Bicriteria Robustness versus Cost Optimisation in Tour of Duty Planning at Air New Zealand

Matthias Ehrgott, David M. Ryan
Department of Engineering Science
University of Auckland
New Zealand
e-mail: {m.ehrgott,d.ryan}@auckland.ac.nz

Abstract

Current optimisation based computer systems to solve all aspects of both planning and rostering processes for both national and international airline operations allow the construction of minimal cost tours of duty and rosters. However, today major airlines do not only require cost effective solutions, but are also very interested in robust solutions. A more robust solution is understood to be one where disruptions in the schedule (due to delays) are less likely to be propagated into the future, causing delays of subsequent flights. Current scheduling systems based solely on cost do not automatically provide robust solutions.

These considerations lead to a multiobjective framework, as the maximisation of robustness will be in conflict with the minimisation of cost, as for example crew changing aircraft between duties is discouraged if inadequate ground time is provided. We develop a bicriteria optimisation framework to generate "efficient" schedules for the domestic airline. An efficient schedule is one which does not allow an improvement in cost and robustness at the same time.

In this paper, we present preliminary results on the incorporation of a robustness objective in the optimisation process for tour of duty planning. We also outline future research to be undertaken in this area.

1 The Tour of Duty Planning Problem

Commercial airlines are required to solve many resource scheduling problems to ensure that aircraft and aircrew are available for each scheduled flight. The aircrew scheduling problem is usually partitioned into two distinct sub-problems. The first problem, called the "Tours of Duty" (ToD) planning problem, involves the construction of sequences of flights to crew the flight schedule. The ToD planning problem has been a subject of considerable interest in the airline industry. It has prompted a great deal of research (see [1]). The second sub-problem associated with aircrew scheduling is referred to as rostering. The rostering process involves the allocation of the planned ToDs from the first sub-problem to individual crew members to form a "Line of Work" (LoW) for each crew member over the rostering period.

In this paper we address the the ToD planning problem. First we describe the problem and introduce the generalised set partitioning model which is used as the underlying mathematical model for the ToD planning problem.

A Tour of Duty (ToD) is an alternating sequence of duty periods and rest periods (or layovers), where duty periods comprise one or more flights, and may also include passengering flights. A passengering flight is one on which a crew member travels as a passenger in order to be positioned at a particular airport for a subsequent operating flight. The first duty period of a ToD must start at a crew base, and the last duty period must end at the same crew base. An airline might have several bases (cities) at which crew are domiciled. Each ToD will have an associated crew complement made up of a number of crew of some ranks.

The set partitioning model provides an underlying mathematical model for the ToD planning and the rostering sub-problems of the aircrew scheduling problem. The set partitioning problem (SPP) is a specially structured zero-one integer linear programme with the form

$$\begin{aligned} \min z &= c^T x \\ \text{subject to } Ax &= e \\ x &\in \{0,1\}, \end{aligned} \qquad \text{SPP}$$

where $e = (1,1,1,\ldots,1)^T$ and A is a matrix of zeros and ones.

In the basic ToD planning model, each column or variable in SPP corresponds to one possible ToD that could be flown by some crew member. Each constraint in SPP corresponds to a particular flight and ensures that the flight is included in exactly one ToD. The elements of the A matrix can then be defined as

$$a_{ij} = \begin{cases} 1 & \text{if the } j\text{'th ToD (variable) includes the } i\text{'th flight (constraint)} \\ 0 & \text{otherwise.} \end{cases}$$

The ToD planning model is usually augmented with additional constraints that permit restrictions to be imposed on the number of ToDs included from each crew base. Because these constraints typically have non-unit right-hand-side values, we describe the ToD planning model as a generalised set partitioning model. At Air New Zealand, the ToD planning model is formed and solved independently for each crew type and the flights they operate.

2 Measuring Robustness of Tours of Duty

Air New Zealand do not only require minimum cost solutions, but are also very interested in robust solutions. A robust solution is understood to be one where disruptions in the schedule (due to delays) are less likely to be propagated into the future, causing delays of subsequent flights. A feature of a robust solution would be that ground time between subsequent flights be greater than the minimum required ground time and crew changing aircraft occurs less frequently. Current scheduling systems based solely on cost do not automatically provide robust solutions. In fact, the natural preference in a minimum cost solution will involve minimum ground time connections between subsequent flights.

The importance of considering robust solutions can be illustrated by looking at the average delay of flights plotted over the time of the day. Figure 1 shows the average delay in two hour periods, starting from 6 am to 8 pm until 10 pm to midnight.

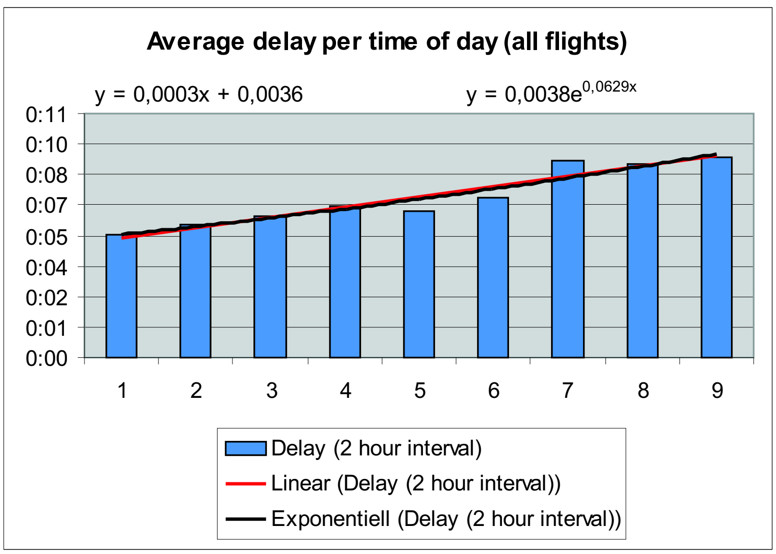

Figure 1: Increasing Average Delay During the Day

Figure 1 clearly shows a basically linear increase in the expected delay. The cause of this increase may be explained through the propagation of delays through the flight schedule. If a flight arrives late at its destination port and the ToD schedule requires crew to change aircraft the subsequent flight on that aircraft will also be delayed, and might in turn cause further delays. If, however, the crew follow the aircraft on a subsequent sector, delays will only effect flights on the same aircraft. Therefore a robust solution is one in which crew changing aircraft is discouraged if insufficient ground time occurs to compensate late arrivals.

Based on this observation we develop an objective function by penalizing ToDs which are not robust. We then try to minimize this objective function while at the same time maintaining a cost effective solution. The robustness measure for each ToD is obtained by considering each sector pair in any given ToD. A penalty will be incurred, if the scheduled ground time minus minimal ground time, minus meal breaks, if any, is less than the expected delay. A measure of the violation of a robust solution by a given sector pair is then the amount by which the expected delay of the incoming flight exceeds the remaining idle time before the departure of the outgoing flight. This will only be done, if an aircraft change is required, whereas if the sectors follow on the same aircraft, no penalty is incurred, as the delay only effects flights on this same aircraft, which are inevitable anyway. In this way we obtain the robustness objective function $r^t x$, where r_j is the penalty for the j'th ToD, obtained by accumulating the penalty for each consecutive sector pair in the ToD. For the preliminary results we report here, we disregarded ground time and simply added expected delays on non-follow on flights.

3 Bicriteria Optimisation in Tour of Duty Planning

The consideration of both a cost objective function and a robustness objective function leads us to the following bicriteria optimization, which is obtained from including a second objective in SPP.

$$
\begin{array}{rcl}
\min z^1 & = & c^T x \\
\min z^2 & = & r^T x \\
\text{subject to } A_1 x & = & e \\
A_2 x & = & b \\
x & \in & \{0,1\},
\end{array}
\qquad \text{2SPP}
$$

where the first set of constraints are the constraints from the scheduled flights and the second are the base constraints.

In this model, instead of optimal solutions we look for efficient solutions. A solution x^* is called efficient, if there is no other solution which is at least as good as x^* with respect to both objectives and strictly better with respect to one. In other words, we want to identify sets of ToDs, which have the property, that if we want to improve (decrease) either their robustness measure or their cost value, the other objective would necessarily have to increase. In multicriteria optimization there will usually exist several, often many efficient solutions, all with different cost and robustness values.

We now have to choose an appropriate methodology to solve the bicriteria problem. This method should first of all be able to find all potentially efficient schedules, but also be flexible enough to allow management to select the efficient solution they will eventually implement. Let us remark here that the popular method of combining both objectives in a weighted sum will not achieve these goals. First of all there is no interpretation of the sum of (dollar) cost and the robustness measure (cumulative delay in minutes). Secondly, the specification of the weights is a difficult task, in particular because the optimal solution will be very sensitive to slight changes of the weights. Perhaps the most important argument against the method is that it is well known that a certain class of efficient solutions can never be found in discrete problems like the ToD planning problem, see [4].

We decided to use the so called ε-constraint methods [3]. This technique is based on the idea of using one of the objectives as a constraint, specifying an upper bound of ε on its value. It is well known that by varying the value of ε all efficient solutions can be found. Figure 2 shows how different values of ε yield different efficient solutions in a diagram symbolically plotting objective $r^T x$ versus objective $c^T x$.

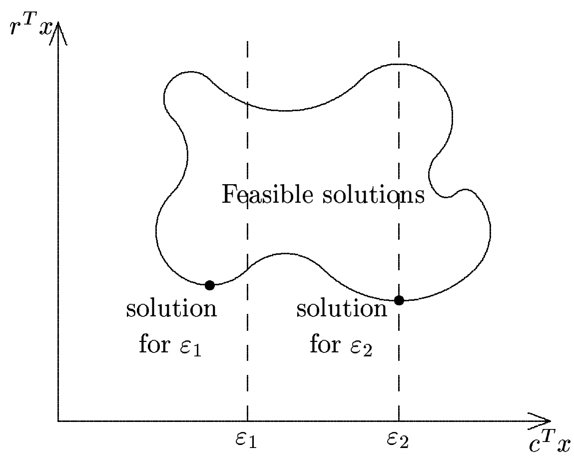

Figure 2: Illustrating the ε-Constraint Method

For the ToD planning problem, we obtain the following reformulation of the bicriteria problem 2SPP.

$$\begin{aligned} \min z^2 &= r^T x \\ \text{subject to } A_1 x &= e \\ A_2 x &= b \\ c^T x &\leq (1 + oc/100)cip \\ x &\in \{0,1\}. \end{aligned} \qquad \varepsilon\text{SPP}$$

In the additional constraint derived from the cost objective function we use an upper bound depending on the optimal objective value *cip* of the integer optimal solution obtained from solving the SPP with the cost objective alone. The parameter *oc* specifies the desired percentage increase in the cost objective management might want to consider. Its usage and therefore the required interaction is straightforward.

4 Results

The model εSPP was implemented in the existing software used by Air New Zealand to solve its ToD planning problems [2]. We show the results obtained from sample runs on the ToD problem for technical crew of the domestic airline, using values of *oc* between 1 and 7 (in steps of 1). For values 1,2, and 3 of *oc* the integer solution could not be found, indicating that the strategies employed currently for minimisation of cost are not appropriate for robustness optimisation. At $oc = 7$ the optimal solution of the pure robustness optimization was obtained, as the cost constraint became inactive. Figure 3 shows the resulting trade off, both for the LP and the IP optimal solutions.

The red line ("LP solution") shows the LP solutions obtained (with some additional values for different *oc* values added in). The form of the curve was obtained because the column generation process did not return existing negative reduced (robustness) cost variables. The column generation process will have to be modified so that promising (robustness) columns are not eliminated in early stages of the dynamic shortest path problem solved. The true efficient frontier obtained from solving LP relaxations of εSPP will have the form indicated by the green line ("Expected"). The pink squares ("IP solution") show the integer solutions obtained for $oc = 4, 5, 6, 100$ and the optimal cost solution. Note that the cost constraint is always active for the LP solutions ($oc < 7$), but usually not for the IP solutions.

Some more information is shown in the following table

The first line in Table 1 shows the optimal solution with the cost objective alone. The second line is the first run with the robustness objective, where the allowable

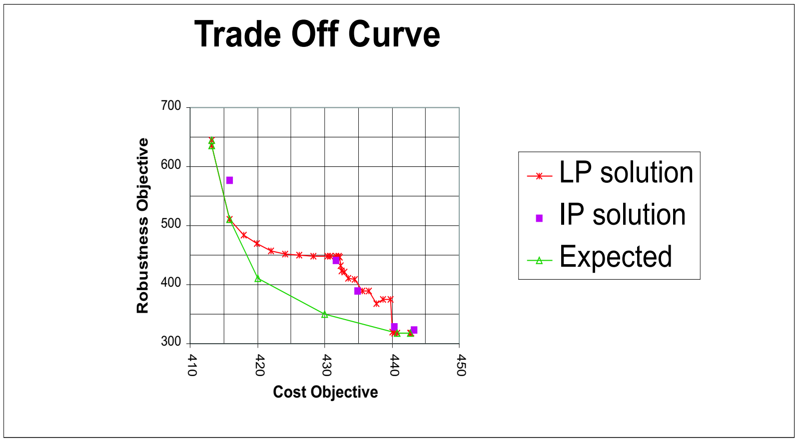

Figure 3: Trade Off Between Cost and Robustness Objective

oc	c(LP)	r(LP)	c(IP)	r(IP)	LP time	IP time
n/a	413.215	645.061	415.830	577.32	277.55	22293.48
0	415.840	510.599			28.8	
1	419.988	469.061			29.17	
2	424.147	451.979			29.64	
3	428.304	448.305			29.93	
4	432.463	422.430	431.624	440.78	33.83	7729.75
5	436.621	389.641	434.96	389.73	51.03	129.87
6	440.780	318.483	440.328	329.25	144.46	12237.92
100	442.762	318.341	443.225	322.58	194.75	4465.91

Table 1: Objective values and computation times

value of the cost objective is set to the optimal value of the optimal cost IP solution. For any lower value, no integer solution will exist when solving the problem with the robustness objective. The last line is the optimal solution obtained with a allowable increase of the cost objective of 100%, i.e. where this constraint is not constraining at all, and the optimal robustness solution is therefore found. Note that in the IP solution, robustness deteriorates, therefore cost improves compared to the LP solution.

To illustrate the difference between the optimal cost and optimal robustness solution, we show a plot of one day in these solutions. As expected the solutions have changed considerably, by including more follow on sector pairs than the original cost optimal solution, more passengering and a different distribution of ToDs between bases.

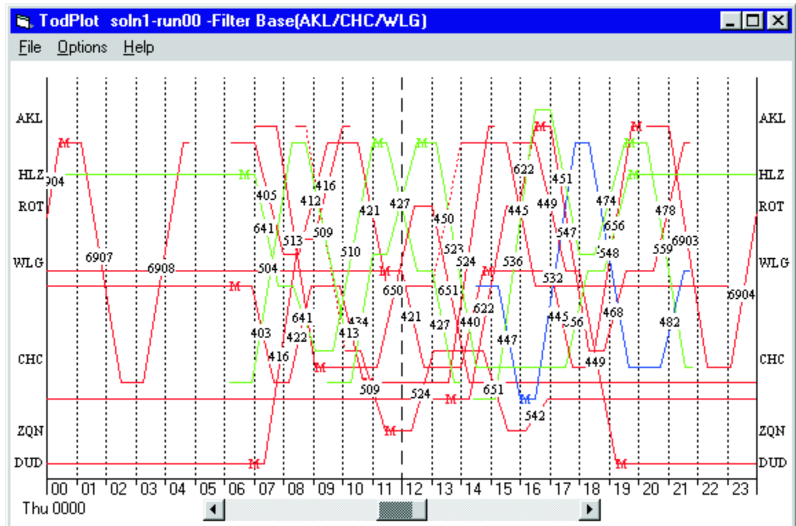

Figure 4: Optimal Solution with Cost Objective

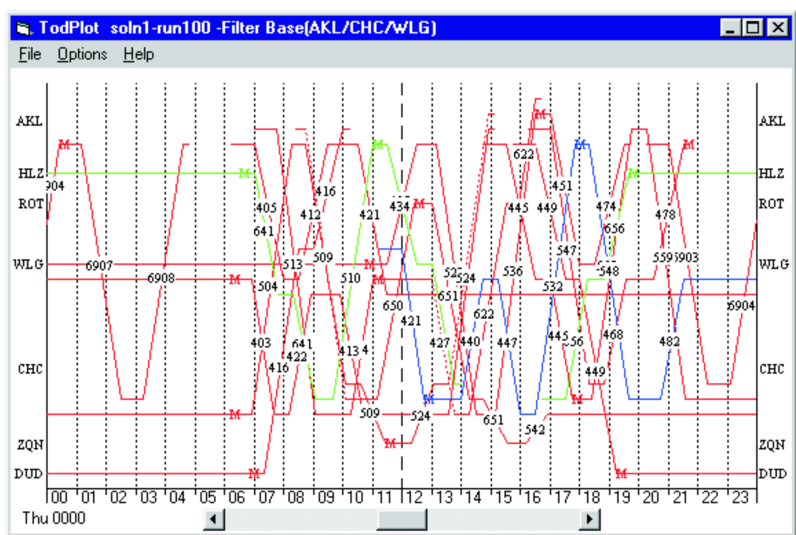

Figure 5: Optimal Solution with Robustness Objective

5 Directions of Future Research

As far as we know the research reported in this paper is the first attempt to solve the ToD planning problem of an airline in a bicriteria context of minimising cost and minimising lack of robustness to obtain exact efficient solutions.

The results are still of a preliminary nature, but clearly show that the goal can be achieved. In the future we will extend our research in several directions. First of all the currently crude and simple measure of robustness can be improved by considering meals, ground time, considering penalties for passengering, etc. Other factors that influence robustness will be considered, too.

As far as the implementation is considered, we observed that the branching rules to select constraint branches in the branch and bound process used so far are not necessarily the best for optimising robustness. We will work on improved branch selection rules.

References

[1] R. Anbil, J. Forrest, and W. Pulleyblank. Column generation and the airline crew pairing problem. *Documenta Mathematica, Extra Volume ICM, III*, pages 677–686, 1998.

[2] E.R. Butchers, P.R. Day, A.P. Goldie, S. Miller, J.A. Meyer, D.M. Ryan, A.C. Scott, and C.A. Wallace. Optimised crew scheduling at air new zealand. *Interfaces*, 2000. to appear.

[3] V. Chankong and Y.Y. Haimes. *Multiobjective Decision Making: Theory and Methodology.* Elsevier Science Pulishing Co., Inc, New York, NY, 1983.

[4] M. Ehrgott and X. Gandibleux. A survey and annotated bibliography of multi-objective combinatorial optimization. *Operations Research Spektrum*, 2000. to appear.