A Statistical Computer Experiments Approach to Airline Fleet Assignment

Venkata L. Pilla[1], Jay M. Rosenberger[1], Victoria C. P. Chen[1], Barry C. Smith[2]

[1]Department of Industrial and Manufacturing Systems Engineering

The University of Texas at Arlington

Campus Box 19017

Arlington, TX 76019 USA

[2]Sabre Research Group

3150 Sabre Drive, MD 8203

Southlake, TX 76092

December 27, 2005

Abstract

The fleet assignment model allocates a fleet of aircraft to scheduled flight legs in an airline timetable. The robust fleet assignment model addressed in this paper utilizes a two-stage stochastic programming framework along with the Boeing concept of demand driven dispatch to assign crew-compatible aircraft in the first stage, so as to enhance the demand capturing potential of swapping in the second stage. A design and analysis of computer experiments approach is utilized to reduce the computation involved in solving the problem. A Latin hypercube-based design is developed within the feasible region, and regression splines are employed to obtain the approximation for the second-stage recourse function. The results on the accuracy of the fit for a real airline carrier are presented and future work is discussed.

1 Introduction

As airline woes continue to persist, mainly due to rising fuel costs, major domestic airlines are battling to reduce costs, increase load factors, and improve revenues. The International Air Transport Association reports that North American carriers are expected to lose $8 billion dollars in the year 2005 (IATA 2005b). One way to reduce costs and increase revenue is to strike a balance between supply (seats) and demand (passengers). Utilizing a smaller capacity aircraft would result in spill due to seat shortage, and usage of a larger capacity aircraft would result in flying empty seats. To avoid this, airline Fleet Assignment Models (FAMs) are used to assign aircraft to the scheduled flights in order to maximize profit (revenue − cost). FAMs have been credited with an annual savings of $100 million at Delta (Subramanian et al. 1994), $15 million annually at USAirways (Rushmeier and Kontogiorgis 1997), and a 1.4% improvement in operating margins at American Airlines (Abara 1989).

The quality of a FAM solution depends upon the accuracy of cost and revenue estimates. While cost estimates are relatively stable and well known, revenue estimates depend on demand forecasts. In addition to seasonal variations, airline passenger demand has been affected by various events, such as the terrorist attacks on September 11, 2001, the outbreak of SARS, and recently rising fuel costs forcing the airlines to increase fares and indirectly affect demand (IATA 2005a). This uncertainty in demand makes it challenging to assign the right type of aircraft to each flight leg in the schedule, which is published 90 days prior to the departure of the flight. Consequently, modeling demand stochastically and delaying the fleet assignment decision closer to departure would likely improve profit.

In this regard, stochastic programming and the Boeing concept of Demand Driven Dispatch are very useful. A brief introduction to the two concepts are given in Sections 1.1 and 1.2.

1.1 Demand Driven Dispatch

Since most demand for flights is realized after the schedule is published, a *robust* FAM approach would include reallocation of aircraft much closer to departure. Berge and Hopperstad (1993) introduced the concept of Demand Driven Dispatch (D^3), in order to match demand to aircraft close to departure of the flight. In this approach, they use two stages of decision-making. The first stage occurs 90 days prior to the departure of the flight, when the flight schedules are published. During this stage, crew-compatible families of aircraft are assigned to flights in the airline timetable. Two aircraft are said to be *crew-compatible*, if they have the same cockpit model; hence, the same crew could operate either aircraft type. The second stage occurs two weeks prior to the departure of flights, when most of the demand is realized and individual aircraft are assigned within the crew-compatible families based on the demand. Swapping can take place with the assignment of specific aircraft in the second stage. For example, Boeing 757 and 767 models are crew-compatible, but a 767 can fly more passengers than a 757. Suppose flights A and B are initially allocated a 757 and 767, respectively. If two weeks before departure, flight A has realized a higher demand than expected, while demand is lower than expected for flight B, then the airline can swap the 757 and 767 without affecting the crew schedule. Higher profit is achieved via this swapping since more revenue is captured, and the the cost of swapping is usually insignificant for the airline.

Using this concept, Berge and Hooperstad have reported 1% - 5% improvement in profits. The difficulty in implementing a D^3-FAM approach is assigning aircraft types to flights such that swapping is feasible within the practical constraints of the airline network. Most notably, two planes can only be swapped if they depart from the same airport and have similar departure times.

1.2 Stochastic Programming

The uncertainty in demand, which is realized close to departure of the aircraft, can be modeled using two-stage Stochastic Programming (SP). SP models use probability distributions for random events, and their goal is to maximize the expectation of some function of the decisions based on these random events. In two-stage SP, a decision is taken in the first stage, after which a random event with a known probability distribution is realized, and a second-stage *recourse* decision is taken based on the random event and the decision from the first stage. The optimal decision consists of a single first-stage decision and a set of *recourse* decisions to be taken based on the random outcome.

As described in Birge and Louveaux (1997), the basic two-stage stochastic linear program with fixed recourse is given by:

$$\min \; z = c^T x + E_\xi \left[\min q(\omega)^T y(\omega)\right] \tag{1}$$

$$\text{s.t.} \quad Ax = b, \tag{2}$$

$$T(\omega)x + Wy(\omega) = h(\omega), \tag{3}$$

$$x \; \geq \; 0, \;\; y(\omega) \geq 0, \tag{4}$$

where $x \in R^{n_1}$ is the first-stage decision vector with linear costs $c \in R^{n_1}, y(\omega) \in R^{n_2}$ is the second-stage decision vector with linear costs $q(\omega) \in R^{n_2}$, A is an m_1 x n_1 first-stage linear constraint matrix with right hand side $b \in R^{m_1}$, and $T(\omega)$ and W are, respectively, m_2 x n_1 and m_2 x n_2 matrices specifying the second-stage linear constraints on x and y with right hand side $h(\omega) \in R^{m_2}$. For a given realization of the stochastic variables, $\omega \in \Omega$, the second-stage problem data, $q(\omega), h(\omega)$ and $T(\omega)$, become known, and the second-stage decision $y(\omega, x)$ can be obtained. Let Ξ denote a set of scenarios, and let the vector $\xi^T(\omega)$ represent a scenario with different components of the second stage, i.e., $q(\omega)^T, h(\omega)^T$ and $T(\omega)$, such that $\xi \in \Xi$. The objective function represented by (1) contains a deterministic term $c^T x$ and the expectation of the second-stage objective $q(\omega)^T y(\omega)$ taken over all realizations of the random event ω. For a given ω, we can write the second-stage value function as

$$Q(x, \xi(\omega)) = \min_y \{q(\omega)^T y | Wy = h(\omega) - T(\omega)x, y \geq 0\}. \tag{5}$$

Suppose, the expected second-stage recourse function is defined as:

$$\Im(x) = E_\xi Q(x, \xi(\omega)), \tag{6}$$

then the deterministic equivalent of the two-stage stochastic linear program can be written as:

$$\begin{aligned} \min \quad & z = c^T x + \Im(x) \\ \text{s.t. } & Ax = b, \\ & x \geq 0. \end{aligned} \tag{7}$$

The primary difficulty is determining the recourse function $\Im(x)$. As stated above, each evaluation of $\Im(x)$ at just one value of x requires calculation of an expected value of an optimized objective. For large, complex problems, like fleet assignment, the iterative approximation methods described by Birge and Louveaux (1997) can be very slow to converge because they typically require a large number of scenarios to adequately represent the stochasticity of the second-stage optimization in (5). Thus, in each optimization iteration of the first stage, there is a high computational cost for evaluating $\Im(x)$ in the second stage, making the problem defined in (7) even harder to solve.

In order to control the computational requirement, Chen (2001) proposed discretization of the x-space to a finite set of points and solving for $\Im(x)$ only at those points, followed by a function approximation technique to estimate the entire surface of $\Im(x)$. This approximation, $\hat{\Im}(x)$, will be computationally trivial to evaluate in (7). In selecting a discretization, it is important to select only those x values which result in a feasible solution in the second stage.

1.3 Contribution

The FAM problem is modeled in this paper using a two-stage SP framework. The two stages correspond to the stages of decision-making in the D^3 concept. Similar to Berge and Hopperstad (1993), the FAM model presented here assigns crew-compatible aircraft to flights in the first stage. Given an initial Crew-Compatible Allocation (CCA), the second stage obtains many equally likely scenarios using a probability distribution, and for each scenario, a deterministic FAM, using the current CCA, solves for the Linear Programming (LP) relaxation of assigning aircraft within the crew-compatible families. This LP is separable by crew-compatible family. The estimate of the expected revenue is determined as the average over the scenarios. As mentioned in Section 1.2, the basic SP solution approach can be computationally intensive if an excessive number of scenarios is required to adequately represent the stochasticity of the second-stage optimization.

A computationally tractable solution method is presented that employs statistical methods from Design and Analysis of Computer Experiments (DACE, see Chen, Tsui, Barton and Allen 2003; Sacks et al. 1989). An experimental design is used to generate the initial CCA and then for each CCA and for each scenario (demand realization) in the second stage, a *computer experiment* (in our case, an optimization problem) is solved. The average over the scenarios is used to estimate the expected revenue value of the recourse function for the two-stage SP model. A statistical model is fit to these data to provide a continuous approximation of expected revenue over the initial CCA space. In our case, the experimental design was based on a Latin hypercube, and the statistical model was based on Multivariate Adaptive Regression Splines (MARS, Friedman 1991). Optimization of the recourse function is not considered in this paper.

A literature review regarding the FAM problem is presented in Section 2. Section 3 discusses more on DACE and how it is applied in this paper, and in Section 4, the two-stage SP model for fleet assignment is presented. In Sections 5 and 6, we demonstrate the methodology of generating the first-stage experimental design and second-stage recourse values, respectively. In Section 7, we present results of the application of this methodology to a real airline carrier, and finally in Section 8, conclusions and future research are discussed.

2 Literature Review

The fleet assignment problem has been a well researched topic for the past fifty years and has been credited for increasing profits at several airlines. Ferguson and Dantzig formulated a combined FAM and aircraft routing problem with deterministic demand (1955) and stochastic demand (1956). Abara (1989) solved the FAM problem using an integer LP model. He defined "turn" as the successive assignment of an aircraft to two consecutive flights and this decision variable causes practical limitations to the FAM problem.

Hane et al. (1995) presented a detailed description on solving a fleet assignment problem as a multi-commodity network flow problem, which formed the basis for a large portion of later FAM research. They modeled the FAM problem as a Mixed Integer Programming (MIP) problem and reported run times twice as fast as the standard LP based branch-and-bound code. Some of the concepts addressed in the Hane et al. (1995), like node aggregation and islands at stations with low activity, are employed in this paper to preprocess the FAM model before solving.

Subramanian et al. (1994) implemented the Coldstart model to solve the fleet assignment problem at Delta Airlines. Theoretical properties of the FAM problem were presented in Gu et al. (1994). Since the solution of FAM affects subsequent planning decisions like maintenance requirements, aircraft routing and crew scheduling, extensions of FAM were considered in Clarke et al. (1996), Rushmeier and Kontogiorgis (1997), Barnhart et al. (1998), Rexing et al. (2000), Ahuja and Orlin (2002), and Klabjan et al. (2002). Disruptions during the operational phase of the problem were addressed in Rosenberger et al. (2003).

An important component of a successful FAM is modeling the objective function. The objective function can be maximizing revenue, minimizing passenger spill (lost revenue due to assigning smaller aircraft) or minimizing the number of aircraft being used. In this paper the objective is modeled as maximizing revenue and, similar to the research mentioned earlier, we assume demand for each flight leg is independent. In practice, this is not true, since in a multi-leg itinerary, the demand on one leg affects the others. This is also referred to as Origin Destination (O-D) FAM. The network affect will be considered in future work. Jacobs et al. (1999) presented an O-D FAM formulation and solved it using Benders' decomposition. Barnhart et al. (2002) developed a Passenger Mix Model (PMM) that, given a schedule with known flight capacities and a set of passenger demands with known fares, determines optimal demand and revenue. They proposed a combined PMM and FAM problem. Farkas (1996) demonstrated that revenue management has a significant impact on passenger volume and mix, and by ignoring these effects FAM produces sub-optimal solutions.

The stochastic nature of passenger demand was recently addressed in Listes and Dekker (2002) for determining an optimal airline fleet composition. Given an airline schedule and a set of aircraft types, the fleet composition problem determines the number of aircraft of each type that the fleet requires to maximize profit. They developed a two-stage SP model to determine a single fleet composition that maximizes profit in the first stage across all demand scenarios generated in the second stage. In the second stage, they solved a deterministic FAM model for each demand scenario allowing swapping to occur (similar to Berge and Hopperstad (1993)) and employed a scenario aggregation approach (Rockafellar and Wets 1991) to reduce the computational complexity.

3 Design and Analysis of Computer Experiments Approach

A Design and Analysis of Computer Experiments (DACE) approach is useful when a computer experiment is the only means for representing a complex system (see recent reviews Chen, Tsui, Barton and Allen 2003; Chen et al. 2005; Kleijnen 2005). DACE-related methods have been successfully employed for solving stochastic dynamic programming and Markov decision problems (Werbos 1998; Chen et al. 1999; Chen 1999; Chen, Günther and Johnson 2003; Tsai et al. 2003; Si et al. 2004; Cervellera et al. 2005), and their use was first suggested for SP by Chen (2001). This paper represents the first implementation of these ideas for SP. Typically, the computer experiment is a simulation model; however, in this paper it is an optimization model.

Specifically, our *DACE Phase* conducts the following:

- An optimization model (computer experiment) of system performance is constructed based on knowledge of how the system operates.
- Design of Experiments (DoE) is used to select the set of sample points as input to the optimization model, which then provides the corresponding responses.
- A statistical model is to fit these data.

For our robust FAM, the DACE Phase is illustrated in Figure 1. In particular, DoE points for the CCA space must be feasible, which is an issue not typically handled by DACE methods; thus, a new approach is devised and described in Section 5. In order to estimate the second-stage recourse function, scenarios are generated based on known probability distributions, and for each CCA, average revenues from the second-stage FAM optimization are collected as responses y (see Section 6 for details). A MARS statistical model, $\hat{y}$(CCA), is fit to these data to generate an approximate second-stage recourse function, which can then be employed for more efficient future evaluations in the first-stage optimization. The run time for MARS is dependent upon the number of basis functions that the user specifies with the parameter M_{max}. A variant of MARS that automatically selects M_{max}, developed by Tsai and Chen (2005), is used to decrease the time required to fit MARS.

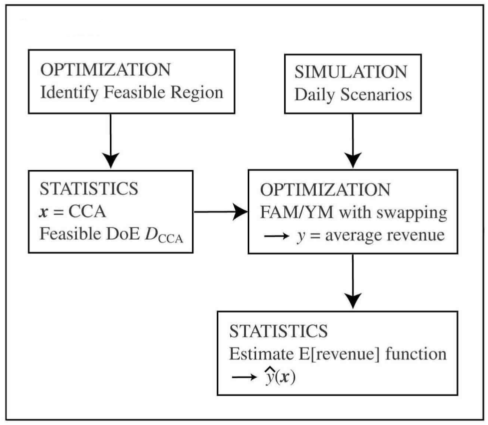

Figure 1: DACE Phase of the Two-Stage SP Framework

4 Two-Stage SP FAM Formulation

Given an airline schedule and a set of fleets of different aircraft that can fly each flight leg, the fleet assignment problem allocates the fleet of aircraft to the scheduled flights subject to the following operational constraints:

- Balance: Aircraft cannot appear or disappear from the network.
- Cover: Each flight in the schedule must be assigned to exactly one aircraft type.
- Plane Count: The total number of aircraft assigned cannot exceed the number of available aircraft in the fleet.

The objective is to find a feasible assignment that maximizes profit. In this paper, the fleet assignment problem is formulated as an integer multicommodity flow problem on a time line network similar to Berge and Hopperstad (1993) and Hane et al. (1995). For a given fleet type, a time line is a graph that represents the arrival and departure events occurring at each station (airport serviced by the carrier) over a specified time period as shown in Figure 2.

Flights below the time line indicate arrivals, flights above the time line indicate departures, and the numbers indicate the corresponding flight legs. Any flight, which arrives at a particular station, will not be available for departure immediately because of the time required for fueling, loading passengers/baggage etc. As such, a *turn time* is added to all the arrivals before they are ready to take off. The turn time is dependent on the particular fleet type and the station. A *node* in a time line begins with an arrival and ends before the next arrival, with at least one departure in between. In Figure 2, BC, DE, and FA represent nodes. The arcs that connect within these nodes are called *ground arcs,* and the arc that connects the last arrival on the time line to the first departure is called the *overnight arc*. These arcs denote at least one plane being on the ground at a station and are defined as continuous variables. Because all flight variables are integral the values corresponding to these arcs will be integral as well. The sum of all the times corresponding to the arcs represent the *total ground time* of the planes at that particular station. The overnight arc includes a *plane count*

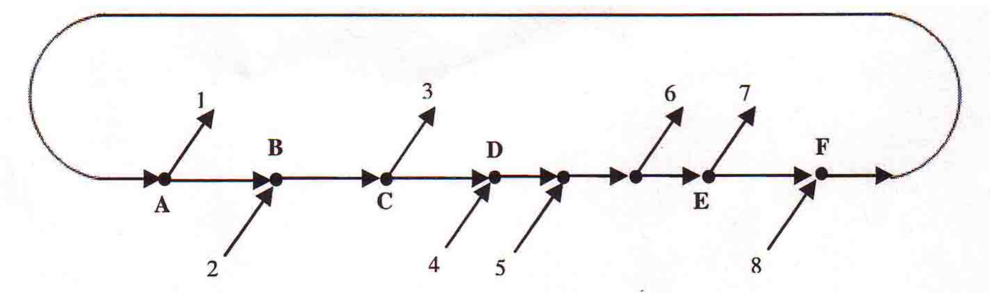

Figure 2: Time Line

hour (typically 4 A.M. EST) that is used for developing the plane count constraints. A detailed description of how to generate the constraints based on the time line is given in Section 5.1.

4.1 Model Assumptions

The following assumptions are taken to model the two-stage SP FAM problem:

- The revenue coefficients are calculated assuming that the demand for each flight is independent.
- All passengers for a flight have equal revenue. This is inconsistent with yield management practices because as the capacity is added, incremental passengers have lower average revenue.
- Spilled passengers are assumed to be lost by the carrier and are not recaptured.
- The decision variables are relaxed to be continuous.
- Fleets within a crew-compatible family have same turn time. In practice, we can choose the maximum turn time within a crew-compatible family.

These assumptions tend to bias FAM solutions to the over use of large aircraft. Considering an OD-FAM is more reasonable and a subject of future research.

4.2 Stochastic Model

Let L be the set of flight legs (indexed by l). Let F denote the set of fleet types (indexed by f), and G be the set of crew-compatible families (indexed by g), which can be used for each of the legs $l \in L$. Since we assign, crew-compatible families in the first stage, for each leg $l \in L$ and for each crew-compatible family type $g \in G$, let a binary variable x_{gl} be defined such that

$$x_{gl} = \begin{cases} 1 & \text{if crew-compatible family } g \text{ is assigned to flight leg } l, \\ 0 & \text{otherwise.} \end{cases}$$

In the second stage, we assign specific aircraft within the crew-compatible family. As such, for each leg $l \in L$, for each aircraft $f \in F$, and for scenario $\xi \in \Xi$, let a binary variable x^{ξ}_{fl} be defined such that

$$x_{fl}^{\xi} = \begin{cases} 1 & \text{if aircraft } f \text{ is assigned to the leg } l \text{ for scenario } \xi, \\ 0 & \text{otherwise.} \end{cases}$$

Consider the following FAM parameters:

- S = set of stations, indexed by s,
- V = set of nodes in the entire network, indexed by v,
- $f(v)$ = fleet type associated with node v,
- A_v = set of flights arriving at node v,
- D_v = set of flights departing at node v,
- M_f = number of aircraft of type f,
- $R(x_{fl}^{\xi})$ = revenue if flight leg l is assigned to aircraft type f for scenario ξ,
- $C(x_{fl}^{\xi})$ = cost if flight leg l is assigned to aircraft type f for scenario ξ,
- a_{v+}^{ξ} = value of ground arc leaving node v for scenario ξ,
- a_{v-}^{ξ} = value of ground arc entering node v for scenario ξ,
- O_f = set of arcs that include the plane count hour for fleet type f, indexed by o,
- L_0 = set of flight legs in the air at the plane count hour.

The two-stage formulation can be represented as:

$$\max \quad \sum_{l \in L} \sum_{f \in F} E[\, R(x_{fl}^{\xi}) \; - \; C(x_{fl}^{\xi})] \tag{8}$$

$$\text{s.t. } a_{v-}^{\xi} + \sum_{l \in A_v} x_{f(v)l}^{\xi} - \sum_{l \in D_v} x_{f(v)l}^{\xi} - a_{v+}^{\xi} = 0 \qquad \forall v \in V, \xi \in \Xi, \tag{9}$$

$$\sum_{g \in G} x_{gl} = 1 \qquad \forall l \in L, \tag{10}$$

$$\sum_{f \in g} x_{fl}^{\xi} = x_{gl} \qquad \forall l \in L, g \in G, \xi \in \Xi, \tag{11}$$

$$\sum_{o \in O_f} a_o^{\xi} + \sum_{l \in L_0} x_{fl}^{\xi} \leq M_f \qquad \forall f \in F, \xi \in \Xi, \tag{12}$$

$$x_{fl}^{\xi} \in \{0,1\} \qquad \forall l \in L, \xi \in \Xi, \tag{13}$$

$$x_{gl} \in \{0,1\} \qquad \forall l \in L, g \in G, \tag{14}$$

$$a_{v+}^{\xi} \geq 0 \qquad \forall v \in V, \xi \in \Xi. \tag{15}$$

The objective is to maximize profit (revenue $-$ cost) in the second stage by assigning aircraft within the crew-compatible allocation made in the first stage. For each flight leg $l \in L$, for an aircraft $f \in g$ and in scenario $\xi \in \Xi$, revenue is calculated as:

$$R(x_{fl}^{\xi}) = (\min\ \{[\text{Demand}]_{\xi},\ \ [\text{Capacity}]_f\}) * (\text{Average fare per passenger})_l.$$

The *block time* of a flight leg l is defined as the length of time from the moment the plane leaves the origin station until it arrives at the destination station. Let b_l be the scheduled block time for flight leg l. The cost for each flight leg is calculated as a function of block time and operating cost of a particular fleet type per block hour, and is given by:

$$C(x_{fl}^{\xi}) = b_l * (\text{Operating cost per block hour})_f.$$

Constraints in set (9) represent the balance constraints needed to maintain the circulation of aircraft throughout the network. Cover constraints (10) represent the first-stage crew-compatible assignment, and set (11) guarantees that aircraft within the crew-compatible family (assigned in the first stage) are allocated. For formulating the plane count constraints (12), we need to count the number of aircraft of each fleet being used at any particular point of day (generally when there are fewer planes in the air). As such, the *ground arcs* that cross the time line at the plane count hour and the flights in air during that time are summed to assure that the total number of aircraft of a particular fleet type do not exceed the number available. Binary constraints (13) and (14) are relaxed to model the decision variables as continuous in this paper. As in practice, most crew-compatible families include only one or two aircraft types, so integer solutions result, as mentioned in Berge and Hopperstad (1993). For families with more than two aircraft types, an upper bound on the objective function is obtained.

Since the variables are relaxed, the resulting fit for the recourse function obtained using MARS provides an upper bound for the profit generated. A true recourse function can be estimated with MARS using integral values only at the expense of higher computational time. This is difficult in the case of solving the two-stage SP with traditional Benders' decomposition because there are integrality issues during the generation of cuts.

5 Design of Experiments Approach

The main objective in the first stage is to assign the initial crew-compatible allocation (CCA) using DoE. The design is generated within the experimental region formed by the first-stage FAM constraints. Since at this stage we are concerned only with the CCA decision variables, the first-stage constraints are similar to the constraints described earlier except that they are based on crew-compatible family variables as defined below:

(Balance) $$a_{v^-} + \sum_{l \in A_v} x_{gl} - \sum_{l \in D_v} x_{gl} - a_{v^+} = 0 \qquad \forall v \in V, \tag{16}$$

(Cover) $$\sum_{g \in G} x_{gl} = 1 \qquad \forall g \in G, l \in L, \tag{17}$$

(Plane Count) $$\sum_{o \in O_g} a_o + \sum_{l \in L_0} x_{gl} \leq M_g \qquad \forall g \in G, \tag{18}$$

$$0 \leq x_{gl} \leq 1 \qquad \forall g \in G, l \in L, \tag{19}$$

where O_g represents the set of arcs that include the plane count hour for crew-compatible family g, and M_g denotes the number of aircraft of crew group g.

In order to generate an appropriate experimental design, the above constraints must be preprocessed, removing any redundant variables, so as to specify the feasible *polytope*, defined by a system of linear inequalities $Ax \leq b$, in the lowest dimensional space. Preprocessing is conducted by exploiting the explicit as well as implicit equalities present in the constraints as defined in Section 5.1. Savelsbergh (1994) demonstrated the use of preprocessing techniques to reduce MIP problems.

Let $P := \{x \in R | Ax \leq b\}$ be a nonempty convex *polytope* formed by the first-stage constraints. Discretized points within this *polytope* can be generated as shown in Section 5.2 to represent the initial CCA decisions. Any infeasible design points can be projected into the feasible polytope as described in detail in Section 5.2.2.

5.1 Reducing the Decision Space

Before generating the design, it is important to reduce the first-stage FAM to the minimum number of variables because the precision of the estimates is adversely affected by multicollinearity (or correlations) among the input variables (Myers and Montgomery 1986). Reducing to the minimum number of variables in the first stage removes the redundancy as well as dramatically improves the efficiency of the design. In order to generate the feasible polytope P, we start with the traditional FAM defined by Hane et al. (1995) and pivot out variables using the equality constraints as demonstrated in Sections 5.1.1 and 5.1.2, and then remove the implicit equalities as discussed in Section 5.1.3. In particular, Section 5.1.1 discusses the application of the Boeing concept of D^3, and Section 5.1.2 takes advantage of the first-stage constraints to reduce the number of variables. Once the equalities are removed, we are left with a set of minimum variables that define the first-stage decision space.

5.1.1 Demand Driven Dispatch

The robust FAM approach presented in this paper incorporates the Boeing concept of D^3 as defined in Section 1.1. In the first stage (90 days prior to departure), before most demand is realized, crew-compatible families of aircraft are assigned to flights. As such, in the first stage, only one variable needs to be considered when there is more than one aircraft belonging to the same crew-compatible family that can be allocated to a flight leg $l \in L$. For example, suppose flight leg l can be assigned to six different fleet types; that is $F = \{1, 2, 3, 4, 5, 6\}$. The cover constraint for flight leg l is given by:

$$x_{1l} + x_{2l} + x_{3l} + x_{4l} + x_{5l} + x_{6l} = 1. \tag{20}$$

Suppose aircraft 1 belongs to crew-compatible family 1, aircraft 2 and 3 belong to crew-compatible family 2, and aircraft 4, 5, and 6 belong to crew-compatible family 3; that is $G = \{1, 2, 3\}$. In this case we substitute x_{2l} and x_{3l} with just x_{2l}, and we replace x_{4l}, x_{5l} and x_{6l} with x_{3l}. Then, the cover constraint for flight leg l becomes:

$$x_{1l} + x_{2l} + x_{3l} = 1. \tag{21}$$

Thus, the number of variables has been reduced from six to three.

5.1.2 Constraints

The special structure of each of the constraints in the first stage can be exploited to reduce the number of variables over which the design has to be generated. Section 5.1.2.1 addresses the reduction due to cover constraints, and Section 5.1.2.2 describes the reduction achieved due to balance constraints.

5.1.2.1 Cover Constraints

For every leg l, the cover constraint obtained after reduction through the Boeing concept can be manipulated to reduce the dimensionality by one. For example, for equation (21) we have:

$$x_{1l} + x_{2l} + x_{3l} = 1 \qquad \Rightarrow \qquad x_{3l} = 1 - x_{1l} - x_{2l}. \tag{22}$$

We also have the constraints:

$$x_{gl} \in \{0, 1\} \text{ and } g \in G = \{1, 2, 3\}. \tag{23}$$

Since we are modeling the variables as continuous, we have

$$0 \leq x_{3l} \leq 1 \qquad \Rightarrow \qquad 0 \leq 1 - x_{1l} - x_{2l} \leq 1 \qquad \Rightarrow \qquad 0 \leq x_{1l} + x_{2l} \leq 1. \tag{24}$$

5.1.2.2 Balance Constraints

The balance constraints maintain the circulation of aircraft throughout the entire flight network. Based on the ground time of the planes and the traffic intensity, stations are typically classified as hubs and spokes.

Balance Constraints for Spokes

Stations with low traffic intensity are called $spokes$. A spoke time line consists of sudden activity during a period of time where we can see equal numbers of arrivals and departures. Therefore, we can find time periods in a spoke time line when there are no planes on the ground. The ground arcs corresponding to these periods can be dropped. *Islands* can be defined as a set of nodes in which the incoming and outgoing ground arcs are zero, and a *complex island* has two or more nodes. Figure 3 represents a spoke time line. In Figure 3, A, B, C, and E represent nodes; B and D represent islands while A forms an *overnight island*. Island D is a complex island with nodes C and E in it. The time line for each crew-compatible family assists us in framing the balance constraints at each station. Consider a spoke time line for crew-compatible family g as shown in Figure 3. The balance constraint for node B is:

$$x_{g3} - x_{g4} = 0 \qquad \Rightarrow \qquad x_{g3} = x_{g4}. \tag{25}$$

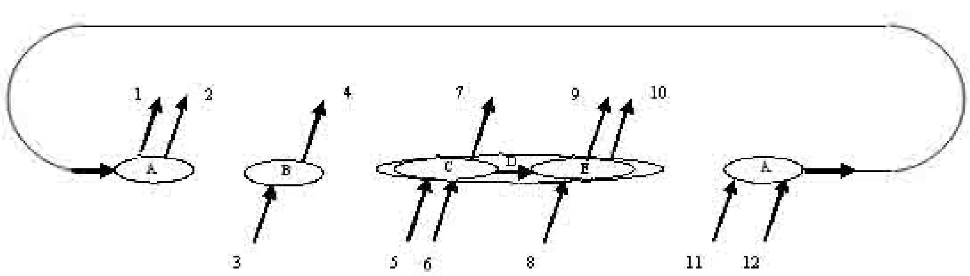

Figure 3: Spoke Time Line

Thus, the crew group type of x_{g3} will determine that of x_{g4}. For node C, the balance constraints are:

$$x_{g5} + x_{g6} - x_{g7} - a_g = 0 \tag{26}$$

$$a_g \geq 0 \tag{27}$$

where a_g represents the ground arc from node C to E. For node E, the balance constraint is:

$$a_g + x_{g8} - x_{g9} - x_{g10} = 0. \tag{28}$$

From (26), (27) and (28), by dropping the ground arc variable a_g the balance constraints for the island D are:

$$x_{g5} + x_{g6} - x_{g7} \geq 0 \tag{29}$$

$$x_{g5} + x_{g6} - x_{g7} + x_{g8} - x_{g9} - x_{g10} = 0 \quad \Rightarrow \quad x_{g10} = x_{g5} + x_{g6} - x_{g7} + x_{g8} - x_{g9}. \tag{30}$$

Similarly, for node A:

$$x_{g11} + x_{g12} - x_{g1} - x_{g2} = 0 \quad \Rightarrow \quad x_{g2} = x_{g11} + x_{g12} - x_{g1}. \tag{31}$$

Thus, the total number of decision variables is reduced from twelve to nine.

Balance Constraints for Hubs

A *hub* consists of continuous activity with periodic flights coming from the spokes. Because of the high traffic intensity at hubs, we cannot eliminate all of the ground arcs as we did at spokes. Consider a hub time line for crew-compatible family g with nodes A, B and C as represented in Figure 4. The balance constraints for nodes A and B are given by:

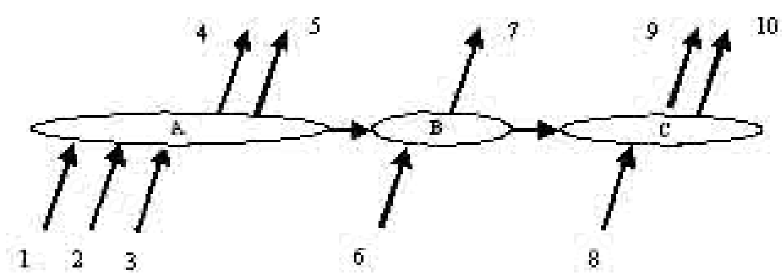

Figure 4: Hub Time Line

$$a_g + x_{g1} + x_{g2} + x_{g3} - x_{g4} - x_{g5} \geq 0 \tag{32}$$

$$a_g + x_{g1} + x_{g2} + x_{g3} - x_{g4} - x_{g5} + x_{g6} - x_{g7} \geq 0 \tag{33}$$

$$a_g \geq 0. \tag{34}$$

Since the time line ends with the last departure at node C, the constraints for node C will determine the overall balance constraint for the hub. Therefore, we have:

$$x_{g1} + x_{g2} + x_{g3} - x_{g4} - x_{g5} + x_{g6} - x_{g7} + x_{g8} - x_{g9} - x_{g10} = 0$$

$$\Rightarrow \quad x_{g10} = x_{g1} + x_{g2} + x_{g3} - x_{g4} - x_{g5} + x_{g6} - x_{g7} + x_{g8} - x_{g9}. \tag{35}$$

For each of the hubs there will be one equality constraint for the entire time line to maintain the balance of flights arriving and departing. Hence, the number of decision variables is reduced by one, and there is one ground arc variable for each crew-compatible family.

5.1.3 Implicit Inequalities

In this section, we generate equations based on implicit opposing inequalities from the set of constraints $Ax \leq b$. This is achieved by formulating a LP problem that finds two implicit opposing inequality constraints using convex combinations of the explicit inequalities. Any feasible solution to the LP refers to an implied equation that can be generated.

An inequality $\alpha x \leq \beta$ from $Ax \leq b$ is called an implicit equality (in $Ax \leq b$), if $\alpha x = \beta$ for all x satisfying $Ax \leq b$. To check if there are any implicit equalities among the system of constraints defined by

$$\textstyle\sum_{j=1}^{n} a_{ij}x_j \leq b_i, \qquad \forall i = 1, 2, ..., m, \tag{36}$$

we formulate a LP. Let $z \geq 0$ be a multiplier for constraint (36), so

$$\textstyle\sum_{j=1}^{n} a_{ij}x_j z_i \leq b_i z_i, \qquad \forall i = 1, 2, ..., m.$$

P is full-dimensional if and only if $P^{'} := \{z | A^T z = 0, b^T z = 0, z \geq 0\} = \{0\}$. The proof for this statement is detailed in Gao and Zhang (2002), Sierksma and Tijssen (2003), and Pilla et al. (2005). The formulated LP is a maximization problem as we are looking for a non-zero solution. Since the constraints in $P^{'}$ form a cone, we need to add a constraint of the form $\sum_{i=1}^{m} z_i \leq 1$, to get a feasible solution. Thus, the LP can be represented as:

$$\max \sum_{i=1}^{m} z_i \tag{37}$$

$$\text{s.t.} \quad A^T z = 0, \tag{38}$$

$$b^T z = 0, \tag{39}$$

$$\sum_{i=1}^{m} z_i \leq 1, \tag{40}$$

$$z \geq 0. \tag{41}$$

If a nonzero solution exists for the LP, then an implicit equality can be generated. Figure 5 shows the flow chart used for reducing the dimensionality of the FAM problem.

5.2 Generating the Experimental Design

An underlying form for the expected revenue function of a FAM model cannot be assumed; thus, we desire an experimental design that adequately represents the CCA space. This is challenging for several reasons that will become apparent in the design generation process. After reducing dimensionality, the design is constructed in three steps:

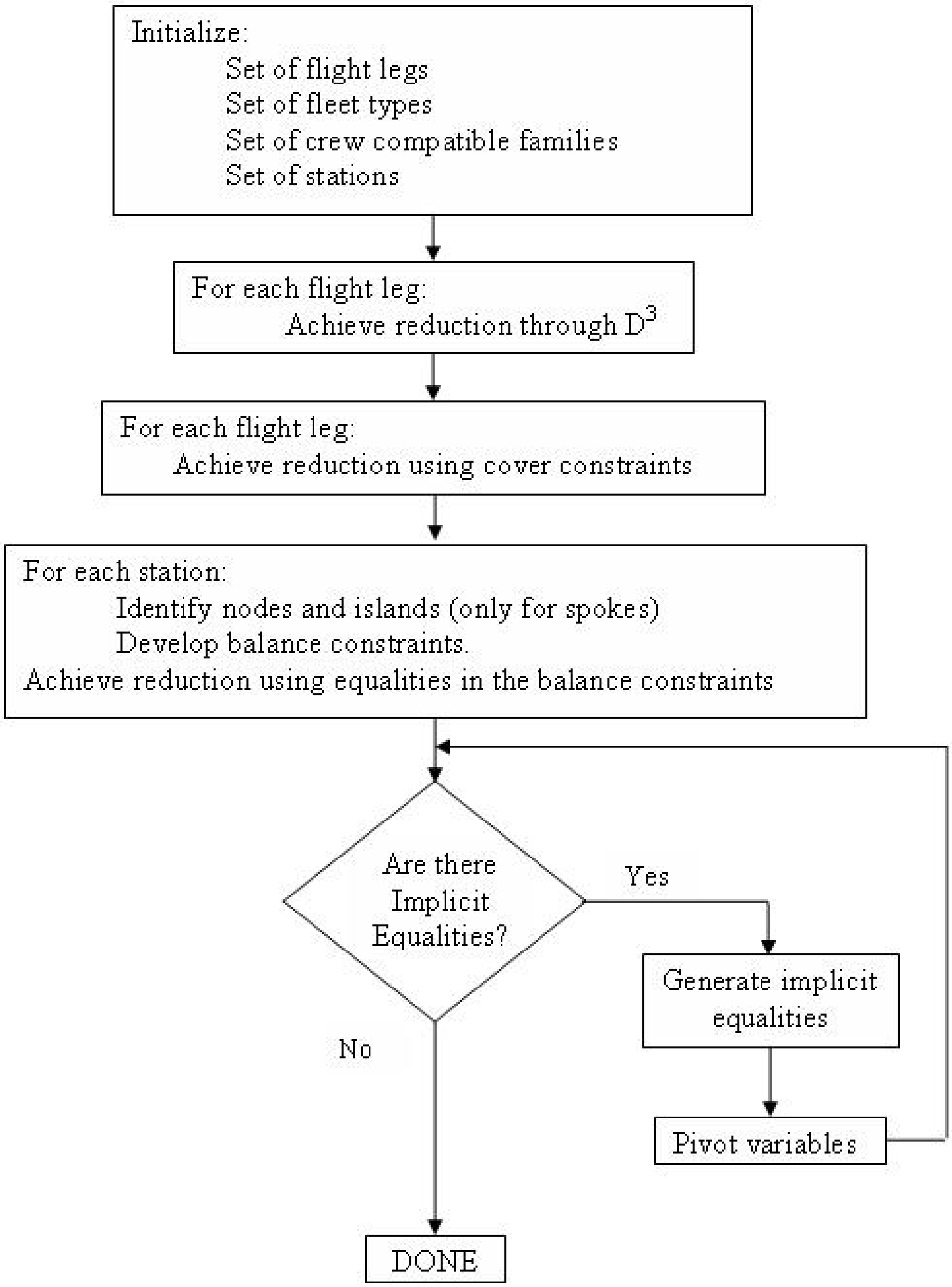

Figure 5: Decision Space Reduction Flow Chart

- A Latin hypercube algorithm is devised to generate DoE points that satisfy the triangular FAM cover constraints. This will provide a set of well-distributed points in the CCA space that are close to the feasible region.
- The current DoE points are projected to extreme points of the (convex) feasible region.
- Another Latin hypercube is used to generate interior feasible points as convex combinations of the extreme points.

Given the high dimensionality of the CCA space, the Latin hypercube is a practical choice of design. Because perfect correlation between two input variables is possible in a Latin hypercube design, our algorithm generates a larger design and columns are eliminated so that the final design has low input variable correlations.

5.2.1 Latin Hypercube Algorithm for the Reduced Cover Constraints

After dimensionality reduction, suppose m is the number of crew-compatible types and n is the number of legs. Let the total number of design points be $N = \lambda p > mn$, where p represents design "levels" and λ is a frequency parameter.

- Draw a "grid" of p points in a m-dimensional hyper-triangle representing a reduced cover constraint over crew-compatible types.
- Order the p points according to distance from origin (randomly break ties) $\rightarrow 1, 2, \ldots, p$. Each is an m-tuple.
- Generate a Latin hypercube with $N = \lambda p$ in n dimensions.
- Randomly assign the Latin hypercube levels for each dimension:

  $1, \ldots, \lambda \rightarrow 1$

  $\lambda + 1, \ldots, 2\lambda \rightarrow 2$

  $\vdots$

  $(i-1)\lambda + 1, \ldots, i\lambda \rightarrow i, \qquad \text{for } i = 1, 2, \ldots, p$

Since each of the reduced cover constraints has a different number of crew-compatible types (m), our design generates $N = \lambda p_1 p_2 \ldots p_m$ points.

To determine the p points in hyper-triangles of the cover constraints, we maximized the minimum Euclidean distance between points (Johnson et al. 1990). For a constraint with a single variable, we spread the points evenly across the dimension, but for a constraint with more than one variable (two variables form a triangle, three or more form a hyper-triangle), we spread them evenly across the design space. The problem of maximizing the minimum pairwise distance among n points in a unit square is equivalent to the problem of finding the maximum diameter of equal non-overlapping circles contained in a unit square (Maranas et al. 1995). For example, a constraint with two variables can be represented by a triangle, and its six equidistant points can be found as shown in Figure 6.

5.2.2 Projecting to the Feasible Region

It is critical that the final design points lie within the (convex) feasible region. Different options were explored, including the omission of any infeasible points and the projection of points into the feasible region. The most promising

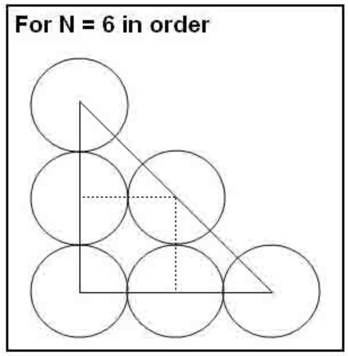

Figure 6: Overlapping circles for generating design

option was projecting points to nearby extreme points of the feasible region. Given that the original set of design points is well-distributed, the resulting set of extreme points should provide good coverage. However, it should be noted that it is not computationally practical to identify all extreme points, so it is possible that some regions of the feasible region will not be covered. The infeasible points (x^0) are projected onto the feasible region using the L1-norm as represented below:

$$\min \ \|x - x^0\| \tag{42}$$

$$\text{s.t.} \quad Ax \leq b, \tag{43}$$

$$x \geq 0. \tag{44}$$

Suppose the solution to the above LP is $(\tilde{x})$. The solution $(\tilde{x})$ actually lies on the face of the feasible polytope $P := \{x | Ax \leq b, x \geq 0\}$. Finding a nearby extreme point to a point on the face of the polytope is a difficult task.

We searched for an extreme point in the direction of the gradient of the L2-norm given by $\|x^0 - \tilde{x}\|^2$ and exploited the features in ILOG CPLEX 8.0. Since every iteration of primal simplex provides a basic feasible solution that is an extreme point, we can search in the direction of the gradient of the L2-norm by providing $\tilde{x}$ as the starting information and stopping the primal simplex method after the first iteration. This will not guarantee the nearest extreme point, but by providing $\tilde{x}$ as the initial primal values and iterating primal simplex once, it is likely that an extreme point close to $\tilde{x}$ is obtained. The LP to optimize the gradient of the L2-norm at the point $\tilde{x}$ is represented as:

$$\min \ (x^0 - \tilde{x})^T x \tag{45}$$

$$\text{s.t.} \qquad Ax \leq b, \tag{46}$$

$$x \geq 0. \tag{47}$$

Nearby extreme points could also be obtained using neighborhood search methods like *breadth first search* (Chvatal 1983).

5.2.3 Interior Feasible Points

Although a MARS approximation could be fit to the extreme points alone, interior points would enable better representation of the shape of the second-stage recourse function. By Minkowski's finite basis theorem (Minkowski 1896), interior points can be obtained by using convex combinations of the extreme points. A second Latin hypercube design is used to specify the coefficients of the convex combinations.

6 Optimization

Once the initial CCA values in the first stage are obtained via the experimental design, random scenarios are generated, and a deterministic second-stage FAM problem is solved for each CCA design point and scenario. The two-stage optimization model was presented in Section 4. An estimate of the revenue was calculated as an average over the scenarios. The objective of the second stage is to maximize profit (Revenue $-$ Cost) of assigning aircraft to each flight leg in the schedule. For calculating the revenue, average fares are multiplied with the minimum of demand and capacity for each scenario. Passengers spilled (customers who cannot be accommodated due to insufficient capacity) are assumed to be lost revenue and are not captured.

6.1 Scenarios

The purpose of scenarios is to provide estimation of an expected value, which is equivalent to numerical integration. Swan (1983) developed an approach to estimate true demand from historical booking patterns and showed that booking data from American Airlines and Swissaire followed either a normal or gamma distribution. In this paper, demand scenarios for each flight are generated based on the normal distribution assuming that the mean and variance are known. Since an average of the scenarios is estimated in the second stage, one would expect the performance of the estimation in the second stage to improve as the number of scenarios increases.

To identify the appropriate number of scenarios, a sample size calculation based on confidence level information is used. Suppose $\bar{y}$ is used as an estimate of average revenue (μ). Then given an estimate for the error variance σ, we can be approximately $100(1-\alpha)$% confident that the error $(|\bar{y}-\mu|)$ will not exceed a specified amount E when the sample size is:

$$\left\lceil \left(\frac{2\, z_{\alpha/2}\, \sigma}{E} \right)^2 \right\rceil .$$

It should be noted that the iterative approximation methods in Birge and Louveaux (1997) require enough scenarios to adequately estimate the expected value and an *expected supergradient* of the recourse function, and it is this latter expectation that ultimately determines the necessary sample size for their methods. By contrast, our DACE approach only requires accurate estimation of the expected revenue, which is a much smoother function than the supergradient.

7 Case Study Results

We applied the methodology described in the previous sections to a real airline carrier with a weekly schedule containing 50 stations, 2358 legs, seven fleet types and four crew-compatible families. The traditional FAM model addressed in Hane et al. (1995) included 6537 integer variables. The turn time was taken as 29 minutes and was assumed constant irrespective of the crew-compatible family. This assumption can be relaxed to make the problem more realistic, but this will make the problem even more complex. Based on the ground time for each station and the traffic intensity, six hubs were identified. The results were computed using a Dual 2.8 GHz Intel Xeon workstation and ILOG CPLEX 8.0 software.

In order to use DoE for generating the first-stage CCA values, the reduced CCA decision space was obtained using the approach presented in Section 5.1. As discussed in Section 5.1.1, the Boeing concept of D^3 was used to create the cover constraints with only crew-compatible families as variables. The obtained cover constraints were used to pivot out one variable for each flight, as mentioned in Section 5.1.2.1. The balance constraints were generated for each station using a time line, as discussed in Section 5.1.2.2, and the equalities obtained were used to pivot out more variables. The plane count constraints were added to the model to maintain feasibility in the first stage. Finally, after reducing the model to the least number of variables using the equalities in the operational constraints, the model was checked for implicit equalities. In our first-stage problem, we did not find any implicit equalities. Pilla et al. (2005) showed that it is possible to generate implicit equalities using the formulation defined in Section 5.1.3. The results of the first-stage dimensionality reduction are presented in Table 1.

Table 1: Computation Results

Total number of variables	6537
Variable reduction due to	
Demand Driven Dispatch (D^3)	2441
Cover Constraint	2358
Balance Constraint	474
Implicit Equalities	0
Remaining variables	1264

The obtained 1264 CCA variables represented 972 legs, with the corresponding cover constraints forming a polytope. Following the methodology discussed in Section 5.2, a Latin hypercube design was generated based on the cover constraints. Out of the 972 legs, the cover constraints of 680 legs involved one variable, and the remaining 292 legs involved two variables. Since n = 972, m = 1 or 2 and N should be greater than 1264 (680*1 + 292*2), we considered N = 1980 design points, with three levels for 680 legs, and six levels for 292 legs as shown below:

- p_1 = $\{$0, 0.5, 1$\}$ (corresponding to three levels)
- p_2 = $\{$(0,0), (0,1), (1,0), (0, 0.5), (0.5, 0), (0.5, 0.5)$\}$ (corresponding to six levels).

These six levels correspond to the six centers of the circles as shown in Figure 6.

The generated Latin hypercube design was mapped to the corresponding levels, which resulted in 1980 first-stage CCA decision values. However, none of these satisfied the first-stage operational constraints. Following the methods in Section 5.2.2, the 1980 infeasible points (x^0) were projected onto the feasible polytope formed by the cover constraints, resulting in 141 points (more than one infeasible point getting projected to the same point, $\tilde{x}$). Since the optimal solution lies at an extreme point, nearby extreme points were obtained by considering the gradient of the L2-norm and using one iteration of ILOG CPLEX 8.0. This resulted in 1525 extreme points (x^*). The spread of these points with respect to the original infeasible DoE points was measured using the ratio

$$\frac{||x^0 - x^*||}{||x^*||}, \tag{48}$$

and the histogram of these ratios is shown in Figure 7. There are chances of some corners in the polytope being left behind during this process as there are a total of $\binom{3695}{1264}$ combinations of extreme points, and it is impossible to take into account all of these. Given that there are no gaps in the histogram, the points appear to be well spread across the polytope. Finally, interior points were obtained using convex combinations of these 1525 extreme points. Thus,

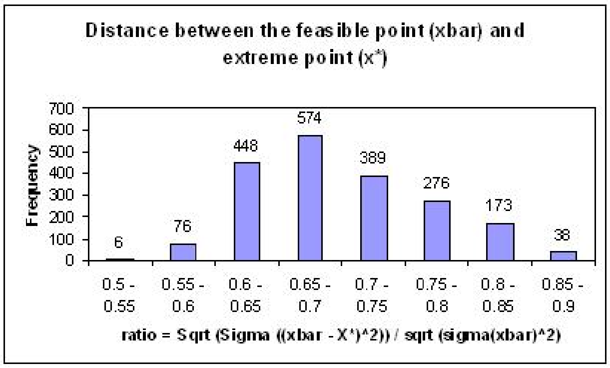

Figure 7: Histogram to check the spread of points

around 3562 total first-stage assignments were obtained, which included 141 feasible projections, 1525 extreme points and 1896 interior points.

In order to solve the second stage, demand scenarios were generated using a normal distribution with known mean and standard deviation values for each flight leg. As discussed in Section 6, for a desired error of approximately 0.1% of the average revenue, we obtained the sample size of scenarios to be 30. For each design point and for each scenario, the second-stage LP as discussed in Section 4 was solved, and the expected revenue was calculated as the average over the scenarios.

Using the average revenue response values at the CCA design values, a MARS approximation was fit using an automatic stopping rule as discussed in Tsai and Chen (2005). The fit resulted in 320 basis functions with a coefficient of determination (R^2) of 99.013%. A validation data set was generated (utilizing new convex combinations of 1525 extreme points) to test the MARS approximation, and relative errors were calculated using the formula

$$\frac{|y - \hat{y}|}{|y|}. \tag{49}$$

The maximum relative error was obtained as 3 x 10^{-5} and the median relative error was 6 x 10^{-6}, both of which are very small. A new set of extreme points was generated to evaluate the MARS approximation in regions possibly outside of that represented by the experimental design, and the relative error was on the order of 10^{-3}. Thus, the MARS approximation is a very good fit for the second-stage recourse function.

8 Future Work

The main contribution of this paper is a method to estimate the recourse function for the two-stage SP model using a MARS approximation over a discretized first-stage decision space based on a Latin hypercube design. The obtained fit can be used for future evaluations and optimized so as to generate a single first-stage decision. In this aspect, derivatives of the MARS function can be utilized to generate revenue cuts. The revenue generated from using our approach will be compared to the present methods utilized.

References

Abara, J. (1989). Applying integer linear programming to the fleet assignment problem, *Interfaces* **19**(4): 20 – 28.

Ahuja, R. and Orlin, J. (2002). Very large scale neighborhood search in airline fleet scheduling, *SIAM News* **35**(9).

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

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

Berge, M. and Hopperstad, C. (1993). Demand driven dispatch: A method of dynamic aircraft capacity assignment, models and algorithms, *Operations Research* **41**(1): 153 – 168.

Birge, J. R. and Louveaux, F. (1997). *Introduction to Stochastic Programming*, Springer, New York, New York.

Cervellera, C., Chen, V. and Wen, A. (2005). Optimization of a large-scale water reservoir network by stochastic dynamic programming with efficient state space discretization, *European Journal of Operational Research, to appear* .

Chen, V. (1999). Application of mars and orthogonal arrays to inventory forecasting stochastic dynamic programs, *Computational Statistics and Data Analysis* **30**: 317 – 341.

Chen, V. (2001). Measuring the goodness of orthogonal array discretizations for stochastic programming and stochastic dynamic programming, *SIAM Journal of Optimization* **12**(2): 322 – 344.

Chen, V. C. P., Tsui, K.-L., Barton, R. R. and Meckesheimer, M. (2005). Design, modeling, and applications of computer experiments, *IIE Transactions, to appear* .

Chen, V., Günther, D. and Johnson, E. (2003). Solving for an optimal airline yield management policy via statistical learning, *Journal of the Royal Statistical Society, Series C* **52**(1): 1 – 12.

Chen, V., Ruppert, D. and Shoemaker, C. (1999). Applying experimental design and regression splines to high-dimensional continuous-state stochastic dynamic programming, *Operations Research* **47**: 38 – 53.

Chen, V., Tsui, K., Barton, R. and Allen, J. (2003). A review of design and modeling in computer experiments, *Handbook of Statistics* **22**: 231 – 261.

Chvatal, V. (1983). *Linear Programming*, W. H. Freeman and Company, New York, New York.

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

Farkas, A. (1996). *The Influence of Network Effects and Yield Management on Airline Assignment Decisions*, PhD thesis, Massachusetts Institute of Technology.

Friedman, J. H. (1991). Multivariate adaptive regression splines, *The Annals of Statistics* **19**(1): 1 – 141.

Gao, L. and Zhang, Y. (2002). Computational experience with lenstra's algorithm, *Technical Report TR02-12*, Rice University, Department of Computational and Applied Mathematics.

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

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

IATA (2005a). Passenger growth slows in august - fuel prices begin to dampen demand, http://www.iata.org/pressroom/pr/2005-09-30-02.htm.

IATA (2005b). Us$7.4 billion losses for global aviation in 2005, http://www.iata.org/pressroom/pr/2005-09-12-01.htm.

Jacobs, T., Johnson, E. and Smith, B. (1999). O-d fam: Incorporating passenger flows into the fleeting process, *AGIFORS Symposium Proceedings, 39*, New Orleans, LA, USA.

Johnson, M., Moore, L. and Ylvisaker, D. (1990). Minimax and maximin distance designs, *Journal of Statistical Planning and Inference* **26**: 131 – 148.

Klabjan, D., Schaefer, A., Johnson, E., Kleywegt, A. and Nemhauser, G. (2002). Airline crew scheduling with time windows and plane-count constraints, *Transportation Science* **36**: 337 – 348.

Kleijnen, J. P. C. (2005). An overview of the design and analysis of simulation experiments for sensitivity analysis, *European Journal of Operational Research* **164**(2): 287 – 300.

Listes, O. and Dekker, R. (2002). A scenario aggregation based approach for determining a robust airline fleet composition, *Technical Report Econometric Institute Report EI 2002-17*, Erasmus University Rotterdam, The Netherlands.

Maranas, C., Floudas, C. and Pardalos, P. (1995). New results in the packing of equal circles in a square, *DMATH: Discrete Mathematics* **142**.

Minkowski, H. (1896). *Geometrie der Zahlen*, Teubner Leipzig.

Myers, R. and Montgomery, D. (1986). *Response Surface Methodology*, John Wiley and Sons Inc, New York, New York.

Pilla, V., Rosenberger, J. and Chen, V. (2005). An lp formulation for implicit dimensionality reduction, *Technical Report COSMOS-05-02*, University of Texas at Arlington, Department of Industrial Engineering.

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

Rockafellar, R. and Wets, R. (1991). Scenarios and policy aggregation in optimization under uncertainty, *Mathematics of Operations Research* **16**(1): 119 – 147.

Rosenberger, J., Johnson, E. and Nemhauser, G. (2003). A robust fleet assignment model with hub isolation and short cycles, *Transportation Science* **38**(3): 357 – 368.

Rushmeier, R. and Kontogiorgis, S. (1997). Advances in the optimization of airline fleet assignment, *Transportation Science* **31**(2): 159 – 169.

Sacks, J., Schiller, S. and Welch, W. (1989). Designs for computer experiments, *Technometrics* **31**: 41 – 47.

Savelsbergh, M. (1994). Preprocessing and probing techniques for mixed integer programming problems, *ORSA Journal on Computing* **6**(4): 445 – 454.

Si, J., Barto, A., Powell, W. and Wunsch, D. (2004). *Handbook of Learning and Approximate Dynamic Programming*, Wiley, New York: Wiley.

Sierksma, G. and Tijssen, G. (2003). Degeneracy degrees of constraint collections, *Mathematical Methods of Operations Research* **57**: 437 – 448.

Subramanian, R., Scheff, R., Quillinan, J., Wiper, D. and Marsten, R. (1994). Coldstart: fleet assignment at delta air lines, *Interfaces* **24**(1): 104 – 120.

Swan, W. (1983). Traffic losses at high load factors, *AGIFORS Symposium Proceedings, 23*, Olive Branch, MS, USA.

Tsai, J. and Chen, V. (2005). Flexible and robust implementations of multivariate adaptive regression splines within a wastewater treatment stochastic dynamic program, *Quality and Reliability Engineering International* **21**(7): 689 – 699.

Tsai, J., Chen, V., Chen, J. and Beck, M. (2003). Stochastic dynamic programming formulation for a wastewater treatment decision-making framework, *Annals of Operations Research, Special Issue on Applied Optimization Under Uncertainty* **132**: 207 – 221.

Werbos, P. (1998). *A Brain-Like Design To Learn Optimal Decision Strategies in Complex Environments*, Springer, Springer, London.

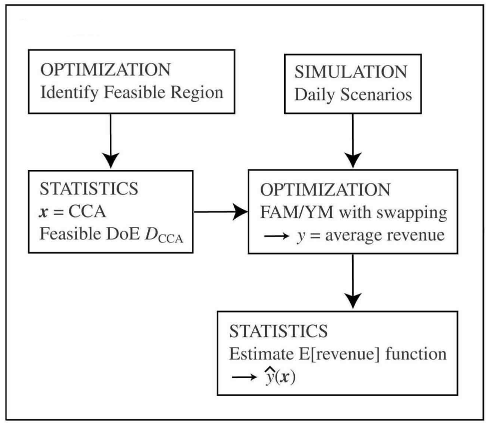

Figure 1: DACE Phase of the Two-Stage SP Framework

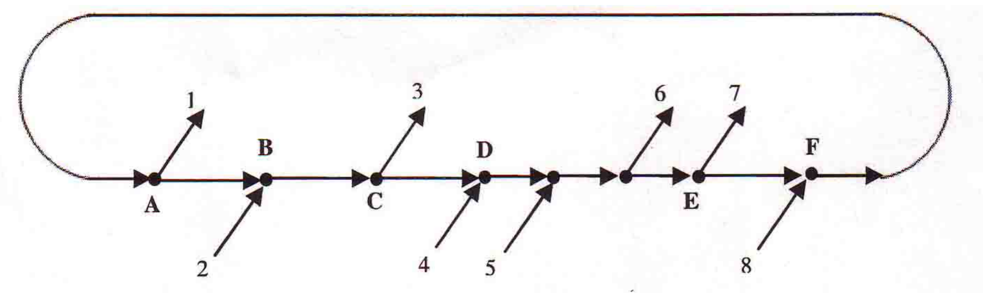

Figure 2: Time Line

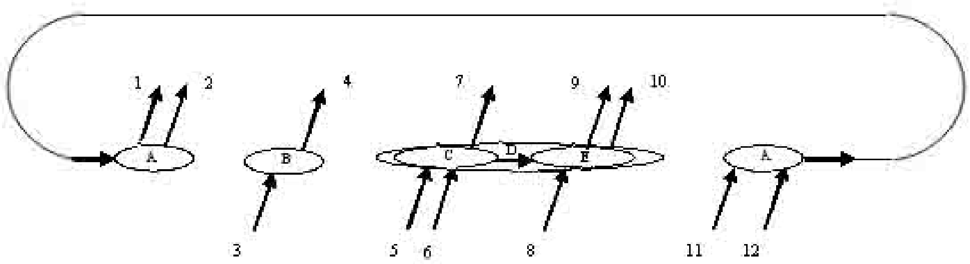

Figure 3: Spoke Time Line

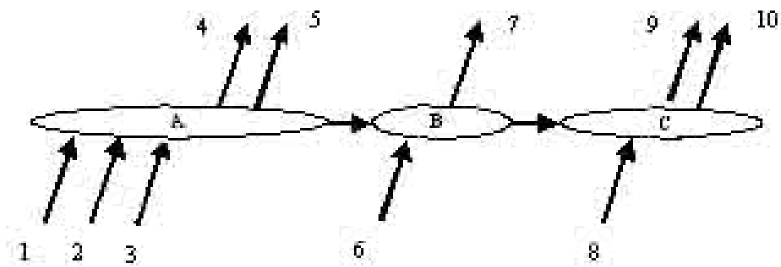

Figure 4: Hub Time Line

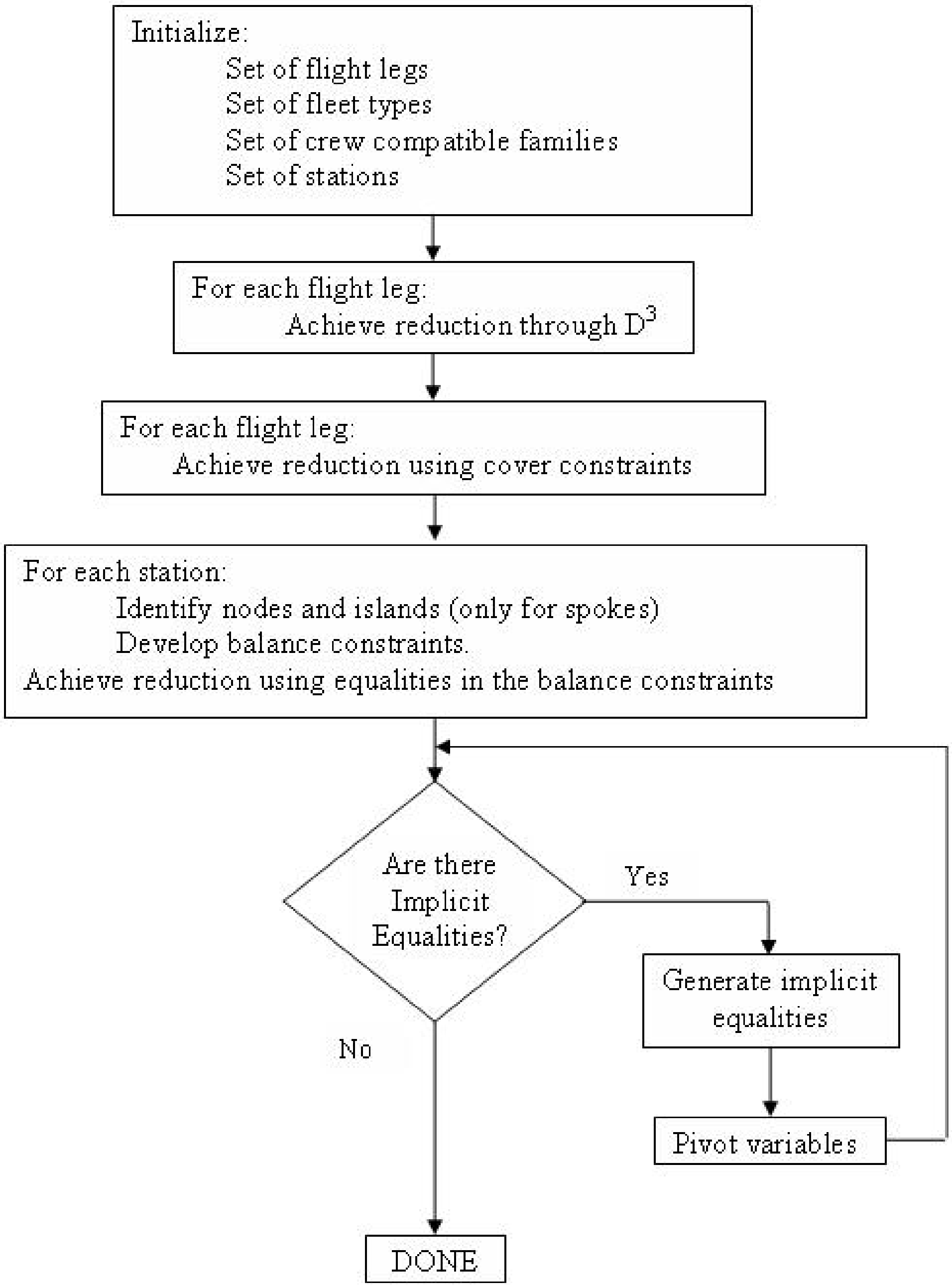

Figure 5: Decision Space Reduction Flow Chart

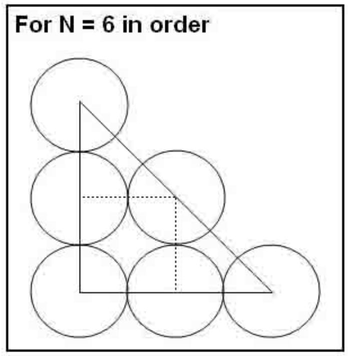

Figure 6: Overlapping circles for generating design

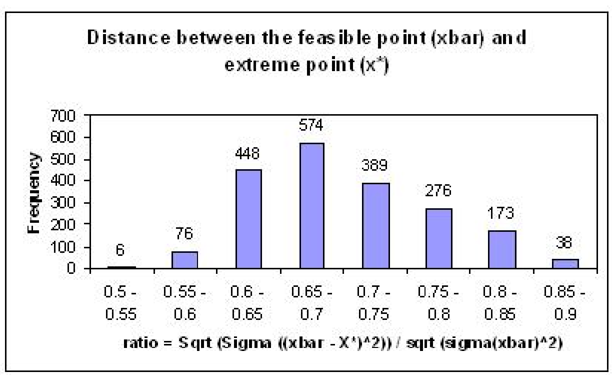

Figure 7: Histogram to check the spread of points