Estimating Flight Departure Delay Distributions —A Statistical Approach With Long-term Trend and Short-term Pattern

Yufeng Tu, Michael Ball

Robert H. Smith School of Business and
The Institute for Systems Research,
University of Maryland,
College Park, MD, 20742

Wolfgang Jank †

Robert H. Smith School of Business,
University of Maryland,
College Park, MD, 20742

Summary. In this paper, we develop a model for estimating flight departure delay distributions. Such distributions are required by air traffic congestion prediction models. We identify and study major factors influencing flight departure delays, and develop a strategic departure delay prediction model. This model employs nonparametric methods for daily and seasonal trends. In addition, the model uses a mixture distribution to estimate the residual error. In order to overcome problems with local optima in the mixture distribution, we develop a global optimization version of the Expectation Maximization algorithm borrowing ideas from Genetic Algorithms. The model demonstrates reasonable goodness of fit and robustness. We use flight data from Denver International Airport in the year 2000 for a case study.

Keywords: smoothing spline, Expectation Maximization (EM), Genetic Algorithm (GA), airline delay, airspace congestion, delay distribution

1. Introduction

The U.S. National Airspace System (NAS) is inherently highly stochastic. Yet, many decision support tools for air traffic flow management take a deterministic approach to problem solving. For example, to predict when an airspace sector will become overloaded, the Federal Aviation Administration (FAA) employs a module called *Monitor Alert*. This tool predicts airspace traffic levels by projecting, for each planned flight, time/space epochs through the airspace based on a single flight plan (route) and a single estimated departure time. The estimated departure time used

†Corresponding author: Wolfgang Jank, Department of Decision and Information Technologies, Robert H. Smith School of Business, College Park, MD 20742-1815 (Email: wjank@rhsmith.umd.edu).

is typically the flight's scheduled departure time. This deterministic approach fails to capture three significant stochastic factors: i) uncertainty in a flight's departure time (including the possibility of flight cancellation), ii) changes in a flight's route immediately before takeoff or after the flight is airborne and iii) airspace queueing effects. On-going research and development activities are seeking to develop stochastic models to replace this deterministic system, see e.g. Chandran (2002). This paper represents a component of that research that is addressing factor i). That is, in this paper, we describe a model for estimating flight departure delay distributions. We emphasize that a major objective is to produce not just point estimates but estimates of the entire distribution since the congestion estimation models envisioned require delay distribution functions. It is perhaps unnecessary to emphasize the potential benefits of reducing airspace congestion and delays. As an example, delays directly attributed to air traffic control actions are estimated to cost airlines 2.9 billion dollars in 1998 in addition to the cost of delays borne by passengers (ATA, 1999).

The Bureau of Transportation Statistics (BTS) releases summary statistics and basic analysis on airline performance each month. Most of its delay analysis focuses on arrival delays rather than the departure delays since arrival delays are more closely related to ultimate passenger satisfaction. On the other hand, when trying to understand the source of arrival delays and airspace congestion in general, study of departure delays becomes quite relevant. We should also note that the BTS analysis and most prior studies of airspace delays typically only provide average delay statistics and do not focus on estimates of distribution functions. SimAir, a modular airline simulation tool developed in the year 2000, is based on the raw historical aggregate distribution (Rosenberger et al., 2000). Although the raw historical distribution is a simple way to capture departure delays, it can potentially be too sensitive to specific random variations in the data set on which it is based. In our analysis, we try to characterize the underlying mechanisms behind the delays, then model and regenerate the delays by using functional characterizations.

The specific delay value of importance as input to advanced air traffic prediction models is the push-back delay, which measures the discrepancy between the scheduled departure time and the actual departure time from the gate (push-back time). Other delays, such as taxi-out delay, delay in the air, taxi-in delay, and arrival delay, are all generated after the flight leaves the gate. There is a body of related, prior research that uses models to estimate departure delays or employs departure delay estimates within broader models. These models typically address problems involving airport surface congestion. For example, Odoni et al. (1994) developed a non-homogeneous queueing model to analyze congested airports. Shumsky (1997) extended this model and estimated take-off times under non-steady state conditions. Idris et al. (2002) developed a queueing model for taxi-out time estimation. The results of our work could potentially be used as inputs into these models. A key component of our model is the estimation of the delay propagation effect. Delay built-up from previous flights is known as the delay propagation and its effects on delays have been studied in several prior papers (see for example, Beatty (1998), Schaefer and Millner (2001) and Wang et al. (2003)). Our work provides a functional characterization of this effect at a single airport and uses the underlying function as input into the departure delay estimates.

In addition to the daily propagation effect, many other factors influence departure delay, such as weather conditions, holiday demand surges, luggage problems, mechanical problems, airline policies, airport congestion, etc. Instead of studying the impact of each factor alone, we group factors into three major categories: seasonal trend, daily propagation pattern and random residues. Then we build our delay estimation model based upon these categories. To estimate the seasonal trend and the daily propagation pattern, we employ a smoothing spline model. Its nonparametric nature eliminates the need for assuming a rigid form for the dependence of the response on the predictors (Hastie and Tibshirani, 1990). In our analysis, we did not have prior knowledge of the form of the seasonal trend or the daily propagation pattern. In addition, by using a smoothing spline, we can treat time as a continuous factor, which is clearly appropriate since the delay at the end of one month will not vary significantly from the delay at the beginning of the next month; there is a similar smooth fluctuation in delay over the course of a day making the smoothing spline an advantageous approach for addressing the daily propagation effect. Finally, we assume a mixture model for the residuals and estimate the mixture-components using the EM (Expectation Maximization) algorithm. The EM algorithm is known for its fast convergence, stability and convenience in mixture density estimation (Bilmes, 1998). However, the EM algorithm typically results only in a local optimum instead of finding the global optimal value.

In an effort to find the global optimum, we develop a global version of EM by combining EM with the ideas of a Genetic Algorithm (GA). GAs were first introduced by Holland (1975) based on the principles of natural selection or “survival of the fittest”, the evolution of species . The GA approach has been applied to many areas including marketing, biology, engineering, etc. In this paper, we apply GA to the maximum likelihood estimation of mixture distributions to overcome the local maximum problems associated with the traditional EM algorithm.

For this case study, we select Denver International Airport, which is a hub for United Airlines (UA). Our model shows promising results in estimating the delay distributions and predicting the departure delay probabilities. Although the case study is for Denver International Airport and UA only, our model can be readily generalized to other airports or other airlines as well.

Section 2 describes the application that motivates our work. Section 3 introduces the model structure and assumptions. Section 4 proposes a Genetic Algorithm version of the EM algorithm. In Section 5, we present the case study, the data and computational results. Section 6 summarizes our findings and points out areas for further research.

2. Application

In order to manage air traffic flows within the U.S., the Federal Aviation Administration (FAA) has contracted with the Volpe National Transportation Systems Center to operate the enhanced traffic management system (ETMS). Airspace sectors are three-dimensional volumes of airspace managed by a single team of controllers. Safety concerns dictate that controller workload should be kept within certain bounds and so limits are placed on the number of aircraft that can simultaneously occupy a sector. The *Monitor Alert* function within ETMS provides predictions when such overloads will occur (VNTSC, 2003). The goal of our work is to replace the current deterministic model for providing such predictions with a stochastic one.

We now provide a slightly simplified version of how *Monitor Alert* operates and then describe our approach to enhance it. We start by defining a set of variables defining future states, which we initially assume are deterministic. Later we will relax this assumption, by treating them as random variables.

$$\begin{aligned} F &= \text{ set of flights under consideration} \\ N(w,t) &= \text{ the number of flights occupying sector } w \text{ at time } t \\ I_i(w,t) &= 1 \text{ if flight } i \text{ occupies sector } w \text{ at time } t \\ & \quad 0 \text{ otherwise} \end{aligned}$$

ETMS continuously updates estimates of $N(w,t)$. The monitor alert function then compares these with sector capacities so as to determine if an alert is necessary. Since $N(w,t) = \sum_{i \in F} I_i(w,t)$, the process of computing $N(w,t)$ can be reduced to computing $I_i(w,t)$ for each flight i. ETMS maintains a prediction of the *flight plan* for each flight. Given an estimate of flight i's departure time, t^i_{dep}, the flight plan provides a deterministic prediction of the times at which the flight will pass through a series of airspace locations along its planned route. Specifically, it predicts the time at which the flight will pass over sector boundaries, and thus determines $I_i(w,t)$. Let τ denote the present time and t^i_{sch} the scheduled departure time of flight i, then ETMS and monitor alert operate as follows: if $\tau \leq t^i_{sch}$, t^i_{dep} is set equal to t^i_{sch} and if the flight has not departed but $\tau > t^i_{sch}$, t^i_{dep} is set equal to τ. Once the flight has departed, its airspace position and flight plan are dynamically updated based on current information.

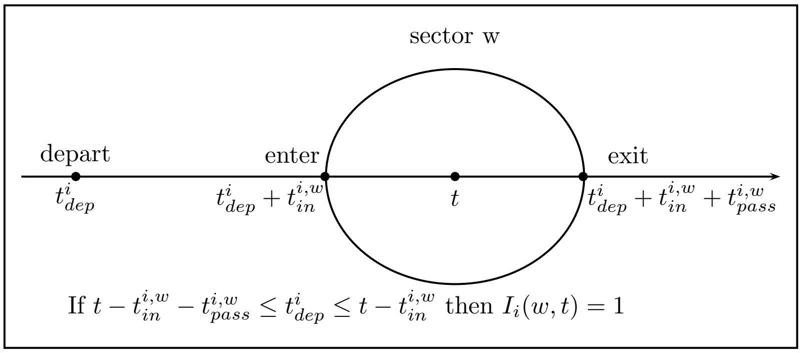
sector w
depart
enter
exit
t^i_{dep}
$t^i_{dep} + t^{i,w}_{in}$
t
$t^i_{dep} + t^{i,w}_{in} + t^{i,w}_{pass}$
If $t - t^{i,w}_{in} - t^{i,w}_{pass} \leq t^i_{dep} \leq t - t^{i,w}_{in}$ then $I_i(w,t) = 1$

Fig. 1. A Typical Flight Path Through Sector w

There are many stochastic elements to this problem—our goal here is to address one of them, namely the possible variation in the flight's departure time. Specifically, for the case where $\tau \leq t^i_{sch}$, we treat t^i_{dep} as a random variable, which implies that $I_i(w,t)$ and $N(w,t)$ are also random variables. Then, in the above procedure we can use $E[N(w,t)] = \sum_{i \in F} E[I_i(w,t)]$. We note that generally flights have three states: on ground when $\tau \leq t^i_{sch}$, on ground when $\tau > t^i_{sch}$, and airborne. Our modifications only apply to flights in the first category. For these flights, since $E[I_i(w,t)] = Pr[I_i(w,t) = 1]$, we need to consider the problem of computing the probability a flight is in a sector at a given time. Now, let $t^{i,w}_{in}$ be the time required for flight i to reach the sector boundary of w under the current flight plan estimate and $t^{i,w}_{pass}$ be the time required for fight i to pass through sector w under the current flight plan (see Figure 1). Then,

$$Pr[I_i(w,t) = 1] = Pr(t - t^{i,w}_{in} - t^{i,w}_{pass} \leq t^i_{dep} \leq t - t^{i,w}_{in})$$

In this paper, we provide methods for estimating the departure time t^i_{dep}, which allows computation of the probability given above. Specifically, we model the departure delay distribution, which measures the discrepancy between the actual departure time and the scheduled departure time.

3. The Model Structure

Our model takes into account two types of delay structures: the seasonal trend and the daily propagation pattern. The delay follows the daily propagation pattern every day and at the same time it follows the seasonal trend throughout the year. Random residuals capture the additional variation not accounted for by these two structures (see Figure 2).

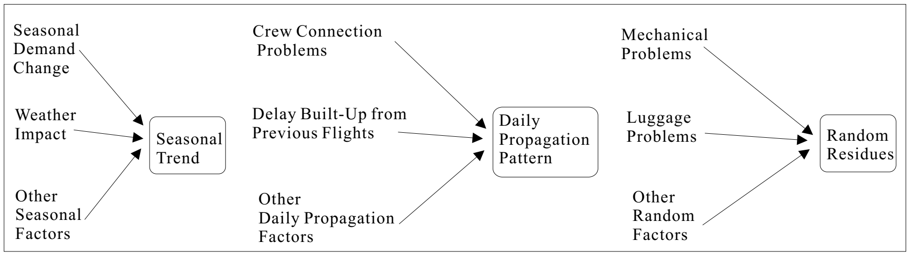

Fig. 2. Factors Influencing Departure Delay

Instead of attempting to enumerate all the possible factors on the left hand side in Figure 2, we use the structures on the right hand side to model the delay. Therefore, the departure delay for each individual flight may be split into three parts: the main effect of seasonal trend, the main effect of the daily delay propagation and random errors.

The model formulation is as follows: Let y_{sti} be the departure delay for the flight i, on day s at scheduled departure time t. Let $f(s)$ be the seasonal trend, $\varphi(t)$ be the daily delay pattern, and ϵ_i denote the random error. We propose an additive model of the form

$$y_{sti} = f(s) + \varphi(t) + \epsilon_i \tag{1}$$

where the seasonal trend is a function of day s and the daily delay pattern is function of time t. We assume that the random error is independent of day s and time t.

Note that in this model we assume the effects of seasonal trend and the daily propagation pattern are additive. This model also assumes that the seasonal trend and the daily propagation pattern are independent from each other and that the random errors only have an additive effect on the delays. That is, without the seasonal trend, each day does not have an impact on the other; without either the seasonal trend or the propagation pattern, the residuals distribute as iid (identical independently distributed).

3.1. Seasonal Trend

We model the seasonal trend using a smoothing spline. This method allows us to trace the seasonal trend without having to assume a rigid form for the dependence of the response on the predictors. It also provides a good fit to the data without exhibiting excessive local variability (Green and Silverman, 1994).

Let π_v be a set of knots chosen from the observed time series of data, then a polynomial spline of order d is given by

$$f(s) = \beta_0 + \beta_1 s + \beta_2 s^2 + \cdots + \beta_d s^d + \sum_{v=1}^{V} \beta_{dv}(s - \pi_v)_+^d \tag{2}$$

where $a_+ = aI_{[a \geq 0]}$ denotes the positive part of the function a. Let β= $(\beta_0, ...\beta_d, \beta_{d1}, ..., \beta_{dv})^T$ be the vector of coefficients in (2). The choice of V and d strongly influence the local variability of the function f. One can measure the degree of departure from a straight line by defining a roughness penalty

$$PEN_m = \int_H (D^m f(s))^2 ds \tag{3}$$

where D^m, $m = 1, 2, ...$, denotes the mth derivative of the function f. Using $m = 2$ and $d = 3$ leads to the commonly known cubic smoothing spline. We find $f(s)$ by minimizing the penalized residual sum of squares:

$$\sum_{s=1}^{365} (\bar{x}_s - f(s))^2 + \lambda_s \int_H (f''(s))^2 ds \tag{4}$$

where $[1, 365] \subseteq H \subseteq \Re$ and λ_s is the smoothing parameter. $\bar{x}_s$ denotes the average daily delay and is calculated by

$$\bar{x}_s = \frac{\sum_t \sum_i x_{sti}}{\sum_t n_{st}} \qquad s = 1, 2, 3, \ldots, 365. \tag{5}$$

where x_{sti} denote the observed departure delay for flight i, on day s at time t; n_{st} denotes the number of flights on day s at time t.

The parameter λ_s controls the smoothness of the spline. Large values of λ_s produce smoother curves while smaller values produce more wiggly curves. In our study, we balance data-fit and smoothness by choosing an equilibrium value of λ_s (see Section 5.4). As to the number of knots V, we set knots at the unique values of $\bar{x}_s$.

3.2. Daily Propagation Pattern

Since the airline operating resources are linked together, delaying one flight can affect other flights. Among the inter-connected resources affected by delayed flight operations are crews, aircrafts, passengers, and gate spaces. Because of this connectivity, airline departures are quite sensitive to delays earlier in the day—the delay of one flight tends to propagate in time to many others.

The same smoothing technique is employed to model the daily propagation pattern. We define the daily propagation function $\varphi(t)$ to be the one that minimizes the penalized residual sum of squares:

$$\sum_{t=00:00}^{24:00} (\bar{x}_t - \varphi(t))^2 + \lambda_t \int_P (\varphi''(t))^2 dt \tag{6}$$

where $[00:00, 24:00] \subseteq P \subseteq \Re$, and λ_t is the roughness penalty for the daily delay propagation pattern. The daily delay propagation function is calculated from the deseasonalized data and $\bar{x}_t$ denotes the average departure delay during time interval T. Let x'_{sti} denote the delays after the seasonal trend is removed,

$$x'_{sti} = x_{sti} - \hat{f}(s) \qquad \forall s, t, i \tag{7}$$

$\bar{x}_t$ can be obtained by

$$\bar{x}_t = \frac{\sum_t^{t+T} \sum_{s=1}^{365} \sum_i x'_{sti}}{\sum_t^{t+T} \sum_{s=1}^{365} n_{st}} \qquad t = 00:00, T, 2T, \ldots, 24:00. \tag{8}$$

We choose λ_t and V in a similar manner as before.

3.3. Finite Mixture Distribution for Residuals

The residuals are defined as the errors without either the seasonal trend or the daily propagation patterns. These residuals originate from many random factors such as, a customer running late, mechanical problems, etc. To capture the residual delay distribution, we apply a finite mixture model. Some of the underlying mechanisms suggest the use of a mixture model: some flights depart earlier than the scheduled departure time; many flights depart around the scheduled time and some fights have long delays.

We model the density distribution g of these residuals as a function of J-components mixture in $\Re^d$. The random residuals ϵ_i are calculated by removing the daily propagation pattern and the seasonal trend from the original data,

$$\epsilon_i = x_{sti} - \hat{f}(s) - \hat{\varphi}(t) \tag{9}$$

The mixture density of the ith point $(i = 1, ..., n)$ can be written as

$$g(\epsilon_i|\theta) = \sum_{j=1}^{J} p_j \psi_j(\epsilon_i|\alpha_j) \tag{10}$$

where p_j ($p_j \in [0,1], \sum_{j=1}^{J} p_j = 1$) is the mixing proportion and $\psi_j(\epsilon|\alpha_j)$ is a density function depending on parameter α.

The vector parameter to be estimated is $\theta = (p_1, ..., p_J, \alpha_1, ..., \alpha_J)$. In this paper, we assume normal group-conditional densities for which we write

$$\psi_j(\epsilon_i|\alpha_j) = \psi_j(\epsilon_i|\mu_j, \Sigma_j), \alpha_j = (\mu_j, \Sigma_j) \tag{11}$$

where μ denotes the mean and Σ denotes the covariance matrix. The log-likelihood is given by

$$logL(\theta|\epsilon) = \sum_{i=1}^{n} log \left\{ \sum_{j=1}^{J} p_j \psi_j(\epsilon_i|\alpha_j) \right\}. \tag{12}$$

One can maximize this log-likelihood by assuming that some of the information is not observed. Specifically, we assume that the residual ϵ_i arise from one of the J groups. Let $z = (z_1, ..., z_n)$ be the corresponding J-dimensional group indicator vectors, which is unobserved (or missing). Let us add the observed data $\epsilon = (\epsilon_1, ..., \epsilon_n)$ and form the complete data, then we have $\omega = (\epsilon, z)$. The log-likelihood of θ can be written as

$$logL_c(\theta|\omega) = \sum_{i=1}^{n} \sum_{j=1}^{J} z_{ij} \left\{ logp_j + log\psi_j(\epsilon_i|\alpha_j) \right\} \tag{13}$$

The root of this likelihood function generally ends up with a local maximum in the interior of the parameter space (McLachlan and Peel, 2000). In the following section, we discuss how we use a variant of the Expectation Maximization (EM) algorithm to approximate the global maximum of the likelihood function.

4. A Genetic Algorithm Version of the EM Algorithm

The traditional Expectation Maximization (EM) algorithm is a very popular tool for maximizing an objective function (Dempster et al., 1977). This popularity stems from its simple implementation and guaranteed monotonic increase of the likelihood during optimization. However, one known limitation is its local nature: it can be easily trapped in a local maximum of the likelihood function (Boyles, 1983). In this paper, we propose a new version of the EM algorithm by employing the Genetic Algorithm for the search of a global optimum.

The EM algorithm is an iterative procedure with two steps: E-step and M-step. The E-step computes the expectation of the complete data log likelihood, conditional on the observed data (flight departure delays) and the current parameter values.

$$Q(\theta|\theta^{(k-1)}) = E[logL_c(\theta|\omega)|\epsilon; \theta^{(k-1)}] \tag{14}$$

where k denotes the kth iteration.

This conditional expectation can be simplified using Equation (13)

$$Q(\theta|\theta^{(k-1)}) = \sum_{i=1}^{n}\sum_{j=1}^{J}\eta_{ij}^{(k-1)}\left\{logp_j + log\psi_j(\epsilon_i|\alpha_j)\right\} \tag{15}$$

where $\eta_{ij}^{(k-1)} = E(z_{ij}|\epsilon_i;\theta^{(k-1)})$ is the posterior probability that ϵ_i belongs to the jth component in the mixture.

The M-step updates the corresponding parameter estimates. In the kth iteration, EM finds the value θ^k which satisfies

$$Q(\theta^k|\theta^{(k-1)}) \geq Q(\theta|\theta^{(k-1)}) \tag{16}$$

for all θ in the parameter space.

Since we assume normal mixture distributions in Equation (11), we have a closed form for the EM algorithm (McLachlan and Peel, 2000):

- E-step : For $i = 1, ..., n$ and $j = 1, ..., J$ compute

$$\eta_{ij}(\theta^k) = \frac{p_j^k\psi(\epsilon_i|\mu_j^k, \Sigma_j^k)}{\sum_{j=1}^{J} p_j^k\psi(\epsilon_i|\mu_j^k, \Sigma_j^k)} \tag{17}$$

- M-step : Set $\theta^{k+1} = (p_1^{k+1}, ..., p_J^{k+1}, \mu_1^{k+1}, ..., \mu_J^{k+1}, \Sigma_1^{k+1}, ..., \Sigma_J^{k+1})$ with

$$\begin{aligned} p_j^{k+1} &= \frac{1}{n}\sum_{i=1}^{n}\eta_{ij}(\theta^k) && (18)\\ \mu_j^{k+1} &= \frac{\sum_{i=1}^{n}\eta_{ij}(\theta^k)\epsilon_i}{\sum_{i=1}^{n}\eta_{ij}(\theta^k)} && (19)\\ \Sigma_j^{k+1} &= \frac{\sum_{i=1}^{n}\eta_{ij}(\theta^k)(\epsilon_i - \mu_j^{k+1})(\epsilon_i - \mu_j^{k+1})^T}{\sum_{i=1}^{n}\eta_{ij}(\theta^k)}. && (20) \end{aligned}$$

This iterative method produces the local optimal value for $\theta = (p_1, ..., p_J, \alpha_1, ..., \alpha_J)$, from which we construct the mixture density distribution g.

Genetic Algorithm (GA) has been applied to many areas including marketing, biology, engineering, etc., for finding a global optimum (Goldberg, 1989). It was first proposed by Holland (1975). The basis for this algorithm comes from the observation that a combination of sexual reproduction and natural selection allows nature to develop living species that are highly adapted to the natural environment.

An implementation of the a Genetic Algorithm begins with a population of chromosomes. One then evaluates these structures and allocates reproductive opportunities in such a way that those chromosomes, which represent good solutions to the target problem, are given a chance to produce their offspring. It is expected that some members of this new population who acquire the best characteristics of both parents can better adapt to the environmental conditions, providing an improved solution to the problem.

In this study, we represent each parameter in $(p_1, ..., p_J, \alpha_1, ..., \alpha_J)$ as one gene. Therefore the vector $\theta = (p_1, ..., p_J, \alpha_1, ..., \alpha_J)$ is a string of parameters just as a chromosome consists of a string of genes. The fitness function is our target function that evaluates the maximum likelihood value in the EM algorithm. This GA process can be visualized as follows:

Step1 Randomly generate a initial population of l chromosomes, which serves as the pool of parents. Initial Pool $= \{\theta_1^0, ...\theta_l^0\}$.

Step2 Evaluation: Evaluate the fitness of each chromosome by calculating $logL(\theta|\epsilon)$ in Equation (12) using θ as the starting value in the EM algorithm. Record the corresponding maximum likelihood value MLK $= \{MLK_1, ...MLK_l\}$

Step3 Crossover: Randomly choose a pair of parents from the initial pool, exchange the genes between two strings (parents) at random positions to generate a pair of children. Specifically, crossover p_j or α_j between two parents randomly. Repeat this step until we get l children. Children Pool $= \{\theta_1^c, ...\theta_l^c\}$.

Step4 Mutation: Specify a fixed, very small probability of mutation p_m in advance. A random number between 0 and 1 is generated; if it falls within the p_m range, the new child organism's chromosome is randomly mutated, which means p_j or α_j could be randomly changed.

Step5 Update: Go to Step 2, record the fitness of all parents as $MLK^0 = \{MLK_1^0, ...MLK_l^0\}$. Record the fitness of children as $MLK^c = \{MLK_1^c, ..., MLK_l^c\}$. Choose the best l chromosomes from all parents and children to remain in the gene pool. Update MLK from $\{MLK^0 \cup MLK^c\}$; update the gene pool correspondingly.

Step6 Repeat Step 2-4 until the Nth generation is produced.

Please note that $\sum_{j=1}^{J} p_j = 1$, so we only need to calculate $(J-1)$ probabilities in this algorithm. The Genetic Algorithm approach has the benefit of seeking a global optimum: by crossover, it conducts a search from a population of points rather than a single point; by mutation, it creates random small deviations, thus exploring the nearby areas. The average degree of fitness generally increases by this procedure for the later generations, since only the best organisms from the previous generation are selected for breeding. GAs are very useful in problem domains that have a complex fitness landscape since this recombination procedure is designed to move the search away from a local optimal value where a traditional hill climbing algorithm might stall.

5. Case Study

We select Denver International Airport and United Airlines with data from the year 2000 for our case study, but other airports and airlines also fit into our modeling framework.

5.1. Data

The data used in this paper is based on Airline Service Quality Performance (ASQP) data, which are collected by DOT (US Department of Transportation) under authority of 14 Code Federal Regulations (CFR). Any airline with more than 1 percent of total domestic enplanements is required to report performance data to DOT.

In the Year 2000, 10 carriers met the reporting requirement threshold. Among them, American, Northwest, United, and US Airways use ACARS (Aircraft Communications Addressing and Reporting System) exclusively; Continental, Delta, and Trans World Airlines used a combination of ACARS and manual reporting systems; and America West, Southwest, and Alaska Airlines relied solely on their pilots, gate agents and/or ground crews to record arrival times manually (FAA, 2002).

We chose the Year 2000 to avoid the September 11th attacks and their consequential impacts on airline performance.

To validate our model, we set 70% of the total data as the training set and the rest 30% as the holdout set. All parameter estimates were calculated from the training set. The prediction outputs and robustness tests were computed based on the holdout data set.

5.2. Calculating the Delay

The delay in this study is the pushback delay which measures the difference between the actual departure time (LVETIME) and the OAG scheduled departure time (OAG_DEP). The ASQP database does not contain the pushback delay field. So we calculate the delay using the following algorithm: $DELAY = LVETIME - OAG\_DEP$. However there are two special situations that merit discussion:

First, a flight may be originally scheduled to depart late at night, but the actual departure time is pushed to early in the morning of the next day. For example, on January 23, 2000, United Airlines (UA) had a flight from Denver (DEN) to Seattle-Tacoma International Airport (SEA). The scheduled departure time was 20:00 in the evening, but the actual departure time turned out to be 00:18 in the morning next day. The simple algorithm shows that this flight took off 19 hours earlier than its scheduled time when actually it is a delayed flight. To address such instances, we

Table 1. Summary Statistics of the Actual Delay

Min	1st Quartile	Median	Mean	3rd Quartile	Max	Std.
18.00	-1.00	3.00	18.16	20.00	802.00	37.16

add the following conditions to our delay calculation algorithm: If $DELAY < -200$ minutes, then $DELAY = LVETIME - OAG\_DEP + 24 * 60$. We assume that usually a flight will not take off 3 hours earlier than the scheduled time (The dataset did not specify whether the actually departure date is the same as the scheduled departure date).

Second, there is a chance that a flight that is scheduled to depart very early in the morning actually departs very late at night the preceding day. Although this situation never happened in the Denver-UA case study, we still build this situation into our delay calculation algorithm as it may happen in other cases: if $DELAY >$ $(23 * 60)$ minutes, then $DELAY = LVETIME - OAG\_DEP - 24 * 60$, since these flights are probably early flights instead of being delayed for more than 23 hours.

5.3. Data Preparation

In the Year 2000, based on the ASQP database information, 92,865 UA flights departed from Denver in total, which is about 254 flights a day on average, excluding some of the missing data (both OAG_DEP and OAG_ARR are 0000). The general descriptive statistics for *delay* (in minutes) are shown in Table 1.

By plotting average daily delay throughout the whole year, we readily identify March 20th as an outlier.

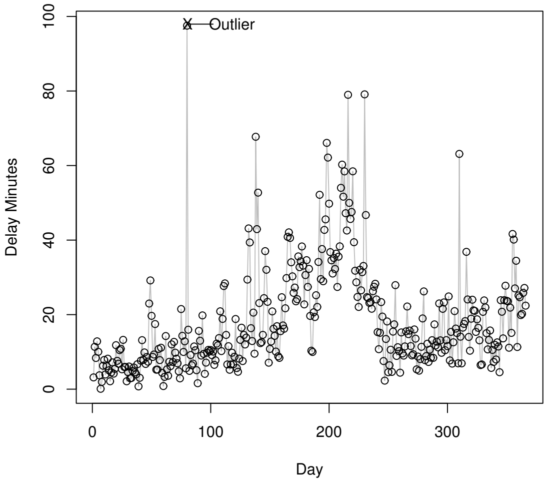

Fig. 3. Average Daily Delay in Year 2000

The average delay on that day is significantly larger than that of other days. The following excerpt from the NCAR (the National Center for Atmospheric Research) News Release explains what happened on that day:

> *Cancellations and delays due to icy weather can cost airlines millions of dollars in a single day. On March 20, 2000, icing conditions at Denver International Airport forced Air Wisconsin to cancel 152 flights. United canceled 159 outbound and 140 inbound flights the same day, most because of weather (NCAR, 2002).*

March 20th was a special case with extreme icing condition. Politovich et al. (2002), in their paper describe the results of a survey sent out to pilots that flew in and out of Denver. One of the question was "Was March 20th an extremely unusual event for DEN?", 23 out of 26 pilots answered *Yes*.

Therefore we exclude March 20th in our study since regular seasonal patterns and daily propagation patterns may not apply on that particular day.

5.4. Estimating the Seasonal Trend

Please note that Year 2000 has 366 days. Since we exclude March 20th, there are only 365 days remaining in the dataset. A smoothing spline that represents the seasonal trend is extracted from the 365 daily delay means. Figure 4(a) shows this trend.

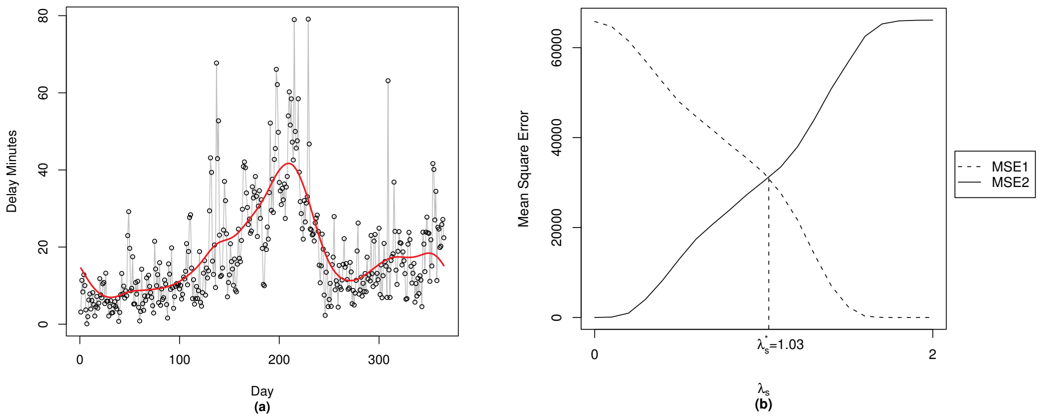

Fig. 4. Estimating the Seasonal Trend: (a) A fitted smoothing spline that represents the seasonal trend; (b) The compromise between the goodness of fit and fluctuation in curve estimation.

The y-axis gives the average delay in minutes and the x-axis shows all days in Year 2000. The delays are high in summer and winter while low in spring and fall, which suggests a clear seasonal trend. The solid line is the cubic smoothing spline $\hat{f}(s)$, for the seasonal trend. This spline is calculated with $\lambda_s = 1.03$.

We choose λ_s in the following way. We want $f(s)$ to show a reasonable data fit and not to display too much rapid fluctuation. To achieve both goals, we calculate the mean square error (MSE) between the fitted spline and a simple regression to measure the amount of fluctuation; we also calculate the MSE between the spline and all data points to measure the goodness of fit. Figure 4(b) shows us the compromise between the goodness of fit and the fluctuation.

MSE1 measures fluctuation, which is the difference between the fitted spline and a simple linear regression; MSE2 measures the data fit, which is the difference between the fitted spline and all original data points. As λ_s varies from small to large, MSE1 decreases and MSE2 increases. The sum of MSE1 and MSE2 is a constant which equals to the difference between the simple regression and the all data points. Figure 4(b) shows that a good fit of the data and less fluctuation are two conflicting aims in the curve fitting. We have to compromise between these two aims when choosing a value. When $\lambda_s = 1.03$, these two curves meet and we use this point in our analysis. We also explore a range of alternative values for λ_s and test the robustness of our results in Section 5.7.

5.5. *Estimating the Daily Propagation Pattern*

After the seasonal trend is removed, we use the same techniques in estimating the the daily propagation pattern. Figure 5 presents a fitted smoothing spline $\hat{\varphi}(t)$ and its corresponding roughness penalty.

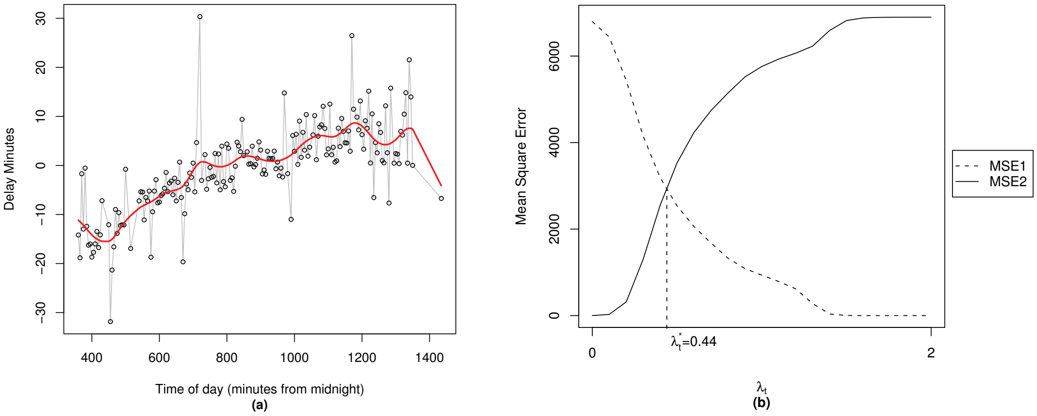

Fig. 5. Estimating the Daily Propagation Pattern: **(a)** A fitted smoothing spline that represents the daily propagation pattern; **(b)** The compromise between the goodness of fit and fluctuation in curve estimation.

In Figure 5(a) the x axis is the scheduled departure time (from 00:00 to 24:00 calculated in minutes), and y axis is the delay in minutes. Each point represents $\bar{x}_t$ at time t. Please note that no flight was scheduled to depart before 6:00 and after 24:00 in Year 2000. So the x axis does not cover all the domain of [00:00, 24:00]. We can see that the average delay gradually builds up as time goes by and decreases when it is deep into the night. The solid line represents $\hat{\varphi}(t)$ the estimated daily propagation pattern. The roughness penalty λ_t is set at 0.44 where MSE1 (measure of fluctuation) and MSE2 (measure of the goodness of fit) meet. See Figure 5(b).

Figure 6 illustrates an interesting pattern in our data. We observed a very distinctive curve when plotting the average delay against the *actual* departure time: during the day, the delay follows a certain kind of pattern—it increases sharply within a certain time interval and then drops; this up-and-down impulse keeps repeating itself over and over again. One may notice that the delay is extremely high in the very early morning. Please note that the x-axis of this plot is the *actual* departure time. Since no flight is scheduled to depart in the very early morning, a flight that actually departs at that time indicates that the flight has been delayed for a long time. When random sampling 30% of the total data, we notice that this pattern stays more or less the same (Figure 6(b)). This suggests that the pattern does not depend on a few extreme values.

Airline scheduling and National Air Space (NAS) queueing effects may contribute to this pattern. When many flights are scheduled to depart in a very short time interval, limitations on the airport departure rate results in long queues. There are several spikes above 1,500 in Figure 6(c), suggesting that more than 1,500 flights were scheduled to depart during those 2-minute intervals aggregated over Year 2000. However, less than 800 flights were actually able to depart during those intervals, explaining the peak reduction effect shown in the actual departure numbers in Figure 6(d).

The queueing effects and "flight banks" in scheduling are well known among airline studies. However, it is surprising to see this well shaped pattern exists when we aggregate the days throughout the year. Usually one might expect the queueing delays on different days to cancel each other, since different days may have different propagation patterns.

This interesting pattern also sheds light on how to choose the roughness penalty when estimating the daily propagation pattern. A smaller penalty value may be more desirable because it captures the similar but more muted peaking phenomenon that exists in the delay v.s. scheduled departure time plot (Figure 5).

A final note in this section: the Daily Propagation Pattern is not really daily. This pattern actually occurs across two consecutive days. The break point is in the early morning around 5:00 am or 6:00 am, when the airport finally consumes all delays and no flight departs anymore. However, if we carefully choose the definition of *day*, for example, from 6:00 am till 5:59 am of the next day, then it is fairly

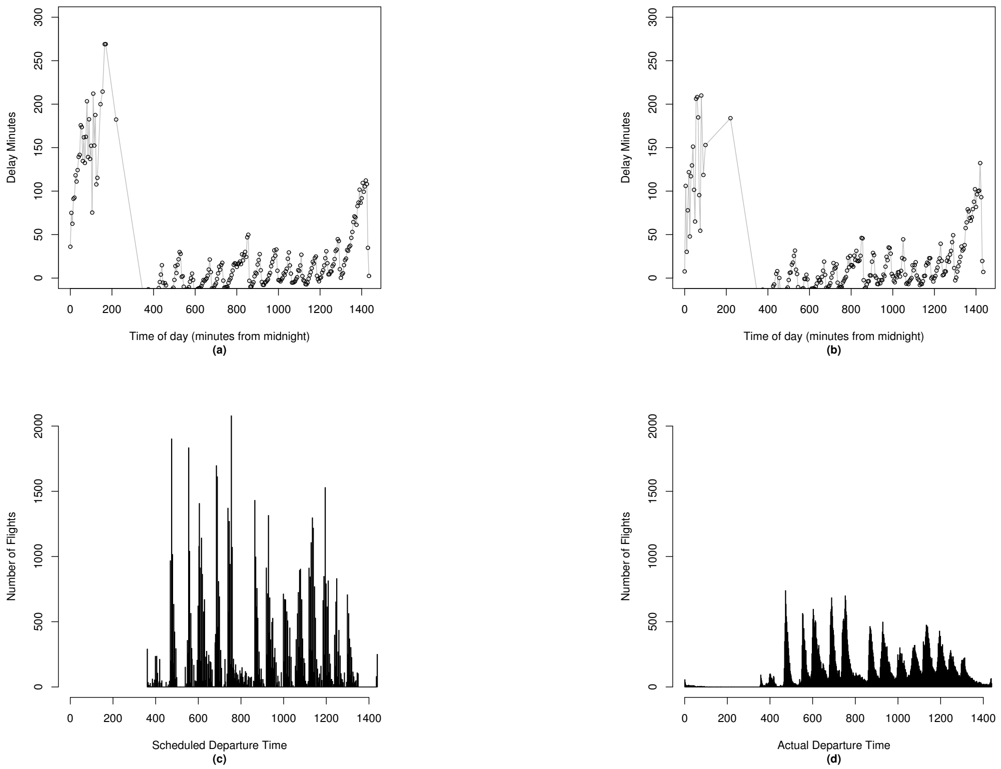

Fig. 6. Pattern in Delays vs Actual Departure Times: **(a)** Plotting $\bar{x}_t$ using actual departure time **(b)** Plotting graph (a) after random sampling 30% of the data **(c)** Number of flights scheduled to depart during every 2-minute interval aggregated throughout Year 2000 **(d)** Number of flights that actually departed during every 2-minute interval aggregated throughout Year 2000

reasonable to name this pattern the Daily Propagation Pattern.

5.6. Density Estimation for the Random Residuals Using a Genetic Algorithm

After both the seasonal trend and the daily propagation pattern are removed, we use the Expectation Maximization (EM) algorithm to find the distribution of the residuals. Since the EM algorithm usually ends up with a local optimal value, we incorporate a Genetic Algorithm (GA) in search of a global optimal value.

We specified 100 parents and 100 generations in our experiment. Random starting values were generated to form the pool of parents/chromosomes. The chance of mutation were controlled at a very low level by setting p_m= 1/(number of parameters+1) (Willighagen, 2005). This iterative procedure shows satisfying results in Figure 7: the best value and the overall fitness of all parents improve as the generation evolves; the convergence rate is reasonably fast since both curves decline quickly. The roughness of the upper curve comes from the mutation which allows

Table 2. Values of the Parameters in Mixture Density Fitting

	p_1, p_2, p_3, p_4	$\mu_1, \mu_2, \mu_3, \mu_4$	$\sigma_1^2, \sigma_2^2, \sigma_3^2, \sigma_4^2$
Solution	0.37,0.40,0.15,0.08	-17.15,-7.31,19.57,69.13	88.20,89.33,1007.73,3926.00

Table 3. Quantile-Quantile Table in Mixture Density Fitting

Percentile	10%	20%	30%	40%	50%	60%	70%	80%	90%
Original residuals	-24.86	-19.52	-15.72	-12.32	-8.98	-5.35	-0.75	7.99	35.17
Fitted residuals	-25.10	-19.84	-15.90	-12.44	-9.05	-5.33	-0.68	7.26	37.58

the algorithm to avoid local optimal values by preventing the chromosomes from becoming too similar to each other.

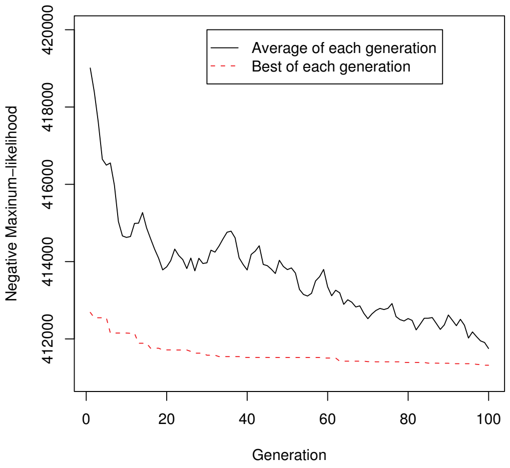

Fig. 7. Finding the Global Maximum via Genetic Algorithm

Table 2 shows the optimal parameter values. A fitted residual distribution is plotted in Figure 8 using the set of optimal parameters. The original residual distributions is shown on the left. One may notice some negative values on the left side of the distributions. This happens because we removed the seasonal trend and the daily propagation pattern. In other words, the negative values mean that these flights have less delays compared with the seasonal average and the daily average values.

We obtain four normal distributions in the mixture to approximate the original residual distribution, which means J=4. As one can see in Figure 8(b), two normal distributions are mixed together to form the high spike: one is slightly to the left to cover more negative values, while the other has a higher peak to help shape the skewness. The third normal distribution captures some medium delays and the fourth one accounts for the very large delays.

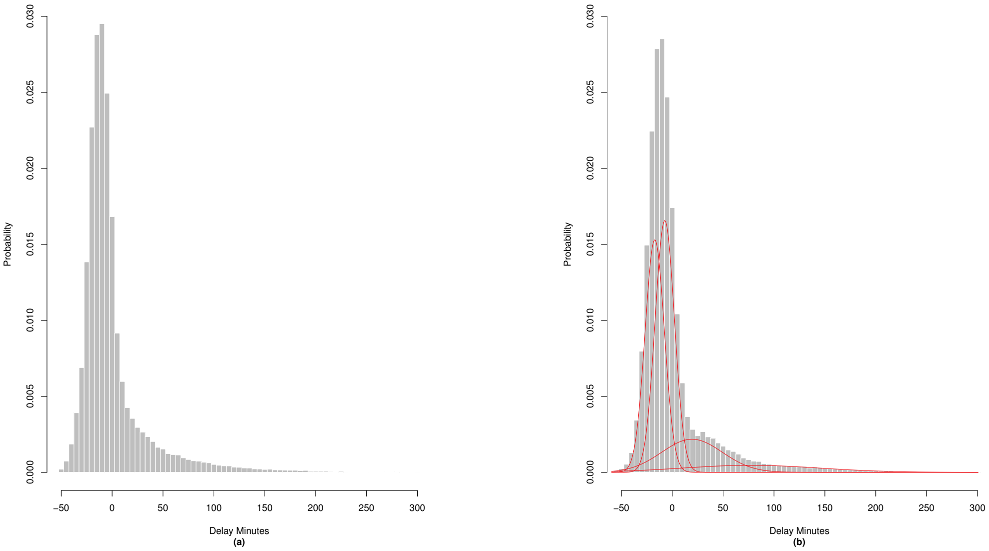

Fig. 8. Fitting the Residuals: **(a)** Density distribution of the original residuals **(b)** The fitted distribution with its four components

The original distributions and the fitted distributions share a similar shape—a skewed distribution with a high spike in the middle, a short left tail and a long right tail. Our fitting is reasonably close by looking at the distribution shapes and comparing the Quantile-Quantile values (Table 3). 8 out of the 9 pairs of quantiles closely follow each other and the difference is less than 1 minute.

5.7. Model Validation and Robustness

We employ a cross validation method in our study, which means parameters are calculated from the training set (70% of total data), and the validation is carried out in the holdout set (30% of total data). In many applications, practitioners are interested in knowing whether the delay will fall within a certain range or the probability of a large delay. Therefore we calculate the 80% confidence interval (CI), 90% CI and also the probability that the delay is larger than 120 minutes to test our model. We compare the difference between the results derived from our model and results calculated from the validation data. A smaller difference indicates a better model. Table 4 lists the model performance with different choices of smoothing penalties.

In Table 4, one may find two bold numbers in the fourth row, which are the equilibrium smoothing penalties in the manner described in previous sections ($\lambda_t = 1.03$ and $\lambda_s = 0.44$). When we derive a 80% CI from our model and apply in the validation set, we find 81.35% of the validation data fall into this interval. Then we

Table 4. Model Robustness With Different Smoothing Penalties: Parameter Sensitivity Test

λ_s	λ_t	80%CI	90%CI	$p(delay \geq 120min) = 2.95\%$
1.03	0.41	81.30%	90.33%	2.88%
1.03	0.47	81.37%	90.33%	3.01%
1.03	**0.44**	81.35%	90.34%	2.91%
1.00	**0.44**	81.41%	90.27%	2.91%
1.06	**0.44**	81.28%	90.32%	3.04%

derive a 90% CI and find that 90.34% of the validation data fall into this interval. We also calculate the probability that a flight will be delayed for more than 120 minutes. Our model gives a probability of 2.91% while actually the probability is 2.95%. The difference is in either the third or fourth decimal place, thus it is reasonably small.

We also explore the robustness of λ in Table 4. We slightly change the value of λ_s and λ_t and estimate the mixture distribution. We find the performance of our model is not significantly impacted, suggesting that the model is very robust.

6. Conclusions and Future Research

We believe our approach to estimating a fight departure delay distribution has several distinctive features. 1) The model includes three components: seasonal trend, daily propagation pattern and random residuals. This structure is intuitive, clear and easy to follow. 2) The model shows reasonable fit to the original data in both the mixture density distribution fitting and the cross validation test. 3) The model demonstrates strong robustness in the parameter sensitivity test: slight variations of the parameter values do not have a significant impact on the performance of the model. 4) Practical model outputs such as the confidence interval and probability of large delays address the needs of the practitioners. 5) The Genetic Algorithm (GA) improves the performance of traditional EM algorithm, thus parameters of a good fit can be found in a more efficient manner.

Our case study focused on the performance of United Airlines in Denver International Airport. Of course, to apply our approach to the application described in Section 2, we would need such a model for all major U.S. airlines and airports. It is not unreasonable that one could carry out a similar analysis for all such airline/airport pairs. On the other hand, an interesting research direction would be to generate a NAS-wide model that could be applied across all major airline/airport combinations.

A second major extension of our work would be to create a dynamic version of

Table 5. Fields and Descriptions

Fields	Data Item	Type	Comments
FAA_CARR	Carrier Code	Character	
FLTNO	Flight Number	Numeric	Range 0001-9999
LEAVE	Departure Airport	Character	
ARRIVE	Arrival Airport	Character	
YYMMDD	Year Month Date	Numeric	
DAYOFWK	Day of Week Indicator	Numeric	
OAG_DEP	OAG Departure Time	Character	Format: HHMM
LVETIME	Actual Departure Time	Character	
OAG_ARR	OAG Arrival Time	Character	
ARRTIME	Actual Arrival Time	Character	
OAG_G2G	OAG Elapsed Time - G2G	Numeric	
ASQP_G2G	Actual Elapsed Time - G2G	Numeric	
FLT_DEL	Actual - OAG Elapsed Time Diff-G2G	Numeric	Non-Negative
TI_DEL	Taxi-In Delay Minutes	Numeric	In Minutes
TO_DEL	Taxi-Out Delay Minutes	Numeric	In Minutes
TAXI_OUT	Taxi-Out Minutes	Numeric	In Minutes
TAXI_IN	Taxi-In Minutes	Numeric	
AIRBORNE	Airborne Minutes	Numeric	

our model. Specifically, a model, which responds to changes in real time parameter measurements, could be developed by incorporating a variety of dynamic factors. For example, such a model might adjust departure delay distributions in response to factors such as weather conditions, estimates of airport surface congestion as well as others.

While a dynamic model clearly could provide improved results, we feel our current model, if incorporated into ETMS and Monitor Alert, would provide improved predictions and traffic flow management. With this in mind, we are pursuing tests of its practical impact.

7. Acknowledgment

The work of the first two authors was supported by NEXTOR, the National Center of Excellence for Aviation Operations Research, under Federal Aviation Administration Research Grant Number 96-C-001 and contract number DFTA03-97-D00004. Any opinions expressed herein do not necessarily reflect those of the FAA or the U.S. Department of Transportation.

Appendix A

Listed are the data fields we obtained from DOT and the Bureau of Transportation Statistics (BTS). See Table 5.

References

ATA (1999). *Approaching Gridlock Air Traffic Control Delays.* Washington DC: Air Transport Association, Departments of Air Traffic Management and Economics.

Beatty, R. (1998). Preliminary evaluation of flight delay propagation through an airline schedule. *2nd USA/EUROPE Air Traffic Management R&D Seminar*, Orlando, 1st–4th.

Bilmes, J. A. (1998). A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *Technical Report TR-97-021*, Computer Science Division, Department of Electrical Engineering and Computer Science, U. C. Berkeley.

Boyles, R. A. (1983). On the convergence of the em algorithm. *Journal of the Royal Statistical Society B 45*, 47–50.

Chandran, B. G. (2002). *Predicting Airspace Congestion using Approximate Queueing Models.* University of Maryland, College Park: Master Thesis.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B 39*, 1–22.

FAA (2002). *Performance Monitoring Analysis Capability V3.1 Addendum Document.* Washington DC: U.S. Department of Transportation.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning.* Reading, MA: Addison-Wesley.

Green, P. J. and B. W. Silverman (1994). *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach.* London: Chapman and Hall.

Hastie, T. J. and R. J. Tibshirani (1990). *Generalized Additive Models.* London: Chapman and Hall.

Holland, J. (1975). *Adaptation in Natural and Artificial Systems.* Ann Arbor, MI: University of Michigan Press.

Idris, H., J. P. Clarke, R. Bhuva, and L. Kang (2002). Queueing model for taxi-out time estimation. *Air Traffic Control Quarterly 10(1)*, 1–22.

McLachlan, G. and D. Peel (2000). *Finite Mixture Models.* New York: John Wiley & Sons.

NCAR (2002). Airlines get new tools to avoid in-flight icing. *NCAR (National Center for Atmospheric Research) News Release*, March 26, Boulder, Colorado, available at http://www.ucar.edu/ communications/newsreleases/2002/icing.html.

Odoni, A. R., J. M. Rousseau, and N. H. M. Wilson (1994). Models in urban and air transportation. *Handbooks in Operations Research and Management Science 5*, 107–128.

Politovich, M. K., B. C. Bernstein, J. Hopewell, T. Lindholm, L. Gaeurke, C. Knable, D. Hazen, and B. Martner (2002). An unusual icing case: 20 march 2000, denver, colorado. *9th AMS Conference on Aviation, Range and Aerospace Meteorology*, Portland, May.

Rosenberger, J. M., A. J. Schaefer, D. Goldsman, E. L. Johnson, A. Kleywegt, and G. L. Nemhauser (2000). Simair: A stochastic model of airline operations. *Proceedings of the 2000 Winter Simulation Conference*, Orlando, December.

Schaefer, L. and D. Millner (2001). Flight delay propagation analysis with the detailed policy assessment tool. *IEEE Systems, Man, and Cybernetics Conference*, Arizona, October.

Shumsky, R. A. (1997). Real-time forecast of aircraft departure queues. *Air Traffic Control Quarterly 5(4)*, 281–308.

VNTSC (2003). *Enhanced Traffic Management System, Functional Description, Version 7.6.* Cambridge, MA: Volpe National Transportation System Center, U.S. Department of Transportation.

Wang, P., L. Schaefer, and L. Wojcik (2003). Flight connections and their impact on delay propagation. *22nd Digital Avionics Systems Conference*, Indiana, October.

Willighagen, E. L. (2005). *genalg - R based genetic algorithm.* version 0.0.6.