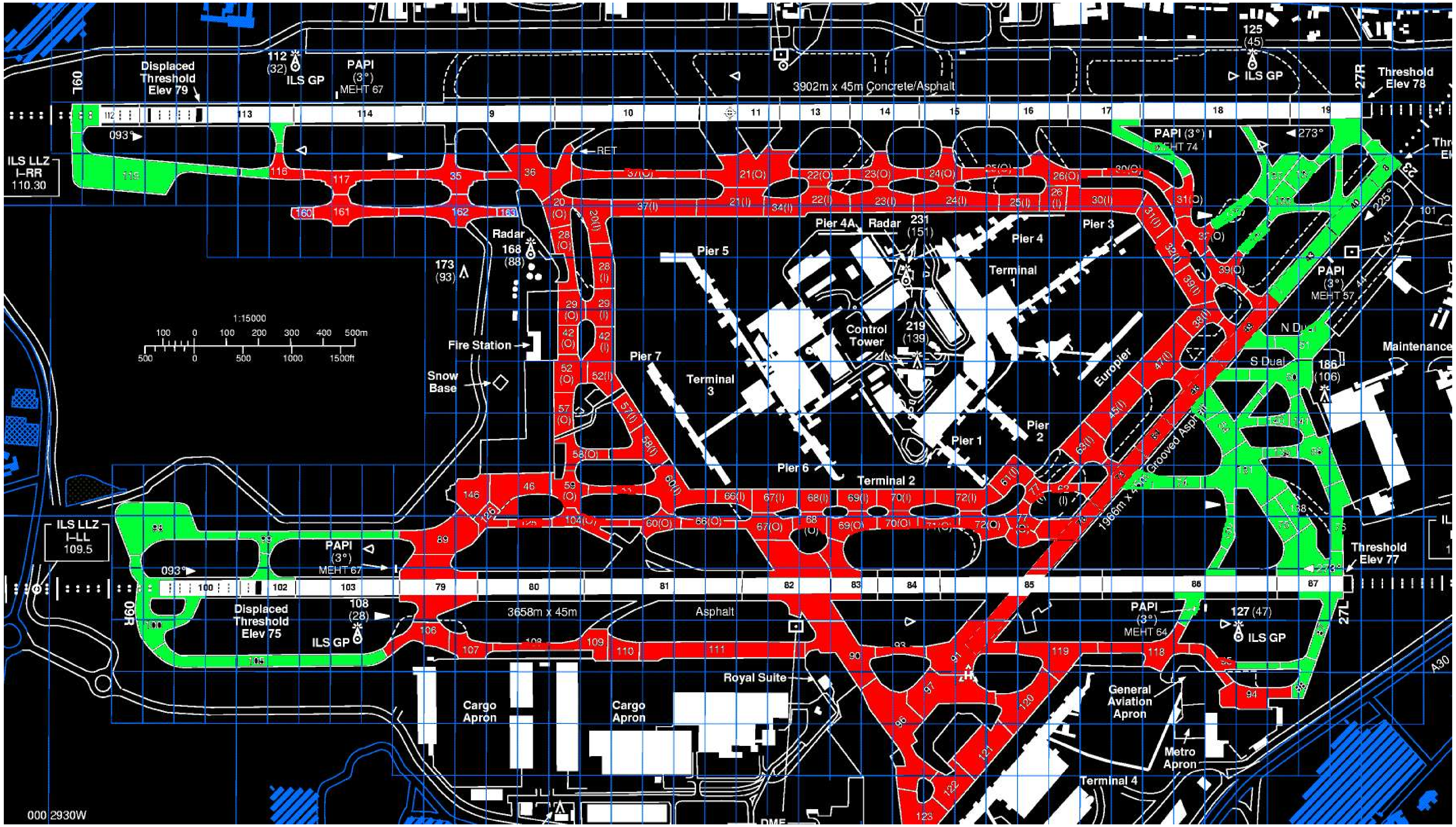
Hybrid Meta-heuristics For Departure Runway Scheduling At London Heathrow Airport

Jason Atkin & Edmund Burke (University of Nottingham)

John Greenwood & Dale Reeson (National Air Traffic Services Ltd)

Melvin Brown (Smith Institute for Industrial Mathematics and System Engineering)

Aim: To reduce delay and improve throughput of the departure runway while meeting all real-world constraints.



Departure System

Aircraft taxi from the stands, by the terminals, along the taxiways, to the holding points by the runways before entering the runways.

Left: Map of London Heathrow airport showing the taxi paths for aircraft.

Key:

White Lines: Runways

White: Terminal buildings

Red: Taxiways

Green: Holding points

Terminals 1, 2 and 3 are between the runways.

Terminal 4 is near the lower right corner.

The departure system

• Only one runway is used for departures at any time. There is an alternation schedule to provide relief from noise for the residences on the flight paths.

• The direction of use of the runway will depend upon the direction of the wind as it is best for an aircraft to take off into a headwind.

• Aircraft taxi from their stand, around the taxiways, to the holding point at the end of the runway that is currently being used for departures.

• The runway controller re-orders the aircraft within the holding point.

• The holding point configuration limits the reordering that is possible.

• The four runway ends each have different holding point configurations so different re-ordering is possible at each.

Separations at take-off

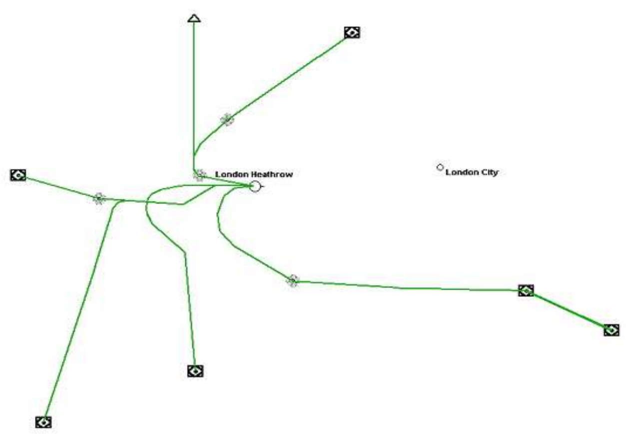
• A delay needs to be imposed on aircraft to maintain a separation from previous departures.

• The separation required depends upon the aircraft characteristics.

• Re-ordering the aircraft can change the separations and affect the throughput.

• Separations depend upon weight category, departure route and speed group.

• By knowing all three, a minimum separation time can be determined.



Departure Flight Paths - Westerly

• Aircraft need to maintain separation distances when in flight. If aircraft leave on similar departure paths they need a greater time separation at take-off.

• Some pairs of routes need an even larger separation to account for congested airspace or routes that cross.

• If the following aircraft is faster then the separation may need to be increased. If the following aircraft is slower then sometimes the separation can be decreased.

• Aircraft taking off have to leave time for the wake vortices left by the previous aircraft to dissipate. Larger aircraft leave worse wake vortices and lighter aircraft can cope less well. A larger separation is needed when a lighter category aircraft follows a heavier category aircraft.

Physical restrictions

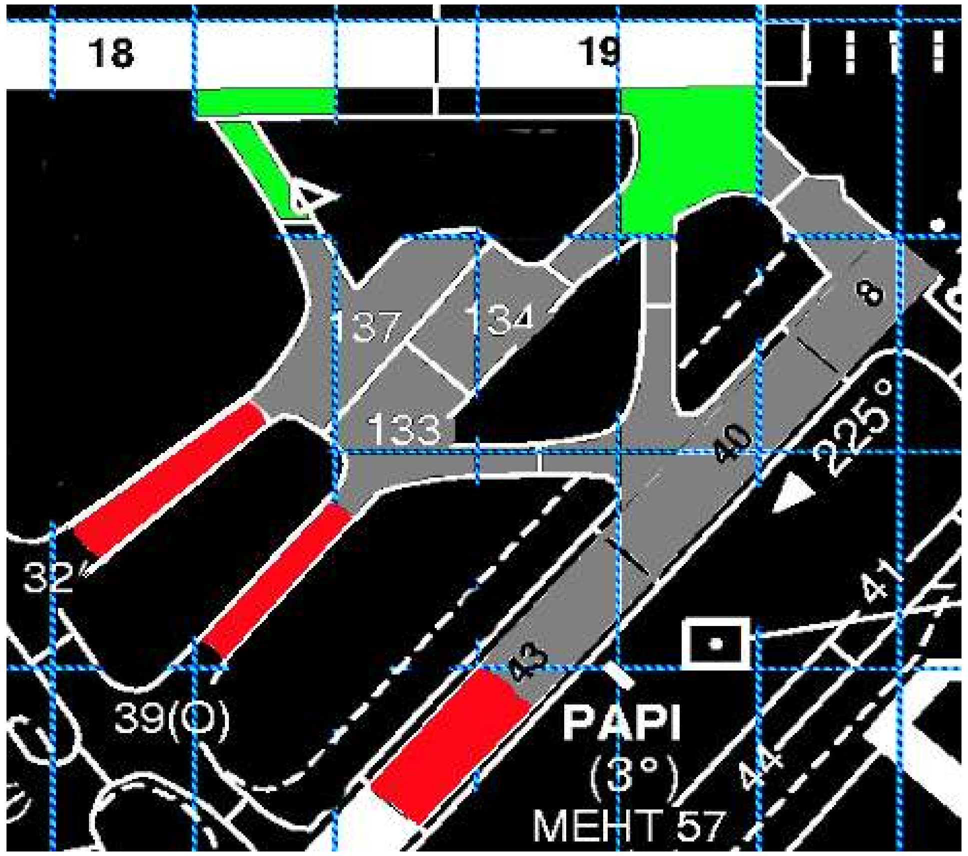
• Physical restrictions limit the reordering that is possible.

• Some reordering won't be physically possible within the holding point.

• Some reordering may involve too much manoeuvring.

• Aircraft taxiing through the holding point need to take good paths.

• Some aircraft need the entire runway for a take-off so must enter at the end.



An example holding point structure. Aircraft enter in three queues (red). Within the grey area aircraft are reordered by a runway controller. Aircraft enter the runway via the green areas.

Other considerations

Calculated Time Of Take-off (CTOT)

Some aircraft have a fifteen minute take-off timeslot within which the controller must ensure the aircraft takes off. This is assigned to control congestion in busy airspace and at busy destination airports.

Minimum Departure Interval (MDI)

At times the frequency aircraft can fly along certain departure routes is temporarily decreased to control temporary congestion in that airspace.

Equity of delay

Equity of delay between aircraft is one of the desirable features of a good sequence.

This work was supported by EPSRC (The Engineering and Physical Sciences Research Council) and NATS (National Air Traffic Services) Ltd. from a grant awarded via The Smith Institute for Industrial Mathematics and Systems Engineering. For further details please contact Jason Atkin, jaa@cs.nott.ac.uk, T: 0115 951 4234

Decision support

Two questions arose.

1) Could a decision support system help? Could it improve the situation?

2) Would a decision support system be fast enough? To be of use it would have to find solutions extremely quickly. The time to find solutions will be perceived as a lag between the situation changing and the suggested order reflecting the change.

Two necessary requirements

To answer these questions two things were needed.

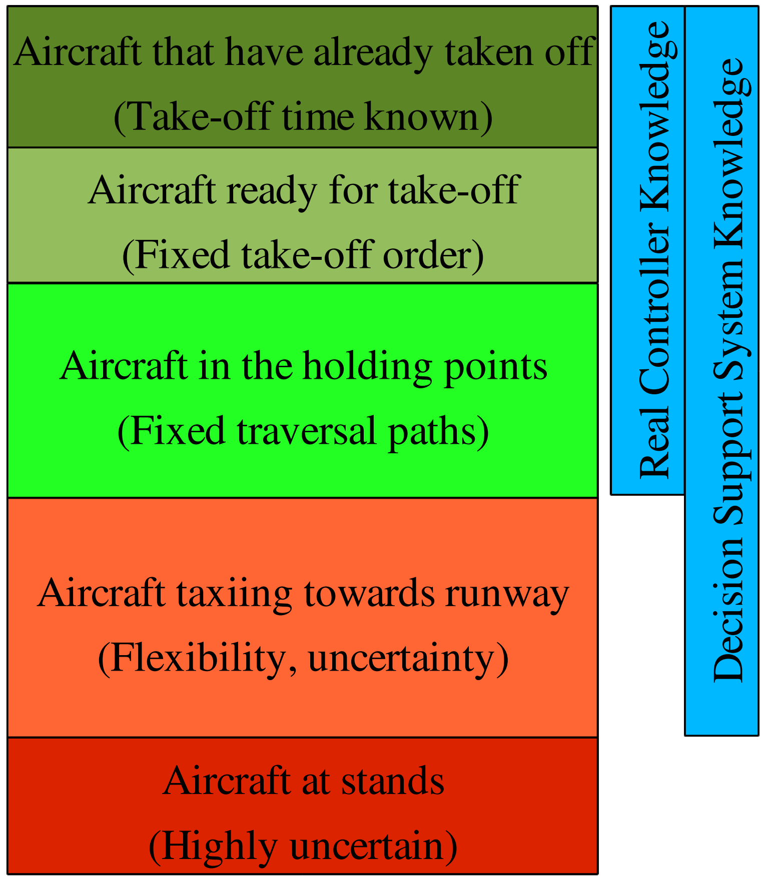
Firstly we needed a model for the situation at Heathrow.

- The model had to take account of the real constraints upon the real controllers.
- It had to be possible, using the model, to determine how good a take-off schedule was and whether it was achievable or not.

Secondly we needed a method for automatically searching for good take-off schedules, to determine whether a computer could do this quickly enough.

The situation changes over time

- Over time aircraft take off and the controller becomes aware of new aircraft.
- Using real data supplied by NATS we aimed to simulate the amount of knowledge a controller had and the amount a decision support system could have had in order to evaluate the merits of a decision support system.
- The system needed to include aircraft in a number of different states.



- A real controller has easy visibility of the aircraft already at the holding points.
- A decision support system could also account for aircraft that have left the stands and are taxiing around the taxiways.
- Arrival times at the holding point can be predicted for aircraft on the taxiways. Uncertainty involved in this prediction means that flexibility has to be left in the schedule.
- We do not assume any knowledge of aircraft still at the stands.
- Separations need to be maintained from aircraft that have already taken off so these need to be included in the system, even after take-off.

Evaluating a schedule

Evaluating a schedule involves a number of steps.

1) Determine whether the schedule is achievable given the physical constraints.

A network model of the holding point is used and paths are allocated to aircraft.

2) Predict take-off times for those aircraft which have not yet taken off.

Take-off times are known for aircraft which have taken off.

Take-off times need to be predicted for the other aircraft in the take-off schedule.

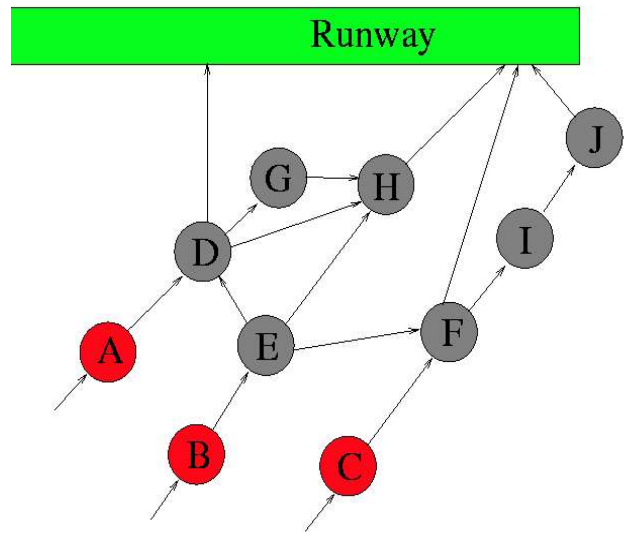
To do so we assume that all aircraft take off as early as possible, given that:

- Separations must be maintained from all previous aircraft.
- The aircraft must be able to reach the runway along its holding point path.
- Aircraft with a CTOT cannot take off before the start of the CTOT slot.

3) Assign a cost to the schedule. The schedule cost is a combination of:

- The total delay for the aircraft, in terms of seconds delay in the holding point.
- The amount by which any CTOT slots are missed.
- A factor to measure the amount of overtaking involved.

Testing the physical constraints



1) Paths through the holding point are assigned to each aircraft heuristically. These ensure that:

- If an aircraft needs to overtake it will have a faster, more direct path assigned to it.
- Larger aircraft enter the runway at the end.
- Only straightforward paths are used.

2) The aircraft are fed through the network model to verify whether the reordering is possible. Each node in the model may be occupied by only one aircraft at any instant in time.

Searching for good take-off schedules

The searches work on complete take-off schedules for all of the aircraft in the system.

The searches move between the different schedules seeking ones with a low cost.

The algorithm can be considered to be working as follows:

1) Start from a schedule which is known to be achievable, e.g. the FCFS schedule.

2) Make a change to the current schedule to create a new schedule.

3) Evaluate the worth of the new schedule and whether it is achievable.

4) Decide whether to move to the new schedule, making it the current schedule.

5) If the desired number of iterations have been completed then end and return the best schedule found, otherwise return to step 2 to find another possible schedule.

The automated searches

We used two different meta-heuristic searches, designed to investigate more of the possible schedules and to escape local optima. A local optimum is a solution better than all of the solutions the search could move to investigate next. It may not, however, be the overall best solution.

A Tabu search algorithm has a short-term memory where it has been. This allows it to avoid going back to schedules it has recently investigated.

A Simulated Annealing algorithm has the ability to sometimes move to higher cost solutions to escape local optima.

For comparison, a limited knowledge exhaustive search was also tested. This test considered only the first seven aircraft at the holding point that had not yet had their take-off time fixed. It investigated all possible ways of reordering these aircraft. As only seven aircraft were considered this search could be performed in the same time as the metaheuristic searches.

Testing the automated searches

- Starting at the beginning of the data period, all of the aircraft which a decision system would have known about at that time were added to the system.
- The automated search under test was applied to reorder the aircraft.
- The test then moved forward in time in 15 seconds intervals and another search was performed at each stage to reorder the aircraft in the system at that time.
    - Aircraft were added to the system as they left their stands.
    - The order of take-off and the take-off times were fixed for aircraft close to their predicted time of take-off.
    - The holding point traversal paths were fixed for all aircraft within the holding point. This simulates having already given the pilots instructions.
- At each stage the automated search took less than 1 second to complete.
- At the end of the dataset all aircraft had been assigned a take-off order and time.
- The final take-off schedule was evaluated and the results are given in the table.

Evaluating a take-off schedule

As Heathrow is so busy the controllers are permitted to miss a few CTOT slots every day by up to five minutes. It is important to miss as few as possible so our primary aim is to reduce the number of CTOT slots missed.

As a secondary objective we attempt to reduce the total time the aircraft spend in the holding point. We refer to this as the holding point delay.

Given the time at which each aircraft reached the holding point and a real or predicted take-off time we can determine the amount of time spent in the holding point for each aircraft and whether it took off within its CTOT slot.

Results

The table below shows the results for the manually and automatically produced schedules. 'Manual, real times' shows the results for the manual schedule, real take-off times. 'Manual, predicted times' shows the results for the manual take-off order, predicting take-off times. Controllers can at times reduce some, unreasonably large separations but we don't assume they will do this so the predicted results will usually lag behind the real results.

FCFS shows the results for a first-come-first-served schedule with no reordering of aircraft. There is a random element involved in both Tabu Search and Simulated Annealing so the results can differ between executions. Each was executed 10 times and the mean is shown. The standard deviation for the delay is also shown, in brackets.

	Dataset 1		***Dataset 2***	
	CTOT	Delay	CTOT	Delay
Manual, real times	10	99805	6	117894
Manual, predicted times	13	110248	11	121037
FCFS, predicted times	94	413350	94	421168
Simulated Annealing	**3**	87195 (291)	**3**	**91529** (430)
Tabu Search	**3**	**87103 (0)**	**3**	91536 (**80**)
7 aircraft exhaustive	9	109081	4	100559

The number of CTOT extensions needed and the total delay in seconds for manually and automatically produced schedules.

- The controllers substantially reduce the total delay by reordering the aircraft.
- The metaheuristic searches perform much better than the 7 aircraft exhaustive search. This can only be due to the increased knowledge and flexibility. The searches also perform better than the controllers, meaning that a decision support system could help.
- To manage their workload the controllers often consider less than 7 aircraft at once. It should therefore be no surprise that the predicted schedule for the 7 aircraft exhaustive search is slightly better than the predicted manual one. Reducing the flexibility more (not shown here) soon makes the schedule worse than that the controllers created as the controllers build significant flexibility into their plans to account for their lack of information.

Conclusions

- The runway controllers do well given their knowledge and workload at the time.
- Considering more aircraft could improve the schedules.
- Heuristically assigning paths through holding points ensures that meta-heuristic searches can find solutions fast enough to be of use in a decision support system.