



A decision support model for establishing an air taxi service: a case study

K Fagerholt^{1*}, BA Foss² and OJ Horgen²

¹*Department of Industrial Economics and Technology Management, Norwegian University of Science and Technology, Trondheim, Norway;* and ²*Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway*

Recent availability of relatively cheap small jet aircraft creates opportunities for a new air transport business: Air taxi, an on-demand service in which travellers call in one or a few days in advance to book transportation. In this paper, we present a methodology and simulation study supporting important strategic decisions, like for instance determining the required number of aircraft, for a company planning to establish an air taxi service in Norway. The methodology is based on a module simulating incoming bookings, built around a heuristic for solving the underlying dial-a-flight problem. The heuristic includes a separate method for solving the important subproblem of determining the best distribution of waiting time along a single aircraft schedule. The methodology has proved to provide reliable decision support to the company.

Journal of the Operational Research Society advance online publication, 9 July 2008
doi:10.1057/palgrave.jors.2602635

Keywords: air transport; decision analysis; heuristics; practice of OR; simulation

Introduction

To go from one airport to another, travellers typically have to connect through 'hub' airports. These hubs are often heavily congested and missed flights and delays are common. This causes a lot of extra flying and waiting time for the passengers. As an example, between Trondheim and Kristiansund in Norway, the direct flight time is less than 30 min (see Figure 1). However, this direct flight is not operated by any of the airline companies in Norway. In this case, the passengers have to connect through Oslo to get to Kristiansund. This takes at least five times longer than the direct flight between Trondheim and Kristiansund.

Furthermore, airlines offer scheduled services only between major airports implying that travellers often have to drive significant distances to and from their origins and destinations. Add to this the limited schedules offered by airlines and the additional risk of missed connections and delays, therefore air travel may sometimes be cumbersome. However, new relatively cheap small jet aircraft create opportunities for a new air transport business: Air taxi, an on-demand service in which travellers call one or a few days in advance to book transportation. These aircraft can also operate on small regional airports not serviced by regular airlines. Already, a few examples on such services exist

in the US today, see for instance DayJet (2008) and SATSair (2008).

FlySmart Airtaxi Service (FlySmart) is a company with plans to establish a new air taxi service in Norway. This new business concept resembles an ordinary taxi car service and is based on the idea that business travellers should themselves decide where and when to travel. Initially, FlySmart wants to establish an air taxi service between airports in Southern Norway by using a fleet of light aircraft, each carrying up to four passengers. At many Norwegian airports, light aircraft have check-in and security control through separate desks. This can be utilized to reduce travel time even more for the passengers. The price level is expected to be somewhat above the full fare tickets offered by schedule operated airlines.

Since this is the first time such a business concept is planned in Norway, there are many uncertain factors when establishing such a service. Therefore, FlySmart wanted a methodology for analysing and providing support for strategic decisions. They had already made some preliminary market analysis to obtain an estimate of the total number of passengers. However, there were still some important strategic decisions to analyse, for instance

- the required number of aircraft to handle the estimated number of passengers;
- the trade-off between fleet size and service that could be offered to the passengers and
- the effect of introducing price differentiation according to customers' requirements on flight time flexibility (time windows).

*Correspondence: K Fagerholt, Department of Industrial Economics and Technology Management, Norwegian University of Science and Technology, Trondheim N-7491, Norway.
E-mail: kjetil.fagerholt@iot.ntnu.no

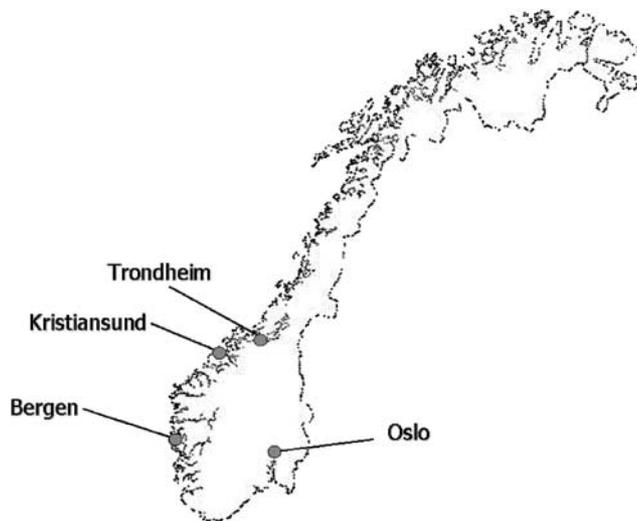


Figure 1 A few airports in Southern Norway.

The aim of this paper is to present a methodology and simulation study supporting the strategic decisions raised above. The methodology is based on a module simulating the incoming bookings for a given day, built around a heuristic for solving the underlying daily routing and scheduling problem, which sometimes is referred to as the dial-a-flight problem (DAFP). Since our methodology and simulation study is based on solving a version of the DAFP, we elaborate upon this in the next section together with a brief literature review on related subjects. In the following two sections, the proposed heuristic solution method for the DAFP is described. This also includes solving a schedule optimization problem for fixed routes, which occurs as a subproblem in our version of the DAFP. We then present a Monte Carlo simulation framework, applying the heuristic for the DAFP, to analyse strategic decisions. Finally, we present the computational study and results, before drawing concluding remarks.

Managing booking requests—the DAFP

Problem description

FlySmart's business concept includes an aircraft fleet located at given home bases in a few airports in Southern Norway. For each aircraft, there will for a given day be a pilot starting around 07.00, and there will be a pilot change around 15.00. The second pilot will then service the aircraft until around 23.00. There is a meal break with a given duration about half way through each shift. All these activities will be defined as *pre-allocated activities*, each with a given time window, in which the activity must start. Pre-allocated activities like start of the day, pilot shift and end of the day are assumed to take no time, but must be performed at the pilots' home airport. The meal breaks are assumed to have a given duration, but

can take place at any airport. A pilot schedule has to satisfy FAA regulations governing flying hours and duty periods, for example, a single pilot cannot fly more than 8 h a day and the duty period cannot extend 14 h. This will be maintained by the proposed two-shift system.

A *booking request* from a traveller will include a pickup and a delivery airport, with a time window, in which the flight must start. Here, in contrast to the problem described in Cordeau *et al* (2007), we assume that only direct flights are allowed, since this is the policy that FlySmart initially wants to introduce. A booking request can contain one to four passengers since the aircraft that will be used only have four passenger seats. The problem of managing booking requests is dynamic in the sense that new requests occur in real time. With each new request, the current schedule must be reconfigured to adjust for the new request as well as those already accepted. When a new request arrives, one has to check if there is capacity to handle it within the desired time window. To do this, one must also consider the existing customer requests which have already been accepted. When a booking request is confirmed, two alternative *booking confirmation policies* are available:

1. Confirm a given time (within the time window) for the pickup.
2. Confirm that the customer will be picked up within the time window, but that final confirmation on pickup time will be given at a later stage (eg the night before departure).

Confirmation *policy 1* restricts the planning flexibility more than *policy 2*. However, the customers may perceive this as better service giving better predictability. For most studies performed in this paper, we have assumed that policy 1 is chosen, though it should be emphasized that the solution method presented in the following sections also handles policy 2. Assuming policy 1 also means that the daily planning problem of managing booking requests becomes more restricted (we cannot change pickup times of already accepted requests). The process of managing incoming requests, following policy 1, is illustrated in Figure 2.

The problem of managing incoming booking requests corresponds to a routing and scheduling problem of assigning requests to the available aircraft and to simultaneously decide the aircraft routes and schedules. This problem is, as already mentioned, sometimes referred to as the DAFP. The objective is to handle all requests, or as many as possible, while minimizing cost. In dynamic routing and scheduling problems, like the problem presented herein, an important question is to decide how to distribute an aircraft's idle time in order to achieve a good solution. Should an aircraft delay as much as possible its departure from its current location, should it move as soon as it can to its next location, or choose some intermediate *waiting strategy*? This part of the problem will be elaborated upon in a later section.

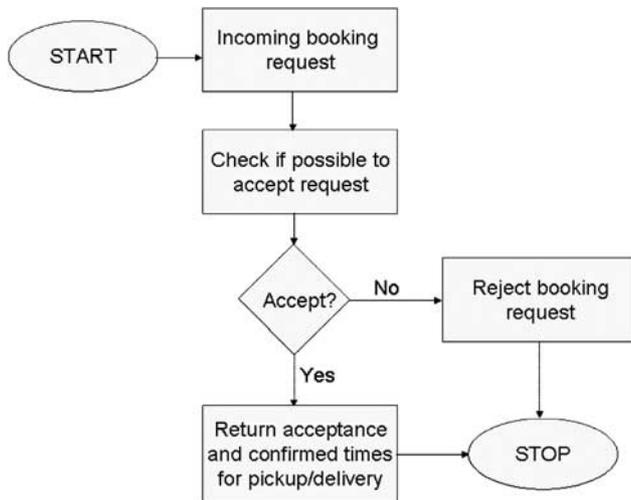


Figure 2 Booking process—confirmation policy 1.

Relevant literature

There is an abundance of research and commercial contributions on various Operational Research (OR) studies for the aviation industry. In a review paper by Barnhart *et al* (2003), an overview of several important OR applications for schedule operated airlines is given, like for instance schedule design, fleet assignment, aircraft maintenance routing, and crew scheduling. Yao *et al* (2008) present a methodology for scheduling the available resources of a fractional aircraft ownership company, where several partial owners of an aircraft are entitled to certain flight hours per year. However, to the authors' knowledge there are only few contributions to the OR literature on air taxi services. Cordeau *et al* (2007) describe what is referred to as the DAFP. The DAFP corresponds to taking transportation requests for a given day and schedule aircraft (and pilots) in a cost-effective way to satisfy these requests. Espinoza *et al* (2008a) presents an integer multi-commodity network flow model with side constraints for the DAFP. Techniques to control the size of the network and to strengthen the linear programming relaxation are developed, which allows the solution of small to medium size instances by commercial solvers. Espinoza *et al* (2008b) demonstrate that high-quality solutions for large-scale instances can be produced by embedding the core optimization technology in a local search scheme. However, there are a few differences between the DAFP considered here and the one considered by Espinoza *et al* (2008a, b). The latter assumes booking confirmation policy 2 described in the previous section, while in this paper we assume that policy 1 will be used (except for one test). Policy 1 also means that the customer must receive immediate feedback regarding the pickup time. To achieve this, a very fast heuristic will be appropriate for solving the DAFP. Another difference between the DAFP here and by Espinoza *et al* (2008a, b), is

that we only allow direct flights for the customers (since this is a company policy).

The DAFP is a version of the Pickup and Delivery Problem with Time Windows (PDPTW), a routing problem that has received much attention in the literature, see for instance Dumas *et al* (1991) and Desrosiers *et al* (1995). In the classical PDPTW, there are a given number of requests, each consisting of a specified quantity to be transported between given a pickup and delivery node. There is a time window attached to each node, in which pickup and/or delivery must start. The PDPTW usually has no pre-allocated activities as in FlySmart's booking problem. Much of the literature on the PDPTW is limited to the dial-a-ride problem (DARP), which is similar to the DAFP. However, all applications of DARPs are on land transport. In the DARP, the requests consist of people who want to be picked up at a given pickup point and delivered at a given delivery point. Time windows or desired pickup and/or delivery times are also usually imposed in the DARP. Typical applications for the DARP are routing and scheduling of taxis and transportation of elderly and/or handicapped persons, see for instance Cordeau and Laporte (2003a), Toth and Vigo (1997), Ioachim *et al* (1995) and Madsen *et al* (1995). A recent survey presenting variants of the DARP and solution methods can be found in Cordeau and Laporte (2003b). While the objective function in PDPTW usually is to minimize cost, the DARP's objective function is often much more compound. For instance, in cases where customers have a desired time for pickup and/or delivery it is usual to also take customer inconvenience into account, see for instance Sexton and Bodin (1985), Sexton and Choi (1986) and Dumas *et al* (1990).

Our version of the DAFP is a dynamic routing and scheduling problem, where the planning must be performed before all booking requests are known. Hence, the solution quality will be affected by how the waiting time is distributed along the aircraft routes. Mitrović-Minić and Laporte (2004) define four alternative waiting strategies for the dynamic PDPTW. A survey on dynamic routing can be found in Gendreau and Potvin (1998).

A heuristic for solving the DAFP

In order to immediately return a pickup time for each request, we propose a fast heuristic for solving the underlying DAFP described previously. The heuristic consists of an insertion method combined with a local search improvement phase.

Let b_i be the new incoming booking request. b_i consists of two nodes i and $i+N$, which represent the pickup and delivery nodes, respectively. There is a time window associated with the pickup node i , in which the flight must start. Let B_A be the set of the already accepted requests (including pre-allocated activities), while K is the number of aircraft in the fleet considered. Now, the heuristic can be described by the pseudo-code given below (applicable both for confirmation policy 1 and 2).

```

k = 1;
while k ≤ K do
begin
  apply insertion heuristic to find best insertion of  $b_i$ 
  in schedule of aircraft k;
  update best insertion so far;
  k = k + 1;
end;
apply local search heuristic to improve existing plan (and to
see if it is possible to insert  $b_i$  if this was not achieved by the
insertion heuristic, see Figure 3);
if feasible insertion of  $b_i$  exists then
begin
   $B_A = B_A U \{b_i\}$ ;
  accept request  $b_i$  (including associated pickup time
  if policy 1 is applied);
end
else
begin
  reject request  $b_i$ ;
end
end

```

Initially, before any requests have been inserted into an aircraft schedule, B_A only consists of the pre-allocated activities with their associated time windows. Then, we go through an insertion heuristic similar to the one Madsen *et al* (1995) proposed for the DARP. For each aircraft k , we determine the best insertion of b_i into the aircraft schedule. Here, we use a compound objective based on real flight cost and a measure estimating the probability of being able to accept future booking requests (if policy 1 is applied). The latter will depend on how waiting time is distributed along a given aircraft schedule, and will be elaborated upon in the next

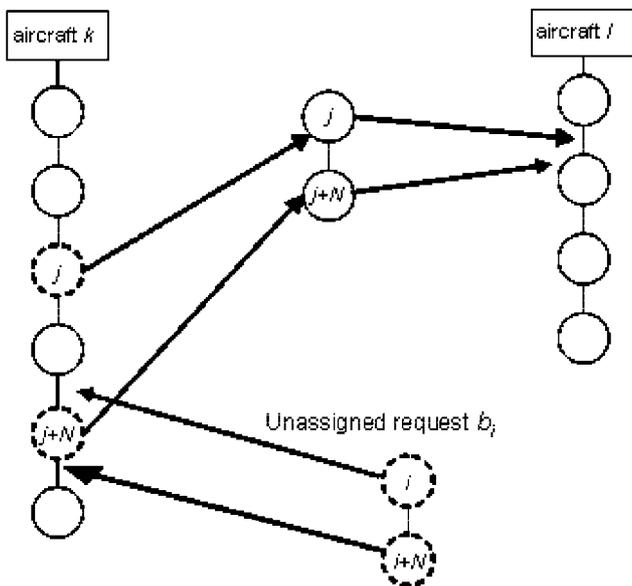


Figure 3 The reassign local search operator.

section. Since we only consider direct flights, we only need to consider subsequent insertions of the pickup and delivery nodes (though an insertion may split pickup and delivery nodes for already accepted requests in the schedule if the corresponding travel equals the travel for the new booking b_i). When testing alternative insertions, we have to maintain feasibility regarding aircraft capacity (number of passengers). We must also ensure feasibility for the time window for the new incoming request, as well as for the already accepted requests including the pre-allocated activities. In the case of confirmation policy 1, the time windows for the already accepted requests will reduce to a given pickup time (except for the pre-allocated activities).

Following the insertion heuristics, we apply a simple, but fast, local search heuristic based on the *reassign* operator, see Figure 3. Any given request $b_j \in B_A$, consisting of the nodes j and $j + N$, is removed from the schedule of aircraft k . Then the best feasible insertion into each of the other aircraft is found. Request b_j is then reassigned to the aircraft that gives the best feasible insertion—aircraft l in Figure 3 (given that this yields an improved solution). Here, we use the same compound objective as described above. If the insertion heuristic was not able to find a feasible insertion for the new request b_i (with nodes i and $i + N$), then this will be an unassigned request. The local search heuristic will now also try to insert b_i into the schedule of the aircraft that we just removed a request from (aircraft k in Figure 3). This will increase the possibility of finding a feasible insertion for b_i .

The heuristic can easily be adapted also for reoptimizing the whole plan, for instance the night before the departure when all requests are known.

Scheduling for a given sequence—waiting strategies

When receiving a new booking request, following confirmation policy 1, we immediately have to return the confirmed pickup time to the customer (if not rejected). This may affect the possibilities of accepting future requests. Assume there are several alternative feasible insertions of a new request. In order to choose the best one, we have to consider the probability of being able to accept requests that will arrive in the future. This will depend on how the waiting time along a given aircraft schedule is distributed, see also the discussion on alternative waiting strategies by Mitrović-Minić and Laporte (2004). The following subsection provides a formal description of the problem of determining the optimal single aircraft schedule (ie the optimal distribution of waiting time). This appears as an important subproblem for the heuristic described in the previous section. Then, a solution method for the problem is presented. It should be emphasized that the waiting time distribution problem only is of interest for confirmation policy 1, since policy 2 allows the pickup times to be changed later within the time windows.

Single aircraft schedule optimization problem

In the following we will present a mathematical model for the single aircraft schedule optimization problem to find the best distribution of waiting time along the schedule. Let $V = \{0, 1, 2, \dots, m\}$ be the set of nodes in a given aircraft sequence $0-1-2-\dots-m$. Define $T_{i,i+1}$ to be the required time to travel between the nodes i and $i+1$ in the sequence, $i \in V \setminus \{m\}$. We assume that $T_{i,i+1}$ includes the service time at i . For each node $i \in V$, a time window $[A_i, B_i]$ is defined. The variable w_i , $i \in V$, represents the waiting time at i before service begins, while t_i , $i \in V$, represents the time for start of service (pickup of the associated customer). Let us further define \hat{P} as an estimate of the expected profit based on the probability of being able to accept additional future booking requests. This profit will be a function of how the waiting time is distributed along the schedule. The calculation of \hat{P} will be discussed in more detail below.

The problem of optimizing the schedule for a given sequence can now be formulated as:

$$\max \hat{P} = \sum_{i \in V} \hat{P}_i(w_i) \quad (1)$$

$$t_{i+1} - w_{i+1} - t_i - T_{i,i+1} = 0 \quad \forall i \in V \setminus \{m\} \quad (2)$$

$$A_i \leq t_i \leq B_i \quad \forall i \in V \quad (3)$$

$$w_i \geq 0 \quad \forall i \in V \quad (4)$$

The objective function (1) maximizes the sum of the expected profits from utilizing the waiting times along the sequence to accept future booking requests. For a given sequence, the flight cost is constant as it does not depend on the timing of stops, and hence not the distribution of waiting time along the sequence. Therefore, we have not included the flight cost in (1). However, it should be emphasized that the evaluation criteria used by the insertion and local search heuristic is a compound objective consisting of a weighted sum of costs and the objective in (1). The waiting times and the times for start of service are calculated by constraints (2), while (3) ensure that service starts within the specified time window. This time window will reduce to a single value for nodes representing pickups of already confirmed bookings. Equation (4) are non-negativity constraints for the waiting variables. We should also remark that waiting is only allowed before the nodes in the sequence that represent pickups and not the delivery nodes. This can easily be handled by forcing the corresponding waiting time variables to zero (or by omitting them from the variable definition).

The objective function (1) requires further explanation. $\hat{P}_i(w_i)$ expresses the expected value of utilizing the waiting time at node i , w_i , to insert a future request into the aircraft schedule. Let us define p_i as the probability per time unit a new request with pickup at the airport corresponding to node i will arrive. $p_{i,i+1}$ is the probability per time unit that a new

request with pickup at the airport of i and delivery at the airport of $i+1$ will arrive. To simplify the presentation, we have assumed that the probabilities p_i and $p_{i,i+1}$ do not vary over time. \bar{t} is the average flight time (including service or boarding time) between any two of the actual airports operated. I is defined as the estimated gross margin for any new request that may arrive.

Now, let us further define $D_{i,i+1}$ as the flight distance between the airports corresponding to the nodes i and $i+1$ in the sequence. Then, we will have two situations to consider: (a) $D_{i,i+1} = 0$ (ie i and $i+1$ represent the same airport) and (b) $D_{i,i+1} \neq 0$ (ie i and $i+1$ represent different airports). Now, we can give the following mathematical expressions for $\hat{P}_i(w_i)$ for situations (a) and (b):

$$(a) D_{i,i+1} = 0 \Rightarrow \hat{P}_i(w_i) = I \max\{p_i(w_i - 2\bar{t}), 0\} \quad (5)$$

$$(b) D_{i,i+1} \neq 0 \Rightarrow \hat{P}_i(w_i) = I \max\{p_i(w_i - 2\bar{t}), p_{i,i+1}w_i\} \quad (6)$$

In (5) for situation (a), we assume that when the waiting time, w_i , is less than twice the average flight time, \bar{t} , we will not be able to utilize the waiting time (since we have to return for the next customer pickup). However, when the waiting time exceeds $2\bar{t}$, there is a chance that a future request from i can be accepted. Using the average flight time, \bar{t} , for calculating the expected profit is a simplification, since the real flight times between any two airports may differ from this. In (6) for situation (b), we also introduce the expected profit based on the probability that a new booking directly between i and $i+1$ will appear.

Figure 4 illustrates the expected profit function $\hat{P}_i(w_i)$ for situations (a) and (b). In situation (b) the bold line represents the expected profit function.

Solution of schedule optimization problem

The single aircraft schedule optimization problem (1)–(4) is an important subproblem that must be solved for each insertion or local search move that is tested in the heuristic. In order to return with an immediate response by the heuristic, this subproblem must also be solved swiftly. Therefore, to solve the problem efficiently, we have used ideas from Fagerholt (2001). Each node in the sequence is duplicated into all possible times for start of service at that node. By discretizing time into steps of a given time unit, the number of possible start times can be reduced and the problem can be solved as a shortest path problem (SPP) on a defined network (this method also allows for any type of non-linear profit function in (1)).

Figure 5 illustrates the principles for generating the shortest path network based on a small example sequence 0–1–2–3–4. Node 0 represents the start node. In Figure 5, we have discretized time into steps of 15 min. The time in the nodes represent arrival times. To simplify the presentation, nodes for the pre-allocated activities other than the start node are not included in Figure 5. We have also assumed that the flight times between any of the airports in this example are

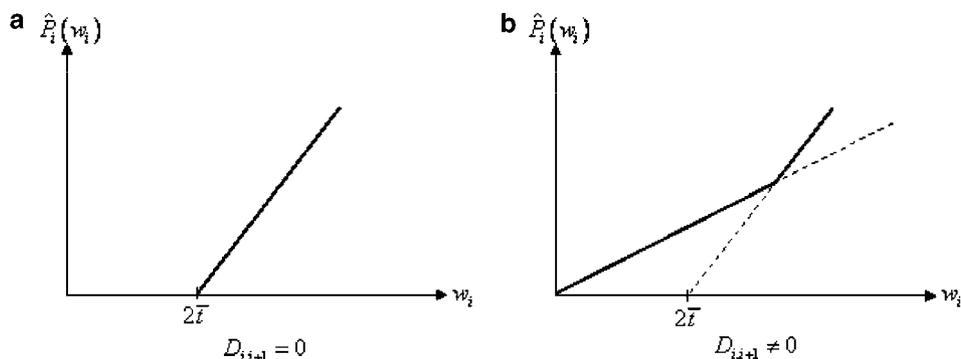


Figure 4 Expected profit function $\hat{P}_i(w_i)$: (a) same airport and (b) different airports.

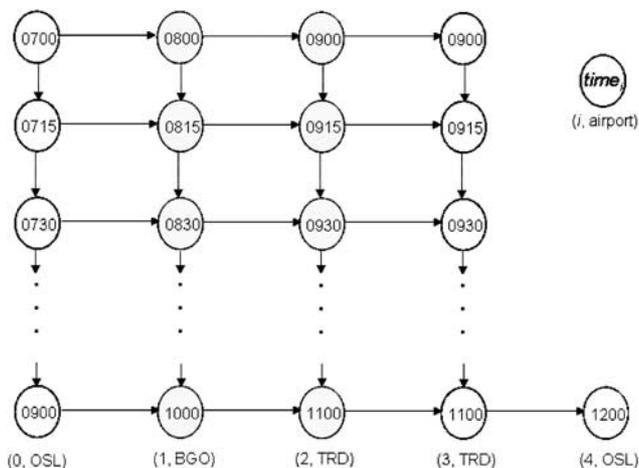


Figure 5 Generation of shortest path network.

1 hour (just to simplify the example in Figure 5). Nodes 1 and 2 represent the pickup and delivery for the new request, respectively. Node 1 still has a time window (width of 2h). The start node 0 has a time window derived by the first pickup node and the earliest allowed start time (0700). Node 3 has a confirmed pickup time (1100).

The horizontal arcs represent the transition from one activity to another (eg pickup to delivery), while the vertical arcs in the network will represent waiting times. The expected profit function $\hat{P}_i(w_i)$ will be calculated for the arcs in the network, as shown previously. The total expected profit for the sequence will here depend on the path selected. Following the upper path in Figure 5 implies that all waiting time (2h) will take place in node 3 (TRD), while the lower path aggregates all waiting in node 0 (in OSL). The paths in the middle will result in some intermediate waiting strategy. The optimal path depends on the probabilities that a new booking request will arrive at the different airports. The optimization problem (1)–(4) can now easily be solved as a SPP by dynamic programming on this network, as illustrated in Figure 5.

Design of the simulation study

We have performed a simulation study to support the strategic decisions mentioned in the introduction, such as optimal/required fleet size and price differentiation strategies regarding width of time windows. We have designed a simulation methodology built around the heuristic for the DAFP described in the previous two sections. The methodology is programmed in Java.

Since this is the first attempt to establish an air taxi service in Norway, there exist no good data on the demand for such services. FlySmart has, however, done some preliminary market analysis, and based on this they have stated a goal on how many people to service. To generate test cases for the simulation study, we have used aviation statistics, available from Avinor (2008), on the number of daily passengers between any pair of 12 selected airports in Southern Norway. This has been combined with FlySmart’s own estimates. The distances between the airports, again found from aviation statistics, made the basis for calculating the flight times.

We generated 100 instances for each analysed scenario. Each instance consisted typically of approximately 200 daily bookings and each booking consisted of 1–4 passengers (uniformly distributed). All bookings were assumed to have a given time window regarding pickup and delivery times.

In most runs we used a fleet consisting of 10 aircraft. Four of these had their home base in Oslo, three in Bergen and three in Trondheim (see Figure 1). Each aircraft schedule had five pre-allocated activities on a given day; start (no earlier than 07.00), meal break (between 10.00 and 13.00), change of pilot (between 14.00 and 16.00), meal break (between 18.00 and 21.00) and end of day (no later than 23.30).

In each of the 100 instances, we used a Monte Carlo simulation procedure to simulate incoming booking requests as realistically as possible, following the processes illustrated by Figure 2. The requests were randomly selected one by one from the complete set of requests. We used the heuristic algorithm for the DAFP, and tried to insert each request in the schedule for one of the aircraft in the fleet. However, for some of the requests, there were no feasible

insertion in any aircraft schedule, hence these requests were rejected.

Computational results

In this section, we present some of the simulation studies performed. The first subsection presents a study testing alternative aircraft fleet sizes. In the following subsection, the effect of flexibility in customer time windows is evaluated, while we thereafter discuss the effect of alternative booking confirmation policies. In the first two studies confirmation policy 1 is used. In the latter two studies, we have used a fixed fleet with 10 aircraft. In all studies we have used a time discretization of 10 min for solving the single aircraft schedule problem. The CPU-time to solve an instance with 10 aircraft and 200 received booking requests is about 1 min with a Pentium 4 2.8 GHz processor. Hence, solving a complete scenario consisting of 100 instances then takes 1–2 h. The time for handling one booking request is less than 1 s.

In each of the studies, we present figures of *Accepted bookings*, *Time utilization*, *Distance utilization* and *Travellers per flight* as a function of received booking requests. Accepted bookings are the number of bookings we were able to accept or insert into any aircraft schedule. Time utilization is the total time minus the time spent on waiting or empty flights divided by the total time. Distance utilization is the flight distance with travellers onboard divided by the total flight distance for all aircraft. Travellers per flight is the average number of travellers onboard the aircraft per flight.

Fleet size

FlySmart wanted to estimate the number of aircraft required to handle approximately 100 daily bookings. Therefore, in this study we test four alternative scenarios with a varying number of aircraft, that is, 5, 10, 15 and 20.

Figure 6 shows the results from these simulations. We observe that 10 aircraft are too low to handle 100 daily bookings without having to reject a large percentage of the received bookings. By using 15 aircraft, the results show that on average it is possible to accept 98.9 of 100 received bookings.

We also note that a small fleet with only five aircraft will have a low time and distance utilization. With such a small fleet, more empty flights have to be made since available aircraft always will be spread in a more restricted geographical area. Further, it should be noted that the utilization does not vary much for a fleet of 10 aircraft or more.

Flexibility in customer time windows

In this study we evaluate the effect of varying the time window width. Here we assume that all booking requests have the same time window width. We have tested four different widths of time windows: 2, 3, 4 and 5 h, respectively. Figure 7 presents the results from this study.

Let us study the behaviour after 100 received booking requests in Figure 7 and compare the time window widths of 2 and 3 h. We observe that increasing the width from 2 to 3 h results in 7.3 more accepted bookings in average, an increase

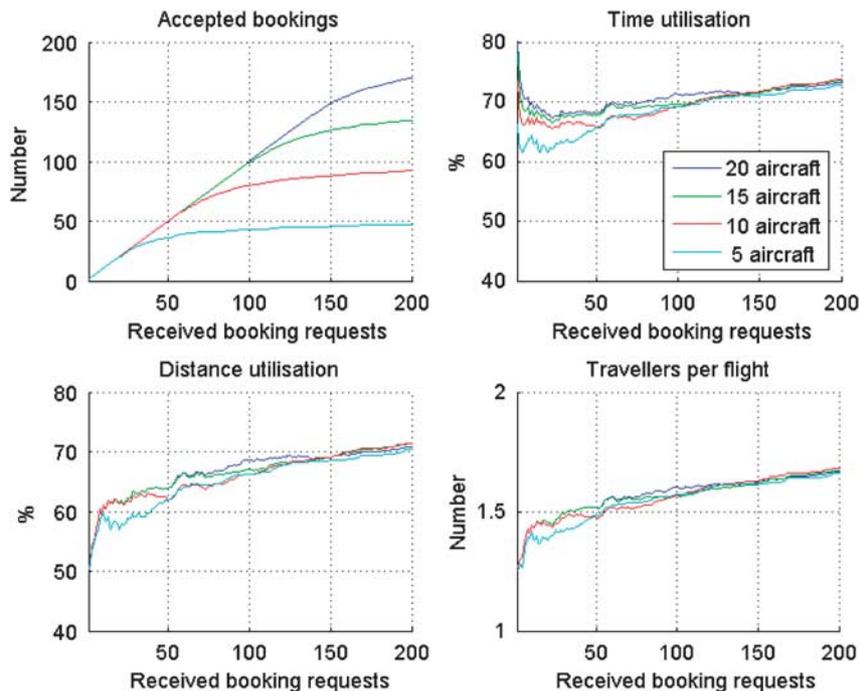


Figure 6 Comparison of alternative fleet sizes.

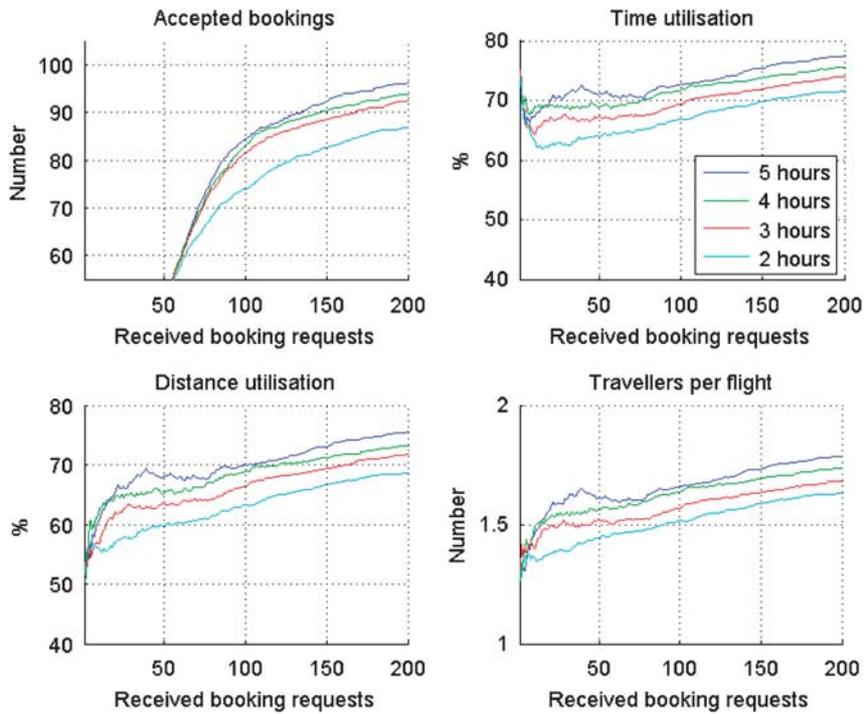


Figure 7 Comparison of various time window widths.

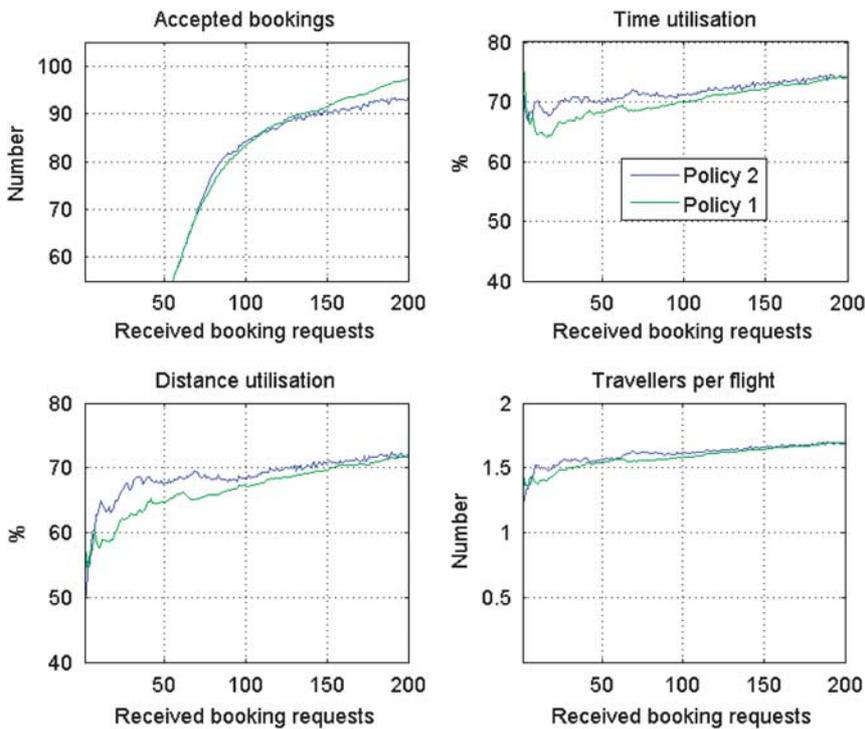


Figure 8 Comparison of alternative confirmation policies.

of approximately 10%. If we increase the time window width further, there are only minor improvements in the solutions. These results indicate that FlySmart should consider

differentiating the prices according to time window width, for instance by giving a price discount when the customer gives a time window width of 3 h or more.

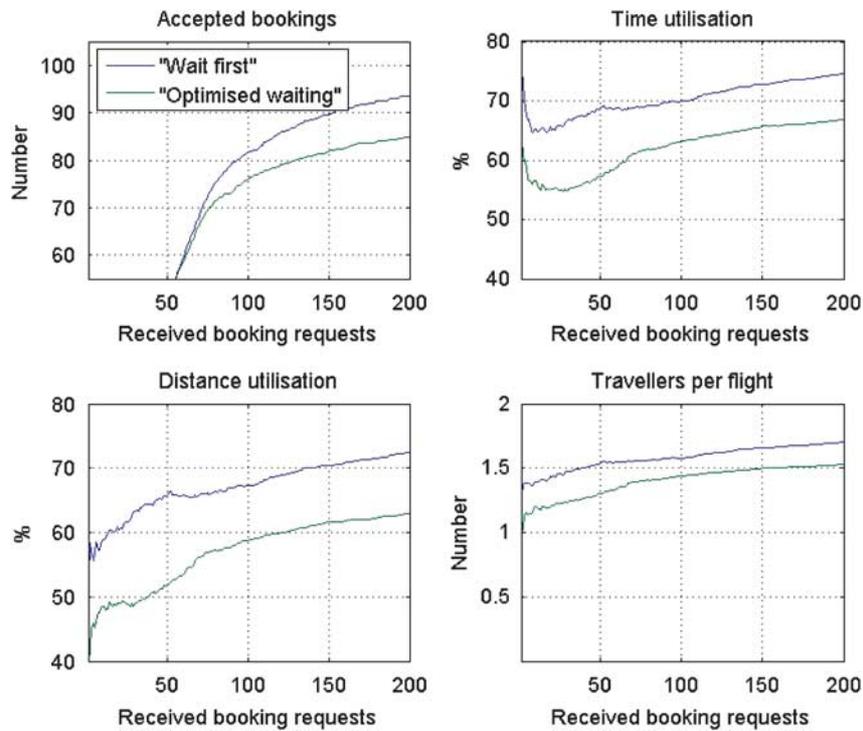


Figure 9 Comparison of waiting strategies.

Confirmation booking policies and waiting strategies

Earlier we discussed two alternative policies on how to confirm booking requests:

1. Confirm a given time (within the time window) for pickup and delivery.
2. Confirm that the customer will be picked up within the time window, but that final confirmation on pickup time will be given at a later stage (eg the night before departure).

In all simulation studies presented so far, we have applied booking confirmation policy 1. However, policy 2 is less restrictive, since we postpone confirming pickup and delivery times, and is therefore expected to give better fleet utilization.

The effect of alternative confirmation policies is studied in Figure 8, where we can observe that the time and distance utilization is, as expected, slightly improved by policy 2. However, regarding the number of accepted bookings, there is not much difference between the two policies. As a matter of fact, when the number of received bookings becomes large (more than 120), confirmation policy 2 performs worse than policy 1. One possible explanation for this is that in the second scenario (policy 2), we have at earlier stages during a simulation been able to accept more of the ‘difficult customers’ that may impose tighter restrictions later during the simulation period.

The relatively small impact that confirmation policies have on the solution quality indicates that FlySmart’s choice of

selecting policy 1 seems to be reasonable. This is also due to the fact that this policy will be perceived as a better customer service and hence more profitable in the long run.

The results summarized in Figure 8 also indicate that the way the distribution of waiting times is handled gives good results. We have also tested this strategy against a ‘wait first-strategy’, where all repositioning flights are performed as late as possible, while all flights with paying travellers are started as soon as they can. The method applied here gives much better results, see the results summarized in Figure 9. Here, the results referred to as ‘Optimized waiting’ are achieved by following the procedure described in the section ‘Scheduling for a given sequence—waiting strategies’. At 100 received booking requests, the results show approximately 10% increase in the number of accepted bookings, as well as a significant increase in both time and distance utilization compared with the simple ‘wait first-strategy’. We also observe more travellers per flight on average.

Summary and concluding remarks

Recent availability of relatively cheap small jet aircraft creates opportunities for a new air transport business: Air taxi, an on-demand service in which travellers call in one or a few days in advance to book transportation. FlySmart is a company planning to establish such a new air taxi service in Norway. In this paper we have presented a methodology and simulation study supporting strategic decisions like fleet size, price

differentiation strategies and booking confirmation policies. The methodology is based on a module simulating the incoming requests for a given day, built around a heuristic for solving the underlying DAFP. The heuristic includes a separate method for solving the subproblem of determining the best distribution of waiting time along a single aircraft schedule.

The simulation study has proved to provide important decision support for FlySmart. For instance, the study showed that the required fleet to handle approximately 100 booking requests, with satisfactorily customer service, is close to 15. This is higher than what was estimated in the first place. The study also showed that confirming the pickup times immediately to the customers can give as good solutions as postponing the confirmation of these times to the night before. One of the main reasons is probably that the proposed method for distributing waiting times along an aircraft schedule seems to give good results. Since this confirmation policy is perceived as a better customer service, these results indicate that FlySmart's choice of policy seems reasonable.

The methodology presented here can also be used for testing alternative scenarios, other than the ones described here, such as alternative depot locations for aircraft and allowing intermediate stops *versus* price reduction (the heuristic can easily be extended to handle this).

When FlySmart's concept for an air taxi service becomes a reality, it will also be useful to improve or extend the heuristics for solving the DAFP as well as to test alternative heuristics. As these new air taxi services are emerging, this will probably become an interesting area of research in the future.

Acknowledgements—We are grateful to Mr Tomas Adrian at FlySmart Airtaxi Service for giving us access to required data and for being enthusiastic about the project. We also appreciate the comments from the referees, which have improved the paper.

References

- Avinor (2008). <http://www.avinor.no>, accessed 25 March 2008.
- Barnhart C, Belobaba P and Odoni AR (2003). Applications of operations research in the air transport industry. *Transp Sci* **37**: 368–391.
- Cordeau J-F and Laporte G (2003a). A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transp Res B* **37**: 579–594.
- Cordeau J-F and Laporte G (2003b). The dial-a-ride problem (DARP): Variants, modelling issues and algorithms. *4OR-Quart J Belgian French and Italian Opns Res Soc* **1**: 89–101.
- Cordeau J-F, Laporte G, Potvin J-Y and Savelsbergh MVP (2007). Transportation on demand. In: Barnhart C and Laporte G (eds). *Transportation*. Handbooks in Operations Research and Management Science, vol. 14. North-Holland: Amsterdam, pp 429–466.
- DayJet Corporation (2008). <http://www.dayjet.com>, accessed 25 March 2008.
- Desrosiers J, Dumas Y, Solomon MM and Soumis F (1995). Time constrained routing and scheduling. In: Ball MO, Magnanti TL, Monma CL and Nemhauser GL (eds). *Network Routing*. Handbooks in Operations Research and Management Science, vol. 8. North-Holland: Amsterdam, pp 35–139.
- Dumas Y, Soumis F and Desrosiers J (1990). Optimizing the schedule for a fixed vehicle path with convex inconvenience costs. *Transp Sci* **24**: 145–152.
- Dumas Y, Desrosiers J and Soumis F (1991). The pickup and delivery problem with time windows. *Eur J Opl Res* **54**: 7–22.
- Espinoza D, Garcia R, Goycoolea M, Nemhauser GL, Savelsbergh MVP (2008a). Per-seat, on-demand air transportation. Part I: Problem description and an integer multi-commodity flow model. *Transp Sci* doi:10.1287/trsc.1070.0227.
- Espinoza D, Garcia R, Goycoolea M, Nemhauser GL, Savelsbergh MVP (2008b). Per-seat, on-demand air transportation. Part II: Parallel local search. *Transp Sci* doi:10.1287/trsc.1070.0228.
- Fagerholt K (2001). Ship scheduling with soft time windows—An optimisation based approach. *Eur J Opl Res* **131**: 559–571.
- Gendreau M and Potvin J-Y (1998). Dynamic vehicle routing and dispatching. In: Crainic TG and Laporte G (eds). *Fleet Management and Logistics*. Kluwer: Boston. pp 115–126.
- Ioachim I, Desrosiers J, Dumas Y, Solomon MM and Villeneuve D (1995). A request clustering algorithm for door-to-door handicapped transportation. *Transp Sci* **29**: 63–78.
- Madsen OBG, Ravn HF and Rygaard JM (1995). A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities and multiple objectives. *Ann Opns Res* **60**: 193–208.
- Mitrović-Minić S and Laporte G (2004). Waiting strategies for the dynamic pickup and delivery problem with time windows. *Transp Res B* **38**: 635–655.
- SATSair (2008). <http://www.satsair.com>, accessed 25 March 2008.
- Sexton TR and Bodin LD (1985). Optimizing single vehicle many-to-many operations with desired delivery times: I. Scheduling. *Transp Sci* **19**: 378–410.
- Sexton TR and Choi Y (1986). Pickup and delivery of partial loads with time windows. *Am J Math Mngt Sci* **6**: 369–398.
- Toth P and Vigo D (1997). Heuristic algorithms for the handicapped persons transportation problem. *Transp Sci* **31**: 60–71.
- Yao Y, Ergun Ö, Johnson E, Schultz W and Singleton SM (2008). Strategic planning in fractional aircraft ownership programs. *Eur J Opl Res* **189**: 526–539.

*Received December 2006;
accepted March 2008 after two revisions*