A Note on "A Stochastic Programming Approach to the Airline Crew Scheduling Problem" by J.W. Yen and J.R. Birge, Transportation Science, 40:3-14, 2006

Bassy Tam[1], Matthias Ehrgott[1,2], David Ryan[1], Golbon Zakeri[1]

[1]Department of Engineering Science, The University of Auckland,

Private Bag 92019, Auckland, New Zealand

email: bassytam@gmail.com, {m.ehrgott,d.ryan,g.zakeri}@auckland.ac.nz,

[2]Laboratoire d'Informatique de Nantes Atlantique, Université de Nantes,

2, rue de la Houssinière BP 92208, 44322 Nantes Cedex 03, France

email: matthias.ehrgott@univ-nantes.fr

Abstract

Yen and Birge (2006) formulate the airline crew scheduling problem as a two-stage stochastic integer programme with recourse. They develop an algorithm, the flight-pair branching algorithm, to solve the problem. In this note we show, by means of an example, that the algorithm does not necessarily terminate with an optimal solution.

Keywords: Airline crew scheduling; stochastic programming.

1 Airline Crew Scheduling

The Tour-of-Duty (ToD) or pairings problem in airline scheduling consists in partitioning the scheduled flights into sequences of flights that crew members can operate. Each such sequence is referred to as a Tour-of-Duty (ToD) or pairing and must obey a set of legal and contractual rules. A ToD consists of one or more duty periods, normally starting and ending at a specified airport, the crew's home base (crew base). A duty period is a crew's working day which consists of a sequence of flights with ground times that cannot be shorter than a certain time period, e.g. 30 minutes, between consecutive flights, meal times and maybe passengering flights, in which a crew member travels as a passenger to get to a particular airport for a subsequent flight or back to the crew base.

The ToD problem can be formulated as a generalized set partitioning problem (GSPP) with the objective of minimizing the crew cost incurred by operating the flight schedule (Butchers *et al.*, 2001; Barnhart *et al.*, 2003):

$$\begin{array}{lrcl} \text{minimize} & c^T x & & \\ \text{subject to} & \boldsymbol{A}x & = & e \\ & \boldsymbol{M}x & \left\{\begin{matrix} \geq \\ = \\ \leq \end{matrix}\right\} & b \\ & x & \in & \{0,1\}^n. \end{array} \tag{1}$$

Each column or variable corresponds to a legal ToD. The value of c_j, the cost of variable j, reflects the dollar cost of operating the j^{th} ToD. The decision variable x_j is equal to 1 if pairing j is included in the solution and 0 otherwise.

The flight constraints $Ax = e$, where e is a vector of ones, ensure that each flight sector is included in exactly one ToD, the elements of the A matrix are

$$a_{ij} = \begin{cases} 1, & \text{if pairing } j \text{ contains flight } i \text{ as operating sector,} \\ 0, & \text{otherwise.} \end{cases}$$

The crew base balancing constraints $Mx\{\leq,=,\geq\}b$ ensure that the distribution of work over the set of crew bases matches the crew resources by imposing restrictions on the number of crew resources included from each crew base.

Optimization approaches to solve problem (1) usually use column generation to generate pairings (Anbil *et al.*, 1998). Because the number of legal pairings is extremely large even for problems with relatively few flights this is embedded in the optimization as dynamic column generation (Barnhart *et al.*, 1998). To obtain an integer solution, a branch and price approach with constraint branching is used (Ryan and Foster, 1981). Follow-on branching (Ryan and Falkner, 1988), which is a constraint branching strategy commonly used for this type of problem, is to force or ban two flights to be operated as a subsequence in a pairing. The flights F_r, F_s are operated as subsequence if a crew assigned to operate flight F_s after the operation of the flight F_r, with no other operating sector in between flights F_r and F_s. On the one branch, all pairings that operate only one of the two flights are eliminated. On the zero branch, all pairings that operate both flights are eliminated.

Traditionally, the ToD planning problem is solved well before the flight schedule becomes operational. In this planning stage, all flights are assumed to have departure times that are both fixed and known. This assumption is often proven wrong when the crew schedule is actually implemented. Optimal solutions to the ToD problem tend to allow only short ground times between consecutive flights. With such solutions, initial minor delays can quickly result in major problems through chain impacts: A late arriving flight does not only cause the following flight on the same aircraft to depart late, but also those flights which are operated by late arriving crew members on a different aircraft. To recover from such disruptions can cause large costs (Ehrgott and Ryan, 2002).

To address this problem and incorporate the short term problem of recovery from disruptions in the ToD planning problem, Yen and Birge (2006) formulate the ToD planning problem as a two-stage stochastic programme with recourse. The model takes the uncertainty in flight departure times and flight duration into account and allows to find ToD solutions the total cost (crew cost plus recovery cost) of which is lower than that of solutions of the traditional deterministic model.

Given a crew schedule, the recourse problem is a large-scale LP to measure the cost of delays, with the first stage problem being the traditional ToD planning (1). Yen and Birge (2006) develop a method based on follow-on branching to solve the model. They sample 100 disruption scenarios and evaluate the solution of the second stage LP for each scenario to determine the "switching cost" associated with aircraft changes. The "switching cost" is then passed back to the first stage problem to remove any "expensive" aircraft changes, by branching on the sector pair with the highest "switching cost".

In the next section we describe the stochastic programming model and the algorithm proposed by Yen and Birge (2006). In Section 3 we show that, contrary to the statement in Yen and Birge (2006), the algorithm does not always find an optimal solution of the model.

2 The Stochastic Programming Approach

A crew schedule x obtained from solving the ToD planning problem (1) can be evaluated under some disruption scenarios. Let a disruption scenario ω be a random element of some space Ω, that occurs with probability $\mathcal{P}(\omega)$. The crew schedule x incurs a recovery cost $\mathcal{Q}(x,\omega)$. The expected value of the cost of future action to operate the crew schedule x is denoted by $\mathcal{Q}(x)$ and it is defined as

$$\mathcal{Q}(x) = \sum_{\omega\in\Omega} \mathcal{P}(\omega)\mathcal{Q}(x,\omega).$$

Thus, the stochastic programming formulation of the robust ToD planning problem is

$$\begin{array}{llcl} \text{minimize} & z = c^T x + \mathcal{Q}(x) & & \\ \text{subject to} & \boldsymbol{A}x & = & e \\ & \boldsymbol{M}x & \left\{\begin{matrix} \geq \\ = \\ \leq \end{matrix}\right\} & b \\ & x & \in & \{0,1\}^n. \end{array} \tag{2}$$

To evaluate a crew schedule under a disruption scenario ω, the pushback recovery procedure is used. Pushback recovery means that a flight is delayed until all resources (crew members and aircraft) are available. Below is a summary of our notation. Let

- SDT_{F_i} be the scheduled departure time of the flight F_i,
- SAT_{F_i} be the scheduled arrival time of the flight F_i,
- FT_{F_i} be the scheduled flight time of F_i, i.e. $\mathrm{FT}_{F_i} = \mathrm{SAT}_{F_i} - \mathrm{SDT}_{F_i}$,
- F_i^p be the plane predecessor flight of F_i,
- $F_i^{c_n}$ be the crew predecessor flight of F_i under pairing n,
- $\mathrm{PGT}_{F_i}^{F_i^p}$ be the minimum required plane ground time between the flights F_i^p and F_i,
- $\mathrm{GDT}_{F_i}^{F_i^{c_n}}$ be the minimum required ground duty time between the flights $F_i^{c_n}$ and F_i,
- $\mathrm{DT}_{F_i}^{\omega}$ be the delay time associated with the flight F_i under scenario ω, $\mathrm{DT}_{F_i}^{\omega}$ is a random variable,
- $\mathrm{ADT}_{F_i}^{\omega}$ be the actual departure time of the flight F_i under scenario ω,

- $\mathrm{AAT}^{\omega}_{F_i}$ be the actual arrival time of the flight F_i under scenario ω,
- $\widehat{\mathrm{ADT}}^{\omega}_{F_i}$ be the actual departure time of the flight F_i under scenario ω without crew interactions,
- $\widehat{\mathrm{AAT}}^{\omega}_{F_i}$ be the actual arrival time of the flight F_i under scenario ω without crew interactions,
- $\mathrm{TD}^{\omega}_{F_i}$ be the total delay to the flight F_i under scenario ω,
- $\mathrm{ND}^{\omega}_{F_i}$ be the non-crew induced delay to the flight F_i under scenario ω.

Considering a crew schedule x consisting of N pairings, the actual departure time of a flight F_i under a disruption scenario ω is

$$\mathrm{ADT}^{\omega}_{F_i} = \max\left\{\mathrm{SDT}_{F_i}, \mathrm{AAT}^{\omega}_{F_i^p} + \mathrm{PGT}^{F_i^p}_{F_i}, \mathrm{AAT}^{\omega}_{F_i^{cn}} + \mathrm{GDT}^{F_i^{cn}}_{F_i}\right\}$$

while the actual departure time without crew interactions is

$$\widehat{\mathrm{ADT}}^{\omega}_{F_i} = \max\left\{\mathrm{SDT}_{F_i}, \widehat{\mathrm{AAT}}^{\omega}_{F_i^p} + \mathrm{PGT}^{F_i^p}_{F_i}\right\}.$$

The actual arrival time of a flight F_i under a disruption scenario ω is

$$\mathrm{AAT}^{\omega}_{F_i} = \mathrm{ADT}^{\omega}_{F_i} + \mathrm{FT}_{F_i} + \mathrm{DT}^{\omega}_{F_i}$$

while the actual arrival time without crew interactions is

$$\widehat{\mathrm{AAT}}^{\omega}_{F_i} = \widehat{\mathrm{ADT}}^{\omega}_{F_i} + \mathrm{FT}_{F_i} + \mathrm{DT}^{\omega}_{F_i}.$$

The total delay, including plane induced delay and crew induced delay, to the flight F_i under scenario ω is defined by the actual arrival time of the flight F_i under scenario ω minus the scheduled arrival time of the flight F_i, that is

$$\mathrm{TD}^{\omega}_{F_i} = \mathrm{AAT}^{\omega}_{F_i} - \mathrm{SAT}_{F_i}$$

and the non-crew induced delay to the flight F_i under scenario ω is

$$\mathrm{ND}^{\omega}_{F_i} = \widehat{\mathrm{AAT}}^{\omega}_{F_i} - \mathrm{SAT}_{F_i}.$$

The recovery cost of operating crew schedule x for a flight schedule consisting of S flights under the disruption scenario ω is thus

$$\mathcal{Q}(x,\omega) = \sum_{i=1}^{S} p_{F_i}(\mathrm{TD}^{\omega}_{F_i} - \mathrm{ND}^{\omega}_{F_i}),$$

where p_{F_i} is the penalty cost for each minute of delay of flight F_i.

After the crew schedule is evaluated, flight pairs are priced by switch delay. Switch delay is a delay due to crew changing aircraft.

Given that the flights F_i and F_j are not on the same aircraft and crew were assigned to perform the connection $(F_i,\ F_j)$ in the ToD, switch delay for connection $(F_i,\ F_j)$ under scenario ω is the total delay of flight F_j under scenario ω minus the delay incurred if the crew were assigned to perform the connection $(F_j^p,\ F_j)$ under scenario ω. Hence, if we consider a crew schedule x consisting of N pairings, the switch delay for a flight connection $(F_i,\ F_j)$ over all scenarios in Ω is

$$\mathrm{SD}_{F_j}^{F_i} = \sum_{n=1}^{N} \delta_{nij} \sum_{\omega \in \Omega} \mathcal{P}(\omega) \max\left\{0, \mathrm{TD}_{F_j}^{\omega} - \left(\mathrm{AAT}_{F_j^p}^{\omega} + \max\left\{\mathrm{PGT}_{F_j}^{F_j^p}, \mathrm{GDT}_{F_j}^{F_j^p}\right\} - \mathrm{SDT}_{F_j}\right)\right\},$$

where

$$\delta_{nij} = \begin{cases} 1, & \text{if} \begin{cases} x_n &= 1, \\ a_{in} &= 1, \\ a_{jn} &= 1, \\ F_j^{c_n} &= F_i, \\ \text{plane}(F_i) &\neq \text{plane}(F_j) \end{cases} \\ 0, & \text{otherwise.} \end{cases}$$

Once the switch delay for each flight pair is calculated for a crew schedule x, the flight pair with the highest switch delay cost is banned to appear from any pairing selected in the next GSPP solution. This is equivalent to imposing a 0-branch on the flight pair with highest switch delay and is called called flight-pair branching. If the total cost (crew cost and recovery cost) of the new crew schedule is worse than that of the previous one, a 1-branch is imposed on that highest switch delay flight pair, and a 0-branch is imposed on the next highest switch delay flight pair.

The processes of solving (1) to find a new crew schedule, evaluating the cost minimal crew schedule, calculating switch cost and flight-pair branching are repeated until no flight pair with positive switch delay cost is found or the increase of planned crew cost is larger than the decrease of the expected recovery cost.

3 The Example

Yen and Birge (2006) state that the algorithm must terminate with an optimal solution. We argue that in some circumstances the true optimal solution cannot be found. We demonstrate this by an example.

Suppose our flight schedule consists of one day of operation with six flights $\{F_1, F_2, \ldots, F_6\}$ operated by two planes (P_1 and P_2) serving three cities (A, B and C). Suppose we only have one delay scenario that always happens. The flight schedule and delay details are as follows.

Flight	**Plane**	**Origin**	**Destination**	**SDT** (hh:mm)	**SAT** (hh:mm)	**DT** (min)
1	1	A	B	12:00	13:00	5
2	2	C	B	12:00	13:00	0
3	2	B	C	13:30	15:00	0
4	1	B	C	13:45	15:15	15
5	2	C	B	15:45	17:15	0
6	1	C	A	16:00	18:00	0

Delay time (DT) is the independent delay time associated with each flight. We assume the minimum plane ground time (PGT) and minimum ground duty time (GDT) to be 30 minutes between any two flights and that passengering is not allowed. For this flight schedule eight different duty periods $\{D_1, D_2, \ldots, D_8\}$ are possible. Note that duty periods coincide with ToDs in this example.

Duty Period	**Path**
1	$F_1 \rightarrow F_3 \rightarrow F_5$
2	$F_1 \rightarrow F_3 \rightarrow F_6$
3	$F_1 \rightarrow F_4 \rightarrow F_5$
4	$F_1 \rightarrow F_4 \rightarrow F_6$
5	$F_2 \rightarrow F_3 \rightarrow F_5$
6	$F_2 \rightarrow F_3 \rightarrow F_6$
7	$F_2 \rightarrow F_4 \rightarrow F_5$
8	$F_2 \rightarrow F_4 \rightarrow F_6$

With these eight duty periods, four possible solutions $\{x^1, x^2, \ldots, x^4\}$ of the pairings problem can be constructed.

Solution	**Duty Periods**
1	D_1, D_8
2	D_2, D_7
3	D_3, D_6
4	D_4, D_5

Let us assume that c is such that

$$c^T x^1 < c^T x^3 < c^T x^4 < c^T x^1 + \mathcal{Q}(x^1) < c^T x^3 + \mathcal{Q}(x^3)$$

and that $c^T x^1 < c^T x^2$. We also assume that the penalty cost p_{F_i} is 1 for all flights F_i.

In particular, x^1 is the unique optimal solution of (1) and we evaluate its on-time performance.

Duty Period	**Flight**	**ADT** (hh:mm)	**AAT** (hh:mm)	**TD** (min)	**ND** (min)
1	1	12:00	13:05	5	5
	3	13:35	15:05	5	0
	5	15:45	17:15	0	0
8	2	12:00	13:00	0	0
	4	13:45	15:30	15	15
	6	16:00	18:00	0	0

The recovery cost $\mathcal{Q}(x^1)$ of this solution is 5. The switching cost of the flight pair (F_1, F_3) is 5 and the switching cost of all other flight pairs is 0. According to the flight-pair branching algorithm, the flight pair (F_1, F_3) is banned from appearing in any pairing selected in the next solution, so the duty periods D_1 and D_2 are banned, and hence solutions x^1 and x^2 are no longer feasible.

The crew schedule x^3 is the next solution. The on-time performance of x^3 is as follows.

Duty Period	**Flight**	**ADT**	**AAT**	**TD**	**ND**
		(hh:mm)	(hh:mm)	(min)	(min)
3	1	12:00	13:05	5	5
	4	13:45	15:30	15	15
	5	16:00	17:30	15	0
6	2	12:00	13:00	0	0
	3	13:30	15:00	0	0
	6	16:00	18:00	0	0

The recovery cost $\mathcal{Q}(x^3)$ of this solution is 15. By our assumption, $c^T x^1 + \mathcal{Q}(x^1) < c^T x^3 + \mathcal{Q}(x^3)$, i.e. the difference in crew cost between x^1 and x^3 is small compared to the difference of 10 of the recovery cost. Following the algorithm this node is fathomed, the 0-branch on the flight pair (F_1, F_3) is removed, and a 1-branch is imposed. In other words, the flight pair (F_1, F_3) is forced to appear in the next solution, so the duty periods D_3, D_4, D_5 and D_6 are banned, and hence solutions x^3 and x^4 are infeasible.

After the 1-branch is imposed on the flight pair (F_1, F_3), x^1 is of course the optimal solution. In this solution, the switching cost for all other flight pairs is 0, i.e. the algorithm terminates with solution x^1.

However, the true optimal solution, x^4, cannot be identified with the algorithm. The on-time performance of x^4 is as follows.

Duty Period	**Flight**	**ADT**	**AAT**	**TD**	**ND**
		(hh:mm)	(hh:mm)	(min)	(min)
4	1	12:00	13:05	5	5
	4	13:45	15:30	15	15
	6	16:00	18:00	0	0
5	2	12:00	13:00	0	0
	3	13:30	15:00	0	0
	5	15:45	17:15	0	0

The recovery cost $\mathcal{Q}(x^4)$ of this solution is 0. Thus, by our assumption, $c^T x^4 + \mathcal{Q}(x^4) = c^T x^4 < c^T x^1 + \mathcal{Q}(x^1)$. Solution x^4 must be obtained by a ban on the flight pair (F_1, F_3), yielding solution x^3, followed by a ban on the flight pair (F_4, F_5). Since the node with a 0-branch on (F_1, F_3) is fathomed, x^4 can never be obtained. Note that solution x^2 has $\mathcal{Q}(x^2) = 20$ and is hence worse than x^1 in both the deterministic and stochastic model.

The example shows that one of the stopping criteria for the flight-pair branching algorithm is invalid, namely that the increase of planned crew cost is larger than the decrease of the recovery cost. The planned crew cost is an increasing function along a branch of the tree constructed by the flight-pair branching algorithm, but since the recovery cost is not a decreasing function, the upper bound of the algorithm is not valid.

References

Anbil, R., Forrest, J., and Pulleyblank, W. (1998). Column generation and the airline crew pairing problem. *Documenta Mathematica*, **Extra volume ICM III**, 677 – 686.

Barnhart, C., Johnson, E. L., Savelsbergh, M. W. P., and Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, **46**, 316 – 329.

Barnhart, C., Cohn, A., Johnson, E., Klabjan, D., Nemhauser, G., and Vance, P. (2003). Airline crew scheduling. In R. Hall, editor, *Handbook of Transportation Science*, pages 493 – 521. Kluwer Scientific Publishers.

Butchers, E. R., Day, P. R., Goldie, A. P., Miller, S., Meyer, J. A., Ryan, D. M., Scott, A. C., and A., W. C. (2001). Optimized crew scheduling at Air Zealand. *Interfaces*, **31**, 30 – 56.

Ehrgott, M. and Ryan, D. (2002). Constructing robust crew schedules with bicriteria optimization. *Journal of Multi-Criteria Decision Analysis*, **11**, 139 – 150.

Ryan, D. M. and Falkner, J. (1988). On the integer properties of scheduling set partitioning models. *European Journal of Operational Research*, **35**, 442 – 456.

Ryan, D. M. and Foster, B. A. (1981). An integer programming approach to scheduling. In A. Wren, editor, *Computer Scheduling of Public Transport*, pages 269–280. North-Holland, Amsterdam.

Yen, J. W. and Birge, J. R. (2006). A stochastic programming approach to the airline crew scheduling problem. *Transportation Science*, **40**, 3 – 14.