Dynamic Programming with Strengthened Dominance for the Multi-Runway Sequencing and Allocation Problem

Geert De Maere · Jason A.D. Atkin · Elizabeth M. Argyle

Keywords Runway Sequencing · Multiple Runways · Runway allocation · Machine Scheduling with Sequence Dependent Setup Times · Dominance Rules

Extended Abstract

Due to the significant increase in traffic over recent years, pressure on runway capacity has increased for many of the world's leading airports. This has sparked increasing interest amongst practitioners and academics in the development and the application of highly advanced decision support to maintain and improve the efficiency of runway operations, reduce delay and fuel burn, and improve passenger satisfaction. The algorithms underpinning such systems are often based on dynamic programming (for instance [4,6,8,9,10,12]), mixed integer linear programming (for instance [3,5]), and heuristics (for instance [11,13]). The exact methods are often applied within a rolling window to maintain scalability and tractability (for instance [2,7]).

The computational efficiency of many of these approaches can often be improved by exploiting problem specific characteristics. Examples of such characteristics include structures in the aircraft separations [9], properties of the objective function [9], the distribution of earliest take-off/landing times in real world instances [9], and the presence of aircraft classes [9,10,12]. These characteristics can significantly reduce the size of the solution space and improve the

Geert De Maere, Jason A.D. Atkin, Elizabeth M. Argyle
ASAP Research Group,
School of Computer Science,
University of Nottingham,
Jubilee Campus,
Wollaton Road,
NG8 1BB Nottingham,
United Kingdom
E-mail: {Geert.DeMaere,Jason.Atkin,Elizabeth.Argyle}@Nottingham.ac.uk

average or worst case complexity of the algorithm. This is illustrated in [9], where a wide range of characteristics for runway sequencing problems with a single runway, and with multiple runways and pre-determined runway allocation, were considered. The exploitation of problem characteristics in [9] resulted in the currently fastest known algorithm to solve complex real world problem instances to optimality.

Lieder and Stolletz [10] studied the runway sequencing problem with multiple runways and flexible runway allocation. The authors considered scenarios for both independent and interacting runways (in which case separations on and between runways must be maintained), as well as different runway allocation policies (ranging from fully flexible to partially constrained). Their algorithm exploits the presence of aircraft classes in combination with an additive monotonic increasing objective function, in which case a first come first served order can be inferred between aircraft belonging to the same aircraft class/type [9,10,12]. The concept of an *availability matrix* was introduced, containing the *"the earliest possible times for the next operation of each **aircraft class** on each runway"*. The availability av_{wr} for aircraft class w on runway r (if it has preceding aircraft) is defined as:

$$av_{wr} = max\{p_{w'r'} + \delta^{r',r}_{w',w}\} \qquad w \in W, r \in R \tag{1}$$

in which $p_{w'r'}$ denotes the take off time of the last aircraft of class w' on runway r', and $\delta^{r',r}_{w',w}$ denotes the minimum separation that must be maintained between aircraft class w' on runway r' followed by aircraft class w on r.

Availability is used in Lieder and Stolletz's [10] dynamic program to infer dominance. A state s is no worse than a state s' if (a) *"at least the same number of operations of all classes have been scheduled"* [10] in state s; (b) all non-identical runways are available for all classes of aircraft for state s no later than for state s'; (c) there exists a permutation of the availability matrix for the identical runways in state s' such that state s is available for each class of operations no later than for state s'; (d) the objective value for state s is no worse than the objective value for state s'.

In this paper, we propose that the dominance rule in [10] can be strengthened by redefining availability to take into account the earliest time for the ***first remaining aircraft*** in every aircraft class. Let $f(w)$ denote the function that returns the first remaining aircraft in the ordered set for aircraft type w. Availability can then be redefined by equation 2, in which $et_{f(w)}$ denotes the earliest take-off time for the first available aircraft in w, $f(w)$.

$$av_{wr} = max\{et_{f(w)}, p_{w'r'} + \delta^{r',r}_{w',w}\} \qquad w \in W, r \in R \tag{2}$$

In contrast to Equation 1, Equation 2 accounts for the fact that the minimum separation $\delta^{r',r}_{w',w}$ will have no influence if $et_{f(w)} \geq p_{w'r'} + \delta^{r',r}_{w',w}$. In layman's terms, if the last aircraft on runway r is a heavy aircraft that requires a large separation (due to the strong wake turbulence that it causes), or a small aircraft that requires a small separation (since it causes less wake turbulence), the difference in this separation would not influence the time for the first remaining aircraft of type w on runway r if it was not yet available for sequencing anyway. That is, if the separation constraints for two states s and s' are non-binding upon the take-off time for the first remaining aircraft, the value of av_{rw} will be the same for both states, increasing the chance that either state will be dominated and can be pruned from the search space. In contrast to Equation 1, Equation 2 thereby takes into account the properties of the actual aircraft, in addition to the presence of aircraft classes.

In a fashion similar to the dominance rule described above, the other pruning rules introduced in [9] can also be extended to the multi-runway sequencing problem with flexible runway allocation. For instance, insertion dominance which states for a monotonic increasing objective function and a single runway that, *"if an aircraft x can be inserted into a partial sequence s (i.e. not appended) without delaying any of the subsequent and remaining aircraft, the sequence s can be pruned without compromising optimality"* [9], can be modified to *"without delaying any of the subsequent aircraft and remaining aircraft on any of the runways"*. The delay for aircraft remaining to be added to the sequence represented by s can be measured in terms of the *availability* defined above.

As a proof of concept, both of the rules above were implemented in the forward dynamic program presented by Lieder and Stolletz in [10], and tested on the publicly available benchmark instances that were introduced by the authors [1]. The datasets contain different numbers of aircraft (N), different runway loads (RL), and different planning horizons (PH), as reflected by their naming. Each dataset contains 10 different subsets, for which the averages are presented in Table 1. The results reported for Lieder and Stolletz's approach and our strengthened formulated were both generated on an Intel(R) Xeon(R) CPU E5-1620 v2 @ 3.70GHz processor, sharing as much of the code as possible in both implementations to ensure a fair comparison. The number of states is reduced by a factor 27.6 and 34.9, and the CPU times are improved by a factor 20.9 and 20.3 for interacting and independent runways, respectively. The superior performance of our strengthened dynamic program clearly illustrates the benefit of exploiting problem characteristics in runway sequencing problems, and provides a strong motivation for our future work on exploring the applicability of the other dominance rules introduced in [9] in multi-runway sequencing problems with flexible runway allocation. The results for these other rules will be included in our conference presentation, together with an in depth explanation of the work already reported in this abstract.

References

1. Benchmark instances. https://stolletz.bwl.uni-mannheim.de/en/research/or_instances/aircraft_scheduling_problem/. Accessed: 2018-03-08
2. Atkin, J.A.D., G. De Maere, Burke, E.K., Greenwood, J.S.: Addressing the pushback time allocation problem at heathrow airport. Transportation Science **47**(4), 584–602 (2013). DOI 10.1287/trsc.1120.0446. URL https://doi.org/10.1287/trsc.1120.0446

Dataset	Lieder and Stolletz [10]		Strengthened		Ratios	
	#States	Time (s)	#States	Time (s)	#States	Time
Interacting Runways						
N40-RL80-PH1800	250881.5	2.42	5031.5	0.09	83.5	72.8
N46-RL90-PH1800	535080.7	5.48	45907.9	0.98	21.6	13.5
N50-RL100-PH1800	944804.6	16.38	230592.0	7.14	5.2	3.2
N30-RL90-PH1200	62363.3	0.73	7405.0	0.10	11.4	10.7
N40-RL90-PH1500	282187.0	3.07	17982.7	0.27	22.4	18.4
N54-RL90-PH2100	1338637.6	14.66	126988.0	3.16	27.0	16.5
N60-RL90-PH2400	2124832.7	20.81	173580.6	4.31	21.8	11.2
Average ratio:					27.6	20.9
Independent Runways						
N40-RL80-PH1800	250621.1	2.28	3465.5	0.06	113.9	73.4
N46-RL90-PH1800	549803.3	5.77	36170.6	0.80	42.8	22.2
N50-RL100-PH1800	1008321.2	36.00	275265.0	22.19	5.7	2.9
N30-RL90-PH1200	60105.3	0.59	5050.3	0.07	15.5	11.9
N40-RL90-PH1500	278754.2	2.56	18014.9	0.28	17.7	10.6
N54-RL90-PH2100	1353247.3	16.48	121358.9	4.53	23.2	10.0
N60-RL90-PH2400	2167773.2	29.44	161797.4	8.39	25.7	11.2
Average ratio:					34.9	20.3

Table 1 A comparison of the average number of states and average runtime in seconds for the dynamic program introduce by Lieder and Stolletz [10] and the strengthened dynamic program introduced here on the benchmark instances available in [1]. The last two columns denote the improvement ratios (reduction in states and CPU time)

3. Avella, P., Boccia, M., Mannino, C., Vasilyev, I.: Time-indexed formulations for the runway scheduling problem. Transportation Science **51**(4), 1196–1209 (2017). DOI 10.1287/trsc.2017.0750. URL `https://doi.org/10.1287/trsc.2017.0750`
4. Balakrishnan, H., Chandran, B.G.: Algorithms for scheduling runway operations under constrained position shifting. Operations Research **58**(6), 1650–1665 (2010). DOI 10.1287/opre.1100.0869. URL `https://doi.org/10.1287/opre.1100.0869`
5. Beasley, J.E., Krishnamoorthy, M., Sharaiha, Y.M., Abramson, D.: Scheduling aircraft landings – the static case. Transportation Science **34**(2), 180–197 (2000). DOI 10.1287/trsc.34.2.180.12302. URL `https://doi.org/10.1287/trsc.34.2.180.12302`
6. Bennell, J.A., Mesgarpour, M., Potts, C.N.: Dynamic scheduling of aircraft landings. European Journal of Operational Research **258**(1), 315 – 327 (2017). DOI https://doi.org/10.1016/j.ejor.2016.08.015. URL `http://www.sciencedirect.com/science/article/pii/S0377221716306300`
7. Furini, F., Persiani, C., Toth, P.: Aircraft sequencing problems via a rolling horizon algorithm. In: A. Mahjoub, V. Markakis, I. Milis, V. Paschos (eds.) Combinatorial Optimization, *Lecture Notes in Computer Science*, vol. 7422, pp. 273–284. Springer Berlin / Heidelberg (2012)
8. G. De Maere, J.A.D. Atkin: Pruning Rules for Optimal Runway Sequencing With Airline Preferences, *Lecture Notes in Management Science: Proceedings of the 7th International Conference on Applied Operational Research, Vienna, Austria*, vol. 7, pp. 76 – 82 (2015)
9. G. De Maere, J.A.D. Atkin, E.K. Burke: Pruning rules for optimal runway sequencing. Transportation Science (2017). DOI 10.1287/trsc.2016.0733. URL `https://doi.org/10.1287/trsc.2016.0733`
10. Lieder, A., Stolletz, R.: Scheduling aircraft take-offs and landings on interdependent and heterogeneous runways. Transportation Research Part E: Logistics and Transportation Review **88**, 167 – 188 (2016). DOI 10.1016/j.tre.2016.01.015. URL `https://doi.org/10.1016/j.tre.2016.01.015`
11. N.R. Sabar, G. Kendall: An iterated local search with multiple perturbation operators and time varying perturbation strength for the aircraft landing problem. Omega **56**, 88 – 98 (2015). DOI 10.1016/j.omega.2015.03.007. URL `http://www.sciencedirect.com/science/article/pii/S0305048315000523`

12. Psaraftis, H.N.: A dynamic programming approach for sequencing groups of identical jobs. Operations Research **28**(6), 1347–1359 (1980). DOI 10.1287/opre.28.6.1347. URL https://doi.org/10.1287/opre.28.6.1347
13. U. Benlic, Brownlee, A.E., Burke, E.K.: Heuristic search for the coupled runway sequencing and taxiway routing problem. Transportation Research Part C: Emerging Technologies **71**, 333 – 355 (2016). DOI 10.1016/j.trc.2016.08.004. URL http://www.sciencedirect.com/science/article/pii/S0968090X16301322