Les Cahiers du GERAD ISSN: 0711–2440

Passenger Flow Model for Airline Networks

J. Dumas
F. Soumis

G–2007–80

November 2007

Les textes publiés dans la série des rapports de recherche HEC n'engagent que la responsabilité de leurs auteurs. La publication de ces rapports de recherche bénéficie d'une subvention du Fonds québécois de la recherche sur la nature et les technologies.

Passenger Flow Model for Airline Networks

Jonathan Dumas
François Soumis

GERAD and
Département de mathématiques et de génie industriel
École Polytechnique de Montréal
C.P. 6079, Succ. Centre-ville
Montréal (Québec) Canada, H3C 3A

dumas@crt.umontreal.ca
francois.soumis@gerad.ca

November 2007

Les Cahiers du GERAD

G–2007–80

Abstract

We present a model that rapidly finds an approximation of the expected passenger flow on an airline network, given forecast data concerning 1) the distribution of the demand for each itinerary, seen as a random variable; 2) the time distribution of booking requests for each itinerary; and 3) the proportion of spill (from an itinerary) that is attracted to a given alternative itinerary. Solutions are found in a few seconds for a 30 000 itinerary network. Results differ from the expected passenger flow found by a simulation by about 0.1% for load factors below 80%.

Résumé

Nous décrivons un modèle de flot de passagers pour réseaux aériens qui, étant données des prévisions concernant 1) la demande pour chaque itinéraire, considérée comme variable aléatoire, 2) la distribution dans le temps des requêtes de réservations pour chaque itinéraire, 3) les proportions de débordement de chaque itinéraire vers tout autre, fournit une approximation de l'espérance du nombre de passagers sur chaque itinéraire et chaque vol. L'algorithme de résolution traite un réseau de 30 000 itinéraires en quelques secondes. Pour des taux d'occupation de 80% et moins, l'espérance de flot ainsi trouvée s'accorde avec celle fournie par une simulation extensive dans une marge de 0,1%.

1 Introduction

The huge problem of planning airline operations in order to maximize profit is currently split into several decision problems. The first one, the flight schedule problem, consists in deciding which flights to offer. The next problem is that of assigning an aircraft type to each flight of a tentative flight schedule. That is the *Fleet Assignment Problem* (FAP). Fleet assignment is subject to constraints arising from the airline's fleet, flying regulations, maintenance necessity, physical realizability, etc., while the objective of the FAP is maximizing expected profit.

Airline companies are therefore interested in having a good estimate of the revenue they may expect if they chose a particular FA for a given flight schedule. Clearly, the part of this problem that deals with demand forecast and customer behavior belongs to the realm of economics. We will instead be concerned here with the influence of the FAP decision variables on the expected revenue, given such demand forecasts and customer behavior description. That is, we model how the assignment of fleet types affects the placement of passengers through capacity constraints.

Our model has the following main features:

1. It considers spill and recapture as occurring between itineraries, not flight legs;
2. It is stochastic, meaning that it takes stochastic demand prediction as inputs, and aims at computing expected numbers of passengers on each itinerary;
3. It is temporal in nature, in the sense that it splits the booking period into time slices and computes passenger placement one time slice after the other. That allows the model to take into account the time distribution of the bookings for each itinerary;
4. It doesn't assume airline control over passengers' behavior.

To our knowledge, no other publicly available passenger flow model possesses all these characteristics.

Old FAP models had a flight-based estimation of loss of revenue due to spill (passenger loss due to capacity constraints) (Abara (1989), Subramanian et al. (1994)). The existence of multi-flight itineraries causes flight-based spill models to ignore the dependency between flights inherent to the spill phenomenon. The effects of this dependency are often referred to as *network effects*; see Barnhart et al. (2002) for a detailed description.

In Phillips et al. (1991) an algorithm is presented that takes into account the fact that the booking process leading to the placement of passengers unfolds during a period of time, and that what happens at the end of it is influenced by what happened before. This algorithm, however, considers the demand as static.

The Passenger Mix Model (PMM) of Barnhart and Kniker (see Barnhart et al. (2002), Kniker (1998)) models passenger flow through a linear program whose objective is to find the most profitable mix of passengers, assuming that the airline company has some control on where (which itineraries) spilled passengers should be redirected.

Jacobs, Smith and Johnson (Jacobs et al. (1999), Smith (2004)) integrate an OD revenue management scheme into a FA solver. The underlying passenger flow model is a linear program that respects the stochastic nature of the demand and that seeks the RM seat allocation that maximizes the expected revenue.

Paper Organization. We give a detailed description of our problem in Section 2. We describe our model in Section 3. Section 4 contains results of a test made on two large networks. In Section 5, we discuss the flexibility of the model with respect to the demand distribution, the possibility of integrating reservation levels in the model, and that of integrating it with a FAP solver.

2 The Problem

In this section, we present our working hypotheses on the nature of the data that should serve as inputs to the model, and on customers behavior. These do not constitute the model. Rather, our model aims at computing a good estimation of the expected number of passengers on each itinerary of a network, *assuming that the hypotheses listed below hold*.

2.1 Network

Consider a flight schedule that an airline company is planning to operate over a typical period $\mathcal{P}$ (e.g., a week) in a season, and assume that aircraft types, and hence, capacities, have been assigned to each flight.

We use the term *itinerary* to denote a physical itinerary, linking an origin to a destination through one or more flights, together with a fare class. A set $\mathcal{I}$ of possible itineraries is offered to the customers.

We subdivide each flight into cabins corresponding to different fare classes, and call them *arcs*. Hence, an itinerary is a set of arcs. We denote by $\mathcal{A}$ the set of all arcs, and write $a \in i$ to mean that arc a is part of itinerary i. Each arc a has a capacity cap_a. For a given flight, arc capacities sum to the capacity of the aircraft type assigned to it. How to integrate nested reservation levels in the problem, and in the model, is explained in Subsection 5.2.

2.2 Demand

We assume that we have forecasts for the unconstrained demand D_i for each itinerary $i \in \mathcal{I}$. This demand D_i is assumed to follow a normal law, truncated at 0, of expectation d_i and coefficient of variation cv_i (so that it has variance $(d_i cv_i)^2$).

It is known that small demands are better modeled by gamma or other non-Gaussian probability laws (Swan (2002)). We will discuss the relaxation of this hypothesis in Section 5.

2.3 Synthetic booking process

Without loss of generality, the whole booking process may be assumed to unfold during the time interval $[0, 1]$. Note that the mapping from the actual period of time to the interval $[0, 1]$ need not be linear; it may rather be such that the booking rate for low fare classes is constant.

Time distribution of booking requests. Consider one occurrence of the typical period $\mathcal{P}$, for which the actual unconstrained demand δ_i for each itinerary i is determined. That is, δ_i is an occurrence of the random variable D_i. For each itinerary i, we have a function $b_i : [0, 1] \to \mathbb{R}$ determining the rate at which the demand for i manifests itself: at time t, this rate is

$$\delta_i \cdot b_i(t).$$

We must have $\int_0^1 b_i(t)dt = 1$. Moreover, we require the function b_i to be piecewise constant (to accommodate the model). These functions should reflect the typical time distribution of bookings for the fare class of itinerary i.

Spill. When an arc is booked to its capacity, all itineraries containing this arc are closed and the remaining demand for them is partly spilled to other itineraries, and partly lost. We assume that the proportion of customers spilled from a blocked itinerary i to an alternative itinerary j is a fixed number $\lambda_{j,i}$. We call it the *spill coefficient* from i to j. A non null proportion of customers is lost, so that $(\sum_{j \in \mathcal{I}} \lambda_{j,i}) < 1$.

If an alternative itinerary j is also closed, the spill is transfered to alternative itineraries for j that are still open, and different from i. Such an itinerary k receives booking requests from customers originally attracted to i at a rate of $\delta_i \cdot b_i(t) \cdot \lambda_{k,j}\lambda_{j,i}$. Spill is allowed to be transfered in that manner no more than three times.

Typically, the spill coefficients will allow passengers to spill from some itinerary to another one of the same class, or to the corresponding itinerary of a higher class.

Passengers. The total number of passengers (successful booking requests) on a given itinerary is the integral, from 0 to the time it closed (or to 1 if it did not), of the booking request rate.

2.4 Statement of the problem

Our problem is:

> *Given a transportation network $(\mathcal{A}, \mathcal{I})$ as described above, assuming customer behavior governed by the synthetic booking process described above, and with given stochastic demand forecasts, find the expected number of passengers on each itinerary.*

Simulation. Solving this problem exactly is not feasible: one would have to integrate large products of density functions over a huge number (on the order of $|\mathcal{A}|!$) of subsets of $\mathbb{R}^{|\mathcal{I}|}$. Hence, to evaluate the quality of our model's solutions, we have implemented an efficient simulation algorithm that computes the outcome of several occurrences of the typical period $\mathcal{P}$.

For each occurrence, it first randomly generates the demand for each itinerary, according to the law it is assumed to follow. It initializes booking rates and finds which arc is due to be full first, and when. It updates request rates according to the synthetic booking process hypotheses and finds the next arc due to be full, and so on. Notice that the number of such request-rate updates is equal to the number of arcs that are full at the end of the booking period and is hence smaller than the number of arcs. The algorithm also modifies request rates at points of discontinuity of the functions $b_i, (i \in \mathcal{I})$. It stores the final number of passengers on each itinerary. Several occurrences of the period $\mathcal{P}$ are simulated, and the average number of passengers for each itinerary is computed.

Observe that this simulation algorithm is much faster than one that would assign itineraries to passengers individually. The number of times it computes new passenger placement rates is the number of arcs that are finally fully booked, which is usually not more than 10% of the number of arcs.

The drawback of using this simulation as a passenger flow model is the computing time. For large networks, it took up to 5 days (depending on the network and the load factor) to make 5,000 simulations on a computer that ran our model in 3 to 7 seconds.

3 The Model

3.1 Overview

The model aims at providing a good approximate solution to the passenger flow problem stated above.

Time discretization. We split the time interval $[0, 1]$, during which the booking process unfolds, into smaller time slices $[t_k, t_{k+1}]$, $(k = 0, \ldots, K)$, with $t_0 = 0$ and $t_{K+1} = 1$. Typically, the first time slices can be larger than the last ones, because much of the spilling activity occurs at the end of the booking process.

We want the booking request rate coming from the demand for each itinerary to be constant on each time slice, so the set $\{t_0, \ldots t_K\}$ must contain all discontinuity points of the functions $b_i, (i \in \mathcal{I})$.

For each time slice, sequentially, we compute good estimates of the expected number of passengers accepted on each itinerary (and of several other quantities).

The time discretization allows the model to take account of what is by nature temporal in the booking process; that is:

1. the non-constant distribution in time of the booking requests coming from the demand (through the functions $b_i, (i \in \mathcal{I})$);
2. the effect of the order in which itineraries are closed on spill and recapture.

For a particular time slice, we compute the value of various quantities associated with itineraries and arcs (number of new booking requests, of new passengers, of spilled passengers, etc.) by solving a non-linear system of equations.

A system of equations. The system is described in detail below, in Subsections 3.2 and 3.3.

For now, let us say that its generic equation is based on the following simple observation: each booking request for an itinerary i either comes from the demand for i, or from a closed itinerary j that spills on i. Hence, if we let $\mathbf{r}_i$ be the expected number of booking requests for i between time t and $t + \Delta t$, and $\mathbf{sp}_i$ be the expected number of unsuccessful booking requests for i during the same period of time, we have

$$\mathbf{r}_i = (d_i \cdot b_i(t))\Delta t + \sum_{j \in \mathcal{I}} \lambda_{i,j} \mathbf{sp}_j, \qquad (1)$$

according to the synthetic booking process description (recall that d_i is the expectation of the demand for i). Now, if $\mathbf{sp}_j$ is expressed as a function of the variables $\mathbf{r}_i, (i \in \mathcal{I})$, we obtain a system of $|\mathcal{I}|$ equations and $|\mathcal{I}|$ unknowns. This approach is inspired by the equilibrium passenger flow model in Soumis (1978) and Soumis and Nagurney (1993).

It would be neither convenient nor instructive to write down the actual system of equations. Rather, we will split it into several equalities involving variables whose names indicate which expected quantity they are meant to be estimates of. That is the subject of the two next subsections.

3.2 Variables and equalities related to the itineraries

Below is a list of the model's variables that are attached to the itineraries. Notice that when we write, for example, that $\mathbf{r}_i^k$ is the number of booking requests for itinerary i during $[t_k, t_{k+1}]$, what we really mean is that it is the *model's estimate* of the *expected* number of booking requests for itinerary i during that period.

- $\mathbf{r}_i^k$: number of booking requests for itinerary i over $[t_k, t_{k+1}]$;
- $total\mathbf{r}_i^k$: total number of booking requests for itinerary i up to time t_k;
- $\mathbf{sp}_i^k$: number of unsuccessful booking requests for itinerary i over $[t_k, t_{k+1}]$;
- $\mathbf{pass}_i^k$: number of successful booking requests for itinerary i over $[t_k, t_{k+1}]$;
- $\mathbf{p}_i^k$: probability that itinerary i is closed at time t_k (assumed to be constant over $[t_k, t_{k+1}]$).

Let us now describe how these quantities are tied to one another.

Firstly,

$$total\mathbf{r}_i^k = \sum_{l=0}^{k-1} \mathbf{r}_i^l \quad (i \in \mathcal{I}).$$

Now, from equation 1, we have

$$\mathbf{r}_i^k = d_i b_i(t_k) \cdot (t_{k+1} - t_k) \;+\; \sum_{j \in \mathcal{I}} \lambda_{i,j} \mathbf{sp}_j^k \quad (i \in \mathcal{I}). \tag{2}$$

The actual probability that an itinerary i is closed varies over time. In our model, we assume that it is constant over the time interval $[t_k, t_{k+1}]$, and denote it $\mathbf{p}_i^k$. We have

$$\mathbf{sp}_i^k = \mathbf{r}_i^k \cdot \mathbf{p}_i^k \quad (i \in \mathcal{I}), \tag{3}$$

and

$$\mathbf{pass}_i^k = \mathbf{r}_i^k \cdot (1 - \mathbf{p}_i^k) \quad (i \in \mathcal{I}). \tag{4}$$

Combining equations 2 and 3, we obtain

$$\mathbf{r}_i^k = d_i b_i(t_k) \cdot (t_{k+1} - t_k) \;+\; \sum_{j \in \mathcal{I}} \lambda_{j,i} \mathbf{r}_j^k \cdot \mathbf{p}_j^k \quad (i \in \mathcal{I}). \tag{5}$$

This would be the generic equation of our system of equations if the variables $\mathbf{p}_j^k$ were fixed. Each variable $\mathbf{p}_i^k$, however, is tied to corresponding variables on arcs, which are described in the next subsection.

3.3 Variables and equalities related to the arcs

For further reference, here is the list of variables related to arcs.

- $\mathbf{R}_a^k$: number of booking requests for arc a over $[t_k, t_{k+1}]$;
- $\mathbf{SP}_a^k$: number of unsuccessful booking requests for arc a over $[t_k, t_{k+1}]$;
- $total\mathbf{R}_a^k$: total number of booking requests for arc a up to time t_k;
- $total\mathbf{SP}_a^k$: total number of unsuccessful booking requests for arc a up to time t_k;
- Var_a^k: the variance of the total number of booking requests for arc a at time t_k;
- $\mathbf{P}_a^k$: probability that arc a is closed at time t_k (assumed to be constant over $[t_k, t_{k+1}]$).

According to the synthetic booking process hypotheses, an itinerary is closed if and only if at least one of its arcs is fully booked, so we set

$$\mathbf{p}_i^k = 1 - \left(\prod_{a \in i} (1 - \mathbf{P}_a^k) \right) \quad (i \in \mathcal{I}). \tag{6}$$

According to our working hypotheses, the events "arc a is closed at time t", for arcs $a \in i$, are dependent, since increased flow on any arc of an itinerary implies increased flows on all arcs of that itinerary, but this dependency is only caused by itineraries using more than one arc of i. This corresponds to a small part of the non local traffic on arcs $a \in i$, which itself is small compared to local traffic. Hence, equation 6, which causes no problem for single-leg itineraries, is fairly reasonable for multi-leg itineraries.

The value of $\mathbf{P}_a^k$ is a function of $\mathbf{SP}_a^k$ and $\mathbf{R}_a^k$ (equation 11); we will first see how these variables are related to the others.

Let us first consider $\mathbf{R}_a^k$, the number of requests on arc a during $[t_k, t_{k+1}]$. What comes first to mind is to set it equal to $\sum_{i:a \in i} \mathbf{r}_i^k$. This, however, has proven to work badly. The following example illustrates why. Consider a two-arc itinerary i with arcs a and a', with a being, on average, 10 times more likely to be full than a'. Unsuccessful bookings requests for i are much more likely to be due to a being closed than a' being closed, and in a sense, a' "doesn't see" these unsuccessful booking requests. Hence, in the computation of the number $\mathbf{R}_a^k$, the contribution of the unsuccessful requests for i should be weighted, in order to reflect that phenomenon.

We propose to think of a booking request on an arc a as either: 1) a successful booking request on an itinerary using a; or 2) an unsuccessful booking request on an itinerary using a, *that we attribute to a*. Moreover, we want this attribution of the $\mathbf{sp}_i^k$ unsuccessful booking requests on itinerary i to be such that no unsuccessful request is counted twice.

Consider a two-arc itinerary i consisting of arcs a and a'. We attribute the unsuccessful booking requests on i in proportion to the numbers

$$\mathbf{P}_a^k(1 - \mathbf{P}_{a'}^k) \quad + \quad \frac{\mathbf{P}_a^k \mathbf{P}_{a'}^k}{2} \qquad \text{and}$$

$$\mathbf{P}_{a'}^k(1 - \mathbf{P}_a^k) \quad + \quad \frac{\mathbf{P}_{a'}^k \mathbf{P}_a^k}{2}$$

to a and a' respectively. That is, when only one arc is closed, it is fully responsible, and when both arcs are closed, the responsibility is evenly split. We let $\alpha_{i,a}^k$ and $\alpha_{i,a'}^k$ be the numbers that add up to 1 and that are proportional to those above. For itineraries consisting of more than two arcs, these weights are computed in a similar fashion. For single-arc itineraries, they are set to 1.

Our equality linking $\mathbf{R}_a^k$ to the other variables is then

$$\mathbf{R}_a^k = \sum_{i:a \in i} (\mathbf{pass}_i^k + \alpha_{i,a}^k \mathbf{sp}_i^k) \qquad (a \in \mathcal{A}). \tag{7}$$

We have seen how $\mathbf{R}_a^k$ is computed; let us now consider $\mathbf{SP}_a^k$. We compute it indirectly. It is more convenient to compute $total\mathbf{SP}_a^{k+1}$, the total spill of arc a at time t_{k+1}, and set

$$\mathbf{SP}_a^k = total\mathbf{SP}_a^{k+1} - total\mathbf{SP}_a^k. \tag{8}$$

The computation of the spill must take into account the stochastic nature of the number of requests. Let X_a^{k+1} be the random variable "total number of booking requests on arc a at time t_{k+1}". This variable is not normally distributed, but we assume, in the current model, that it follows a normal law.

This choice is justified by the fact that the large majority of requests are originating from the demand for itineraries using a, which follow normal laws truncated at 0; their sum is approximately normal with negligible truncation at 0. The other kind of requests are the recaptured ones. The number of recaptured requests is not normally distributed, but since recaptured requests originate from several itineraries, it is the sum of several random variables, and its distribution tends to be normal-like.

Our model's estimation for the expectation of X_a^{k+1} is $total\mathbf{R}_a^{k+1}$, and for its variance, we use

$$Var_a^{k+1} = \sum_{i:a\in i} (total\mathbf{r}_i^{k+1} \cdot cv_i)^2. \tag{9}$$

Implicitly, this last equation assumes independence between requests on different itineraries, and assumes that the number of requests for any itinerary i has the same coefficient of variation as the original demand for i.

We obtain $total\mathbf{SP}_a^{k+1}$ as the expectation of the number of requests on a, at time t_{k+1}, in excess of its capacity, that is,

$$total\mathbf{SP}_a^{k+1} = \int_{cap_a}^{\infty} (x - cap_a) f_{X_a^{k+1}}(x)dx, \tag{10}$$

where $f_{X_a^{k+1}}$ is the density function of X_a^{k+1}. Then, we let $\mathbf{SP}_a^k = total\mathbf{SP}_a^{k+1} - total\mathbf{SP}_a^k$.

Finally, the probability $\mathbf{P}_a^k$ that arc a is full during $[t_k, t_{k+1}]$ is set to be the ratio of the spill of a to its number of requests during that time interval:

$$\mathbf{P}_a^k = \begin{cases} 0 & \text{if } \mathbf{R}_a^k = 0, \\ 0 & \text{if } \frac{\mathbf{SP}_a^k}{\mathbf{R}_a^k} < 0, \\ 1 & \text{if } \frac{\mathbf{SP}_a^k}{\mathbf{R}_a^k} > 1, \\ \frac{\mathbf{SP}_a^k}{\mathbf{R}_a^k} & \text{otherwise.} \end{cases} \tag{11}$$

Notice that in practice, in the running of an algorithm that seeks an approximate fixed-point of our system by iteratively assigning to $\mathbf{P}_a^k$ its value as a function of the values of $\mathbf{SP}_a^k$ and $\mathbf{R}_a^k$ at the previous iteration, $\frac{\mathbf{SP}_a^k}{\mathbf{R}_a^k}$ will almost always lie in $[0, 1]$, but none of the cases in equation 11 is impossible. This completes the exposition of our system of equations.

3.4 List of the equations

To give the reader a better view of the whole system, here is a list of the equations. We have combined some of them to make the ensemble more compact. For all $i \in \mathcal{I}$ and $a \in \mathcal{A}$,

$$\begin{aligned}
\mathbf{r}_i^k &= d_i b_i(t_k) \cdot (t_{k+1} - t_k) \ + \ \sum_{j \in \mathcal{I}} \lambda_{i,j} \mathbf{r}_i^k \cdot \mathbf{p}_i^k, \\
\mathbf{p}_i^k &= 1 - \left(\prod_{a \in i} (1 - \mathbf{P}_a^k) \right), \\
\mathbf{P}_a^k &= \frac{\mathbf{SP}_a^k}{\mathbf{R}_a^k}, \qquad \text{(except in some extreme cases)} \\
\mathbf{R}_a^k &= \sum_{i:a \in i} (\mathbf{r}_i^k \cdot (1 - \mathbf{p}_i^k) + \alpha_{i,a}^k \mathbf{r}_i^k \cdot \mathbf{p}_i^k), \\
\mathbf{SP}_a^k &= \int_{cap_a}^{\infty} (x - cap_a) f_{X_a^{k+1}}(x) dx - total\mathbf{SP}_a^k,
\end{aligned}$$

where:

- $\alpha_{i,a}^k$ is the responsibility coefficient described before equation 7,
- X_a^{k+1} is a normal random variable of expectation $total\mathbf{R}_a^{k+1} = \sum_{l=0}^k \mathbf{R}_a^l$ and of variance $\sum_{i:a \in i} (total\mathbf{r}_i^{k+1} \cdot cv_i)^2$,
- $total\mathbf{SP}_a^k = \int_{cap_a}^{\infty} (x - cap_a) f_{X_a^k}(x) dx$.

3.5 Resolution and implementation

We use a simple iterative fixed-point method to find approximate solutions to equations 2 to 11.

The algorithm treats time slices sequentially. For time slice $[t_k, t_{k+1}]$, variables $\mathbf{r}_i^k$ are initialized to $d_i b_i(t_k) \cdot (t_{k+1} - t_k)$, the original demand for i for that time slice. Variables $\mathbf{P}_a^k$ are initialized to the value reached by $\mathbf{P}_a^{k-1}$ after convergence. (variables $\mathbf{P}_a^0$ are initialized to 0). Remaining variables related to the interval are set to 0.

Then, variables' values are computed according to the equations of the previous section, in the following order (for all itineraries i, and all arcs a):

1. $\mathbf{R}_a^k$;
2. Var_a^{k+1};
3. $\mathbf{SP}_a^k$;
4. $\mathbf{P}_a^k$;

5. $\mathbf{p}_i^k$ and $\alpha_{i,a}^k$,
6. $\mathbf{sp}_i^k$ and $\mathbf{pass}_i^k$;
7. $\mathbf{r}_i^k$.

This is repeated until a stopping criterion is met that is specified by a desired degree of stability of the variables. In our implementation, for example, we made the iterative process stop when the quantity $\sum_{i\in\mathcal{I}} |\mathbf{r}_i^k - \mathbf{r}_i^{k-1}|$ was less than one thousandth of the total demand for the time interval. It is an attractive criterion since this value decreases, from one iteration to the next, in a roughly geometric way. In our tests the number of iterations needed to meet it rarely exceeded 15.

To test the robustness of our algorithm, we ran it several thousand times over different networks. We randomly perturbed the networks' data and modified the stopping criterion. It always stopped.

When we imposed very stringent stopping criteria like $\sum_{i\in\mathcal{I}} |\mathbf{r}_i^k - \mathbf{r}_i^{k-1}| < 0.0001$, the value of $\sum_{i\in\mathcal{I}} |\mathbf{r}_i^k - \mathbf{r}_i^{k-1}|$ actually reached 0 for each time slice $[t_k, t_{k+1}]$, within about 50 iterations. In other words, the algorithm actually found a fixed point. This is due to the necessary discretization of the functions related to the normal distribution. We used a table for the cumulative distribution function of a $N(0,1)$, and one for $\int_c^\infty xf(x)dx$, where f is the density function of a $N(0,1)$; both have 10,000 entries.

3.6 Existence of a solution to the system of equations

We do not have a proof that the algorithm of the previous section converges. In this section we prove that there exists a solution to our system of equations. At the end of the section, we will be better able to explain why, in practice, the algorithm does converge.

Let us write $n = |\mathcal{I}|$ and $m = |\mathcal{A}|$. Consider any time slice $[t_k, t_{k+1}]$, and let $\mathbf{d} \in \mathbb{R}_+^n$ be the vector whose component $\mathbf{d}_i$ is the total original demand for i during $[t_k, t_{k+1}]$, that is,

$$\mathbf{d}_i = d_i \cdot b_i(t_k) \cdot (t_{k+1} - t_k).$$

Now, let $\epsilon > 0$ be a small real number. We write $\mathbb{R}_{++}$ to denote the set of positive real numbers. Define

$$\begin{aligned} X &= \prod_{i\in\mathcal{I}} [\epsilon, +\infty) \subseteq \mathbb{R}_{++}^n, \text{ and} \\ Y &= \prod_{a\in\mathcal{A}} [\epsilon, 1] \subseteq \mathbb{R}_{++}^m. \end{aligned}$$

In what follows, $\mathbf{r}_i$, the component i of the vector $\mathbf{r} \in \mathbb{R}^n$, will play the role of $\mathbf{r}_i^k$, and $\mathbf{P}_a$, the component a of the vector $\mathbf{P} \in \mathbb{R}^m$, will play the role of the probability $\mathbf{P}_a^k$ of the previous section. The vectors $\tilde{\mathbf{r}}$ and $\tilde{\mathbf{P}}$ will be the vectors whose components are, respectively, the new number of requests and the new probabilities obtained from $\mathbf{r}$ and $\mathbf{P}$ by the equations of the previous section.

For all $i \in \mathcal{I}, a \in i$, let $\alpha_{i,a} : Y \to [0,1]$ be the function that returns the responsibility weight described before the equation 7 of the previous section, with the entry $\mathbf{P}_a$ of $\mathbf{P} \in Y$ playing the role of $\mathbf{P}_a^k$. That is, $\alpha_{i,a}(\mathbf{P}) = 1$ if i has a single arc a, and

$$\alpha_{i,a}(\mathbf{P}) = \frac{\mathbf{P}_a(1-\mathbf{P}_{a'}) + \frac{\mathbf{P}_a\mathbf{P}_{a'}}{2}}{\mathbf{P}_a(1-\mathbf{P}_{a'}) + \mathbf{P}_{a'}(1-\mathbf{P}_a) + \mathbf{P}_{a'}\mathbf{P}_a}$$

if i consists of arcs a and a' only. The reader can easily write down the expressions for $\alpha_{i,a}$ when i has three or more arcs, and can verify that all these functions are continuous over Y.

For any $(\mathbf{r}, \mathbf{P}) \in X \times Y$, consider the following mappings :

1. $R : X \times Y \to \mathbb{R}^m_{++}$, the new number of arc requests, whose a^{th} component is

$$R_a(\mathbf{r}, \mathbf{P}) = \sum_{i:a\in i} \mathbf{r}_i \left(\prod_{a\in i}(1-\mathbf{P}_a) + \alpha_{i,a}(\mathbf{P}) \cdot \left(1 - \prod_{a\in i}(1-\mathbf{P}_a)\right)\right),$$

   $(a \in \mathcal{A})$. One may recognize that $R_a(\mathbf{r}, \mathbf{P})$ depends on $\mathbf{r}$ and $\mathbf{P}$ in the same way that $\mathbf{R}_a^k$ depends on $\mathbf{r}^k$ and $\mathbf{P}^k$ in the previous section. R is continuous.

2. $Var : X \to \mathbb{R}_{++}$,

$$Var_a(\mathbf{r}) = \sum_{i:a\in i} (\mathbf{r}_i \cdot cv_i)^2, \qquad (a \in \mathcal{A})$$

   where the $cv_i, (i \in \mathcal{I})$ are constant. This is a continuous mapping.

3. $\tilde{\mathbf{P}} : X \times Y \to Y$,

$$\tilde{\mathbf{P}}_a(\mathbf{r}, \mathbf{P}) = \min\left\{1, \max\left\{\epsilon, \quad \frac{\left(\int_{cap_a}^{\infty}(x - cap_a)f_X(x)dx\right) - total\mathbf{SP}_a}{R_a(\mathbf{r}, \mathbf{P})}\right\}\right\},$$

   $(a \in \mathcal{A})$, where the $total\mathbf{SP}_a$ and cap_a are constants, X is a normal $N(total\mathbf{R}_a + R_a(\mathbf{r}, \mathbf{P}), Var_a(\mathbf{r}))$, and $total\mathbf{R}_a$ is a constant. One may verify that $\tilde{\mathbf{P}}$ is continuous.

4. $\tilde{\mathbf{r}} : X \times Y \to X$,

$$\tilde{\mathbf{r}}_i(\mathbf{r}, \mathbf{P}) = \max\left\{\epsilon, \quad \mathbf{d}_i + \sum_{j\in\mathcal{I}} \lambda_{i,j}\mathbf{r}_j \left(1 - \prod_{a\in j}(1 - \tilde{\mathbf{P}}_a(\mathbf{r}, \mathbf{P}))\right)\right\},$$

   $(i \in I)$, where the $\lambda_{i,j}$ are the spill coefficients described earlier. $\tilde{\mathbf{r}}$ is a continuous mapping.

Now, let $\mathbf{\Lambda}$ be the $n \times n$ matrix with $(\mathbf{\Lambda})_{ij} = \lambda_{i,j}$. Recall that there exists $\rho \in (0,1)$ such that, for all $j \in \mathcal{I}$, $\sum_{i\in\mathcal{I}} \lambda_{i,j} \leq \rho$. Moreover, the $\lambda_{i,j}$ are non-negative. Hence, the maximum absolute column sum norm of $\mathbf{\Lambda}$ is smaller than 1, and so is its spectral norm, that is, $\|\mathbf{\Lambda}\| < 1$.

Let $P(\mathbf{r},\mathbf{P})$ be the diagonal $n \times n$ matrix with $(P(\mathbf{r},\mathbf{P}))_{ii} = 1 - \prod_{a\in i}(1 - \tilde{\mathbf{P}}_a(\mathbf{r},\mathbf{P}))$. If we denote by $\max\{\mathbf{u},\mathbf{v}\}$ the vector whose i^{th} component is $\max\{\mathbf{u}_i,\mathbf{v}_i\}$, then

$$\tilde{\mathbf{r}}(\mathbf{r},\mathbf{P}) = \max\{\epsilon\mathbf{1}_n,\ \mathbf{d} + \mathbf{\Lambda}P(\mathbf{r},\mathbf{P})\mathbf{r}\},$$

where $\mathbf{1}_n$ is the all-one vector in $\mathbb{R}^n$.

Let $\mathbf{d}^* = \max\{\epsilon\mathbf{1}_n, \mathbf{d}\}$, and define Z as

$$Z = \{\mathbf{r} \in \mathbb{R}^n | \epsilon\mathbf{1}_n \leq \mathbf{r} \text{ and } \|\mathbf{r}\| \leq \frac{\|\mathbf{d}^*\|}{1 - \|\mathbf{\Lambda}\|}\}.$$

We claim that the following continuous mapping

$$\begin{array}{rcl} F : Z \times Y & \rightarrow & \mathbb{R}^n \times Y \\ (\mathbf{r},\mathbf{P}) & \mapsto & (\tilde{\mathbf{r}}(\mathbf{r},\mathbf{P}), \tilde{\mathbf{P}}(\mathbf{r},\mathbf{P})) \end{array}$$

maps $Z \times Y$ into itself.

Pick $(\mathbf{r},\mathbf{P}) \in Z \times Y$. By construction, $\tilde{\mathbf{P}}(\mathbf{r},\mathbf{P}) \in Y$. Also,

$$\begin{array}{rcl} \|\tilde{\mathbf{r}}(\mathbf{r},\mathbf{P})\| & \leq & \|\mathbf{d}^*\| + \|\mathbf{\Lambda}\| \cdot \|P(\mathbf{r},\mathbf{P})\| \cdot \|\mathbf{r}\| \\ & \leq & \|\mathbf{d}^*\| + \left(\|\mathbf{\Lambda}\| \cdot 1 \cdot \frac{\|\mathbf{d}^*\|}{1-\|\mathbf{\Lambda}\|}\right) \\ & = & \frac{\|\mathbf{d}^*\|}{1-\|\mathbf{\Lambda}\|}, \end{array}$$

which is a first condition $\tilde{\mathbf{r}}$ must satisfy to belong to Z. The second condition is $\tilde{\mathbf{r}}(\mathbf{r},\mathbf{P}) \geq \epsilon\mathbf{1}_n$, and that is satisfied by construction of $\tilde{\mathbf{r}}$. Hence, $F(Z \times Y) \subseteq (Z \times Y)$. This set is homeomorphic to a closed ball in $\mathbb{R}^{n+m}$, so by Brouwer's fixed-point theorem, F has a fixed point.

The utility of ϵ. In the preceding proof, we have imposed the conditions $\mathbf{r}_i \geq \epsilon$ and $\mathbf{P}_a \geq \epsilon$ for some $\epsilon > 0$ to make $\tilde{\mathbf{P}}$ a continuous function, but we haven't found it necessary to do that in our resolution algorithm. In fact, these conditions are *de facto* satisfied after a certain number of iterations. For any time interval, every itinerary i of lower class has a non-zero $\mathbf{p}_i$ at the second iteration, which makes it spill onto other itineraries. If all itineraries have a non-zero proportion of their spill that is destined to the corresponding itinerary of higher class, all arcs have a positive probability of being closed after several iterations, and each $\mathbf{r}_i$ is positive.

Convergence of the algorithm in practice. We can now describe what our resolution algorithm essentially does more synthetically. For a fixed time slice $[t_k, t_{k+1}]$, let $\mathbf{r}^{(n)}$ be the booking requests vector it computes at iteration n. Let $P^{(n)}$ be the diagonal $n \times n$ matrix containing the probabilities that each itinerary be closed, as computed by the algorithm at iteration n, as a function of $\mathbf{r}^{(n-1)}$ and $P^{(n-1)}$. Notice that $\|P^{(n)}\| < 1$ for all n. We have

$$\mathbf{r}^{(n+1)} = \mathbf{d} + \mathbf{\Lambda} P^{(n+1)} \mathbf{r}^{(n)}.$$

Consequently,

$$\begin{aligned} \|\mathbf{r}^{(n+1)} - \mathbf{r}^{(n)}\| &= \|\mathbf{\Lambda} P^{(n+1)} \mathbf{r}^{(n)} - \mathbf{\Lambda} P^{(n)} \mathbf{r}^{(n-1)}\| \\ &\leq \|\mathbf{\Lambda}\| \cdot \|P^{(n+1)} \mathbf{r}^{(n)} - P^{(n)} \mathbf{r}^{(n-1)}\|. \end{aligned}$$

Given the nice behavior of the $P^{(n)}$s and the fact that $\|\mathbf{\Lambda}\| < 1$, it is not surprising that, in practice, we see the quantity $\|\mathbf{r}^{(n+1)} - \mathbf{r}^{(n)}\|$ decreasing roughly geometrically as n increases. Over all our tests with unperturbed spill coefficients (see the next section), we have seen no violation of the inequality

$$\|\mathbf{r}^{(n+1)} - \mathbf{r}^{(n)}\| < 0.75 \cdot \|\mathbf{r}^{(n)} - \mathbf{r}^{(n-1)}\|.$$

4 Results

In this section we analyze the performance of our model on two large networks. They have been constructed using data from Air Canada. Spill coefficients and time distribution functions b_i of the demand do not come from real life data. We assert that this does not affect the validity of our experiment, since its point is to measure the accuracy of our model by comparing its passenger flow estimation with that of our simulation, run on exactly the same network, with the same spill coefficients and the same functions b_i.

4.1 Input description

Networks. Network 1 is made from a part of the Air Canada forecast weekly network for summer 2005. It has 29,715 itineraries and 14,731 arcs (in the sense of Section 2). Its arcs have on average 24.1% of their initial demand that comes from multi-arc itineraries. Its average spill coefficient is 0.125, and the average over all its itineraries i of $\sum_{j \in \mathcal{I}} \lambda_{j,i}$ is 0.627.

Network 2 is made from a part of an Air Canada weekly network of 2002. It has 23,948 itineraries and 5,180 arcs. Its arcs have on average 42.0% of their initial demand that comes from multi-arc itineraries. Its average spill coefficient is 0.077, and the average over all its itineraries i of $\sum_{j \in \mathcal{I}} \lambda_{j,i}$ is 0.697.

We used Air Canada forecasts for the average demands d_i. We used three aggregated fare classes, say A, B and C, with C being the lowest one. Coefficients of variation cv_i of the demand for itinerary i have been set to 0.3 when $d_i \geq 5$ and to 0.5 otherwise. The functions b_i controlling the time distribution of booking requests are one-step functions. For itineraries in class C, the booking request rate from the demand is constant over $[0, 1]$. For itineraries in classes B and A, requests from the demand are assumed to start at time 0.7 and 0.85 respectively, and then arrive at a constant rate. We have computed spill coefficients $\lambda_{i,j}$ that take into account the departure and arrival times of i and j, their number of legs, and their classes. $\sum_{j\in\mathcal{I}} \lambda_{j,i} \leq 0.75$ for any itinerary i.

Algorithm. The time discretization used was $0, 0.3, 0.55, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1$. Our stopping criterion for the iterative process described in Section 3.5 was that the difference, from one iteration to the next, in the total number of booking requests be smaller than one thousandth of the total demand for the time interval.

Simulation. For each test, we have made 5,000 simulations of the synthetic booking process (the simulation is described at the end of Section 2).

Computation time. It took our algorithm between 3 seconds and 7 seconds to terminate, depending on the test. For all tests and all time intervals, the stopping criterion has been met in at most 16 iterations. The simulation algorithm, more sensitive to increased demand, took between 7 hours and 5 days.

4.2 Analysis

We have made two series of tests for each of our networks: one in which the demand is incrementally modified, one in which the spill coefficients are modified, for a fixed demand.

For a given test, let $pass_i^{sim}$ and $pass_i^{mod}$ be the expected number of passengers on itinerary i, as estimated by the simulation and the model, respectively. In the following table, the *signed error* is $\left(\sum_{i\in\mathcal{I}} pass_i^{mod} - pass_i^{sim}\right) / \sum_{i\in\mathcal{I}} pass_i^{sim}$, and the *average deviation* is $\left(\sum_{i\in\mathcal{I}} \left|pass_i^{mod} - pass_i^{sim}\right|\right) / \sum_{i\in\mathcal{I}} pass_i^{sim}$. In the column titled *Spill demand*, we give the proportion of the initial demand that was spilled, according to the simulation (regardless of whether the passengers were lost or recaptured). The column titled *(Spilled requests)/demand* contains the total number of unsuccessful requests divided by the total demand, according to the simulation.

Bearing in mind that the goal is to estimate the expected revenue associated with the network, the most important statistic regarding the accuracy of the model is the signed error, and the values in Table 1 and Table 2 indicate that the model is remarkably accurate.

The two last columns in Table 1 and Table 2 indicate that there was in fact some important spilling activity in the booking processes associated with the networks. Notice also that the simulation results, being averages of 5,000 simulations for each network, are imperfect. The average initial demands generated in the simulations consistently show an average deviation of about 0.36% when compared to the expected values d_i. It is reasonable to infer that the simulation's passenger estimations similarly differ from the actual expected

Table 1: Effect of demand variation. Model passenger flow compared to simulation passenger flow, network 1.

Demand factor	Demand/ cap. (%)	Load factor(%)	Signed error (%) overall	1-arc it.	2-arc it.	Average dev. (%)	Spilled dem. (%)	(Spilled requests)/ dem. (%)
0.80	56.7	55.7	**0.008**	0.05	-0.24	0.49	3.2	4.4
0.90	63.8	61.9	**0.008**	0.07	-0.40	0.53	5.1	7.3
0.95	67.3	64.8	**0.007**	0.08	-0.49	0.57	6.3	9.1
1.00	71.6	68.2	**0.002**	0.09	-0.60	0.61	7.8	11.5
1.05	75.1	70.8	**−0.02**	0.09	-0.68	0.66	9.2	13.8
1.10	78.0	72.8	**−0.02**	0.10	-0.84	0.71	10.4	15.8
1.15	81.5	75.2	**−0.03**	0.10	-1.00	0.76	12.0	18.4
1.20	85.1	77.4	**−0.05**	0.10	-1.13	0.81	13.6	21.3
1.25	88.6	79.4	**−0.08**	0.09	-1.30	0.86	15.3	24.3
1.30	92.2	81.3	**−0.11**	0.07	-1.45	0.91	17.0	27.4
1.35	95.7	82.9	**−0.16**	0.03	-1.54	0.97	18.7	30.7
1.40	99.3	84.5	**−0.20**	0.006	-1.74	1.03	20.5	34.0

Table 2: Effect of demand variation. Model passenger flow compared to simulation passenger flow, network 2.

Demand factor	Demand/ cap. (%)	Load factor(%)	Signed error (%) overall	1-arc it.	2-arc it.	Average dev. (%)	Spilled dem. (%)	(Spilled requests)/ dem. (%)
0.73	60.1	59.2	**0.04**	0.06	-0.01	0.42	2.3	3.5
0.80	65.6	64.0	**0.08**	0.10	-0.01	0.48	3.8	5.6
0.87	71.1	68.4	**0.09**	0.14	-0.04	0.55	5.6	8.7
0.93	76.5	72.2	**0.07**	0.13	-0.12	0.61	7.8	12.4
1.00	82.0	75.8	**0.01**	0.09	-0.23	0.67	10.1	16.8
1.07	87.5	79.0	**−0.07**	0.02	-0.33	0.75	12.6	21.4
1.13	92.9	81.7	**−0.16**	-0.07	-0.44	0.85	15.2	26.4
1.20	98.4	84.2	**−0.27**	-0.17	-0.60	0.95	17.8	31.5
1.27	103.9	86.4	**−0.39**	-0.28	-0.77	1.07	20.3	36.7

passenger flows, according to the synthetic booking process. Hence, a non negligible part of the average deviations is due to the imperfection of the simulation's results.

Table 3 and Table 4 show the results of tests made to exhibit the effect of the magnitude of the spill coefficients on the accuracy of the model. For each test, the original spill coefficients have been multiplied by the number in the column *Spill factor*.

All four series of tests show that the model overestimates the number of passengers on one-arc itineraries and underestimates it on two-arc itineraries. This is particularly clear when spill coefficients are small. We believe that one major cause for the underestimation of passengers on two-arc itineraries is the equation 6, which overestimates their probability of being closed by assuming independence of events that are not independent.

The underestimation of passengers on two-arc itineraries entails an underestimation of arc requests, which lowers the probabilities $\mathbf{P}_a^k$, by lowering both the expectation and the variance of the random variables X_a^{k+1} used to compute them. That is consistent with the

Table 3: Effect of spill variation. Model passenger flow compared to simulation passenger flow, network 1, demand factor 1.1.

Spill factor	Load factor(%)	Signed error (%) overall	1-arc it.	2-arc it.	Average dev. (%)	Spilled dem. (%)	(Spilled requests)/ dem. (%)
1.2	73.6	**0.19**	0.26	-0.32	1.01	10.7	18.4
1.1	73.3	**0.07**	0.17	-0.60	0.80	10.6	17.0
1.0	72.8	**−0.02**	0.10	-0.84	0.71	10.4	15.8
0.9	72.5	**−0.07**	0.07	-1.02	0.69	10.3	14.7
0.8	72.1	**−0.12**	0.04	-1.20	0.67	10.1	13.7
0.6	71.5	**−0.15**	0.04	-1.44	0.65	9.9	12.2
0.4	71.0	**−0.13**	0.08	-1.62	0.65	9.7	11.0
0.2	70.5	**−0.09**	0.15	-1.74	0.69	9.4	10.0
0	70.1	**−0.03**	0.23	-1.82	0.84	9.2	9.2

Table 4: Effect of spill variation. Model passenger flow compared to simulation passenger flow, network 2. demand factor 1.0.

Spill factor	Load factor(%)	Signed error (%) overall	1-arc it.	2-arc it.	Average dev. (%)	Spilled dem. (%)	(Spilled requests)/ dem. (%)
1.2	76.5	**0.15**	0.11	0.26	0.83	10.4	19.8
1.1	76.1	**0.07**	0.10	-0.02	0.73	10.2	18.1
1.0	75.8	**0.01**	0.09	-0.23	0.67	10.1	16.8
0.9	75.5	**−0.03**	0.09	-0.41	0.61	10.0	15.5
0.8	75.2	**−0.06**	0.10	-0.55	0.59	9.8	14.4
0.6	74.8	**−0.10**	0.13	-0.76	0.59	9.6	12.5
0.4	74.4	**−0.10**	0.17	-0.90	0.61	9.3	11.0
0.2	74.1	**−0.07**	0.23	-0.97	0.64	9.1	9.8
0	73.7	**−0.03**	0.29	-1.01	0.81	8.8	8.8

model's overestimation of passengers on one-arc itineraries. There is more to it than that, however, as a close examination of the trends of those two biases, in the lower parts of Table 3 and Table 4, show.

In the upper parts of these tables, we see the model's signed error increasing as the spill factor increases from 0.6 to 1.2. We attribute this phenomenon to the equilibrium equation for booking requests, equation 2. Recall that, in our synthetic booking process as well as in the simulation, unsuccessful requests do not spill more than three times, nor do they rebound to the original itinerary when the alternative is closed. The model ignores this. We therefore expect it to overestimate the passenger flow when there is a sizable spilling activity, and when spill coefficients are high enough that the spill of second and higher orders becomes significant. For both networks, when the spill factor is 1.2, some itineraries i are such that $\sum_{j \in \mathcal{I}} \lambda_{j,i} = 0.9$. The average of $\sum_{j \in \mathcal{I}} \lambda_{j,i}$ is 0.752 for Network 1 and 0.836 for Network 2.

It may be argued, however, that it is the simulation that underestimates the recapture in high spill conditions. After all, the main reason why we limited the number of simulation recapturing waves at three instead of four or more was to keep its computing time reasonable.

Let us return to the first two series of tests, with constant spill coefficients and varying demand. For both networks, the signed error, while staying in the $\pm 0.1\%$ range for load factors below 80%, becomes markedly more negative for very high demands. The explanation we advance for this is quite simple: if an arc is expected to be almost full, the model cannot get it very wrong, and certainly cannot overestimate its passenger flow by much. Because the model's expected flow respects the network's capacity constraints, some of its overestimations are censored. In a normally loaded network, overestimations roughly balance underestimations (see the histograms below, showing data from the test made on network 1, with demand multiplied by 1.25). When, however, the initial demand for arcs is on average close to 100% of their capacity, as is the case for the higher demand tests on each network, this censoring effect is quite perceptible.

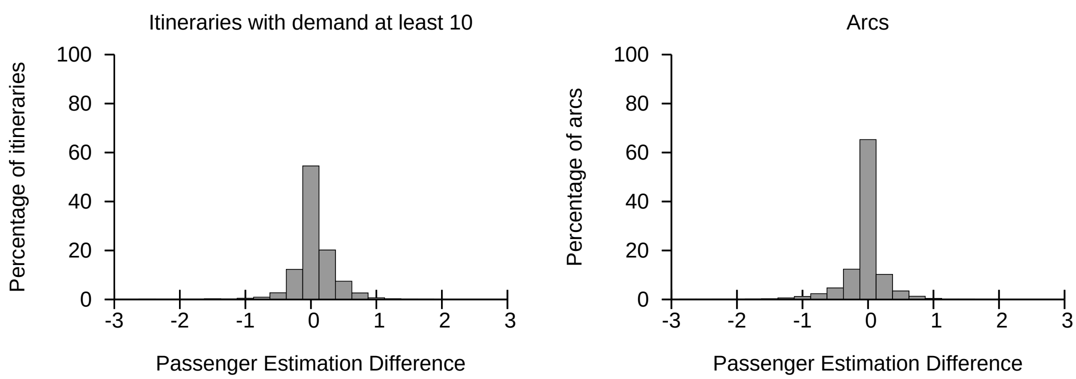

Figure 1: Passenger estimation difference between model and simulation.

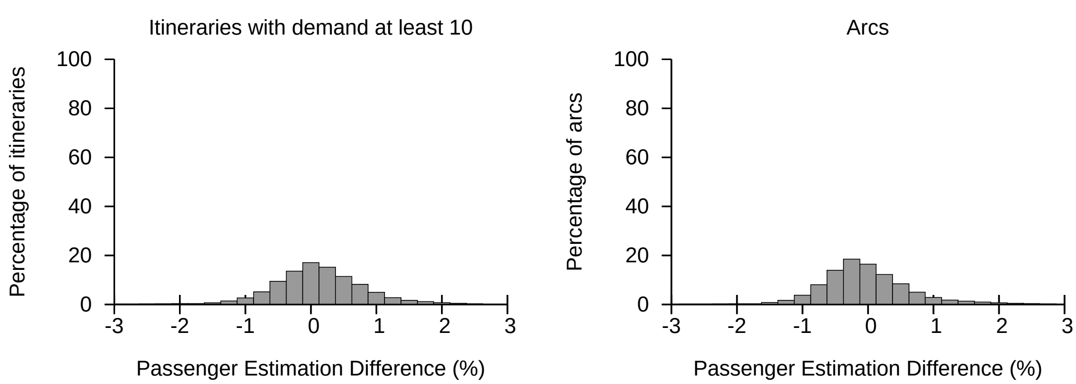

Figure 2: Passenger estimation difference between model and simulation (%).

5 Further discussion

5.1 Demand distribution

In the description of the problem (Section 2), the demand distribution for itineraries are required to be normal. It is, however, possible to do without this requirement.

Equation 10, determining the model's estimation of the expected spill attributed to an arc, is the only one relying on the properties of the normal law. Hence, even if demands for itineraries are assumed to follow laws other than the normal, if one deems the normal law appropriate to model the distribution of requests *on arcs*, (because requests on arcs are the aggregation of requests on several itineraries), then one can use the model as it is.

Moreover, one is free to model the distribution of requests on arcs according to the laws followed by the demands for itineraries using them, by simulating beforehand the densities of the sum of several such random variables, or to use analytic approximation, as it is done in Thom (1968) for gamma distributions, in the context of rain precipitation.

5.2 Revenue management

Reservation level revenue management strategy could be modeled within our framework. Suppose, for example, that arc a is physically suited to accommodate fare classes A and B, and that a number $res_A < cap_a$ is reserved for fare class A. Consider an itinerary i of class B using arc a, whose probability of being closed during the period $[t_k, t_{k+1}]$ is $\mathbf{p}_i^k = 1 - \left(\prod_{a \in i}(1 - \mathbf{P}_a^k)\right)$ in our model, without revenue management. Arc a may cause itinerary i to be closed because:

1. the number of arc requests of class B on arc a is greater than $cap_a - res_A$,
2. the number of arc requests of class B on arc a is less than $cap_a - res_A$, but the total number of arc requests on a is greater than cap_a.

The probability $\mathbf{P}_{a,B}^{k,res}$ of the first event occurring can be estimated the same way $\mathbf{P}_a^k$ is, provided that one carries the required new variables "number of requests of class B on arc a", $(a \in \mathcal{A})$. The probability of the second event occurring is estimated by $(1 - \mathbf{P}_{a,B}^{k,res}) \cdot \mathbf{P}_a^k$. Writing

$$\mathbf{P}_{a,i}^k = \mathbf{P}_{a,B}^{k,res} + (1 - \mathbf{P}_{a,B}^{k,res}) \cdot \mathbf{P}_a^k,$$

we use $\mathbf{p}_i^k = 1 - \left(\prod_{a \in i}(1 - \mathbf{P}_{a,i}^k)\right)$ as the estimation of the probability that i is closed during the period $[t_k, t_{k+1}]$. Several nested reservation levels can be handled similarly.

Notice that no hypothesis on the order in which the booking requests for the various fare classes occur is necessary.

5.3 Applications

Our model, like any other passenger flow model, can be used for fleet assignment revenue estimation and various scenario analyzes involving modification of the demand or perturbations in the network.

One obvious motivation for the conception of a computationally simple and structurally sound passenger flow model is its integration to a fleet assignment model, with the aim of improving its revenue estimation (see Barnhart et al. (2002), Kniker (1998) and Jacobs et al. (1999), Smith (2004)). We believe that our model is promising in this regard.

It would be interesting to see, for example, if the model could be efficiently used to recompute costs assigned to each fleet type on each leg, within an iterative algorithm that alternately solves a FAP and recomputes its objective function. At first sight, this would require several thousand calls of our resolution algorithm (one for each possible fleet type for each leg), but one could surely take advantage of the fact that the perturbation of one leg only does not greatly affect the passenger flow on the entire network.

One may also try to use our model directly in a branch and bound resolution algorithm for the FAP, readjusting the objective function as new sets of FAP variables are fixed.

5.4 Simulation and reality

We have measured our model against a simulation based on the synthetic booking process hypotheses listed in Section 2. Our results should not be taken out of this context. Evaluating the performance of our model in the real world would effectively amount to evaluating the quality of forecasts and the correspondence of the synthetic booking process hypotheses with reality. These hypotheses, although reasonable, could surely be improved and made more consistent with actual customer behavior.

6 Conclusion

We have presented a model whose objective is to find, for middle- and long-term planning purposes, approximation for expected passenger flow under the hypotheses listed in Section 2. These are, essentially, that we have:

1. forecasts of the demand distributions for the airline network's itineraries;
2. forecasts for the time distribution of the booking requests for these itineraries;
3. estimates of the proportion of customers who will settle for some given itinerary when their preferred itinerary happens to be unavailable.

Our model provides passenger flow estimations for a weekly network that differ from those of a simulation by about 0.1% on a normally loaded network, in a computation time of less than 10 seconds on a computer that requires about 3 days to run 5,000 simulations.

Clearly, the uncertainty in the forecast data enumerated above is much larger than 0.1%. It seems to us, however, that none of the three points above can be ignored by whoever tackles the problem of passenger flow forecasting. Hence, we present our model as an efficient tool for planning and demand scenario analysis, a tool that can only be as good as the input it receives.

References

Abara, J. 1989. Applying Integer Linear Programming to the Fleet Assignment Problem. *Interfaces* **19** 20–28.

Barnhart, C., T. Kniker, M. Lohatepanont. 2002. Itinerary-Based Airline Fleet Assignment. *Transportation Science* **36**(2) 199–217.

Jacobs, T.L., B.C. Smith, E.L. Johnson. 1999. O&D FAM: Incorporating Passenger Flows into the Fleeting Process. *AGIFORS Symposium Proc.*. New Orleans, LA.

Kniker, T. 1998. Itinerary-Based Airline Fleet Assignment. Ph.D. thesis, MIT, Cambridge, MA.

Phillips, R. L., D. W. Bowd, T. A. Grossman. 1991. An Algorithm for Calculating Consistent Itinerary Flows. *Transportation Science* **25**(3) 225–239.

Smith, B. C. 2004. Robust Airline Fleet Assignment. Ph.D. thesis, Georgia Institute of Technology.

Soumis, F. 1978. Planification d'une flotte d'avions. Ph.D. thesis, Université de Montréal, Montréal, Canada.

Soumis, F., A. Nagurney. 1993. A Stochastic, Multiclass Airline Network Equilibrium Model. *Operations Research* **41**(4) 721–730.

Subramanian, R., R. P. Scheff Jr., J. D. Quillinan, D. S. Wiper, R. E. Marsten. 1994. Coldstart: Fleet Assignment at Delta Air Lines. *Interfaces* **24** 104–120.

Swan, W. M. 2002. Airline Demand Distributions: Passenger Revenue Management and Spill. *Transportation Research Part E* **38** 253–263.

Thom, H. C. S. 1968. Approximate Convolution of the Gamma and Mixed Gamma Distributions. *Monthly Weather Review* **96**(12) 883–886.