Transportation Research Part C 95 (2018) 165–184

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

A two-stage taxi scheduling strategy at airports with multiple independent runways

X. Zou[a,b], P. Cheng[a,*], W.D. Liu[b], N. Cheng[a], J.P. Zhang[b]

[a] *Institute of Navigation and Control, Department of Automation, Tsinghua University, Beijing 100084, PR China*
[b] *The Second Research Institute of Civil Aviation Administration of China, Chengdu 610041, PR China*

ARTICLE INFO

Keywords:
Airport
Taxi scheduling
Integer programming
Runway exit availability

ABSTRACT

Long taxiing times at large airports lead to fuel wastage and dissatisfied passengers. This paper investigates the 4D taxi scheduling problem in airports to minimize the taxiing time. We propose an iterative two-stage scheduling strategy. In the first stage, all aircrafts in a current schedule period are assigned initial 4D routes. In the second stage, landing aircrafts that are unavailable to fulfil their initially assigned routes are rescheduled using a shortest path algorithm based approach. In this paper, the simplified model used in most existing literature, that depicts a runway as having a single entrance and a single exit or even sets only one point to represent both of them has been discarded. Instead, we model the fact that a runway has multiple entrance and exit points and use an emerging concept—Runway Exit Availability (REA)—to measure the probability of clearing a runway from a specific exit during a specific time interval so that the taxiing scheduling model can be much higher approximation to the practical operation. An integer programming (IP) model factoring REA is proposed for assigning 4D taxiing routes in the first stage. The IP model covers most practical constraints faced in airport taxiing procedures, such as the rear-end/head-on conflict constraint, runway-crossing constraint, take-off/landing separation constraint, and taxi-out constraint. Besides, flight holding patterns at intersections are much more realistically modelled. Furthermore, to accelerate the solving process of the IP model, we have refined the formulation using several tricks. Simulation results by proposed scheduling approach for operations at the Beijing Capital International Airport (PEK) for an entire day demonstrate a surprising taxiing time saving against the empirical data and simulation results based on a strategy similar to what being used now days while showing an acceptable running time of our approach, which supports that our approach may help in real operation in the future.

1. Introduction and literature review

A rapid development of air transportation has resulted in increased flight delays worldwide. Among the factors influencing the air traffic system, airport capacity is considered as one of the bottlenecks (Neufville and Odoni, 2003). To deal with this issue, the favoured approach in all countries is to expand the airport, i.e. to build more runways and taxiways. When an airport has a small number of runways and a simple taxiway structure, runway capacity determines the airport capacity. However, with an increasing complexity of the taxiway system, congestion on the taxiway too increases, so that the efficiency of taxi planning becomes another restricting factor determining airport capacity (Cheng, 2004). Using regression analysis, Kisler and Gupta found that surface traffic

---

[*] Corresponding author.

E-mail addresses: zouxiang@caacsri.com (X. Zou), chengp@tsinghua.edu.cn (P. Cheng), liuweidong@caacsri.com (W.D. Liu), ncheng@tsinghua.edu.cn (N. Cheng), zhangjianping@caacsri.com (J.P. Zhang).

https://doi.org/10.1016/j.trc.2018.07.005
Received 25 December 2017; Received in revised form 20 June 2018; Accepted 5 July 2018

has a great impact on an airport's operational efficiency (Kistler and Gupta, 2009). The airport taxi scheduling (TS) problem has become a focus by many researchers over the past decade and several methods to solve the TS problems have been proposed.

In this work, we sort previous literatures into three groups according to the solving method applied in the literatures. Works in Burgain et al. (2012), Atkin et al. (2013), Yu and Qing (2017) are mainly concerned about deciding the pushback times, the take-off sequence of departing flights, or the estimation of unimpeded taxiing time. They do not provide taxiing routes. Another kind of methods (Baik et al., 2000; Lesire, 2010; Atkin et al., 2014; Weiszer et al., 2015; Zhang et al., 2016; Park et al., 2017) sequentially calculate every flight's shortest feasible taxiing route according to their chronological order along with taxiway occupation by preceding flights, which represents constraints for subsequent flights. This kind of approaches are efficient in computational cost and run very fast. In Atkin et al. (2014), Ravizza et al. report that it only took tens of seconds to schedule all flights for an entire day at the Zürich Airport. However, difficulty in obtaining or the inability to obtain an optimal pushback sequence and pushback times is a common problem encountered in this type of approaches. Most of them simply adopt a first come first served principle. The third kind of methods (Marín, 2006, 2013; Balakrishnan and Jung, 2007; Marín and Codina, 2008; Rathinam et al., 2008; Roling and Visser, 2008; Lee and Balakrishnan, 2010; Clare and Richards, 2011; Anderson and Milutinović, 2013; Mori, 2013; Guepet et al., 2016; Evertse and Visser, 2017) consider several operating flights in a single scheduling instance. They usually build a time–space network representing the taxiway system and model the taxi scheduling problem as a Mix Integer Programming (MIP) or Integer Programming (IP) problem. The method of Roling and Visser (2008) is an earlier version of the IP based method. Balakrishnan et al. published a series of literatures (Balakrishnan and Jung, 2007; Rathinam et al., 2008; Lee and Balakrishnan, 2010) to optimize real operations in different airports. This series assumed one or two alternative routes for each flight. Marín published a triplet of papers (Marín, 2006, 2013; Marín and Codina, 2008) considering several algorithms, such as "Fix and Relax" and "Lagrangian Decomposition" to speed up the solving procedure. The constraints in Clare and Richards (2011) comprising runway crossing conflicts, rear-end conflicts, head-on conflicts, varying taxiing speed, etc. are very comprehensive. However, the highly complex model resulted in a solving time too long to be acceptable. In contrast to the approaches in Marín (2006), Balakrishnan and Jung (2007), Marín and Codina (2008), Rathinam et al. (2008), Roling and Visser (2008), Lee and Balakrishnan (2010), Clare and Richards (2011), Marín (2013) that set discrete time variables, the approaches in Anderson and Milutinović (2013), Mori (2013) used continuous variables to represent taxiing procedures and proposed two MIP models. Authors in Guepet et al. (2016), Evertse and Visser (2017) set the objective of taxi scheduling to be minimizing the emission by aircrafts' surface movement which shows the concerning about the environmental protection. However, there is no essential difference between the objective of minimizing taxi time and the one of minimizing the emission especially in the view of the methods applied to solve the TS problem.

In fact, Air Traffic Flow Management (ATFM) (Bertsimas and Patterson, 1998; Bertsimas et al., 2011), Runway Configuration (RC) (Kim et al., 2014), Departure & Arrival Scheduling (DAS) (Eun et al., 2010; Mori, 2013), Gate Assignment (GA) (Ding et al., 2005), and Taxi Scheduling (TS) are highly dependent on each other. ATFM assigns time slots of runway operation (take-off or landing) to flights at several airports. This time slot is macroscopic (e.g., it is set at 15 min in Bertsimas et al. (2011)), but it defines the flow pattern of the air traffic system. With the output of ATFM, RC, DAS, GA, and TS, operations of flights at airports are made more efficient, e.g. by deciding landing sequence and time, the runway could be operated for landing or take-off, parking position, taxiing route, etc. However, the internal relationship among ATFM, RC, DAS, GA, and TS is beyond the scope of this paper and it was not considered in existing literatures yet. Henceforth, in this work, we assume that the TS problem is being solved on the premise of determined operating runways and parking positions for all flights and available earliest pushback time for departures and known landing time of arrivals.

In this paper, we implement a discrete time IP-based method with the aim of delivering an acceptable solution.

Previous literatures have usually oversimplified runway operations, which represents a runway with only one or two nodes. In fact, a runway usually has multiple entrances and exits. One major contribution of this paper is to use an emerging concept called "Runway Exit Availability" (REA) along with a two-stage scheduling strategy to deal with the problem of uncertain exit and exit times, which in fact was introduced for the first time in one of our papers in 2014 (Cheng et al., 2014). In this paper, we strengthen the mathematical definition comprehensively and propose an operable calculation method for REA. The introduction of REA can help the optimization model to be much closer to the real operation.

Another main contribution of this paper is to make the pattern of holding along taxiing routes more practical. Where block or waiting along a taxiing route is necessary, previous literatures held flights on nodes representing intersections of taxiing routes and introduced node capacity constraints, e.g. the formulation of general network flow problems. However, in reality, taxiing flights do not stop at intersections or on nodes. Instead, they hold around the intersections just like cars waiting for green light. Hence, theoretically, the capacity of each node must be greater than one because two or three flights can wait at the same intersection simultaneously. However, if we set the capacity as simply greater than one, a collision caused by two or more flights on the same side of an intersection cannot be avoided. This is because in such a scenario, the only point that would be considered is whether the gross number of waiting flights is below the stated capacity. In this paper, we introduce new decision-making variables, called the block-and-hold variables, to represent holding behaviours along taxiing routes. With these variables, no node capacity constraints are needed and no extra nodes are required, so that the size of the taxiing network does not need to be expanded. We also integrate the "taxi-out constraint" into the IP model. This kind of constraint considers the pushback and the engine starting procedures of departing flights. Note that, the "taxi-out constraint" in this paper is different from the concept of pushback delay that is common in existing research works. As a matter of fact, our proposal of including such a constraint has been supported by some published works (i.e. Reference Weiszer et al. (2014) and Coupe and Milutinovi (2015)).

Finally, the heuristics for head-on conflict check and avoidance and several tricks to refine the formulation of the IP model result in no Branch-and-Bound operations being needed in simulation instances and that a very few numbers of cuts are applied. This