.2), also called SOS constraints. It consists in branching on half of the variables in the constraints rather than a single one: instead of constraining one single variable to be 0 or 1, the variables of the constraint are partitioned into 2 sets; in one branch, the sum of the variables of the first subset is constrained to be 0 and the sum of the variables in the other subset is constrained to be 1. In the other branch, the opposite choice is made. The branching priority is determined by choosing constraint with the largest expected impact on the objective. Such branching rules are now standard in commercial solvers.

2.3   An Attempt to improve FAM Leg Costs

Kniker and Banhart (1998) review the computation of FAM leg cost. They present a model specialized in computing the spill cost of a Fleet Assignment, and attempt to use this model to improve these costs.

2.3.1   FAM Leg Costs Computation

In FAM, $c_{t,l}$, the costs of assigning the aircraft type t to the leg l, are traditionally decomposed as:

$$c_{t,l} = \underbrace{c_{t,l}^{\text{op}}}_{\text{operating costs}} + \underbrace{c_{t,l}^{\text{sp}}}_{\text{lost revenue}}$$

- Operating costs include fuel, landing and take-off fees, and estimates of crew costs.

- Lost revenue occurs because passengers cannot be accommodated by the airline on some flights, the assigned aircraft type being too small to satisfy the demand. This loss of passenger is called **spill**.

While the operating costs only depend on t and l, the spill costs are obtained through an estimation: Let Q_l be the amount of passengers who

want to travel through the flight leg l regardless of the fleet assignment, and $\text{Fare}_l^{\text{sp}}$ be the average fare associated with spilled passengers on the leg l:

$$c_{t,l}^{\text{sp}} = \underbrace{\text{Fare}_l^{\text{sp}}}_{\text{spill fare}} \cdot \underbrace{\max(0, Q_l - \text{Cap}_t)}_{\text{number of spilled passengers over a leg}} \qquad (2.5)$$

- Q_l is referred to as the unconstrained demand on the leg l. In practice, this unconstrained demand is hard, if not impossible to know, because one only sees the actual bookings, constrained by the network capacity.

- The spill fare is typically lower than the average fare on the leg l, because the airline is assumed to perform revenue management and thus to capture passengers who pay more, or alternatively, to spill low fare customers.

The spill fare on the leg l represents the net revenue loss due to one passenger being spilled from l. It is computed as

$$\text{Fare}_l^{\text{sp}} = \kappa * \text{average\_fare}_l \qquad (2.6)$$

where $\kappa \in [0, 1]$ is a globally adjusted parameter.

Kniker and Banhart (1998) try to determine a good value for κ. To do so, a specialized method for estimating the spill cost of a given fleeted network is proposed. The corresponding problem is referred to as the **Attainable Contribution Problem**, and is described as follows:

Given a fleeted schedule and the unconstrained itinerary demands, find the carrying plus spill cost minimizing flow of passengers over the network, such that (1) the total number of passengers on each flight does not exceed the capacity of the flight, and (2) the total number of passengers on each itinerary does not exceed the unconstrained demand on that itinerary.

This problem is solved through a model called Passenger Mix Model (PMM).

2.3.2 Data

The PMM formulation has been improved since Kniker and Banhart (1998). We present here a enhanced version of Barnhart et al. (2002), that includes recapture, but also differentiates the cabins.

- L: the set of flight legs in the flight schedule (indexed by l).

- C: the set of cabins (indexed by c), e.g. $C = \{\text{economy, business, first}\}$

We expand the concept of itineraries to include cabins: an itinerary is not just a sequence of legs, it is a sequence of leg-cabins. It is assumed that, for every itinerary, an unconstrained demand and a fare are known:

- d_i: the unconstrained demand at the level of the itinerary i. This is different from the unconstrained demand on the leg l. The relationship linking Q_l to d_i is $Q_l = \sum_{c \in C} \sum_{i \ni (l,c)} d_i$, where $i \ni (l,c)$ means that the leg-cabin (l,c) is part of the itinerary i.

- Fare_i is the fare of the itinerary i.

Recapture rates are defined:

- $r_{i \to j}$: the recapture rate from the itinerary i to the itinerary j, i.e. the proportion of customers who would accept the itinerary j if they are refused the itinerary i.

Capacities have to be known:

- $Cap_{l,c}$: the capacity of the cabin c for the leg l

2.3.3   Model

Decision Variables

- $s_{i \to j}$: the number of passengers spilled from i to j by the model.

MIP Formulation

$$\min \sum_{i \in I} \sum_{j \in I} (\text{Fare}_i - r_{i \to j} \text{Fare}_j) s_{i \to j} \qquad (2.7)$$

$$s.t. \sum_{i \ni (l,c)} d_i - \sum_{i \ni (l,c)} \sum_{j \in I} s_{i \to j}$$

$$+ \sum_{i \in I} \sum_{j \ni (l,c)} r_{i \to j} s_{i \to j} \leq \text{Cap}_{l,c} \qquad \forall l \in L, \forall c \in C \qquad (2.8)$$

$$\sum_{j \in I} s_{i \to j} \leq d_i \qquad \forall i \in I \qquad (2.9)$$

$$s_{i \to j} \geq 0$$

This translates to:

- (2.7) Minimize the spill minus the recapture.

- (2.8) Passenger travel is consistent with cabin capacities.

- (2.9) No more passengers are spilled than the unconstrained demand.

Adapting the Spill Fare

With now an accurate way to compute the spill cost of a fleeted network, Kniker and Banhart (1998) attempt to produce a spill fare yielding the best possible fleet assignment.

Finding the best possible spill fare: In order to determine the best value for κ in the equation (2.6), the revenue of FAM is challenged against the Passenger Mix Model. The following procedure is used: the network is fleeted with a fleet assignment solver. This outputs an accurate computation of the operating costs. Then, using a passenger mix model, the passengers are assigned to the flight legs, computing the attainable contribution.

This process is applied on a range of values for κ, allowing to look for an optimal value with the obtained results. The following diagram gives an overview of the method used.
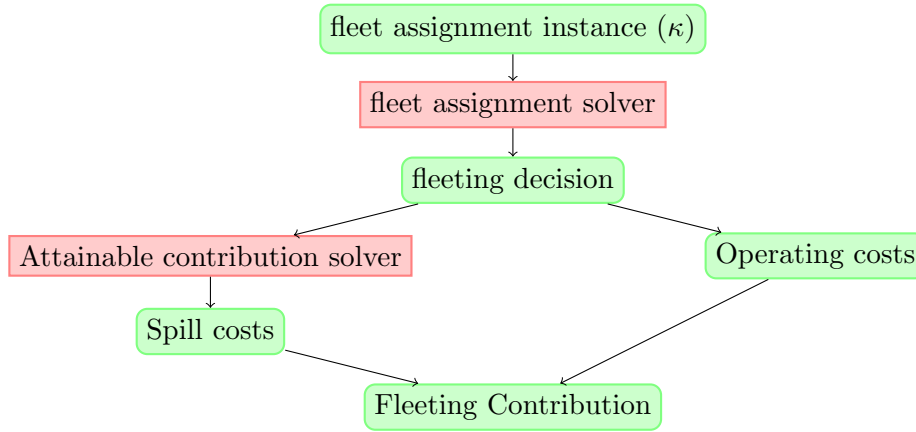
Procedure repeated on a range of different values for κ.

Figure 2.2: Procedure Kniker and Banhart (1998)

Results

The estimated spill (from the fleet assignment model) is plotted against the actual spill (from the passenger mix solver). The aim is to determine the value of κ that results in the greatest contribution of the network.

This experiment yields several outcomes:

- It is observed that the value of κ yielding the most revenue is about 0.7, suggesting that using a spill fare of $0.7.\text{Fare}_l$ to determine the FAM assignment costs would lead to the best fleet assignments.

- Plotting the estimated spill used a priori for building the fleet assignment against the actual spill computed a posteriori from this fleet assignment, it is found that a similar value $\kappa = 0.7$ better estimates the spill.

- The difference between actual and estimated spill is then plotted against the revenue of the fleet assignment. The curve is expected to show that the better the spill estimation is, the better the fleet assignment. However, it is observed that sometimes, perfect estimation of the spill does not necessarily yield best fleet assignment. This is interpreted as a lack of local accuracy: this method only allows to match the FAM spill cost with the spill cost as determined by PMM on the whole network; however, when restricting the network to the leg level, or when considering a subnetwork, these costs will most likely differ.

Conclusion

The result suggests that while FAM can provide a good approximation of the best possible Fleet Assignment, its poor modeling of the spill mechanism prevents it from computing the best possible fleet assignment.

2.4 IFAM

As FAM fails to model accurately the mechanism of spill and recapture, Barnhart et al. (2002) suggest a new model called itinerary based fleet assignment. They propose to merge the Passenger Mix Model and the Basic Fleet Assignment Model in a straightforward process: The two objectives are merged and all the constraints of the two models are put together, with cabin capacity constraints that now depend on the fleet assignment. This article is a major breakthrough in the field of fleet assignment, and obtains community approval. More than 100 articles quote it. It has become the standard model to test against.

2.4.1 Model

Sets, Decision Variables

The same notations are used as in FAM and PMM. Additionally:

- For each leg of the schedule and each aircraft type, the coefficient $c_{t,l}$ stands for the operating cost of assigning an aircraft of type t to the leg l.

- For each aircraft type and each cabin, $\text{Cap}_{t,c}$: the capacity of the cabin c for the aircraft type t is defined.

IFAM mixes decision variables from FAM and PMM:

- $x_{t,l} = \begin{cases} 1 & \text{if aircraft type } t \text{ is assigned to leg } l \\ 0 & \text{otherwise.} \end{cases}$

- $y_{t,g}$: the number of aircraft of type t waiting on the ground arc g.

- $s_{i \to j}$: the number of passengers spilled from i to j by the model.

MIP Formulation

The result of this new formulation is what follows:

$$\min \sum_{l \in L} \sum_{t \in T} c_{t,l} x_{t,l} + \sum_{i \in I} \sum_{j \in I} (\text{Fare}_i - r_{i \to j} \text{Fare}_j) s_{i \to j} \qquad (2.10)$$

$$s.t. \sum_{t \in T} x_{t,l} = 1 \qquad \qquad \forall l \in L$$
$$\qquad (2.11)$$

$$\sum_{l \in I(n)} x_{t,l} + y_{t,g_i(n)} = \sum_{l \in O(n)} x_{t,l} + y_{t,g_o(n)} \qquad \forall n \in N, \forall t \in T$$
$$\qquad (2.12)$$

$$\sum_{g \ni CT} y_{t,g} + \sum_{l \ni CT} x_{t,l} \leq |t| \qquad \qquad \forall t \in T$$
$$\qquad (2.13)$$

$$\sum_{i \ni (l,c)} d_i - \sum_{i \ni (l,c)} \sum_{j \in I} s_{i \to j}$$
$$+ \sum_{i \in I} \sum_{j \ni (l,c)} r_{i \to j} s_{i \to j} \leq \sum_{t \in T} x_{t,l} \text{Cap}_{t,c} \qquad \forall l \in L, \forall c \in C$$
$$\qquad (2.14)$$

$$\sum_{j \in I} s_{i \to j} \leq d_i \qquad \qquad \forall i \in I$$
$$\qquad (2.15)$$

$$x_{t,l} \in \{0,1\}, y_{t,g} \geq 0, s_{i \to j} \geq 0$$

2.4.2 Solving the MIP

Barnhart et al. (2002) conduct an extensive study of IFAM, and develop a general method for solving the problem.

A Column generation is applied to the spill variables to solve the LP relaxation of the problem, and a branch and bound phase is applied to get the integral solution.

The challenge is significant: the number of spill variables is theoretically the square of the number of possible itineraries over the network. However,

this number can be reduced because not all itineraries are relevant. Furthermore, it is not necessary to generate a recapture variable for a couple of itineraries if there is absolutely no chance that recapture could occur between them. The run time of IFAM is reported to be about 10 times the runtime of FAM.

2.5   Conclusion

In terms of Operations Research, the Fleet Assignment problem is more than 60 years old. FAM, one of the first historical models, offered a robust basis for efficiently building assignments. Ever since, more and more efforts have been put into adapting the Fleet Assignment in order to improve the passengers-related part of the profit. Among the diversity of works performed to improve this efficiency, some studies have been largely echoed by the industry. D^3, proposing more a framework than a model, iteratively improves a given assignment with the improved forecasts. Other solutions include to invent various ways of computing better FAM leg costs.

The work of Kniker and Banhart (1998) has provided excellent insights for understanding how to provide good leg costs for FAM, and was the introduction of the Passenger Mix Model, later embedded in FAM to become IFAM. However, we believe the approach of using the results of a PMM to improve a FAM formulation was not fully exploited in Kniker and Banhart (1998) We believe that airlines that are still using FAM would value a work that allows them to reuse their engine to produce IFAM-quality solutions. This is the theme of our next chapter.

Chapter 3

An Attempt to Build Better Fleet Assignments With Only a FAM Solver

The following chapter consists in a study of IFAM oriented towards a reuse of FAM. We start from the review of FAM leg costs performed in Kniker and Banhart (1998), where methods for computing leg-based spill costs are discussed, and we extend their ideas. With the help of a Passenger Mix Model solver, a Lagrangian Relaxation of IFAM is developed in order to produce improved solutions with only a FAM solver. We also detail the Dantzig-Wolfe decomposition associated with this Lagrangian Relaxation. Implementations are performed, and numerical results are given.

3.1 Motivations

The following section lists reasons why we believe that building algorithms that only rely on FAM is a good idea.

Software Leverage   FAM is a long standing standard model widely used by the airlines. Because it is simple, fast and reliable implementations already exist. These implementations carry the experience of years of practice. Basing new code on them would both allow to reuse this experience and reduce development effort.

Customizations and Specific Requirements   Airlines that already use FAM solvers are customizing them. They may enforce various requirements, such as legal constraints specific to the countries where their flights are operated, corporate business rules, local labor union negotiated constraints, or technical constraints of some airports. The final result is more than a

generic solver: it is a full software solution that is adapted to their own needs. Moving out of this solution involves reimplementing all these requirements in the new solver. By reusing FAM as a black box, the cost of reimplementing these requirements can be avoided, and the risk of suffering from bugs is minimized.

FAM as a measure   The universality of FAM makes it a de facto standard. Building around it allows to measure the time efficiency in terms of number of calls to the FAM engine. With such an approach, we obtain a measure which is independent of the hardware and the solver, which everyone can understand, provided they know the running time of FAM.

The idea of iteratively solving traditional FAM with improving inputs is not new. Such an approach might remind the reader the D^3 method from Berge and Hopperstad (1993), where a Fleet Assignment model is solved with increasingly accurate leg costs. More recently, Dumas and Soumis (2008); Dumas et al. (2009) described a model where a custom passenger flow model would interact with an implementation of FAM to converge in a few iterations.

The main source of inspiration of this chapter comes from Kniker and Banhart (1998), where the authors show that, using adjustments of a global coefficient κ for estimating the spill costs, the revenue of the FAM fleet assignments can be improved. The method is not accurate enough to fully optimize the fleet assignments produced. However we believe that more accuracy can be brought, using not one, but several coefficients to describe the spill over the network.

3.2   A Lagrangian Relaxation Approach

3.2.1   Computing leg-based Spill Costs

In Kniker and Banhart (1998), a coefficient κ is used to control globally the spill fare. The spill fare of the leg l is an estimation of the loss to the airline when a passenger is spilled on l.

$$\text{Fare}_l^{\text{sp}} = \kappa * \text{average fare on}(l)$$

This spill fare is used to compute the spill cost for the aircraft type t on the leg l:

$$c_{t,l}^{\text{sp}} = \text{Fare}_l^{\text{sp}} \max(0, Q_l - \text{Cap}_t)$$

Where Q_l is the unconstrained demand on the leg l, Cap_t is the capacity of the aircraft type t, and thus $\max(0, Q_l - \text{Cap}_t)$ is the minimum number of spilled passengers if a type t aircraft is chosen for l. Note that because Q_l is

not constrained by the network, $Q_l - \text{Cap}_t$ is an overestimation of the spill, because part of the passengers of this unconstrained demand might already be spilled on other legs.

Our instances present cabin capacity constraints. We adapt the formula to our needs and decompose this cost by cabin. This decomposition is of limited interest, except that it allows to better see one of the results of this chapter. Let $Q_{l,c}$ be the unconstrained demand on the leg-cabin (l,c):

$$
\begin{aligned}
c_{t,l}^{\text{sp}} &= \sum_{c \in C} C_{t,l,c}^{\text{sp}} \\
&= \sum_{c \in C} \text{Fare}_{l,c}^{\text{sp}} \max(0, Q_{l,c} - \text{Cap}_{t,c}) \\
&= \sum_{c \in C} \kappa.(\text{average fare on}(l,c)) \max(0, Q_{l,c} - \text{Cap}_{t,c})
\end{aligned}
$$

Kniker and Banhart (1998) show that a good choice of κ may improve the revenue, but cannot lead to optimality. Starting from this remark, we wonder whether more accuracy could be reached by directly estimating a spill fare for each leg-cabin: intuitively, instead of using only one single value of κ for the whole network, we would thus use a different $\kappa_{(l,c)}$ for each leg-cabin (l,c). We tried to design an iterative procedure of modification of these spill fares, basing our choice on the spill obtained on the network.

Such adjustments remind us what happens with Lagrangian Relaxations: in such schemes, a constraint is relaxed and is inserted in the objective function, such that the model has to pay for violating this constraint. With this approach, leg-cabin based spill fares could be interpreted as the price the model has to pay for violating passenger flow constraints.

Starting from the former remark, a Lagrangian Relaxation of IFAM has been implemented. In the following part, we describe, starting from the general case, how this relaxation was implemented.

3.2.2 Generic Description of a Lagrangian Relaxation

In this section, we give a general description of Lagrangian Relaxation and of the methods used to solve this kind of problem, with focus on the subgradient method. We then apply the results to IFAM. We mostly apply results already known in the literature. For more information on Lagrangian Relaxations, the reader is referred to Lemaréchal (2001).

Definitions

Provided a problem P defined by

$$
\begin{aligned}
\min \quad & cx + fs \\
\text{s.t.} \quad & x \in X \\
& Ax + Bs \leq a \\
& Ds \leq d \\
& s \geq 0
\end{aligned}
$$

$Ax + Bs \leq a$ being a set of "difficult" constraints, the Lagrangian Relaxation of P is the problem $LR(u)$ ($u \geq 0$) defined by:

$$
\begin{aligned}
\theta(u) = \min \quad & cx + fs + u(Ax + Bs - a) \\
\text{s.t.} \quad & x \in X \\
& Ds \leq d \\
& s \geq 0
\end{aligned}
$$

Although the results presented here are true for any problem of this form, their relevance might appear more clearly to the reader if we say now that P will later refer to IFAM, with x representing the fleet assignment, and s representing the spilled passengers.

Because no constraint links x and s in this formulation, $\theta(u)$ can be rewritten:

$$
\theta(u) = \theta_x(u) + \theta_s(u) - ua
$$

where $\theta_x(u)$ and $\theta_s(u)$ are the optimal values of the following problems:

- $LR_x(u)$:

$$
\begin{aligned}
\theta_x(u) = \min \quad & (c + uA)x \\
\text{s.t.} \quad & x \in X
\end{aligned}
$$

- $LR_s(u)$:

$$
\begin{aligned}
\theta_s(u) = \min \quad & (f + uB)s \\
\text{s.t.} \quad & Ds \leq d \\
& s \geq 0
\end{aligned}
$$

For a given u, we denote $x^*(u)$ (resp. $s^*(u)$) as the optimal solution of $LR_x(u)$ (resp. $LR_s(u)$).

Remarks

- Any feasible solution of P is a feasible solution of $LR(u)$.

- An optimal solution of $LR(u)$ provides a lower bound for P.

- Thus, if one finds an optimal solution $(x^*(u), s^*(u))$ for $LR(u)$ such that:

  - The objective of $LR(u)$ is the same as in P:

  $$u(Ax^*(u) + Bs^*(u) - a) = 0$$

  - The solution is feasible for P:

  $$Ax^*(u) + Bs^*(u) \leq a$$

  then this solution is optimal for P.

We search for the best lower bound that can be obtained by this Lagrangian Relaxation, which leads to solve the Lagrangian Dual, defined by:

$$\theta^* = \max \ \ \theta(u)$$
$$s.t. \ \ u \geq 0$$

Algorithm

To solve the Lagrangian Dual, the general approach is usually to compute a sequence $(u_q)_{q \in \mathbb{N}}$ that converges towards u^*, the value of u that is optimal for the Lagrangian Dual. In this process, we note GUB (Global Upper Bound) and GLB (Global Lower Bound) the best bounds known for the optimal value of P.

1: Initialize GUB
2: Initialize u_0
3: $q \leftarrow 0$
4: **repeat**
5:     solve $LR_x(u_q)$ and $LR_s(u_q)$, obtain $\theta_x(u_q)$, $\theta_s(u_q)$, $x^*(u_q)$, $s^*(u_q)$
6:     **if** $(x^*(u_q), s^*(u_q))$ is feasible for P and
    $u_q(Ax^*(u_q) + Bs^*(u_q) - a) = 0$ **then**
7:       $(x^*(u_q), s^*(u_q))$ is optimal for P, STOP
8:     **end if**
9:     $\theta(u_q) \leftarrow \theta_x(u_q) + \theta_s(u_q) - ua$
10:     **if** $\theta(u_q) > GLB$ **then**
11:       update GLB
12:       use $x^*(u_q)$ to improve GUB
13:     **end if**
14:     compute u_{q+1}
15:     $q \leftarrow q + 1$
16: **until** a stopping criterion is met

Algorithm 1: Lagrangian Dual

- GUB can be initialized by finding a solution of P, or using a heuristic. Additionally, if it is not too expensive, it can be updated on a regular basis (for example when GLB is improved) by turning a Lagrangian solution into a solution for P.

- Each iteration yields a Lagrangian solution $(x*(u_q), s*(u_q))$. However this solution may not be feasible for P. If we are interested in turning it into a feasible solution of P, we can extend $x^*(u_q)$ by solving for s in the following problem:

$$
\begin{aligned}
\min \quad & fs \\
\text{s.t.} \quad & Bs \leq a - Ax^*(u^*) \\
& Ds \leq d \\
& s \geq 0
\end{aligned}
$$

In the general case, this problem may not have a solution, but we will see that in our case, it always does.

- u_0 may be initialized to 0

- Classical stopping criteria might include, but are not limited to the following conditions:

- The maximal expected step between u_q and u_{q+1} is small.
- A time limit has been reached.
- The bounds have not been improved for a given number of iterations.

Subgradient

Several methods can be used to compute u_{q+1}. A simple, memoryless method is to use the subradient. In that method, u_{q+1} is computed from u_q using the following formula:

$$u_{q+1} = \max\{0, u_q + \rho G_q\}$$

G_q is a subgradient and represents the violation of the relaxed constraint:

$$G_q = Ax^*(u_q) + Bs^*(u_q) - a$$

ρ represents the quality of the solution. Usually,

$$\rho = \pi \frac{GUB - \theta(u_q)}{\|G_q\|^2}$$

π being a coefficient initialized to 2 and divided by 2 every 20 or 30 iterations without improvement of GLB. Of course, tuning is required for determining good values and criteria for updating it.

3.2.3   Lagrangian Relaxation, applied to IFAM

We now apply a Lagrangian relaxation to the IFAM formulation. Recall the IFAM formulation:[1]

---

[1] A reader who is familiar with the literature might notice that the quantity $\sum_{i \ni (l,c)} d_i$ could have been merged into one single quantity $Q_{(l,c)}$, the unconstrained demand on the leg-cabin (l, c). Because we believe it makes latter parts of this thesis easier to understand, we prefer to leave this quantity unchanged.

$$\min \sum_{l \in L} \sum_{t \in T} c_{t,l} x_{t,l} + \sum_{i \in I} \sum_{j \in I} (\text{Fare}_i - r_{i \to j} \text{Fare}_j) s_{i \to j} \tag{3.1}$$

$$s.t. \sum_{t \in T} x_{t,l} = 1 \qquad\qquad \forall l \in L$$
$$\tag{3.2}$$

$$\sum_{l \in I(n)} x_{t,l} + y_{t,g_i(n)} = \sum_{l \in O(n)} x_{t,l} + y_{t,g_o(n)} \qquad\qquad \forall n \in N$$
$$\tag{3.3}$$

$$\sum_{g \ni CT} y_{t,g} + \sum_{l \ni CT} x_{t,l} \le |t| \qquad\qquad \forall t \in T$$
$$\tag{3.4}$$

$$\sum_{i \ni (l,c)} d_i - \sum_{i \ni (l,c)} \sum_{j \in I} s_{i \to j} + \sum_{i \in I} \sum_{j \ni (l,c)} r_{i \to j} s_{i \to j} \le \sum_{t \in T} x_{t,l} \text{Cap}_{t,c} \quad \forall l \in L, \forall c \in C$$
$$\tag{3.5}$$

$$\sum_{j \in I} s_{i \to j} \le d_i \qquad\qquad \forall i \in I$$
$$\tag{3.6}$$

$$x_{t,l} \in \{0,1\}, y_{t,g} \ge 0, s_{i \to j} \ge 0 \tag{3.7}$$

The Leg Cover Constraints 3.2 are injected into the Cabin Capacity Constraints 3.5 to obtain:

$$\min \sum_{l \in L} \sum_{t \in T} c_{t,l} x_{t,l} + \sum_{i \in I} \sum_{j \in I} (\text{Fare}_i - r_{i \to j} \text{Fare}_j) s_{i \to j} \qquad (3.8)$$

$$s.t. \sum_{t \in T} x_{t,l} = 1 \qquad \forall l \in L$$
$$(3.9)$$

$$\sum_{l \in I(n)} x_{t,l} + y_{t,g_i(n)} = \sum_{l \in O(n)} x_{t,l} + y_{t,g_o(n)} \qquad \forall n \in N$$
$$(3.10)$$

$$\sum_{g \ni CT} y_{t,g} + \sum_{l \ni CT} x_{t,l} \le |t| \qquad \forall t \in T$$
$$(3.11)$$

$$\sum_{t \in T} \Big(\sum_{i \ni (l,c)} d_i - \text{Cap}_{t,c} \Big) x_{t,l} - \sum_{i \ni (l,c)} \sum_{j \in I} s_{i \to j} + \sum_{i \in I} \sum_{j \ni (l,c)} r_{i \to j} s_{i \to j} \le 0 \quad \forall l \in L, \forall c \in C$$
$$(3.12)$$

$$\sum_{j \in I} s_{i \to j} \le d_i \qquad \forall i \in I$$
$$(3.13)$$

$$x_{t,l} \in \{0,1\}, y_{t,g} \ge 0, s_{i \to j} \ge 0 \qquad (3.14)$$

We reformulate IFAM as in the precedent section:

$$
\begin{aligned}
\min \quad & cx + fs \\
s.t. \quad & x \in X \\
& Ax + Bs \le a \\
& Ds \le d \\
& s \ge 0
\end{aligned}
$$

- x are the fleet assignment variables, s are the spill variables.

- X designates the Constraints 3.9, 3.10, 3.11, and $x_{t,l} \in \{0,1\}$. In other words, it includes all the constraints of FAM, and represents the set of feasible fleet assignments. Note that X could include additional company specific constraints without changing the formulation.

- $Ax + Bs \le a$ represents the set of Constraints 3.12.

- $Ds \le d$ represents the set of Constraints 3.13

The Cabin Capacity Constraints 3.12, which correspond to $Ax + Bs \leq a$, are relaxed. Following what was previously shown, we can immediately rewrite the problem as two independent problems:

- $LR_x(u)$, a FAM problem with adjusted costs:

$$\min \sum_{l \in L} \sum_{t \in T} (c_{t,l} + \sum_{c \in C} u_{l,c} (\sum_{i \ni (l,c)} d_i - \mathrm{Cap}_{t,c})) x_{t,l}$$

$$s.t. \sum_{t \in T} x_{t,l} = 1 \qquad\qquad\qquad \forall l \in L$$

$$\sum_{l \in I(n)} x_{t,l} + y_{t,g_i(n)} = \sum_{l \in O(n)} x_{t,l} + y_{t,g_o(n)} \qquad \forall n \in N$$

$$\sum_{g \ni CT} y_{t,g} + \sum_{l \ni CT} x_{t,l} \leq |t| \qquad\qquad \forall t \in T$$

$$x_{t,l} \in \{0,1\}, y_{t,g} \geq 0$$

- $LR_s(u)$, a pure itinerary problem:

$$\min \sum_{i \in I} \sum_{j \in I} ((\mathrm{Fare}_i - \sum_{(l,c) \in i} u_{l,c}) - r_{i \to j}(\mathrm{Fare}_j - \sum_{(l,c) \in j} u_{l,c})) s_{i \to j}$$

$$s.t. \sum_{j \in I} s_{i \to j} \leq d_i \qquad\qquad\qquad \forall i \in I$$

$$\tag{3.15}$$

$$s_{i \to j} \geq 0 \qquad\qquad\qquad \forall i \in I, \forall j \in J$$

$LR_s(u)$ can be decomposed in $|I|$ independent problems. Indeed, for a fixed itinerary i, the variables $(s_{i \to j})_{j \in I}$ only appear in the Constraint 3.15. We shall thus define $LR_{s,i}(u), i \in I$ as:

$$\min \sum_{j \in I} ((\mathrm{Fare}_i - \sum_{(l,c) \in i} u_{l,c}) - r_{i \to j}(\mathrm{Fare}_j - \sum_{(l,c) \in j} u_{l,c})) s_{i \to j}$$

$$s.t. \sum_{j \in I} s_{i \to j} \leq d_i$$

$$s_{i \to j} \geq 0 \qquad\qquad\qquad \forall j \in I$$

The obvious optimal solution for each independent problem is to select j_0 such that $f_{i \to j_0} = (\mathrm{Fare}_i - \sum_{(l,c) \in i} u_{l,c}) - r_{i \to j_0}(\mathrm{Fare}_{j_0} - \sum_{(l,c) \in j} u_{l,c})$ is minimal:

- If $f_{i \to j_0} < 0$, spill everything on j_0: $s_{i \to j_0} = d_i$ and $s_{i \to j} = 0 \; \forall j \neq j_0$

- Otherwise do not spill: $s_{i \to j} = 0 \; \forall j$

Extending a Lagrangian Solution to a full Solution

As we said earlier, each iteration of the Lagrangian Relaxation algorithm yields a Lagrangian solution $(x*(u_q), s*(u_q))$, that may not be feasible for P. We explained that in order to turn this solution into a feasible solution of P, we could extend $x^*(u_q)$ by solving for s the following problem:

$$
\begin{aligned}
\min \quad & fs \\
\text{s.t.} \quad & Bs \le a - Ax^*(u^*) \\
& Ds \le d \\
& s \ge 0
\end{aligned}
$$

This problem is always feasible, because given a feasible fleet assignment, one can always find a feasible spill solution. For example, we can spill every passenger out of the network.

Interpretation

In the objective function of $LR(u)$, the quantity

$$
\sum_{c \in C} u_{l,c} \left(\sum_{i \ni (l,c)} d_i - \mathrm{Cap}_{t,c} \right)
$$

has to be compared with the spill cost $c_{t,l}^{\mathrm{sp}}$ presented in 3.2.1

$$
\sum_{c \in C} \mathrm{Fare}_{l,c}^{\mathrm{sp}} \max(0, Q_{l,c} - \mathrm{Cap}_{t,c})
$$

We can see that $u_{l,c}$ plays exactly the role of the spill fare on the leg-cabin (l, c). This statement is obvious when $\sum_{i \ni (l,c)} d_i - \mathrm{Cap}_{t,c} \ge 0$. If, on the contrary, $\sum_{i \ni (l,c)} d_i - \mathrm{Cap}_{t,c} < 0$, it means that there is always enough capacity for every passenger to fit in the aircraft, and that therefore, spill does not happen. As a consequence, in that case, because there is no spill, the spill fare can be anything. With the convention that the spill fare is chosen to be null in such situations, our statement always holds. Note that a subgradient approach will necessarily choose null values for non violated components, and will thus respect this statement.

Thus, FAM can be obtained as a solution of LR(u), with u such that:

$$
u_{l,c} = \begin{cases} \mathrm{Fare}_{l,c}^{\mathrm{sp}} & \text{if } Q_{l,c} \ge \mathrm{Cap}_{t,c} \\ 0 & \text{otherwise.} \end{cases}
$$

This interpretation gives us insights about what could be a good value for $u_{l,c}$. We can adapt the work of Kniker and Banhart (1998)

$$
u_{l,c}^{\mathrm{heuristic}} \simeq 0.7 \text{ average fare on } (l, c) \tag{3.16}
$$

where the value 0.7, as already mentioned, has been experimentally obtained by the authors based on a series of simulations.

Algorithm applied

The algorithm previously described is used, with the following adjustments:

- To initiate GUB, a FAM instance is solved, using spill costs from Kniker and Banhart (1998). This yields a Fleet Assignment, which is extended to a full solution, following the previously described process of Lagrangian solution extension.

- A classic subgradient method is used for obtaining u_{q+1} from u_q. We update the parameter π in a way that is dependant on the size of the instance. More accurately, π is divided by 2 every k iterations without improvement of GLB, where k is the number of Cabin Capacity Constraints (3.5) divided by 3: $k = \frac{|L||C|}{3}$.

3.2.4 Results

We run the obtained algorithm on two instances. The first instance is a small, theoretical instance of 30 legs, 46 itineraries. The second instance is based on data from a small airline, and has 225 legs, 1895 itineraries. We measure the iterations needed to converge. The running time is 18.39 seconds for the small instance, and we let the computation run for 2992.05 seconds for the large one. In the following diagrams:

- The curves labeled FAM and IFAM are the costs of FAM and IFAM fleet assignments. The cost of FAM has been determined by running a Passenger Mix Model on a FAM-based solution.

- The curve GUB refers to the Global Upper Bound. It is the cost of the best fleet assignment generated by this heuristic. It is initialized with the cost of FAM, and it decreases whenever a better solution is found.

- The curve labeled theta is the value of the Lagrangian Relaxation for the current iteration.

- The curve GLB is the Global Lower Bound, and is the highest value taken by theta for the current iteration.

On Figure 3.1, we observe that this algorithm requires at least 10 iterations to find a solution that is better than FAM. The best solution is found at the 94th iteration. On Figure 3.2, our heuristic does not improve the FAM solution. We notice that the overall convergence is slow and requires a lot of iterations. Thus, the subgradient method used to iterate
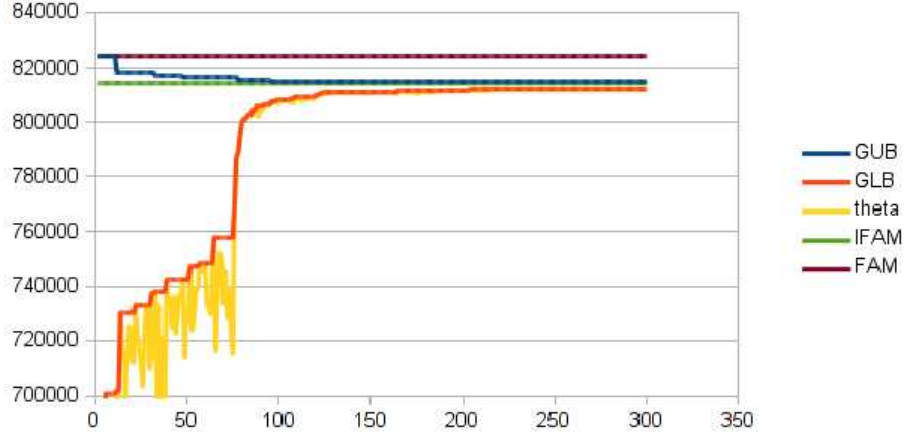
Figure 3.1: Convergence of the Lagrangian Relaxation (30 legs)
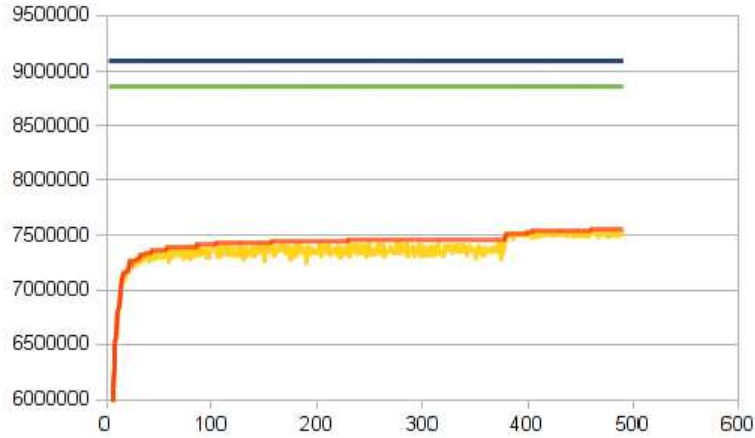
Figure 3.2: Convergence of the Lagrangian Relaxation (225 legs)

is not suited. Theoretically, if running for an infinite amount of time, the subgradient method, provided it has the right parameters, finds the optimal Lagrangian multipliers. However, in practice, the status depends on the stopping criteria applied. It is also well known that the subgradient method converges slowly because it is a memory-less process. We thus turn towards Dantzig-Wolfe Decomposition to improve the speed of this algorithm. The next section explains the relationship between Lagrangian Relaxation and Dantzig-Wolfe Decomposition, and then applies it to Fleet Assignment.

3.3   A Column Generation Approach

The following section first presents the Dantzig-Wolfe Decomposition Relaxation in a general context and enlightens the link with the previously exposed Lagrangian relaxation. Then, an application to IFAM is made, and results are shown.

3.3.1   Dantzig-Wolfe Decomposition Relaxation

In this section we adapt known results to show how, starting from our already implemented Lagrangian Relaxation, one can implement a Dantzig-Wolfe Decomposition Relaxation. For a highlight about how Lagrangian Relaxation and Dantzig-Wolfe Decomposition are generally linked, the reader is invited to refer to Briant et al. (2008).

Definitions

In the previous part, we already explained that, starting from P (P will again refer to IFAM later)

$$
\begin{aligned}
\min \quad & cx + fs \\
s.t. \quad & x \in X \\
& Ax + Bs \leq a \\
& Ds \leq d \\
& s \geq 0
\end{aligned}
$$

and relaxing the Constraints $Ax + Bs \leq a$, would lead to solve the independent problems $LR_x(u)$ and $LR_s(u)$. We now assume that X is a finite set, and that $S = \{s | Ds \leq d, s \geq 0\}$ is a bounded polyhedron (X and S will later be refer to the set of feasible fleet assignment and the set of spill solutions). Let X be indexed as follows:

$$\{x^k, k \in K\}$$

And the set of vertices of the polyhedron S will be denoted:

$$\{s^m, m \in M\}$$

The previous problems can be rewritten as:

- $LR_x(u)_{(u \geq 0)}$:

$$\theta_x(u) = \min_{k \in K}(c + uA)x^k$$

- $LR_s(u)_{(u \geq 0)}$:

$$\theta_s(u) = \min_{m \in M} (f + uB)s^m$$

The Lagrangian Dual Problem is expressed as:

$$\theta^* = \max_{u \geq 0} \theta(u)$$
$$= \max_{u \geq 0} \theta_x(u) + \theta_s(u) - ua$$
$$= \max_{u \geq 0}[\min_{k \in K}(c + uA)x^k + \min_{m \in M}(f + uB)s^m - ua]$$

Since X and S are finite:

$$
\begin{aligned}
\theta^* = \quad & \max \quad \alpha + \beta - ua \\
& s.t. \quad \alpha \leq (c + uA)x^k && \forall k \in K \\
& \quad \quad \beta \leq (f + uB)s^m && \forall m \in M \\
& \quad \quad \alpha \in \mathbb{R}, \ \beta \in \mathbb{R}, \ u \geq 0 \\
\theta^* = \quad & \max \quad \alpha + \beta - ua && (3.17) \\
& s.t. \quad \alpha - uAx^k \leq cx^k && \forall k \in K \\
& \quad \quad \beta - uBs^m \leq fs^m && \forall m \in M \\
& \quad \quad \alpha \in \mathbb{R}, \ \beta \in \mathbb{R}, \ u \geq 0
\end{aligned}
$$

Taking the dual of this linear program:

$$
\begin{aligned}
\theta^* = \quad & \min \quad \sum_{k \in K}(cx^k)\lambda_k + \sum_{m \in M}(fs^m)\mu_m \\
& s.t. \quad \sum_{k \in K}\lambda_k = 1 && (\text{variable } \alpha) \\
& \quad \quad \sum_{m \in M}\mu_m = 1 && (\text{variable } \beta) \\
& \quad \quad \sum_{k \in K}(Ax^k)\lambda_k + \sum_{m \in M}(Bs^m)\mu_m \leq a && (\text{variable u}) \\
& \quad \quad \lambda_k \geq 0, \mu_m \geq 0
\end{aligned}
$$

Because $\{\sum_m s^m \mu_m | \sum_m \mu_m = 1, \mu_m \geq 0\} = conv(S) = \{s | Ds \leq d, s \geq 0\}$, we are left with the following problem, which is the Dantzig-Wolfe De-

36

composition Relaxation associated with the Lagrangian Relaxation previously exposed, noted DW:

$$\theta^* = \quad \min \quad \sum_k (cx^k)\lambda_k + fs$$

$$s.t. \quad \sum_k \lambda_k = 1 \qquad \qquad \text{(dual value w)}$$

$$Ds \leq d \qquad \qquad \text{(dual value v)}$$

$$\sum_k (Ax^k)\lambda_k + Bs \leq a \qquad \text{(dual value u)}$$

$$\lambda_k \geq 0, s \geq 0$$

Algorithm

Building an algorithm to solve DW leads us to consider a problem DW$'$ that contains only a subset X' of $X = \{x^k, k \in K\}$. In such a process, one may repeatedly solve DW$'$ and add elements to X' until it is proven that no additional element can improve the solution. This corresponds to adding columns to the matrix of DW$'$, which is why it is often referred to as a column generation algorithm.

A step in this algorithm is then to determine a new $x \in X$ to insert into X'. To do so, one may apply a steepest edge strategy, i.e. determine the column of most negative reduced cost to introduce X'. Let (u^*, v^*, w^*) be the optimal dual solution of DW$'$, and let $C = (c_k)_{k \in K}$ be the vector of the reduced costs. We try to find the column of most negative reduced cost in C. This means finding:

$$\min_{k \in K} c_k = \min_{k \in K} cx^k - w^* - Ax^k u^*$$
$$= \theta_x(-u^*) - w^*$$

Note also that solving DW$'$ naturally yields the Lagrangian cost. Indeed, let (λ^*, s^*) the optimal primal solution of DW$'$, and $(u*, v*, w*)$ be the optimal dual solution. These solutions being respectively primal and dual feasible, we have:

$$s^* \in \{s | Ds \leq d, s \geq 0\} \text{ and } v^* \in \{v | u^*B + vD \leq f, v \leq 0\}$$

And, from the Complementary Slackness theorem, we have:

$$v^*(Ds^* - d) = 0 \text{ and } (f - u^*B - v^*D)s^* = 0$$

Thus, s^* and v^* are optimal solutions of the dual linear programs:

$$(f - u^*B)s^* = \quad \min \quad (f - u^*B)s \quad = \quad \max \quad vd \qquad \qquad = v^*d$$
$$s.t. \qquad Ds \leq d \qquad \qquad s.t. \quad vD \leq f - u^*B$$
$$s \geq 0 \qquad \qquad \qquad v \leq 0$$

and thus $\theta_s(-u^*) = (f - u^*B)s^* = v^*d$, which means that the Lagrangian bound can be obtained at each step by computing

$$\theta(-u^*) = \theta_x(-u^*) + (f - u^*B)s^* + u^*a$$

The following algorithm naturally ensues:

```
 1: Initialize GLB
 2: Initialize GUB
 3: Initialize X'
 4: repeat
 5:     solve DW(X'), obtain the optimal dual solution (u*, w*).
 6:     solve LR_x(-u*), obtain x*(-u*), θ_x(-u*), θ(-u*).
 7:     if θ(-u*) > GLB then
 8:         update GLB
 9:         complete x*(-u*) to a full solution to see if it improves GUB
10:     end if
11:     if θ_x(-u*) - w* < 0 then
12:         X' ← X' ∪ {x*(-u*)}
13:     else
14:         No column of negative reduced cost can be inserted in X', STOP
15:     end if
16: until A stopping criterion is met
```
<center>**Algorithm 2:** Dantzig-Wolfe</center>

3.3.2 Application to IFAM

With the already stated reformulation of IFAM, we apply the results of the previous section. We verify that X and S are finite:

- X, being the set of feasible fleet assignments, necessarily represents a finite number of possibilities.

- S, being a bounded polyhedron (the spill is limited by the demand, which is finite for every itinerary), has a finite number of vertices.

The algorithm described is applied, with the first column inserted in X determined using a traditional FAM with assignment costs as in Kniker and Banhart (1998).

3.3.3 Results

We run the obtained algorithm on the same two instances. The running times are 0.70 seconds for the small instance, and 1775.55 seconds for the large one. In the following diagrams:

<center>38</center>

- The curves labeled FAM and IFAM denote again the costs of FAM and IFAM fleet assignments.

- The curves GUB and GLB denote the Global Upper and Lower Bounds.

- The curves labelled theta and master denote the values of the lagrangian dual and the master problem.
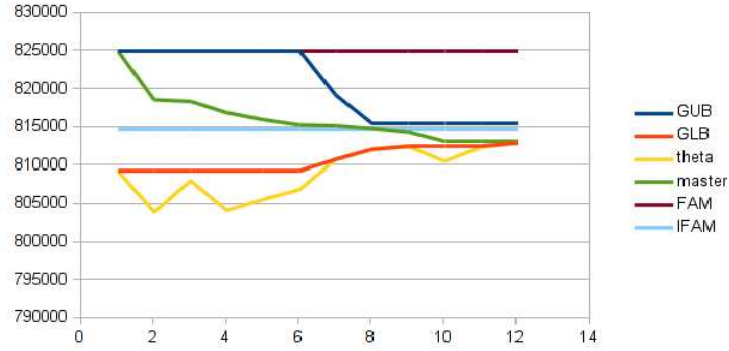
Figure 3.3: Dantzig-Wolfe Decomposition Relaxation Convergence (30 legs)
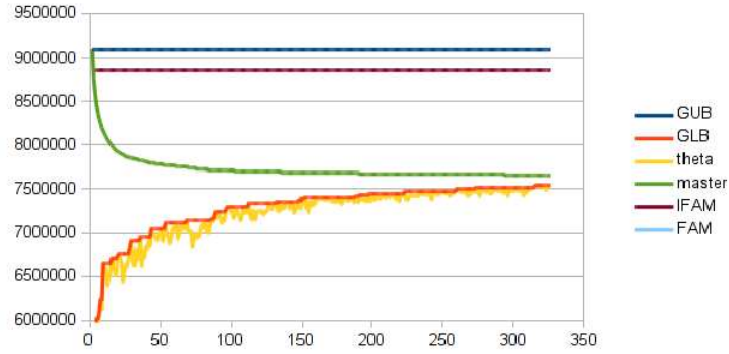
Figure 3.4: Dantzig-Wolfe Decomposition Relaxation Convergence (225 legs) (FAM and GUB are the same line: the solution is not improved)

In terms of computation time, we observe that the results are better. We can see in Figure 3.3 that the best solution is found at the 8th iteration, and the optimality of the solution of the Dantzig-Wolfe linear relaxation is proven at the 12th iteration. However, the Dantzig-Wolfe heuristic does not improve the FAM solution for the large instance after 325 iterations.

To prove the optimality, one has to ensure that no columns of negative reduced cost can be generated anymore. Even if the optimal solution has been found, an important number of additional iterations (and of columns that do not improve the problem) might thus be required until this solution is guaranteed to solve the best the Dantzig-Wolfe decomposition relaxation. In practice, one may want to find criteria to stop the algorithm without such a guarantee, but this requires fine tuning.

3.3.4 Analysis of the Results and Possible improvements

Generating a better Fleet Assignment from the Columns

Our results show that, at least in the second case, our heuristic is unable to improve the Global Upper Bound. This means that, in the process of generating fleet assignments to insert in the relaxation of DW$'$, we do not find a better fleet assignment than FAM. Since we are considering a relaxation of the problem and not the problem itself, it is indeed possible that no better fleet assignment is needed to reach optimality. As a consequence, we might want to improve this process by attempting to find a better solution from the generated set of fleet assignments.

Solving the relaxation of the Dantzig-Wolfe decomposition of this problem is equivalent to finding a set of fleet assignment that collectively solve the relaxed problem better than an individual fleet assignment. The iterations indicate which column may be interesting to insert in order to improve the overall solution. Given $\{x^k | k \in K\}$, the set of fleet assignment generated by our process, the convex combination $\sum_{k \in K} \lambda_k x^k$ is thus the best fleet assignment for solving the linear relaxation of DW'. However, this combination is not itself a feasible fleet assignment.

If we find out that none of these fleet assignments is particularly good individually, an interesting heuristic would be to implement an algorithm which, given this combination of fleet assignments, would find the closest feasible fleet assignment. We would thus be solving a instance of FAM with modified leg costs: The more different from the solution the aircraft type, the more expensive the leg cost.

$$C_{t,l} = \begin{cases} \sum_{k \in K}(1 - \lambda^k x_{t,l}^k) & \text{if } \sum_{k \in K} \lambda_k x_{t,l}^k \neq 0 \\ M & \text{otherwise.} \end{cases}$$

Where M would be a large constant, preventing to choose an aircraft type if no column generated ever picks it. Because we believed that our time would be better spent working on an alternative model, this idea was not implemented.

Stabilization

Another possible improvement could be to add stabilization. Indeed, it is a well-known fact (see, for example, Briant et al. (2008)) that, without stabilization, Dantzig-Wolfe decompositions have a bad performance. Many elaborate stabilization methods exist. We describe here a simplified version of one of the simplest, the so called box-step stabilization. The principle is to prevent the problem to introduce useless columns, by adding limits on the dual values of DW. We thus insert the following constraints in the Lagrangian Dual Problem 3.17:

$$u \leq U$$

Where U is what we consider to be a good estimation for u.

When dualized, this translates into additional columns in DW:

$$\theta^* = \qquad \min \quad \sum_k (cx^k)\lambda_k + fs + U\delta$$
$$s.t. \quad \sum_k \lambda_k = 1$$
$$Ds \leq d$$
$$\sum_k (Ax^k)\lambda_k + Bs - \delta \leq a$$
$$\lambda_k \geq 0, s \geq 0, \delta \geq 0$$

When the algorithm stops generating columns of negative reduced costs, we have to check that none of these artificial δ variables are used. Otherwise, we authorize a wider range of dual costs, which means increasing U over the non-null δ components.

We can make an educated guess about a good starting value for U: since it is a bound on the Lagrangian multipliers, which have the same meaning as the spill fares of FAM, it is very likely that $u^{\text{heuristic}}$ from Formula 3.16 will be a good starting point. The increases over U could correspond to doubling the limits over the non-null delta components. Again, this idea was not implemented, in order to focus on alternative models.

3.4 Conclusion

In this chapter, we exposed an attempt to produce better fleet assignment with only a FAM Solver, using an approach that iteratively enhances the costs of a FAM problem. We have seen that this approach was successful on a toy instance, but it fails to address larger instances. Moreover, because this work is based on relaxing IFAM, we can only hope the solutions produced to be as good as IFAM's. This second point was clearly a problem to be addressed by this thesis, and we thus chose not to push further this study in order to focus on designing an alternative model.

One important lesson this study taught us was that it is difficult to obtain reliable data in order to build fleet assignment instances. Obtaining unconstrained demands and recapture rates requires skills and experience. This raised the issue of testing the reliability of IFAM with uncertain data. Moreover, this made us we believe that it would be interesting to spend time working on a model, that would make it easier to deal with unreliable sources of inputs.

The next chapter focuses on creating an alternative model to IFAM, targeting better usability.

Chapter 4

An alternative Fleet Assignment Model

The goal of the previous chapter was to improve the quality of the solutions of a FAM solver, assuming informations on the itinerary demands. The methods we exposed thus assumed that the airline was able to provide IFAM data to work with. Yet, another challenge for implementing IFAM is to gather the data necessary to build the instances. The data-mining we had to perform to test our own implementations has convinced us that the task of obtaining itinerary based demands and recapture rates is intricate.

IFAM relies on a large amount of different inputs, which may be obtained with more or less reliability. We believe that such exhaustiveness not only makes the model difficult to implement, but may not be necessary. Our goal, in this part, is to propose an alternative fleet assignment model able to deal with less information. In order to do so, we analyze the consequences of using moderately inaccurate to very inaccurate forecasts with IFAM. These experiments lead us to formulate a series of desired features for a Fleet Assignment model with simplified input. At the end, we propose a new model that allows airlines to provide demand forecasts in a more flexible way. Our model is a generalization of a no-recapture version of IFAM, where demands are modeled as variables of a Linear Programming Model. These demands can be constrained according to the forecasts. We illustrate the usage of this model with examples.

4.1   IFAM in an uncertain Context

Barnhart et al. (2002) have shown how FAM poorly models the passenger behavior, creating fleet assignments that do not properly adapt to the passenger demands. As a consequence, IFAM was created to solve this issue. However, 10 years later, at the time this thesis is being written, a lot of airlines still use FAM to assign their fleet. There are plenty of reasons for

such an innovation not to spread to the whole airline industry immediately. Nevertheless, we believe that one of these reasons is a lack of confidence in the model, and more accurately a lack of confidence in one's ability to produce proper demand forecasts for this model.

In this section, we enumerate the different forecasts needed to implement IFAM. We list sources of inaccuracies that effectively make these forecasts hard to obtain with confidence. We then study the influence of using inaccurate inputs for the generated fleet assignments. Our goals are to explain why some airlines report to be uncomfortable with implementing IFAM, and to determine a series of criteria for the elaboration of an alternative model.

4.1.1 Implementing IFAM in real Life

Forecasts required by IFAM

The key idea introduced by Barnhart et al. (2002) with IFAM is that on an airline network, passengers book itineraries, not flight legs. Therefore, changing the capacity of a leg impacts all the itineraries using this leg. This, in turn, impacts the traffic on the other legs used by these itineraries. By propagation, this might induce changes on the whole network traffic. As a consequence, the revenue of the fleet assignment could be modified by a considerable amount. However, FAM, with its leg-based representation of the revenue, does not accurately take this effect into account: modifying the aircraft assigned to a leg does not modify the cost of other legs. This is why it may thus yield suboptimal solutions.

The solution proposed by Barnhart et al. (2002) is to embed a Passenger Mix model within the fleet assignment constraints. In other terms, a simulation of how passengers travel on the flight network is made part of in the Fleet Assignment Model. Passenger data is introduced as an input of the problem:

- For every itinerary i, a demand d_i and a fare Fare_i are known. The demand has to be unconstrained, which means that it represents the number of customers desiring to travel on the itinerary i, regardless of the number of seats available for i. The fare is the price paid to travel on i.

- For every pair of itineraries (i, j), a recapture rate $r_{i \rightarrow j}$ has to be known. The recapture rate represents the ratio of passengers who would accept to purchase j if i is unavailable.

This allows, with the variables $s_{i \rightarrow j}$ standing for the number or passengers spilled from i to j, and I being the set of all itineraries, to accurately express the revenue of the fleet assignment as:

$$\sum_{i \in I} \text{Fare}_i d_i - \sum_{i \in I} \sum_{j \in I} (\text{Fare}_i - r_{i \to j} \text{Fare}_j) s_{i \to j}$$

How accurate can these Forecasts be?

There is no doubt that a company with the knowledge of a 100% accurate forecast of the aforementioned parameters would indeed be able to produce perfect fleet assignments. However, obviously such an input can only be obtained through an estimation, and is thus subject to inaccuracy. Since any company willing to implement IFAM would try to estimate these values, we naturally wonder how accurate would the forecasts be.

The true accuracy of a forecast can only be determined afterwards with a measure of the actual value. However, in a first approximation, the only data at the disposal of an airline is the number of booking on each itinerary. This, of course, gives a lower bound of the demand, since the bookings are constrained by the capacities of the network. If we try to express b_i, the number of bookings over the itinerary i, we obtain:

$$b_i = d_i - \sum_{j \in I} s_{i \to j} + \sum_{j \in I} r_{j \to i} s_{j \to i}$$

Observing b_i is not enough to simultaneously guess the values of d_i, $s_{i \to j}$ and $r_{i \to j}$, since several solutions might fit: for example, one could always assume that $b_i = d_i$ and $s_{i \to j} = 0$ for all (i, j) (the demand exactly fits the number of bookings and there is no spill nor recapture) and get a consistent result. Of course, such an assumption would be wrong, and has been proven by revenue management studies to produce a spiral down effect: under-estimating the demands conduces to assign lower capacities aircraft, which in turns, conduces the demands to actually decrease. However, its consistency indicates that measuring the accuracy of demand and recapture forecasts cannot be done by only observing the bookings, since wrong forecasts can produce good results by this measure.

We believe that there is no way to know exactly how many passengers are spilled from an itinerary to another. In the hypothesis that a company actually tried to accurately estimate these figures, this would require knowing which customers are part of the legitimate demand on an itinerary, and which customers were recaptured. So, for every passenger, one would need to know if, when they bought their ticket, they would actually have preferred buying an alternative itinerary. Not only this would be impractical, but it would probably not make sense:

- The situation of choosing an alternative itinerary as a replacement for a preferred option never actually happens for the passenger, because the airline never tells them that this preferred option exists. Instead,

when itineraries are bought, revenue management has decided which itineraries are available for sale, and might hide the preferable alternatives. Customers pick their preferred deal among this reduced set of choices, without being given the opportunity to tell which demand they are actually part of.

- On a given itinerary, it would be difficult to distinguish between passengers being recaptured from an alternative itinerary, and passengers coming from the original demand. Obviously, everyone wants a cheap deal. Does this make everyone count as spilled from the cheap fare-classes?

The fact that the actual recapture rates and unconstrained demands are potentially impossible to measure does not prevent good estimations to be computed. In fact, every company has elaborate processes to estimate the number of passengers traveling to identify their market share, plan their strategy, and perform revenue management. It has to be made clear that this paragraph is not meant to challenge these methods, and we may assume that the state of the art tools that are involved, are yielding the best possible forecasts for demand and recapture. But our actual claim is that even if a method is yielding the best possible forecast, there is no way to compare this forecast with actual values, because no one can ever observe them. Therefore the real accuracy of these tools remains unknown.

What is the Impact of Inaccuracy?

Because of the reasons we mentioned, we believe that it is difficult to know how good or bad the forecasts about unconstrained demand and recapture rates actually are. The question that naturally arises is: does it really matter? If good accuracy brings little value, airlines can safely ignore any claim about forecast inaccuracy, and use approximate metrics. It can also be a sign that the models can be simplified. What follows is a short study about unconstrained demands, recapture rates, and the influence of an error on the revenue of a fleet assignment.

Previous Work

Barnhart et al. (2002) provide a validation of IFAM based on sensitivity analysis. They show, through empirical testing, that IFAM only requires a rough estimation of recapture rates. A set of recapture rates is used for performing the fleet assignment, and another set of recapture rates is used for estimating the actual benefit of the fleet assignment. The recapture rates used to perform the fleet assignment are linearly scaled by the same factor, from 0.5 to 1.5 times the actual rates. The result is that small variations of this factor do not affect significantly the fleet assignments.

Demand uncertainty is also addressed. A base case scenario of demands is defined. This base case is used to build an IFAM instance, and a FAM instance (the demands in FAM being determined by selecting the appropriate itineraries). Then, an important number of perturbed scenarios of demands are generated from the base case, using it as the average. These scenarios are applied to the fleet assignments obtained with the two models, in order to determine their respective average revenues. To apply a scenario to a fleet assignment, the authors use two different methods. One of them is to directly use PMM, and the other is to use a restricted version of PMM, which models an imperfect control of Passenger Choices. In both cases, it is shown that the average revenue yielded by IFAM is, in most cases, superior to the revenue obtained from applying FAM.

In the next sections, we try to complement their work with other small experiments.

4.1.2 Effect of Recapture Rate Inaccuracy on IFAM Profit

Since there is no way to give an exact measure of the recapture, it is not possible to measure the real accuracy of any recapture rate forecast. This does not mean that a forecast of these recapture rates cannot be good. In fact, it could be possible that forecast tools are very good and produce 100% accurate data, but we believe that even in the presence of a perfect recapture forecaster, while we might be able to confirm that the reality behaves consistently with the forecasts, we would not be able to prove that this forecast is the right one.

Worrying about a forecast is one thing, but what really matters in the end, since we are likely to produce error prone forecasts, is knowing the impact of a bad forecast. If inaccuracy in recapture rate forecasts leads to bad fleet assignments, we would rather know how inaccurate the forecasts need to be to really impact the fleet assignment.

In order to measure this impact on the revenue, we designed an experiment. The goal of this experiment is to determine the impact of bad recapture rate forecasts on profit. A fleet assignment instance for the IFAM model is chosen, and is assumed to perfectly represent what is going to happen in the future. We note this instance I. We know that solving I will lead to the perfect fleet assignment. Our objective is to measure what happens if the forecasts of recapture rates deviate from the perfect forecasts by some known amount. So instead of solving I, we solve I', which is a perturbed version of I. In I', recapture rates have been modified by random amounts, but no more than a given value $p\%$. Solving I' leads to a fleet assignment, noted X'. The assignment X' is most likely not optimal for I. We want to know by what amount. To do so, we measure the efficiency of this fleet assignment, by running a passenger mix model on X', using the recapture rates and the demands of I. We repeat this procedure 20 times for a given

p, and we plot the average revenue for a given perturbation. Our procedure is summarized by the given algorithm:

1: I is an IFAM instance
2: **for** $p \in 0..100$ **do**
3:    **for** $k \in 1..20$ **do**
4:      $I' \leftarrow \text{perturb}(I, p)$: perturb I by $p\%$
5:      $X' \leftarrow \text{assign}(I')$: solve I', get X' the fleet assignment
6:      Use X' and the passenger data of I to compute the revenue X' would actually generate.
7:    **end for**
8:    record the average revenue for a given perturbation
9: **end for**
    Algorithm 3: Solution quality for a given perturbation

Recapture rates are perturbed using the following method: each recapture rate is perturbed by a random amount. This amount follows a uniform distribution centered in zero, of length $2*p$ if p is the perturbation targeted. Obviously, the result is truncated to remain in the interval $[0, 1]$. However, recapture rates are perturbed only if they were originally non null, because it would not make sense to recapture between totally unrelated itineraries. We made sure that every possible recapture rate that could make sense would be considered.

This procedure is applied on a set of different test cases, so that we can see if the impact of bad recapture rate forecasts vary with the instance. These test cases are rated from very pessimistic to very optimistic. They are derived from a small theoretical instance of 30 legs, 46 itineraries, and from a real life instance of 225 legs and 1895 itineraries. For both instances, the reference recapture rates have been generated artificially, following a simple criteria including match in origin and destination, difference of price, flight duration and departure time offset. The demands are such that, in the optimal fleet assignments, the average load factor is 80% over the network. The operating costs are such that the profit of the assignment is near zero (We refer to this situation as "balanced", since the operating cost of the fleet assignment is compensated by the revenue made out of the bookings).

Eight other situations of demand and operating costs are also tested. These situations correspond to combinations of:

- high (resp. low) demands, where each itinerary demand is 30% higher (resp. lower)

- high (resp. low) operating costs where the operating costs are 10% higher (resp. lower)

Our results are reported in the figures 4.1 and 4.2. For each perturbation, we compare the profit obtained on average with the profit of the optimal
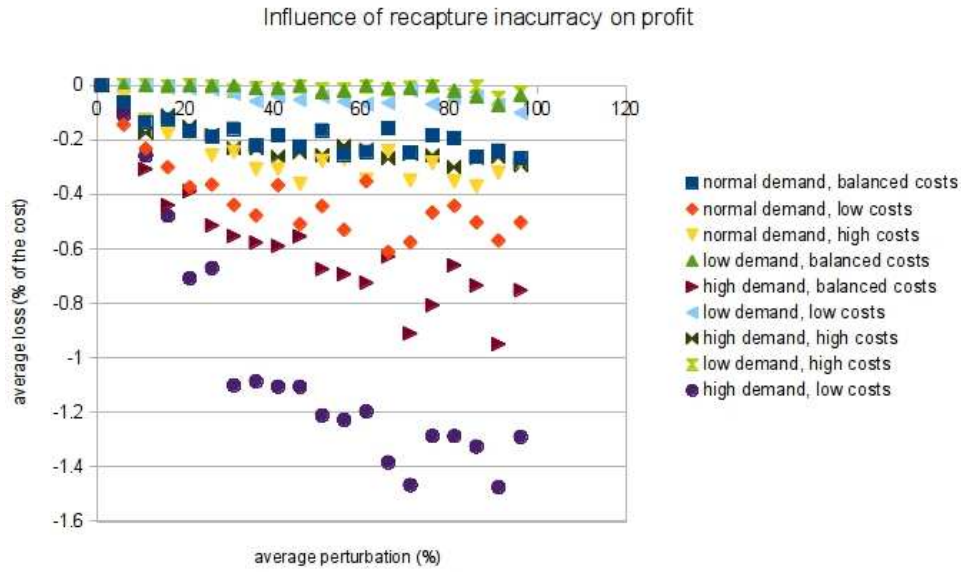
Figure 4.1: Effect of perturbing recapture rates - 30 legs, 46 itineraries
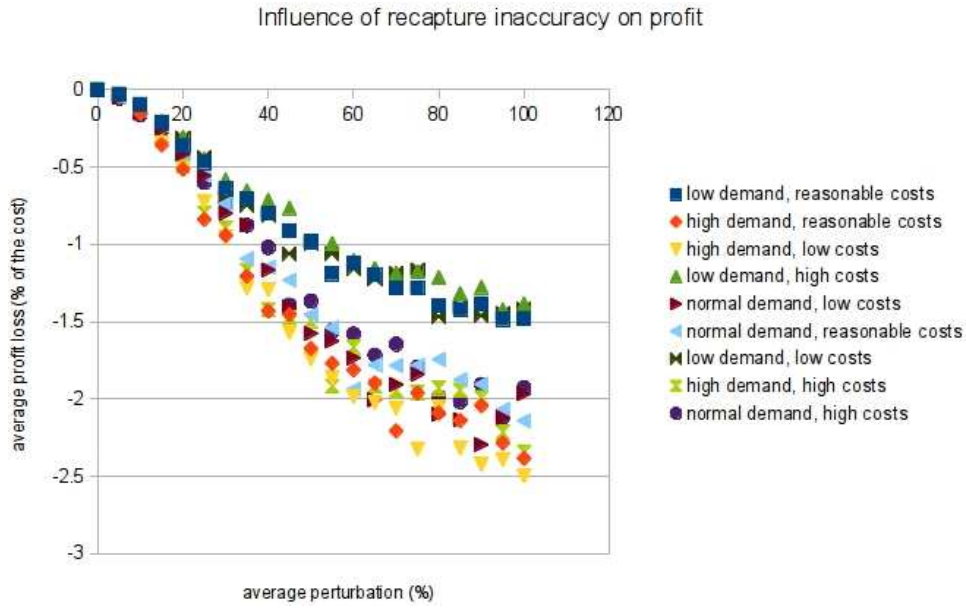

Figure 4.2: Effect of perturbing recapture rates - 225 legs, 1895 itineraries

fleet assignment for the reference instance. The axis "average perturbation" is the average amount of perturbation applied to the recapture rates. The axis "average loss" represents the profit loss due to poor forecasting. This profit loss is not expressed as a percentage of the optimal profit. Instead,

49

it is expressed relatively to the operating cost of the instance. The reason for normalizing to the operating cost and not to the optimal profit is that, as previously mentioned, some of the instances have been specifically designed to yield zero profit, so dividing by this value makes no sense. As a consequence, we choose to normalize this profit loss to the operating cost of the instance, because we expect this value to be a good point of comparison for any airline. We plot this percentage against the average perturbation applied.

As one might expect, the recapture rates are more likely to play a role when there is a high demand and low operating costs. What we can see is that even with highly perturbed recapture rates, the profit of the assignment varies by at most 2.5% of the operating cost. In other terms, it does not seem to matter if the recapture rate forecasts are very bad, the assignment will still have a similar performance. This suggests that the recapture concept may not introduce provable improvements in fleet assignments. However, the bigger base instance of 225 legs is the one with the worse results regarding this aspect, and has a ratio itinerary/leg of $1895/225 = 8.5$, which is more important than the ratio itinerary/leg of the toy instance, which is $46/31 = 1.5$. This suggests that having more itineraries worsens the effect of using poor recapture rates forecasts. Other instances in the literature are reported to have 2000 legs and 75000 itineraries, showing that standard ratios may reach 35 itineraries for one leg. The results of this experiment thus need to be taken carefully, and it would probably be a good idea for airlines using IFAM to reproduce the experiment with their own data.

4.1.3 Measuring small Demand Values versus grouping them

To implement IFAM, one needs to provide demand forecasts for every possible itinerary over the airline network. This leads to a high granularity of demand, making it hard to predict, as described by Culioli (2006), referring to both Air France and KLM. As such, the IFAM approach is subject to large deviations between expected and actual demands, possibly making the resulting fleet assignment decisions inadequate. In a hub-and-spoke network, and on a given leg, airlines report that they are confident about forecasting demand to a set of geographically close destinations, as opposed to one particular destination. Revenue Management literature has verified this claim. What follows is another experiment and a demonstration designed to confirm it with another proof.

In order to determine whether leg demands are actually subject to less variability than itinerary demands, we design a small experiment. The goal of this experiment is to associate a measure of the variability of the itinerary demand to a measure of the variability of the leg demand. To measure the variability of a variable, we use the coefficient of variation, which is defined as the ratio of the standard deviation σ to the mean μ: $CV = \frac{\sigma}{\mu}$. This ratio,

also known as the relative standard deviation, is well suited for comparing the variabilities of variables of same nature, but of different means: we thus adopt it as our measure of variability. We define a process that takes as an input a given coefficient of variation for the itinerary demands, and outputs the average coefficient of variation of the leg demands on the network.

An instance I is chosen. The itinerary demands of I are randomly perturbed, to produce the instance I'. The perturbation applied to obtain I' is such that each itinerary i is perturbed by a random amount ϵ_i, where ϵ_i follows a normal law centered in 0 and of standard deviation σ_i; σ_i is determined such that the coefficient of variation $CV_i = \frac{\sigma_i}{d_i}$ is constant over the network: for all i, $\sigma_i = CV.d_i$. Of course, should the result be negative, the perturbed demand is truncated to 0. The demands in I' are then aggregated at the leg level. For a given input of CV, this process is repeated 20 times, such that for every leg, a sample of 20 aggregated demands is obtained. For each leg l, we then compute the mean d_l and the standard deviation σ_l of the sample, that we combine in the coefficient of variation $CV_l = \frac{\sigma_l}{d_l}$. All the coefficients of variation of all the leg demands are then combined into their average, to form the average leg demand variation.
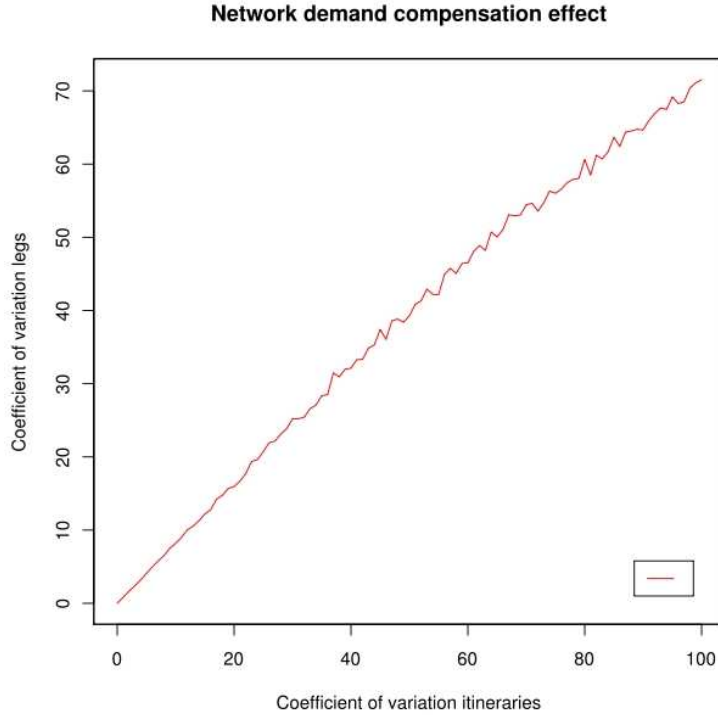

Network demand compensation effect

Figure 4.3: Accuracy of the demand: leg-based versus itinerary-based

Figure 4.3 reports the measured coefficient of variation of the leg demand

51

for a given coefficient of variation on the itinerary demand, for a 30 legs/46 itineraries instance. We can see that the leg-based demand is slightly less variable than the itinerary-based demand, with a coefficient of variation of 70% for the leg-based demand when the itinerary-based demand reaches 100%. What is also interesting about this curve is that it seems to be linear.

In fact, this network compensation effect is simply the effect of the law of large numbers. What follows shows that, under the assumptions we made in this experiment, the coefficient of the curve can be precomputed:

Proof. Assume a network of legs L, run over with a set of itineraries I. Consider any leg $l \in L$. Every itinerary i going through l has a demand that we note D_i. To match the conditions of our experiment, we make the following assumptions:

- D_i follows a normal law of mean d_i and standard deviation σ_i.

- The demands are independent

- Like in the previous experiment, the coefficient of variation of the demands is constant: for all $i \in I$, $\dfrac{\sigma_i}{d_i} = \mathrm{CV}$, or more conveniently, $\sigma_i = \mathrm{CV}.d_i$

We now note D_l the cumulated demand over the leg l:

$$D_l = \sum_{i \ni l} D_i$$

- D_l, as a sum of independent variable of normal law, follows a normal law of mean $\sum_{i \ni L} d_i$ and of standard deviation $\sqrt{\sum_{i \ni l} \sigma_i^2} = \mathrm{CV}\sqrt{\sum_{i \ni l} d_i^2}$.

- The coefficient of variation of D_l is thus $\mathrm{CV}_l = \mathrm{CV}\dfrac{\sqrt{\sum_{i \ni l} d_i^2}}{\sum_{i \in L} d_i}$

- We can thus compute exactly the average coefficient of variation over the network: $\mathrm{CV} = \frac{1}{|L|}\sum_{l \in L} \mathrm{CV}_l$. This is the value we have been plotting.

$\square$

From this result we can also deduce that the more itineraries on a leg, the least variable the aggregated demand. Indeed, assuming the D_i have the same mean, i.e. $d_1 = d_2 = \cdots = d$, we can rewrite:

$$\mathrm{CV}_l \leq \frac{\mathrm{CV}}{\sqrt{n}}$$

where n is the number of itineraries using l.

In other terms, compared to the variability of the demand of a single itinerary, the variability of a leg-based demand is divided by the square root of the number of itineraries on this leg. Such an equation suggests it is easier to measure demands when they are aggregated, confirming the claim made by Culioli (2006). Interestingly, the way to aggregate the demands has no impact on the result. With these hypothesis, any set of itineraries could be considered, and their demands, considered together, would be less variable.

4.1.4 Conclusion about these Experiments

In the first part of this chapter, we explained why some airlines were uncomfortable with implementing IFAM. Most proposals of new models use their own method to measure the revenue, thus positioning themselves in a way that makes them look better. Our approach was different: in every experiment, we assumed that IFAM was the best possible model to describe the reality, and our goal was to see whether using another method would prove the model wrong by its own measure. The strategy we used was the same every time: assume an input deviated from the reality, and measure the effect. We believe that our analysis is thus as unbiased as it can be.

Depending on the beliefs of the reader, we may or may have not succeeded to convince that it is desirable to fix the behavior of IFAM. Our experiments would obviously be best performed by the airlines themselves. We tried to give the elements to make an educated guess. Hopefully, this work will convince engineers and forecasters to test extensively their own data, and will lead to a questioning of industry practices. Collaboration and publication of data is desirable if we want to see convincing academic work in this area.

With the work we performed here, we believe that some elements require to be improved in IFAM:

Recapture Beyond all the effort we made to show that recapture may not be the best possible way to model the behavior of passengers, this work has been done in close collaboration of the Revenue Management department of Amadeus, who opposed the concept. We believe it is worth looking for an alternative.

Demand Granularity There is a growing interest in considering demand as a nested concept. Grouping different demands together will lead to better accuracy in measures, and a greater confidence in the assignment yielded by the models.

The next part of this chapter presents a model that tries to make use of all the lessons learnt with this work on IFAM, and to bring a better

representation of the passenger revenue.

4.2 Market Driven Fleet Assignment

In this section, we present a new Fleet Assignment model. This model has been conceived with a series of requirement in mind, guided with our experience in testing IFAM in various situations. It is a generalization of the no-recapture version of IFAM, and its conception is influenced by Revenue Management.

We start with a summary of our requirements, then we present the model. We give elements of comparison between this model, FAM and IFAM. We conclude with examples to illustrate the expressiveness that can be reached using it.

4.2.1 Desirable Features of a modern Fleet Assignment Model

When we started designing a new model for Fleet Assignment, we had a set of requirements in mind. These requirements came naturally, from the desire to improve the model. We list them here.

Model Recapture differently

Recapture rates are an unclear concept. They are hard to measure, and they are confusing. During the course of this thesis, a joint effort has been made with the Revenue Management team of Amadeus to gather data for building IFAM instances. After presenting them with the concept of recapture, their feedback was clearly negative, and we were strongly advised to drop it. Since Revenue Management is the discipline that now drives optimization of the passengers flow, listening to opinion of experts of this domain made sense to us. Modeling recapture differently has thus become one of our primary objectives.

Be able to deal with only Part of the Demand Information

When gathering data to build IFAM instances, it became clear that the kind of information IFAM requires can push the data mining very far. This task is obviously not made easy by the fact this data is part of critical systems that absolutely need to work for the airlines to do their job. As scientists, we can cope with these problems with a lot of post analysis. But at the airline level, moving to an IFAM-based system requires a complete refactoring of major components of the IT system. Refactoring has a cost, and not every airline will risk it unless they are convinced that it is absolutely necessary. An important goal we set to ourselves was thus to be able to deal with any

source of data the airline would be able to provide, and to make a model that is adaptable.

Minimize disruption

The same reasons for which we wanted to build a model that could adapt to any kind of demand input also made us want to build a model that could be used as soon as possible. Airline processes are numerous, nested, interdependent. Therefore, for a solution to be quickly adopted, it has to fit in the general scheme with the least possible amount of modifications. This does not mean that the model will necessarily use exactly the same inputs, but it means that it has to be at least backward compatible with what is currently used.

With all these requirements in mind, we designed the model that is about to be presented.

4.2.2 Redefining the Market Concept to match mainstream Economics

Traditionally, in the processes linked to schedule planning, markets are defined as pair of cities. While it is true that this definition corresponds to sets of customers who are potentially interested in buying tickets, we find this definition restrictive. The Cambridge dictionary defines a market as *the people who might want to buy something.* Should we redefine the concept of market in complete accordance with this definition, we believe that the definition of the scheduling term **market** should not be reduced to city pairs.

We propose to widen the definition of market. The unit of good sold by airlines is the itinerary. The general term **market**, with no further precision, would thus concern the set of every possible itinerary that can be bought. To only refer to a subset of these itineraries, we can use a criterion: For example, to refer to market as in the common scheduling literature, we could talk about "The market of the itineraries of the city pair A-B". This would allow to define other markets, such as "The market of business-class", or "The market of Australia". Such a definition would match mainstream economics and make the scheduling discipline more accessible to others. More generally, we may simply define a market as a set of itineraries.

Starting from here, and for the rest of this thesis, we use the term market as follows: A market m is any set of itineraries taken together. It is characterized by the unordered list of itineraries it comprises.

4.2.3 The nested Nature of Demand

It has to be noted that Revenue Management models customer behavior differently. One of the major differences is that potential customers for a given product are also subject to buy other flavours of the same product, and will pick their preferred match depending on the availability and their willingness to pay. In contrast, to model the fact that passengers are candidates for buying several itineraries, IFAM uses recapture. There is a probability of recapture between itineraries, and some percentage of the passengers will accept an alternative itinerary if their primary choice is unavailable.

We believe that there is a way to model passengers as candidates for several itineraries at the same time, by grouping demands together. Not only it would match more accurately the kind of things Revenue Management is doing, but it would also allow to use the property that itineraries grouped together are less variable.

4.2.4 Market Driven Fleet Assignment

The idea of Market Driven Fleet Assignment naturally arose from all the ideas exposed previously:

- Itinerary demands seem too uncertain to be usable;

- There is an interest in grouping demands together;

- Recapture would be best expressed as nested in different groups of itineraries.

As a consequence, why not make demand a variable of the model? We would be allowed to use it in equations, to constrain it, and to express freely any knowledge known about it. To give an example, here is what can be done regarding a leg: Rather than specifying all the demands of the itinerary running across the leg separately, we could consider the leg as a market, and constrain the demands on this leg to match a given value. If the demand forecast for the leg l is D_l, it would translate to the constraint.

$$\sum_{i \ni l} d_i = D_l$$

This formulation has the advantage of not forcing us to specify the values of each d_i, and to let the model decide it for us. However, we can see potential problems of feasibility. In this interest, we allow to specify the constraints using a lower and an upper bound. The previous constraint thus translates to:

$$D_l^{\min} \leq \sum_{i \ni l} d_i \leq D_l^{\max}$$

Actually, there is no reason to force any geographical concept into the technique we just introduced. There might be an interest in measuring the demand of any market. We can define more general constraints: given a market m, which is a set of itineraries, we define the following constraint:

$$D_m^{\min} \leq \sum_{i \in m} d_i \leq D_m^{\max} \tag{4.1}$$

Not only this notation is intuitively linked to the notions of market in mainstream economics, but it has a great power of expressiveness. Any group of itineraries can be considered, leaving room for any possible knowledge the airline may know about the demand on their network.

Once again, one could stop here, and use only this kind of constraints to define MDFAM. In fact, early versions of this model included only these constraints. But this would restrict the concept of constraining the demand as part of the fleet assignment. If we start to admit demand as a variable, there is actually no reason to restrict the type of constraints to apply on it. For example, analysis could lead to discover that the demands of two itineraries are always linearly correlated, leading to constraints of the form:

$$\beta - \mathcal{E} \leq D_1 + \alpha D_2 \leq \beta + \mathcal{E}$$

So, in fact, any constraint can be applied to the demand, and the most general form for demand constraints is:

$$\sum_{i \in I} \alpha_i D_i \leq \beta$$

We call the constraints on the demand the market constraints, and we index these constraints in a set M.

Relationship Between MDFAM and Recapture

It was brought to our attention that in constraints such as

$$\sum_{i \in m} d_i = D_m$$

look like they would probably well be expressed with traditional IFAM, assuming a recapture of 100% between itineraries of the market m. Indeed, recapture rates mark the ability of the airline to redirect demand to other itineraries. A recapture rate of 100% means that any attempt to spill passengers towards the alternative itinerary will succeed, which surely happens when demands move freely between the itineraries of the market.

However, this consideration does not stand if one starts to consider that markets can intersect. For example, MDFAM allows constraints of the form

$$d_i + d_j = D_{i,j}$$

. If we assume such a constraint for every pair of itinerary in the network, it only constrains more the MDFAM formulation. On the contrary, assuming that every recapture rates of the network are equal to 1 enables us to fully spill the demand from any itinerary towards any other itinerary, making the demand less constrained.

As a conclusion, while MDFAM allows us to consider intersecting markets, IFAM would not allow to do exactly the same using recapture rates.

4.2.5 Formal Presentation of the model

Data   The notations of IFAM are still in use. Additionally, we note M the set of market constraints (indexed by m).

Variables   The same fleet assignment variables as in FAM and IFAM are used:

- $x_{t,l} = \begin{cases} 1 & \text{if aircraft type } t \text{ is assigned to leg } l \\ 0 & \text{otherwise.} \end{cases}$

- $y_{t,g}$: the number of aircraft of type t waiting on the ground arc g.

Rather than using spill variables and recapture rates, booking variables are used.

- b_i: the number of passengers booking the itinerary i.

The unconstrained demand is not a data anymore, but a variable to be decided from market constraints.

- d_i: the unconstrained demand on the itinerary i.

Model

$$\max \sum_{i \in I} \text{Fare}_i b_i - \sum_{l \in L} \sum_{t \in T} c_{t,l} x_{t,l} \tag{4.2}$$

$$s.t. \sum_{t \in T} x_{t,l} = 1 \qquad\qquad \forall l \in L \tag{4.3}$$

$$\sum_{l \in I(n)} x_{t,l} + y_{t,g_i(n)} = \sum_{l \in O(n)} x_{t,l} + y_{t,g_o(n)} \qquad\qquad \forall n \in N \tag{4.4}$$

$$\sum_{g \ni CT} y_{t,g} + \sum_{l \ni CT} x_{t,l} \le |t| \qquad\qquad \forall t \in T \tag{4.5}$$

$$\sum_{i \ni (l,c)} b_i \le \sum_{t \in T} x_{t,l} \text{Cap}_{t,c} \qquad\qquad \forall l \in L, \forall c \in C \tag{4.6}$$

$$\sum_{i \in I} \alpha_i^m d_i \le \beta^m \qquad\qquad \forall m \in M \tag{4.7}$$

$$b_i \le d_i \qquad\qquad \forall i \in I \tag{4.8}$$

$$x_{t,l} \in \{0,1\}, y_{t,g} \ge 0, d_i \ge 0, b_i \ge 0$$

This translates to:

- (4.2) The objective is to maximize the profit.

- (4.3) Exactly one aircraft type must be assigned to each leg.

- (4.4) The flow of aircraft is conserved at each node of the network.

- (4.5) At the count time, the number of aircraft of every type is conserved.

- (4.6) The cabin capacity are respected.

- (4.7) The demands must respect the market constraints.

- (4.8) The bookings are consistent with the demand.

The market constraints (4.6) can be freely specified. Any type of demand constraint that can be linearly expressed will fit into this model. Their number is undetermined, but the idea is to fit enough constraints here for the demand to be tightly constrained, otherwise, the model will optimize and overestimate the revenue of the fleet assignment. Possible examples of market constraints are discussed in the next sections.

4.2.6 Analysis of MDFAM

MDFAM is a generalization of the no-recapture version of IFAM:
To see it, we define define in MDFAM the variables

$$s_i = d_i - b_i.$$

standing for the number of passengers spilled on itinerary i.

We proceed to the following change of variables:

$$b_i = d_i - s_i$$

After proper substitution, and turning the model in a minimization of the opposite of the objective, we obtain:

$$\min \sum_{l \in L} \sum_{t \in T} c_{t,l} x_{t,l} + \sum_{i \in I} \text{Fare}_i(s_i - d_i) \tag{4.9}$$

$$s.t. \sum_{t \in T} x_{t,l} = 1 \qquad \forall l \in L \qquad (4.10)$$

$$\sum_{l \in I(n)} x_{t,l} + y_{t,g_i(n)} = \sum_{l \in O(n)} x_{t,l} + y_{t,g_o(n)} \qquad \forall n \in N \qquad (4.11)$$

$$\sum_{g \ni CT} y_{t,g} + \sum_{l \ni CT} x_{t,l} \le |t| \qquad \forall t \in T \qquad (4.12)$$

$$\sum_{i \ni (l,c)} (d_i - s_i) \le \sum_{t \in T} x_{t,l} \text{Cap}_{t,c} \qquad \forall l \in L, \forall c \in C \qquad (4.13)$$

$$\sum_{i \in I} \alpha_i^m d_i \le \beta^m \qquad \forall m \in M \qquad (4.14)$$

$$s_i \le d_i \qquad \forall i \in I \qquad (4.15)$$

$$x_{t,l} \in \{0,1\}, y_{t,g} \ge 0, d_i \ge 0, s_i \ge 0$$

Put under this form, it is easy to verify that MDFAM is a generalization of the no-recapture version of IFAM. Let $(\beta_i)_{i \in I}$ be positive real numbers standing for itinerary demands. If we use the following market constraints:

$$d_i \le \beta_i$$

and

$$-d_i \le -\beta_i$$

for each $i \in I$, then it ensues that $d_i = \beta_i$ for each $i \in I$. The constant $\sum_{i \in I} Fare_i d_i$ can be taken out of the objective function, and the model becomes exactly an instance of IFAM, where the recapture rates are all null and thus, where there is only one spill variable per itinerary, denoting the spill towards the null itinerary (out of the network). Since $(\beta_i)_{i \in I}$ can be chosen arbitrarily, we can formulate any instance of IFAM with MDFAM, which proves the point.

Number of variables:

- FAM:

    - Assignment variables $x_{t,l} : |T| \times |L|$

- Ground variables y_{t_g} : $|T| \times |G|$
- **Total:** $|T| \times |L| + |T| \times |G|$

- IFAM:
  - Assignment variables $x_{t,l}$: $|T| \times |L|$
  - Ground variables y_{t_g} : $|T| \times |G|$
  - Spill variables $s_{i \to j}$: $|I|^2$
  - **Total:** $|T| \times |L| + |T| \times |G| + |I|^2$

- MDFAM:
  - Assignment variables $x_{t,l}$: $|T| \times |L|$
  - Ground variables y_{t_g} : $|T| \times |G|$
  - Booking variables b_i : $|I|$
  - Demand variables d_i : $|I|$
  - **Total:** $|T| \times |L| + |T| \times |G| + 2 * |I|$

The difference in terms of number of variables between IFAM and MDFAM is thus $|I|^2 - 2 * |I|$, which means that MDFAM has actually less variables than IFAM. This can be explained by the fact MDFAM actually models recapture as a group phenomenon and thus uses less variables to model it.

Number of constraints:

- FAM:
  - Leg cover: $|L|$
  - Aircraft flow: $|N| \times |T|$
  - Number of aircraft: $|T|$
  - **Total:** $|L| + |N| \times |T| + |T|$

- IFAM:
  - FAM constraints: $|L| + |N| \times |T| + |T|$
  - Cabin capacities: $|L| \times |C|$
  - Spill limits: $|I|$
  - **Total:** FAM $+ |L| \times |C| + |I|$

- MDFAM:
  - FAM constraints: $|L| + |N| \times |T| + |T|$
  - Cabin capacities: $|L| \times |C|$

- Booking limits: $|I|$
- Market constraints (User dependent): $|M|$
- **Total:** $\mathrm{FAM} + |L| \times |C| + |I| + |M|$

The difference between IFAM and MDFAM in terms of number of constraints is thus $|M|$, the number of market constraints.

4.2.7 Examples of Constraints

Here are some possibilities offered by MDFAM.

Matching a geographical Criterion
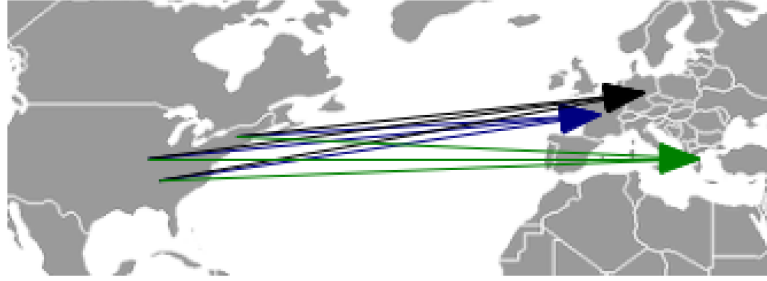
Figure 4.4: Itineraries to Europe: France (blue), Germany (black), Greece (green)

Mainstream economics will often refer to markets according to geographical criteria. For example, the European market would refer to the demand for all flights departing to Europe. Such a market would create constraints that would look as the one that follows. Let I_{Europe} be the set of itineraries that are going to Europe:

$$D_{\mathrm{Europe}}^{\min} \leq \sum_{i \in I_{\mathrm{Europe}}} d_i \leq D_{\mathrm{Europe}}^{\max}$$

Such a constraint may match a potentially large number of itineraries. The concept of Market Driven Fleet Assignment allows it and can include many various nested constraints. These constraints can be combined, and include constraints of various nature. For example, the European market can be split into sub-markets, like France, Germany, Greece, etc. Each of these sub-markets would lead to different sets of itineraries, and as many constraints:

$$D_{\mathrm{France}}^{\min} \leq \sum_{i \in I_{\mathrm{France}}} d_i \leq D_{\mathrm{France}}^{\max}$$

$$D_{\mathrm{Germany}}^{\min} \leq \sum_{i \in I_{\mathrm{Germany}}} d_i \leq D_{\mathrm{Germany}}^{\max}$$

$$D_{\text{Greece}}^{\min} \leq \sum_{i \in I_{\text{Greece}}} d_i \leq D_{\text{Greece}}^{\max}$$

Matching based on the Fare Class
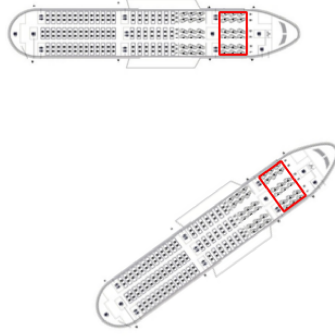
Figure 4.5: Matching based on the Fare class

It may also be interesting to look at passengers based on the fare they are willing to pay for their plane ticket. Some persons are after the quality of service and may prefer traveling in first class over other criteria. If we decide to look at the market of the persons who want to travel in first class over a particular set of flights, we would be led to consider constraints such as:

$$D_{\text{First class}}^{\min} \leq \sum_{i \in I_{\text{First class}}} d_i \leq D_{\text{First class}}^{\max}$$

With this approach, we thus have an alternative manner to model recapture. The fact that some persons may not care about the itinerary they are traveling on make them appear as candidates for several itineraries.

Enforcing Correlations between Demands

When someone flies somewhere, it is more than likely that this person with return back home. We thus expect that observations of demand patterns on the flight network will lead to correlated demands between an itinerary and another that would go in the opposite direction. Let $i_1, \ldots, i_k$ be a set of itineraries going in one direction, and $i'_1, \ldots, i'_l$ a set of itineraries going in the opposite direction. We expect to see a correlation between $i_1, \ldots, i_k, i'_1, \ldots, i'_l$ such that $(d_{i_1} + \cdots + d_{i_k}) - (d_{i'_1} + \cdots + d_{i'_l}) = 0$, with some error margin ϵ. It could be enforced in MDFAM as:

$$-\epsilon \leq \sum_{j=1}^{k} d_{i_j} - \sum_{j=1}^{l} d_{i'_j} \leq \epsilon$$

Figure 4.6: Correlated Demands

Localized Match

Figure 4.7: Matching by leg

Another possibility is to look at smaller markets, such as single legs. Airlines in the traditional FAM approach, have gotten good at measuring demands over legs. Given a leg l, let I_l be the set of itineraries using l. Such constraints would be expressed as:

$$D_l^{\min} \leq \sum_{i \in I_l} d_i \leq D_l^{\max}$$

The nice feature of this approach is that it would allow traditional FAM models, that already use these leg-based demands, to be plugged in MDFAM directly. Thus, MDFAM virtually requires no work from a company using FAM. The same could be said about IFAM. One could use MDFAM on single itineraries, and constrain their demands individually.

4.2.8 Conclusion

In this chapter, we exposed the scientific approach that led us to create a new Fleet Assignment model. This model achieves several features:

- it is a generalization of the no-recapture version of IFAM, and as such, it cannot produce worse results than the former.

- it is adaptable to any possible input of demand, provided this input can be expressed as linear constraints.

- it can be directly used by both a company using FAM and a company using IFAM.

The possibilities are vast. So what are correct Market Constraints? It seems obvious that, because demands have become decision variables, the model will optimize them. As a consequence, to avoid inaccuracies, it seems important to find the right types of Market Constraints to put into MDFAM to obtain the best results.

This is the topic of the next chapter.

Chapter 5

Comparison With Traditional Models and Tuning for Profit

In the previous chapter, we presented MDFAM, a flexible model to build fleet assignments with demand information of multiple forms. This chapter is dedicated to determining what are the best possible Market Constraint for generating the most profit with this model. We compare MDFAM with the two other major models of the literature, FAM and IFAM, in a series of experiments based on real life data.

We start this chapter with an introductory example to illustrate the benefit of MDFAM. In a second part, we show how, using correlations discovered through data analysis, MDFAM can outperform other models on real-life instances. We explain how the data was obtained and post-processed. Finally, we use 2-fold cross validation in order to compare MDFAM with alternative Fleet Assignment Models. A variety of test cases are used to benchmark MDFAM, with various degrees of success. We conclude about the efficiencies of the different models.

5.1   Introductory Example

As opposed to IFAM, which is a model that takes average demand as a given input, MDFAM uses any information about demand that can be expressed with linear constraints. Demands are treated as variables, and as such, are optimized. As a consequence, MDFAM can be seen as a version of IFAM that computes the best case demand scenario to work with.

It may seem unlikely that the best case scenario actually happens in reality. However the following example shows that in some cases, with good inputs, MDFAM is preferable to IFAM.

5.1.1 Description of the Situation

We consider the following network of two flight legs. These flights are both starting from the same city, but are serving unrelated destinations. Because the destinations are completely different, no passenger would consider swapping flights, and thus recapture is assumed to be null.

Flight	Origin	Destination
1	A	B
2	A	C

We also assume that there are only two possible itineraries in the flight network: flying on 1 or on 2. Both itineraries cost the same price to the passengers.

Itinerary	Flight	Fare
1	1	100
2	2	100

The company can assign three different types of aircraft to the flights.

Aircraft	Number of seats	Cost
80	80	4000
90	90	4500
100	100	4900

We assume that the demands follow a pre-determined pattern, and that only two equally likely demand scenarios may occur:

Scenario	Itinerary	Demand
1	1	100
	2	80
2	1	80
	2	100

A possible interpretation of this theoretical example could be that:

- On both flights, 80 customers travel on a regular basis, and are used to book long in advance;

- There are 20 other persons who share a common reason to travel, and know which leg they are interested in at the last minute. For example, they could be invited to a meeting that randomly takes place in B or C.

We are now going to consider two different cases: in the first case, the airline will be performing the fleet assignment using IFAM, and in the second case, the airline will be performing the fleet assignment using MDFAM. Once an assignment is decided, it will be applied all over the season: assuming the Passenger Mix Model applies, we will compute the profit of this assignment on each of the demand scenarios, which will enable us to decide which assignment is, on average, the most profitable.

5.1.2 Average Performance of the IFAM Solution

We assume that the airline is getting started with IFAM. They perform data analysis, and find out what the average demands are. They then assume the average demand scenario to assign their fleet, and use the resulting assignment over the season.

Itinerary	Demand
1	90
2	90

Assuming this average scenario, because any of the other solutions would lead to either have 10 empty seats or spill 10 passengers and, in both cases, being less profitable, the optimal solution is:

Flight	Assigned aircraft
1	90
2	90

The assignment computed on the average scenario is now used over the season. We can compute the average performance of IFAM easily: the cost of the network remains the same, so we just have to compute the revenue made by the airline with the two different demand scenarios. These scenarios are symmetric: on one flight, the demand is 100, and on the other flight, the demand is 80. The number of passengers who travel on the first flight is 90 (and 10 persons are spilled), and the number of passengers who travel on the second flight is 80 (and there are 10 empty seats). Thus, the profit is:

$$(80 + 90) * 100 - 2 * 4500 = 8000$$

5.1.3 Average Performance of the MDFAM Solution

We now assume that a good data analysis is performed. This analysis finds out that the demand always satisfies the following constraints:

$$80 \leq d_1 \leq 100$$
$$80 \leq d_2 \leq 100$$
$$d_1 + d_2 = 180$$

Being optimistic, MDFAM will assume the best case demand scenario. As the optimal solution of a linear programming problem, the demand saturates one of the constraints. This results in assuming the following situation (which turns out to be true half of the time):

Itinerary	Demand
1	100
2	80

Because any other assignment would either spill passengers, or have empty seats and be more expensive, the optimal MFAM solution is:

Flight	Assigned aircraft
1	100
2	80

We now compute the performance of the obtained assignment over the season. Again, the cost of the network remains the same. In one case, all the 180 passengers can use the network, and in the other case, only 160 persons use the network. On average, the same number of passengers use the network. However, the assignment is cheaper, since we now pay $4000 + 4900 = 8900$. On average, the revenue of the fleet assignment is thus:

$$(160 + 180)/2 * 100 - (4000 + 4900) = 8100$$

5.1.4 Conclusion

We can see that, on this particular case, the profit yielded by an MDFAM fleet assignment can be higher on average than the profit of an IFAM fleet assignment, even though the IFAM fleet assignment was computed based on a perfectly accurate forecast of the average demand. The choice made by MDFAM is better because the profit made over the better scenarios compensates the loss over the bad scenarios.

Though our example we may sound silly, we believe that the real world has comparable situations, where demand correlations could be detected. For example, we expect passengers who take a flight to come back, and thus the demand of an itinerary to be correlated with the demand in the opposite direction. Through data analysis and with the knowledge of experts, many

such demand relationships could be established, and MDFAM is meant to exploit them.

5.2 MDFAM Applied to Realistic Situations

In this section, we compare the models in realistic situations. We designed a simulation framework, where an airline decides how to assign its fleet using FAM, IFAM, or MDFAM. The obtained fleet assignment is then applied over a season, providing the average profit of the model. Instead of using a random demand generator, which would oblige us to make possibly biased assumptions upon demand, we base our simulation on real-life data.

5.2.1 Simulation Overview

Evaluating a model essentially consists in simulating two steps:

1. **Forecasting the best fleet assignment**: The airline use historical data to build an instance for the model of their choice, and solve it;

2. **Applying the obtained fleet assignment**: They run the fleet assignment over a season. Each day yields particular demands. They collect the overall profit.

For both of these steps, we assume knowing realistic demand scenarios. These scenarios must refer to the same network, and each of them must provide a vector of demand per itinerary. In other words, the scenarios provide realizations of the demand. In the first step, when forecasting, these scenarios are interpreted as past events of demand an are used to build (to forecast) a fleet assignment instance. In the second step, when applying the fleet assignment, they are interpreted as events to occur in the season, and are used to evaluate the profit of the fleet assignment.

Figure 5.1 describes the protocol overview: assuming that we collected the demands over the network during a season, we obtain a sample of demand scenarios. A subsample of these scenarios (green) is selected in order to predict the best fleet assignment. Then, once this fleet assignment is computed, another subsample (blue) is selected to test it. We make no further assumptions on the subsamples for the moment: they could contain the full sample, or they could intersect (but not be empty).
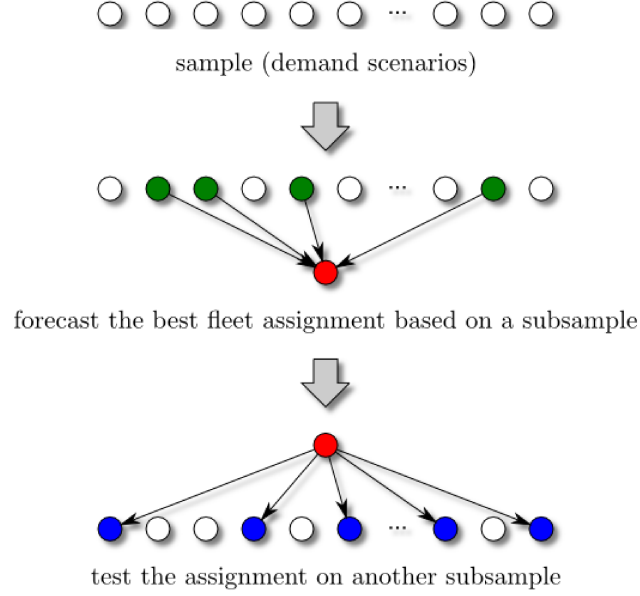
Figure 5.1: Simulation Overview

We are now going to explain how we implement the forecasting step and the test step, assuming the demand scenarios are already provided. Then, we are going to explain how the demand scenarios have been obtained. We will conclude with some results.

5.2.2 Forecast Simulation: Constructing Instances from Demand Scenarios

In this subsection, we assume being provided with a set of demand scenarios over the same flight network. These scenarios are assumed to represent past demand realisations on this network. We want to simulate the fact that an airline is performing data analysis over these scenarios and build an instance for their preferred model. We now explain how to generate FAM, IFAM and MDFAM instances from these demand scenarios.

IFAM Instances

To create an IFAM instance given a set of scenarios, we compute, for every itinerary, the mean demand over every scenario. Let S be the set of demand scenarios, indexed by s. We note $d_i(s)$ the demand on the itinerary i in the scenario s. The algorithm for generating an IFAM instance from various

demand scenarios is:

1: **for** $i \in$ itineraries **do**
2:     $d_i \leftarrow \text{mean}(\{d_i(s) | s \in S\})$
3: **end for**

Algorithm 4: Creating an IFAM instance from demand scenarios

This yields an IFAM instance that is assumed to be representative of the scenarios.

FAM Instances

The process to create FAM instances is based on the formula described in 3.2.1. Recall that FAM uses leg-based costs, which are computed as follows:

$$c_{t,l} = \underbrace{c_{t,l}^{\text{op}}}_{\text{operating costs}} + \underbrace{c_{t,l}^{\text{sp}}}_{\text{lost revenue}}$$

Let $Q_{l,c}$ be the demand on the leg cabin (l, c). The formula of 3.2.1 says that the spill cost of assigning the aircraft type t to the leg l is:

$$C_{t,l}^{\text{sp}} = \sum_{c \in C} \text{Fare}_{l,c}^{\text{sp}} \max(0, Q_{l,c} - \text{Cap}_{t,c})$$

The following spill fare is used:

$$\text{Fare}_{l,c}^{\text{sp}} \simeq 0.7 \text{ average fare on } (l, c)$$

We denote $Q_{l,c}(s)$ the demand on the leg cabin l, c in the scenario s. We compute the spill costs based on the average $Q_{l,c}$ over the season. The algorithm for generating a FAM instance from various demand scenarios is:

1: **for** $l \in$ legs **do**
2:     **for** $t \in$ aircraft types **do**
3:        $C_{t,l}^{\text{sp}} \leftarrow \sum_{c \in C} \text{Fare}_{l,c}^{\text{sp}} \max(0, (\text{mean}(\{Q_{l,c}(s) | s \in S\}) - \text{Cap}_{t,c})$
4:     **end for**
5: **end for**

Algorithm 5: Creating a FAM instance from demand scenarios

We combine the obtained spill costs with the operating costs to obtain a FAM instance that describes the set of scenarios.

MDFAM Instances

The process we use to generate MDFAM instances is slightly more sophisticated. MDFAM requires constraints on the demand as an input. We thus need a way to build these constraints from demand scenarios.

The general form of a Market Constraint is

$$\sum_{i \in I} \alpha_i d_i \leq \beta$$

where I denotes the set of itineraries over the network, and d_i denotes the demand over the itinerary i. We call **direction** the vector of coefficients $(\alpha_i)_{i \in I}$. Given a direction, the linear sum $\sum_{i \in I} \alpha_i d_i$, is called **the demand in the direction** $(\alpha_i)_{i \in I}$.

The following procedure describes how Market Constraint can be iteratively generated from an input set of scenarios:

1. We choose a direction $(\alpha_i)_{i \in I}$ for measuring the demands.

2. We measure the demands according to this direction: for every scenario s, we look at the value of $\sum_{i \in I} \alpha_i d_i(s)$

3. We compute (μ, σ) the mean and the variance of the sample. We then output the constraint $\sum_{i \in I} \alpha_i d_i \in [\beta^{\min}, \beta^{\max}]$, with $\beta_{\min} = \mu - 0.13\sigma$ and $\beta_{\max} = \mu + 0.13\sigma$.

If $\sum_{i \in I} \alpha_i d_i$ was following a normal distribution, $[\beta^{\min}, \beta^{\max}]$ would correspond to the 10% confidence interval of $\sum_{i \in I} \alpha_i d_i$. This interval has been determined empirically, and appears to give good results. This process is repeated for as many directions as necessary. The directions can be arbitrary.

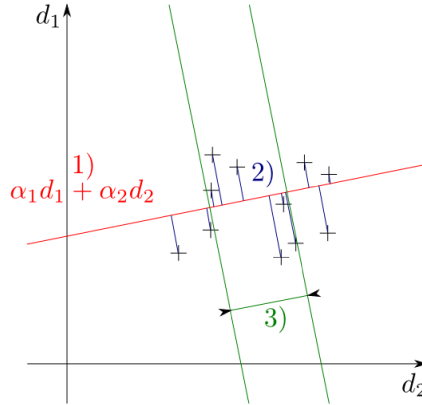Figure 5.2: Generation of a Market Constraint

Figure 5.2 gives a graphical interpretation of this process for 2 demands: occurrences of the demands of the itineraries 1 and 2 are plotted, such that 1 point on the graph represents one occurrence. The red line represents the direction chosen, and the green constraints are deduced from the "confidence interval".

We test different direction types.

73

Possible directions for MDFAM demand Constraints

In this part, we describe the types of directions we have tested for building the Market constraints of MDFAM.

Itinerary-based demand constraints   The simplest direction to use in MDFAM is itinerary-based. It consists in directly mapping itinerary demands to constraints. Given an itinerary i_0, the direction is expressed as $(\alpha_i)_{i \in \text{itineraries}}$, such that

$$\alpha_i = \begin{cases} 1 & \text{if } i = i_0 \\ 0 & \text{otherwise.} \end{cases}$$

Leg-based demand constraints   Another direction that naturally comes to mind for measuring demands is an aggregation of the itinerary demands by legs. Given a leg l of the network, we measure the variable $\sum_{i \ni l} d_i$. The direction can thus be expressed as $(\alpha_i)_{i \in \text{itineraries}}$, such that:

$$\alpha_i = \begin{cases} 1 & \text{if itinerary } i \text{ uses leg } l \\ 0 & \text{otherwise.} \end{cases}$$

Origin and destination-based demand constraints   Another direction, also geographically-based, is the origin and destination based direction. It consists in aggregating the demands of the itineraries that shares the same origin and destination. Given the origin airport O and the destination airport D, this direction is expressed as:

$$\alpha_i = \begin{cases} 1 & \text{if itinerary } i \text{ has the origin } O \text{ and the destination } D \\ 0 & \text{otherwise.} \end{cases}$$

Price-based demand constraints   The optimistic nature of the model makes it likely to maximize the demands of itineraries with high prices. As a consequence, to limit this phenomenon, we constrain the demands of the itineraries with high fares to be limited. We construct these directions such that, given two fares $f_{\min}$ and $f_{\max}$, the itineraries that have a fare between $f_{\min}$ and $f_{\max}$ are grouped together:

$$\alpha_i = \begin{cases} 1 & \text{if itinerary } i \text{ has a fare in the interval } [f_{\min}, f_{\max}] \\ 0 & \text{otherwise.} \end{cases}$$

To determine the bounds of the intervals, we use the following process: we compute the average fare on the network, and order the itineraries by fare. We then group itineraries together such that the average number of

persons in one group multiplied by the average fare of the group are worth the value of 20 times the average fare. This way, itineraries with high fares constitute small groups, while itineraries with low fares are rather large.

Principal Component Analysis Constraints   Principal Component Analysis is a process to determine which parts of a dataset are the most variable. Given a sample, it determines the directions in which the sample varies the most. Mathematically, the procedure determines an orthogonal transformation to convert a set of linearly correlated variables into a set of linearly uncorrelated variables, called components. In the output, the components are ordered from the most variable to the least variable. A comprehensive decription of the process can be found in Jolliffe (2005).

We use an external library to run a principal component analysis on the scenarios of demands, in order to determine the directions in which the demand varies the most. The output of this analysis is a set of orthonormal directions in which the demands vary the most. We measure the demand in these directions exactly.

In MDFAM, we are interested in finding relations between demands with little variability. As a consequence, the interesting components would need to be given in reverse order. There are 26 weeks in a season, and thousands of dimensions in the demand. As a consequence, we are actually left with a number of constraints that is very small, representing all the variability of the sample. This orthonormal basis is then completed through a Graham-Schmidt process. Since all the variability is contained in the 26 first components, the subsequent vectors of the basis are directions where there is absolutely no variation, constituting excellent constraints for MDFAM.
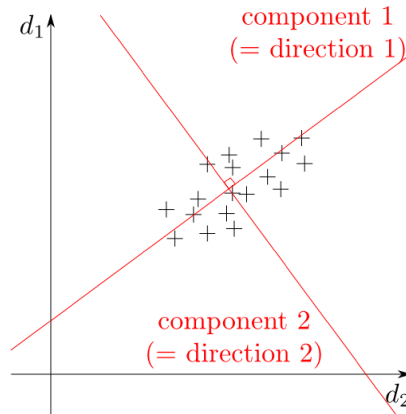

Figure 5.3: Directions from principal component analysis

Figure 5.3 illustrates the process of choosing directions from the com-

ponent analysis. Assuming several realisations of the demand vector (d_1, d_2), the principal component analysis returns in order (component 1, component 2), such that the component 1 is the direction where the observed variation is the largest, and component 2 is the second direction by this criteria, while being perpendicular to component 1. These directions are then used to build Market constraints.

5.2.3 Evaluation Method: Applying a Fleet Assignment to a Set of Demand Scenarios

In this section, we explain how we evaluate the efficiency of a fleet assignment. We still assume being provided with demand scenarios, but we interpret them differently, assuming they are future events of the season.

This process is rather straightforward: for each scenario, we use the passenger mix model (PMM) to compute the attainable contribution of the assigned network. We subtract the operating cost of the fleet assignment, in order to obtain the profit. We compare the cumulated profits of the fleet assignments obtained with each different model.

We now explain how the demand scenarios were obtained.

5.2.4 Obtaining the Demand Scenarios

This section explains how we obtained the demand scenarios used in our simulation framework. Our data source consists in the daily bookings and cancellations on the flights of a major airline. Several processing steps are then applied. We list them here.

Aggregation of daily Bookings

The raw data consists in single transactions, which may be either bookings or cancellations. These transactions, at the time they are made, refer to flights that have not yet been operated (up to one year prior to the event).

This information is aggregated, so that we know the total number of bookings on the various itineraries of the network, for a given day. To do so, it is necessary to go through the previous year of bookings and cancellations. The transactions are filtered with the departure flight date of the targeted day, and aggregated at the itinerary level. We end up, for every day of a season, with a vector of number of bookings per itinerary. For each itinerary, the cabin information is known, and we consider two itineraries using the same flight legs but different cabins to be different. The fare information may also be given, but is not always present.

In order to obtain several instances with consistent networks and demands, we choose several typical parts of the week to repeat this operation,

76

and we separate winter from summer. This separation is done according to the IATA definition, using the dates of Daylight Saving Time (DST) introduction in European Union countries as boundaries. We are left the winter and summer versions of Sundays and Tuesdays. In order to make larger instances, couple of days are also considered: Saturday-Sunday and Tuesday-Wednesday.

We compare our results with public yearly traffic figures published by the airline. Over the whole year, the claimed figures are 10% higher than our measures, which suggests missing data in our source. We adjust by artificially adding 10% to our results.

Deducing daily Schedules from the Bookings

For the sake of simplicity, the schedule information is not part of the input, since it can be deduced from the bookings: any flight booked must necessarily be part of the schedule. Furthermore, using a single source of data avoids mismatches between bookings and schedule. This method prevents us to detect flights with no bookings, but since they don't happen in practice, it remains reliable.

One major issue resides in the fact that even when only considering the same day of the week, the flight network is not consistently the same across the season. In practice, it is perfectly normal that we observe a modified schedule, and it can be perceived as a sign of the flexibility of the airline. But for our experiment, this is a problem, because to simulate that the same fleet assignment is applied over a season, we need the itinerary demands to remain on the same network. In order to circumvent this problem, we compute an average network of the company. Our approach is based on frequency analysis. We sort the legs by number of occurrences in each schedule and keep the most frequent legs.

The size of the resulting schedule is determined by the average size of the schedule over the season (in terms of number of legs). If the process results in an infeasible schedule, caused by an insufficient number of aircraft, we reduce this size. In practice, the reduction needed is never more than 5%.

The bookings are then adapted to this network on every scenario: itineraries using legs outside of the network are split to keep only the part that belongs to the average network.

Collecting, converting and adjusting the Fares

In our raw data, the fares are given in the currency of the buyer. A first step is thus to convert the fares to a reference currency. This conversion is operated using rates adapted to the date of the transaction: some currencies vary a lot during the season and we find that it is preferable to take this phenomenon into account.

77

Then, the average fare over the season is computed for every itinerary. In the rare cases when all the fares are missing for a given itinerary, a fare for an itinerary that shares the same origin and destination, irrespectively of the connection station, is used.

Deducing the Demands out of the Bookings

The process to deduce the demands from the bookings is rather straightforward: we scale everything up (again) by 10%. Obviously, this introduces a bias: we should only increase the demands of itineraries operating on saturated legs. However, we believe that, even though we can probably achieve a better accuracy, we are not seeking the perfect input. The idea of the experiment is to taste what goes on, not to reach perfection. In order to keep the number of transformations small, we do not try to push the extrapolation further.

As for the recapture rates, we simply ignore them, and we consider that there is no recapture in our scenarios (the main reasons for this choice being that we don't know any way for collecting them, and that we believe the impact of adding them would not be significant).

Fleet and operating Costs

The fleet of the airline is public and was obtained from their website. The operating costs are computed using publicly available hourly rates [1].

In order to make the profits of the instances as realistic as possible, we apply a second transformation, which goal is to match the figures with the actual profit reported by the airline.

Public figures indicate that the net income of the airline is about 3% of their revenue. We also have, from our own data, an estimation of this revenue. However, because the net income includes every costs, we have to take crew costs out of the equation. We do this very simply: according to companies specialized in crew optimization [2], on average, crew costs represent 30% of the total costs of an airline. As a consequence, the operating costs are linearly scaled so that, in the optimal IFAM schedule of the average instance, the operating costs are roughly around 67% of the revenue generated by the bookings.

Data Set

The major characteristics of the data set are summarized in the following table. Each instance represents a season (26 weeks) of demands. Along with the main characteristics of the instances (number of legs, number of

---

[1]source: https://www.conklindd.com/CDALibrary/ACCostSummary.aspx

[2]source: http://www.theoptimacorporation.com/airline

itineraries), we give the average itinerary and leg demands over the season. We also give the average standard deviation of these demands, such that each demand is given as an ordered pair (average mean, average standard deviation).

Day	Season	Legs	Itineraries	Demand per leg	Per itinerary
sa-su	summer	420	4047	(87.4, 60.7)	(13.6, 6.5)
sa-su	winter	400	3993	(54.5, 53.2)	(10.9, 5.4)
su	summer	224	2275	(89.6, 59.5)	(13.3, 6.2)
su	winter	222	2284	(55.6, 58.2)	(11.4, 5.8)
tu	summer	260	2630	(71.1, 46)	(10.9, 4.7)
tu	winter	246	2386	(45.9, 43.4)	(9.8, 4.4)
tu-we	summer	526	5427	(72.4, 47.4)	(11.1, 4.7)
tu-we	winter	521	4729	(45.3, 42.6)	(9.9, 4.5)

We can see that the standard deviation is reaching up to 60 persons per leg. This implies that some of the itinerary demands variations add up.

5.2.5   Application of MDFAM over the Set of Instances

We run our simulation on every instances, using the aforementioned process to generate instances of FAM, IFAM, MDFAM. The MDFAM instance is generated using the combined effect of all the constraint types previously mentioned, the separate effects being the object of a study in the next section.

As described in Figure 5.4, we use the full sample for forecasting and testing each fleet assignment. We solve these instances, and we measure the revenue of the fleet assignments obtained over the season. The results are summarized in the following table. The value between parenthesis is the relative deviation to the IFAM profit.
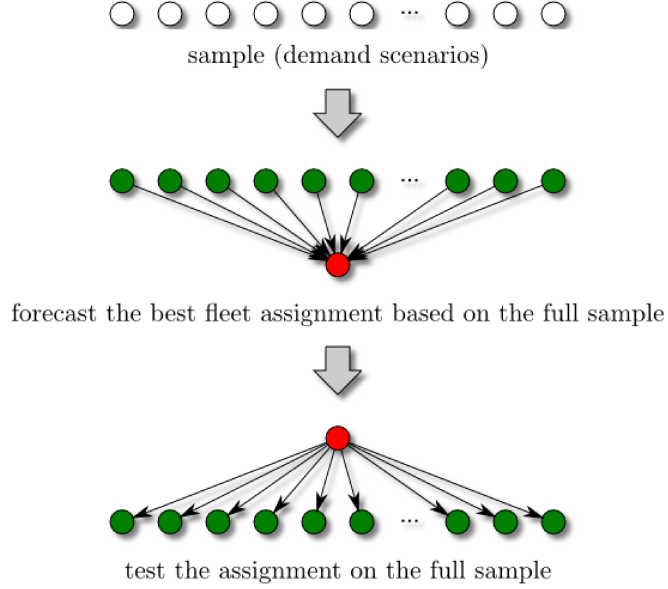
Figure 5.4: Protocol Using the Full Sample

Day	Season	IFAM	FAM	MDFAM
sa-su	summer	2296999	1627772 (-29.13%)	2346989 (+2.17 %)
sa-su	winter	373199	29326 (-92.14%)	416448 (+11.58%)
su	summer	1048246	873928 (-16.62%)	1074634 (+2.51 %)
su	winter	209034	61380 (-70.63%)	232156 (+11.06%)
tu	summer	1339491	1288219 (-3.82 %)	1367657 (+2.10 %)
tu	winter	610759	569462 (-6.76 %)	642108 (+5.13 %)
tu-we	summer	2839544	2574805 (-9.32 %)	2846719 (+0.25 %)
tu-we	winter	1427071	1279817 (-10.31%)	1460123 (+2.31 %)

As we can see, FAM gives results that are consistently inferior to those of IFAM, with a profit that can drop down to 92% less. On the contrary, MDFAM performs consistently better. The revenue of the best possible fleet assignment can be increased by up to 11%, and on average the improvement is 4.6%.

Conclusion

We conclude from the previous experiment that with good knowledge of the implicit correlations of the demands over a flight network, it is possible to build fleet assignments with MDFAM that are better than those obtained

with IFAM. However, we are well aware that the correlations obtained from analyzing the data set are perfect (the same sample being used for predicting and for testing). Statisticians would say that our analysis might be "over-fitting" the data set. As a consequence, we decided to run another experiment to compare the models in uncertain context.

5.3 Comparing the Models in an uncertain Context

In the previous section, we showed that, with a good knowledge of the demand patterns over a season, we were able to produce MDFAM fleet assignments performing better than IFAM fleet assignments. However, such a degree of accuracy may be impossible to reach by forecasts. In order to know if the process proposed would fit in a more realistic context, we need to introduce uncertainty in the experiment.

Comparing models is a difficult task: the major issue is the fairness regarding the way to evaluate the performance of the models. Obviously, the choice of the performance evaluation function impacts the relative performance of the models. For example, if the revenue function of one of the models compared is used, this model gets an obvious advantage, because it is built to optimize based on this revenue function. Besides, most of the time, different models involve different inputs. Therefore, to be fair, the evaluation needs to assume the same amount of information for every model. We explain here how we addressed these issues.

5.3.1 Purpose of the Experiment

To describe our experimental protocol, we shall start with describing what we intended to test. Our goal is to compare models when demand forecasts are not 100% correct. Like in the previous chapter, we assume that, given perfect input, IFAM computes a perfect fleet assignment. More accurately, we assume that the Passenger Mix Model, given the demands on a fleeted network, provides the actual revenue of the fleet assignment. We thus want to assume a demand forecast, and test the efficiency of the fleet assignment when the demand is different from this forecast.

Criterion of Evaluation

The method we have chosen to measure the efficiency of a fleet assignment uses real-life data: We measure the demands of an airline network over a scheduling season. This yields different demand scenarios. Given a fleet assignment, each of these demand scenarios can be then applied, using the Passenger Mix Model (PMM) from IFAM. We obtain the revenue of the fleet

assignment for the given scenario. Doing this for the whole season, we can compute the average revenue of the fleet assignment over the season.

5.3.2   2-fold Cross-Validation

Cross-validation is a method normally used to verify a predictive model. It consists in making a prediction based on part of a dataset, and testing this prediction on a different part of the dataset. In the case of 2-fold cross validation, half of the dataset is randomly selected for making the prediction, and the other half is used to check this prediction. Then the roles of the sets are swapped.

In our case, we do not make a prediction, but we create a fleet assignment instance from half of the dataset, and we compute the revenue of the instance based on the other half of the dataset. The process is summarized by the following procedure:

1: $S = \{$demand scenarios$\}$
2: Randomly partition S in $S1$ and $S2$, two sets of equal size
3: Create I, fleet assignment instance from $S1$
4: Compute X the corresponding fleet assignment
5: Compute the average revenue of X on S2
6: Exchange the roles of $S1$ and $S2$

Algorithm 6: Performance evaluation of a Fleet Assignment model

Figure 5.5 explains how we adapt our simulation to the 2-fold cross validation: The only thing we have to modify is now to randomly select the forecast subsample as half of the initial sample, and then to test on the second subsample. The roles of the subsamples are then exchanged.
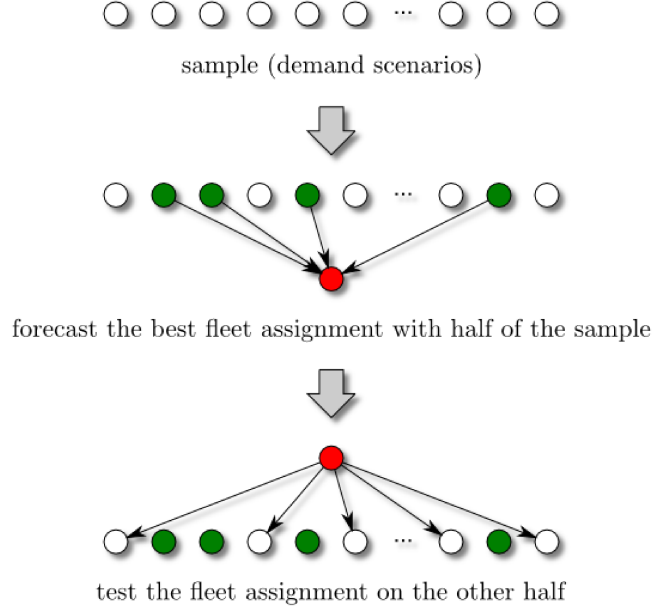
Figure 5.5: Protocol Adapted to 2-fold Cross Validation

5.3.3 Results

The experiment is run over all the set of scenarios previously mentioned, using the aforementioned procedure. In the following table, we summarize the average efficiencies of all the available methods, for all the instances. The profit is relative to IFAM, which consistently performs better than the other models.

Method		Profit			CPU time
	worse	average	best	median	
IFAM	0%	0%	0%	0%	53.3s
FAM	-71.77%	-15.32%	7.51%	-4.99%	32.5s
MDFAM - itinerary	-25.24%	-0.88%	6.99%	0.84%	37.2s
MDFAM - leg	-609.90%	-176.53%	-42.41%	-125.41%	42.1s
MDFAM - price	-468.66%	-171.18%	-45.15%	-143.15%	3068.4s
MDFAM - o/d	-763.22%	-198.24%	-39.74%	-150.28%	90.7s
MDFAM - PCA	-25.73%	-3.11%	5.88%	0.02%	45.6s
MDFAM - all combined	-15.44%	-0.67%	4.84%	0.84%	44.9s

There are several interesting conclusions to this experiment:

1. Though MDFAM comes close to the best profit (0.67% when all the methods are combined), we were not able to find correlations that would enable us to find better fleet assignments with this model. This suggests that correlations found from parts of a data set do not generalize well enough to the whole dataset for MDFAM to beat IFAM. This also suggests that good forecasters for MDFAM may be difficult to build.

2. MDFAM can be faster to solve than IFAM. All the tests that are run here are models entered raw in the solver memory, and no optimisation method has been applied. In particular, it is interesting to notice that the most interesting methods in terms of profit are equally interesting in terms of CPU time.

3. MDFAM can also be very slow to solve. This is particularly true for the price based method, where constraints are based on itineraries of the same price. Our guess here, is that the culprit is the symmetry introduced by such constraints, since all the demands of the same constraint yield a similar revenue.

4. The least constrained the demands are, the worse MDFAM performs. Not enough constraints will leave the model free to choose unrealistic demands, and the decisions taken are thus very bad.

5. FAM performs significantly worse than IFAM. To our knowledge, this is the first study to compare IFAM and FAM in the context of demand volatility. One might have expected that aggregating the demands would make the decisions better, but our results show that it is not the case.

5.3.4 Conclusion

In this chapter, we proposed an extensive comparative study of our model with the models of the literature. To our knowledge, this is the first study to involve as much data over such a long period of time. This is also one of the rare studies to feature a comparison with both of the two most widely used models in the industry. It required the implementation of both of these models, and an extensive amount of data mining.

We showed that the model we have built, under ideal conditions, performs consistently better than the others. The demonstration features both a simple example, and a test on real data. We also tested this model in a more uncertain context, with less success, showing that there was little chance that this technique would be usable if the accuracy obtained in the correlations is not high enough. Finally, these tests incidentally exposed results that may help to settle the FAM versus IFAM debate.

Despite the fact that this study reveals results based on a large amount of data, it is only based on the network on a single airline. To be able to draw conclusive results in the general case, one would need to repeat the same experience with others. This can only be done with the full cooperation of airlines. We hope that this document will bring enough attention on the subject in order for such a cooperation to occur in the future.

The next chapter concludes this thesis.

Chapter 6

Conclusion and Perspectives

Through this thesis, we addressed a variety of problems in the domain of Fleet Assignment.

In a first part, we experimented techniques to solve Fleet Assignment problems with a FAM solver, basing our approach on well known results. We showed that FAM could be seen as a Lagrangian Relaxation of IFAM, with particular Lagrangian multipliers. We implemented this Lagrangian Relaxation in the general case, and we then used it into a Column Generation approach based on a Dantzig-Wolfe decomposition of IFAM. This approach could be used to solve a toy instance, but suffered performance issues on instances of moderate size. We proposed possible enhancements for improving this performance, but we left them unimplemented, in order to focus on other modeling aspects.

In a second part, we discussed possible effects of forecasts inaccuracy over IFAM profit, a topic already largely covered by the literature. We explained why we believe desirable to consider groups of demands rather than single itinerary demands. We then proposed a new definition for the term Market in the context of Airline Scheduling, and formulated MDFAM, a new Fleet Assignment model based on this proposal. We illustrated the flexibility of this model through various examples.

The third part of this work was a study of the major models of the literature along with MDFAM, considering demand volatility. We showed that, when the knowledge of the demand behavior was very good, MDFAM could lead to better results than IFAM. We also showed that, when introducing more uncertainty in the data, the advantage of MDFAM over IFAM was less clear, suggesting that building forecast for the express purpose of applying MDFAM would certainly require a great expertise. This study also showed that IFAM consistently gave better results than FAM, even in a context of high demand volatility.

This works opens the way towards two directions of development:

The first direction would be to address demand likeliness. The model

of this thesis, MDFAM, is fundamentally optimistic about demands, and automatically assumes the best possible demand scenario. We have seen that this is a problem, because unless the constraints are very strict, this demand scenario is unlikely to actually happen. In this perspective, we could study what are the best demand constraints, such that the demand remains realistic. Another possibility could be to penalize the model whenever it chooses an unrealistic demand scenario. The objective function would thus need to include a measure of the demand likeliness, and a cost associated to this measure.

The second direction of development would be to better address fare volatility. Demands are not the only inputs of the Fleet Assignment to suffer volatility. Another parameter that is also highly variable is the fare. Fare pricing systems are a vast, intricate issue of airline industry. One possibility for addressing this volatility could be to turn the fares into constrained variables, like it has been done for the demands. However, this would lead to a quadratic formulation, a sort of problem that is difficult to solve at this scale. Furthermore, we would also be exposed to difficulties of the same class, with a model choosing unrealistic values.

Bibliography

Abara, J. (1989). Applying integer linear programming to the fleet assignment problem. *Interfaces*, pages 20–28.

Ahuja, R., Goodstein, J., Mukherjee, A., Orlin, J., and Sharma, D. (2001). A very large-scale neighborhood search algorithm for the combined through and fleet assignment model. *MIT Sloan Working Paper No. 4388-01*.

Ahuja, R., Liu, J., Goodstein, J., Mukherjee, A., Orlin, J., and Sharma, D. (2003). Solving multi-criteria through-fleet assignment models. *Applied Optimization*, 79:233–256.

Barnhart, C., Belobaba, P., and Odoni, A. (2003). Applications of operations research in the air transport industry. *Transportation science*, 37(4):368–391.

Barnhart, C., Boland, N., Clarke, L., Johnson, E., Nemhauser, G., and Shenoi, R. (1998a). Flight string models for aircraft fleeting and routing. *Transportation science*, 32(3):208–220.

Barnhart, C., Farahat, A., and Lohatepanont, M. (2009). Airline fleet assignment with enhanced revenue modeling. *Operations research*, 57(1):231–244.

Barnhart, C., Johnson, E., Nemhauser, G., Savelsbergh, M., and Vance, P. (1998b). Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, pages 316–329.

Barnhart, C., Kniker, T., and Lohatepanont, M. (2002). Itinerary-based airline fleet assignment. *Transportation Science*, 36(2):199–217.

Belobaba, P. P. and Farkas, A. (1996). *The influence of network effects and yield management on airline fleet assignment decisions*. PhD thesis, Massachusetts Institute of Technology.

Berge, M. and Hopperstad, C. (1993). Demand driven dispatch: A method for dynamic aircraft capacity assignment, models and algorithms. *Operations Research*, pages 153–168.

Briant, O., Lemaréchal, C., Meurdesoif, P., Michel, S., Perrot, N., and Vanderbeck, F. (2008). Comparison of bundle and classical column generation. *Mathematical Programming*, 113(2):299–344.

Clarke, L., Hane, C., Johnson, E., and Nemhauser, G. (1996). Maintenance and crew considerations in fleet assignment. *Transportation Science*, 30(3):249–260.

Culioli, J.-C. (2006). Modèles et problèmes de revenue management pour une compagnie aérienne. Toulouse. 5e rencontre Math-Industrie: Aéronautique et Espace.

Dumas, J., Aithnard, F., and Soumis, F. (2009). Improving the objective function of the fleet assignment problem. *Transportation Research Part B: Methodological*, 43(4):466–475.

Dumas, J. and Soumis, F. (2008). Passenger flow model for airline networks. *Transportation Science*, 42(2):197–207.

Gopalan, R. and Talluri, K. (1998). Mathematical models in airline schedule planning: A survey. *Annals of Operations Research*, 76:155–185.

Gu, Z., Johnson, E., Nemhauser, G., and Wang, Y. (1994). Some properties of the fleet assignment problem. *Operations Research Letters*, 15(2):59–71.

Hane, C., Barnhart, C., Johnson, E., Marsten, R., Nemhauser, G., and Sigismondi, G. (1995). The fleet assignment problem: solving a large-scale integer program. *Mathematical Programming*, 70(1):211–232.

Jacobs, T. L., Smith, B. C., and Johnson, E. L. (2008). Incorporating network flow effects into the airline fleet assignment process. *Transportation Science*, 42(4):514–529.

Jiang, H. and Barnhart, C. (2009). Dynamic airline scheduling. *Transportation Science*, 43(3):336–354.

Jiang, H. and Barnhart, C. (2013). Robust airline schedule design in a dynamic scheduling environment. *Computers & Operations Research*, 40(3):831–840.

Jolliffe, I. (2005). *Principal component analysis*. Wiley Online Library.

Kniker, T. S. and Banhart, C. (1998). Shortcomings of the conventional airline fleet assignment model. Technical report.

Lemaréchal, C. (2001). Lagrangian relaxation. *Computational Combinatorial Optimization*, pages 112–156.

Lohatepanont, M. and Barnhart, C. (2004). Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment. *Transportation Science*, 38(1):19–32.

Rexing, B., Barnhart, C., Kniker, T., Jarrah, A., and Krishnamurthy, N. (2000). Airline fleet assignment with time windows. *Transportation Science*, 34(1):1–20.

Rosenberger, J. M., Johnson, E. L., and Nemhauser, G. L. (2004). A robust fleet-assignment model with hub isolation and short cycles. *Transportation Science*, 38(3):357–368.

Shebalov, S. (2009). Practical overview of demand-driven dispatch. *Journal of Revenue & Pricing Management*, 8(2):166–173.

Smith, B. C. and Johnson, E. L. (2006). Robust airline fleet assignment: Imposing station purity using station decomposition. *Transportation Science*, 40(4):497–516.