



This item was submitted to Loughborough's Institutional Repository (<https://dspace.lboro.ac.uk/>) by the author and is made available under the following Creative Commons Licence conditions.



CC creative commons
COMMONS DEED

Attribution-NonCommercial-NoDerivs 2.5

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

BY: **Attribution.** You must attribute the work in the manner specified by the author or licensor.

Noncommercial. You may not use this work for commercial purposes.

No Derivative Works. You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the [Legal Code \(the full license\)](#).

[Disclaimer](#) 

For the full text of this licence, please go to:
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

Receding Horizon Control for Aircraft Arrival Sequencing and Scheduling

Xiao-Bing Hu and Wen-Hua Chen, *Member, IEEE*

Abstract—Airports, especially busy hub airports, proved to be the bottleneck resources in the air traffic control system. How to carry out arrival scheduling and sequencing effectively and efficiently is one of main concerns to improve the safety, capacity, and efficiency of the airports. This paper introduces the concept of receding horizon control (RHC) to the problem of arrival scheduling and sequencing in a dynamic environment. The potential benefits RHC could bring in terms of airborne delay and computational burden are investigated by means of Monte Carlo simulations. It is pointed out that while achieving similar performance as existing schemes, the new arrival scheduling and sequencing scheme significantly reduces the computational burden and provides potential for developing new optimization algorithms for further reducing airborne delay.

Index Terms—Air traffic control, arrival scheduling and sequencing, first-come-first-served, position shifting, receding horizon control.

I. INTRODUCTION

ONE of the main problems that air traffic control (ATC) has to face nowadays is airborne delays caused by air traffic congestion. To a certain extent, this is due to the traffic distribution that, owing to the requirements of transport users, is mainly concentrated at large airports, particularly so-called busy hub airports, where the management of the region around the airport, i.e., the terminal area (TMA), becomes critical. Air traffic management in the TMA has two components: separation assurance and traffic flow management. Separation assurance is the function of providing safe separation between pairs of aircraft. Traffic flow management, by contrast, manages groups or aggregations of flights, seeking to balance the demand for service with the available capacity. Arrival sequencing and scheduling (ASS) is one of the standard problems in ATC. Simply speaking, ASS is the function of generating efficient landing sequences and landing times for arrivals at the airport such that the safety separation between arrival aircraft is guaranteed, the available capacity at the airport is efficiently used, and airborne delays are significantly reduced [1], [17], [18], [21].

A simple way to perform ASS is to schedule arrival of the aircraft in a first-come-first-served (FCFS) order based on a

predicted landing time (PLT) at the runway. Although FCFS scheduling establishes a fair order based on PLT, it ignores much other useful information which may lead to efficiently make use of the capacity of the airport, reduce airborne delays, and/or improve the service to airlines. In [2], “delay exchange” is introduced into ASS, where a fair method is proposed to accommodate an airline request for an earlier arrival by advancing the landing time of one aircraft while simultaneously delaying another aircraft from the same airline. In [3], individual airline priorities among incoming flights are taken into account in ASS, and the concept of “priority scheduling” is then defined as the scheduling of a bank of arrival aircraft according to a preferred order of arrival. The most widely accepted concept in the practices of ASS is “position shifting” [4]–[11], [17]–[22], which is based on two facts: first, safety regulations state that any two coalitudinal aircraft must maintain a “minimum horizontal separation,” which is a function of the types and of the relative positions of the two aircraft; second, the “landing speed” of a type of aircraft is generally different from that of another type of aircraft. As a consequence of the variability of the above parameters, the landing time interval (LTI), which is the minimum permissible time interval between two successive landings, is a variable quantity. These differences in separation are mandatory and recognized by federal regulations. By shifting the aircraft position, it is possible to significantly reduce the airborne delay and increase the capacity of an airport. This paper focuses on the problem of “position shifting”-based ASS.

There are a number of papers studying the problem of “position shifting” ASS (for example, see [6]–[11], [17]–[22]). Dear [6] presents an excellent investigation of the ASS. In particular, the author points out that, in order to determine the landing sequence, all aircraft in the system need to be considered. This is denoted as the “static” version of the problem where all aircraft are present at the same time in the holding stacks and they can land at any time. In the “dynamic” version, the author considers the composition of the aircraft mix to change over time. The operational constraints on the rearrangement of the sequence mentioned above are modeled by introducing the notion of constrained position shifting (CPS). With respect to this, the final position of an aircraft in the actuated sequence cannot differ from the initial one by a specified parameter, termed maximum position shifting. In [7] and [8], a dynamic programming algorithm is developed for the static case that also implements the CPS concept, with the objective of minimizing the total time required to land a set of aircraft having the same PLT. The sequencing problem is recognized to have a structure

Manuscript received June 10, 2004; revised October 13, 2004. This work was supported in part by an ORS Award from the Overseas Research Students Awards Scheme, Universities U.K. The Associate Editor for this paper was H. Dia.

The authors are with the Department of Aeronautical and Automotive Engineering, Loughborough University, LE11 3TU, U.K. (e-mail: X.Hu@lboro.ac.uk; W.Chen@lboro.ac.uk).

Digital Object Identifier 10.1109/TITS.2005.848365

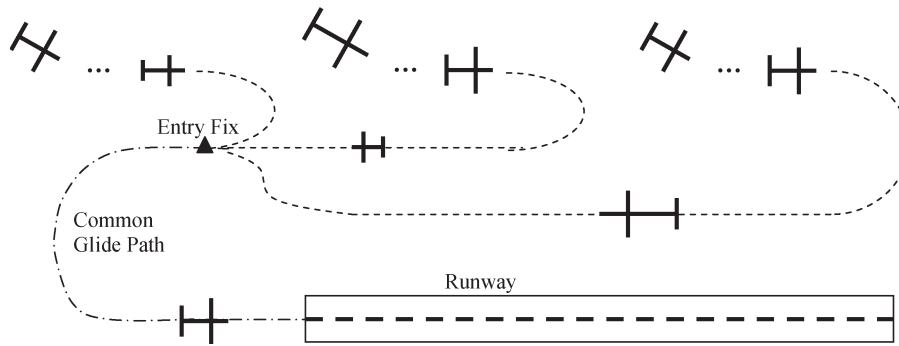


Fig. 1. A set of aircrafts waiting for landing.

similar to the traveling salesman problem (TSP). Under this structure, grouping aircraft by weight class and considering that the number of different classes is fixed, it is possible to show that the algorithm runs in polynomial time. In [11], referring to the same optimization criterion, a combinatorial approach is proposed that, for selecting the aircraft to land next, takes into account not only the weight class but also the preferred landing time of the individual aircraft with the same optimization criterion. In [10], for the static case, the ASS problem is modeled as a cumulative TSP with ready times and two lower bounds are proposed for testing heuristic solutions. For the dynamic case, where only limited knowledge of the arrivals is assumed, a set of constraints are added to the basic model to allow the controller to maintain given patterns of the landing sequences previously generated. Beasley *et al.* [19] report a population heuristic algorithm for static ASS problem. Based on the result of [19], the dynamic case is investigated in [20] by introducing displacement function, which is used to explicitly link each new decision back to the previous decisions. Fuzzy theory is attempted in [22], where experts' knowledge of the ASS problem is used to form the core of fuzzy reasoning mechanism. In [9], three models and corresponding algorithms are developed for the Boston Logan Airport terminal airspace to expedite the landing of incoming aircraft. The paper reports results on real airflow data sets for the Logan Airport which show that better sequencing can reduce delays by 30% in some instances.

Most of the papers above mainly focus on developing effective static algorithms to solve the problem of position shifting-based ASS. Although the dynamic case is also discussed in [7], [8], and [10], the proposed dynamic algorithms are just simple extensions of the associated main static results by introducing some constraints. In the real world of ATC, ASS is always carried out in a dynamic environment. Therefore, modeling and developing algorithms directly based on the dynamic feature of the ASS problem could bring advantages. In [17], [18], [20], and [21], a common methodology based on freeze horizon, influence horizon, optimization interval, or similar concepts is adopted to attack the ASS problem in a dynamic fashion. However, most attention is still paid to the online optimizer, and little insight is provided about how to design methodology-related parameters or what are the influences of these parameters on performance and robustness. In this paper, the concept of receding horizon control (RHC) is introduced to study the dynamic

ASS problem. Besides online optimizer, the RHC-related parameters such as the length of receding horizon and operating interval are variables to be chosen. How to choose them and what are their influences on the overall performance, rather than the development of online optimizer, are the focuses of this paper.

Simply speaking, RHC is an N -step-ahead online optimization strategy. At each time interval, based on current available information, RHC optimizes the concerned problem for the next N intervals in the near future, but only the part of solution corresponding to current interval is implemented. At the next interval, RHC repeats the same optimizing procedure for another N intervals in the near future based on updated information. RHC has now been widely accepted in the area of control engineering and proved to be very successful regarding its many advantages against other control strategies [12]. Recently, attentions have been paid to applications of RHC to areas like management and operations research. For example, theoretical research work on how to apply RHC (also known as model predictive control) to a certain class of discrete event systems was presented in [13] and [14], and many practical implementations of RHC in the area of commercial planning and marketing were reported in [15]. However, as mentioned in [16], research work on applying RHC to areas other than control engineering is just beginning. This paper fully investigates the way and the result of introducing the concept of RHC into the problem of position-shifting-based ASS at a busy hub airport. The main objective of the paper is to investigate any potential benefit with respect to airborne delay and online computational burden by applying RHC in online sequencing and scheduling the arrival traffic flow at the airport in a dynamic environment. Section II describes the details of the problem of arrival scheduling and sequencing, and then an RHC algorithm to solve this problem is presented in Section III. Results of Monte Carlo simulations are given and discussed in Section IV. The paper ends with conclusions in Section V.

II. THE PROBLEM OF ARRIVAL SCHEDULING AND SEQUENCING

A set of aircraft is supposed to land at the same runway of an airport during a period of time of interest, as illustrated by Fig. 1. Assume that the number of the aircraft of concern is N_{AC}

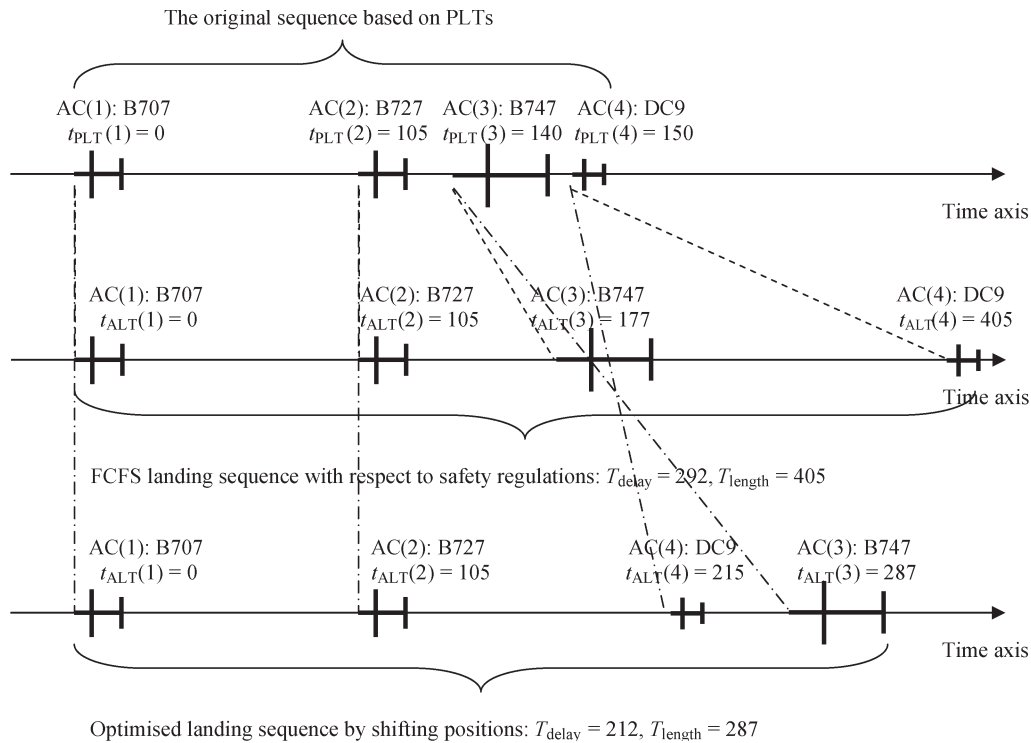


Fig. 2. FCFS landing sequence and position shifting.

 TABLE I
 MINIMUM LANDING TIME INTERVALS [10]

S_{ij} (second)	Category of Following Aircraft: j				
	1	2	3	4	
Category of Leading Aircraft:	1	96	200	181	228
	2	72	80	70	110
	3	72	100	70	130
	4	72	80	70	90

Note: 1 = B747; 2 = B727; 3 = B707; 4 = DC9.

and the period of time is T_{range} minutes long. Then N_{AC} and T_{range} can be used to estimate the degree of congestion at the runway of the airport. For the i th aircraft $\text{AC}(i)$ in the original sequence $i = 1, \dots, N_{\text{AC}}$, there is a PLT at the runway denoted as $t_{\text{PLT}}(i)$. Based on this set of PLTs, i.e., $t_{\text{PLT}}(i), i = 1, \dots, N_{\text{AC}}$, an FCFS landing sequence can be directly worked out with respect to safety regulations.

As mentioned in Section I, the safety separation, i.e., minimum LTI, between a pair of successive aircraft is a function of the type and of the relative positions of the two aircraft. For the sake of simplicity, such as [10], aircraft waiting to land are classified into a relatively small number of distinct categories, according to speed, capacity, weight, and other technical characteristics. Table I shows the minimum LTIs relative to main categories of commercial aircraft. In particular, four categories are considered: category number 1 identifies Boeing 747 (B747), category number 2 corresponds to Boeing 727 (B727), category number 3 identifies Boeing 707 (B707), and finally category number 4 corresponds to Mc Donnell Douglas DC9 (DC9).

It is evident that the LTIs in Table I are asymmetric. For example, a minimum LTI of 200 s is required for a B727 to follow

a B747, while a minimum LTI of only 72 s needs to be satisfied for the same pair of aircraft in reverse order. By taking advantage of the asymmetries of the LTIs, in other words by shifting positions of aircraft in an FCFS landing sequence, it is possible to reduce delays and to improve the capacity of the airport. The potential benefits resulting from position shifting, considering airborne delay, are illustrated in Fig. 2 in an intuitive way.

Suppose by using a certain algorithm to shift positions based on an FCFS landing sequence, a new landing sequence, position shifting (PS) landing sequence, is worked out, where the i th aircraft in the original landing sequence is given an allocated landing time (ALT), denoted as $t_{\text{ALT}}(i), i = 1, \dots, N_{\text{AC}}$. Then the goal of the algorithm for shifting position is usually to minimize the total airborne delay of the PS landing sequence T_{delay} , which is defined as

$$T_{\text{delay}} = \sum_{i=1}^{N_{\text{AC}}} [t_{\text{ALT}}(i) - t_{\text{PLT}}(i)]. \quad (1)$$

Instead of T_{delay} , sometimes the length of the PS landing sequence T_{length} that is calculated as

$$T_{\text{length}} = \max [t_{\text{ALT}}(1), \dots, t_{\text{ALT}}(N_{\text{AC}})] - \min [t_{\text{ALT}}(1), \dots, t_{\text{ALT}}(N_{\text{AC}})] \quad (2)$$

is adopted as the index for optimization. The index T_{delay} emphasizes the operating cost of airlines, while the index T_{length} focuses on the capacity of the airport. In many cases, a minimum T_{delay} occurs simultaneously with a minimum T_{length} . However this does not mean they are equivalent. Due to the

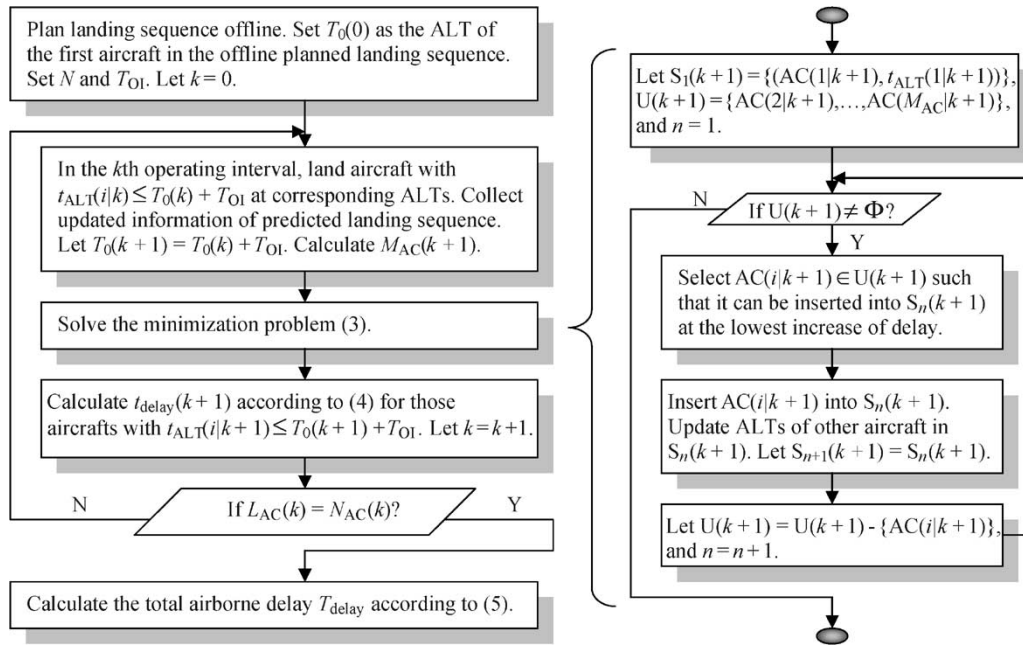


Fig. 3. Flow charts of RHC algorithm (left) and online optimizer (right).

space limit, the proposed RHC algorithm in this paper will be described only based on the index T_{delay} .

No matter which index, T_{delay} or T_{length} , is chosen for optimization, the corresponding minimization problem is proved to be a NP complete problem [10]. As the number of aircraft of concern, i.e., N_{AC} , goes up, the computational burden for optimization will significantly increase. To further reduce airborne delay and improve airport capacity, the aircraft could be classified into more than four instinct categories, which means the optimization will be based on more complicated tables of minimum LTIs. Consequently, the computational efficiency of position shifting ASS is even more challenging.

If the landing sequence only needs to be optimized offline, then the computational burden might be manageable. This is the case when the arrival flow is managed in the offline planning phase, which is generally carried out based on predicted information 1 or 2 days before the arrival flow actually happens. Unfortunately, in real world, air traffic goes on in a dynamic environment and there exist many uncertainties in the arrival flow at the airport. Therefore, the offline plan is not enough and the arrival flow should be resequenced and rescheduled on the ground of updated information in real-time. As indicated in [17], one of the two technical challenges in ATM is the development of real-time scheduler. In this case, the computational burden for optimization is vital to any candidate algorithm. Besides, how to provide a robust performance in a dynamic ATC environment is another important issue. In other words, because of uncertainties on arrival flow and changes of environment and infrastructure such as weather, a good algorithm proposed to attack the ASS problem should consider not only making the most of updated information but also filtering out unnecessary/low-valuable information.

III. RHC ALGORITHM

RHC has proved to be a very effective online optimization strategy in the area of control engineering [12]. It is easy for RHC to handle complex dynamic systems with various constraints. It also naturally exhibits promising robust performance against uncertainties since the online updated information can be sufficiently used to improve the decision. In this framework, decision is made by looking ahead for N steps in terms of a given cost/criterion, and the decision is only implemented by one step. Then the implementation result is checked, and a new decision is made by taking into account of updated information and looking ahead for another N steps.

Before the proposed RHC algorithm for solving the ASS problem is described, some notations are introduced. T_{OI} is the length of an operating interval. N is the length of receding horizon, i.e., the receding horizon has N operating intervals. $N_{AC}(k)$ denotes the number of aircraft in the original predicted landing sequence at the k th operating interval. $t_{PLT}(i|k)$ and $t_{ALT}(i|k)$ stand for the PLT and actual landing time of the i th aircraft in original predicted landing sequence at the k th operating interval, respectively, where $i = 1, \dots, N_{AC}(k)$. $T_0(k)$ denotes the beginning time of the receding horizon at the k th operating interval. $M_{AC}(k)$ is the number of those aircraft whose PLTs are not larger than $T_0(k) + NT_{OI}$ at the k th operating interval. In other words, the first $M_{AC}(k)$ th aircraft in the original predicted landing sequence at the k th operating interval is predicted to land no later than $T_0(k) + NT_{OI}$.

The ASS scheme developed based on the RHC concept is described as follows with the flow chart given in Fig. 3:

- Step 1) Suppose that an offline planned landing sequence is available. Set $T_0(0)$ as the ALT of the first aircraft in the offline planned landing sequence. Set

N and T_{OI} according to each individual airport. Let $k = 0$.

- Step 2) During the k th operating interval, those aircraft with $t_{ALT}(i|k) \leq T_0(k) + T_{OI}$ are cleared to land at the runway according to the optimized ALTs, i.e., implement the previously scheduled sequence for the k th operating interval. Meanwhile, collect updated information of the original predicted landing sequence at the $(k + 1)$ th operating interval, let $T_0(k + 1) = T_0(k) + T_{OI}$, and calculate $M_{AC}(k + 1)$. For those aircraft with $t_{ALT}(i|k) > T_0(k) + T_{OI}$, their PLTs updated at the $(k + 1)$ th operating interval could be different from their PLTs collected at the k th operating interval due to uncertainties in the ATC environment.
- Step 3) Solve the following minimization problem by shifting position and rescheduling aircraft

$$\min_{t_{ALT}(1|k+1), \dots, t_{ALT}[M_{AC}(k+1)|k+1]} \sum_{i=1}^{M_{AC}(k+1)} [t_{ALT}(i|k+1) - t_{PLT}(i|k+1)] \quad (3)$$

subject to the minimum LTIs given in Table I and $AC_{LL}(k)$, which stands for the last aircraft to be cleared to land during the k th operating interval.

- Step 4) Remove those aircraft with $t_{ALT}(i|k+1) \leq T_0(k+1) + T_{OI}$ from the original predicted landing sequence and sum up their airborne delay as

$$t_{\text{delay}}(k+1) = \sum_{i \in \Theta(k+1)} [t_{ALT}(i|k+1) - t_{PLT}(i|k+1)] \quad (4)$$

where $\Theta(k+1)$ is the set of serial numbers of those aircraft with $t_{ALT}(i|k+1) \leq T_0(k+1) + T_{OI}$.

- Step 5) Let $k = k + 1$. If $M_{AC}(k) = N_{AC}(k)$ and there is no aircraft with $t_{ALT}(i|k) > T_0(k) + T_{OI}$, implement the scheduled sequence and go to Step 6. Otherwise, go to Step 2.
- Step 6) Calculate the total airborne delay of the actual PS landing sequence as

$$T_{\text{delay}} = \sum_{i=1}^k t_{\text{delay}}(i). \quad (5)$$

Remark 1: There are many existing algorithms that can be modified and used as online optimizer to solve the minimization problem (3) (for example, see [10], [19], and [22]). Those algorithms are different in terms of computational efficiency and optimality. All those existing optimization algorithms for ASS can be embedded in the above scheduling procedure developed based on the RHC concept. Since the receding horizon with a length of N is much shorter than the whole operation period, as illustrated by the examples in Section IV, the computational burden can be significantly reduced. In this paper, the online

optimizer is based on the algorithm presented in [10] and described as follows:

- Step 1) Let $S_1(k+1) = \{[AC(1|k+1), t_{ALT}(1|k+1)]\}$ denote the initial feasible sequence, $U(k+1) = [AC(2|k+1), \dots, AC(M_{AC}|k+1)]$ denote the set of aircraft waiting to be inserted in $S_1(k+1)$, and set $n = 1$.
- Step 2) While $U(k+1) \neq \Phi$, i.e., $U(k+1)$ is not empty, do
- Step 2.1) Select an aircraft $AC(i|k+1) \in U(k+1)$ such that it can be inserted in the sequence related to $S_n(k+1)$ at the lowest increase of airborne delay; let h be the relative insertion position;
- Step 2.2) Insert $[AC(i|k+1), t_{ALT}(i|k+1)]$ at position h in the sequence related to $S_n(k+1)$, update the ALTs of other aircraft in $S_n(k+1)$, and let $S_{n+1}(k+1)$ be the new feasible schedule related to the new sequence;
- Step 2.3) Let $U(k+1) = U(k+1) - [AC(i|k+1)]$, i.e., remove the aircraft $AC(i|k+1)$ from $U(k+1)$, and $n = n + 1$.

Fig. 3 also gives the flow chart of the above optimizer for solving the minimization problem (3).

Remark 2: The receding horizon is the key technique to enable the algorithm to be not only computationally efficient but also robust in dynamic and uncertain environment. As illustrated in Fig. 4, the receding horizon can be considered as an optimization window sliding along the time axis. At the $(k-1)$ th operating interval, the optimization window is composed of the k th to the $(k+N-1)$ th operating interval. Those aircraft with PLTs within the optimization window are chosen to be sequenced and scheduled at the $(k-1)$ th operating interval. The first operating interval in the optimization window can be considered as a frozen window at the k th operating interval. When the optimized PS landing sequence is worked out, those aircraft with ALTs within the frozen window will be cleared to land at the runway. Then, at the k th operating interval, the optimization window recedes into the future by T_{OI} , a new set of aircraft waiting to land are considered, and the online optimization is repeated.

The successful design of the RHC algorithm depends on a proper choice of the length of receding horizon N . If N is too small, many useful information could be missed out, and therefore the RHC algorithm could be too shortsighted and exhibit poor performance. On the other hand, if N is too large, the computational burden will become heavy, and in addition, more information in the far future, which might not be very accurate or contain uncertainties, is used and could degrade the performance of the algorithm.

IV. SIMULATION RESULTS

A simulation system is set up based on [10] and then modified to randomly generate arrival flow at the airport with a certain degree of uncertainty. There are two arrival flows:

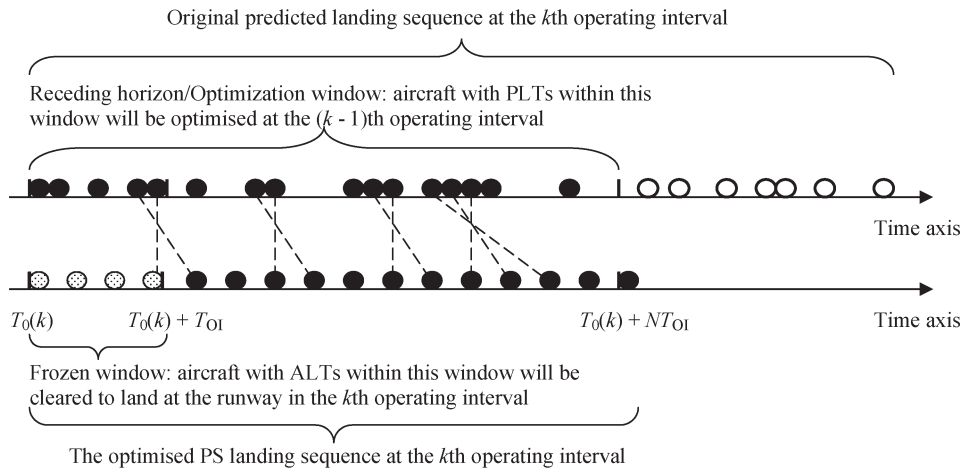


Fig. 4. Receding horizon/optimization window and frozen window.

predicted arrival flow (PAF) and actual arrival flow (AAF). The number of aircraft in a PAF is considered to be 60. The simulation system randomly allocates PLT for each individual aircraft in a PAF. All PLTs are within a specified time range $[T_0, T_0 + T_{\text{range}}]$, where T_{range} is a system parameter to adjust the degree of congestion at the airport. Based on a PAF, the simulation system continues to generate the associated AAF by introducing uncertainties randomly, where each aircraft may have another PLT different from that in the PAF, some aircraft may be cancelled, and some new aircraft may appear. Another two system parameters are used to set the level of uncertainties: P_1 , the percentage of aircraft that have different PLTs in an AAF from those in the associated PAF, and P_2 , the percentage of aircraft that are cancelled or new in the AAF. It is not the PAF but the AAF that turns up at the airport. Based on a PAF, the proposed RHC algorithm and a dynamic optimization algorithm (DOA), which employs the same online optimizer as in the RHC algorithm but optimizes the whole arrival flow, i.e., from current instant to the end of the planning, are used to schedule and sequence the arrival flow and each individual aircraft is given an ALT to land at the runway. Then, these ALTs are applied to the associated AAF. It will be no surprise that these ALTs worked out by the RHC or the DOA based on the PAF at each step probably do not work very well or are even infeasible for the corresponding AAF.

To fully investigate the performance of the proposed RHC algorithm, 12 simulation cases with different degrees of congestion and different levels of uncertainties are defined in Table II. Cases 1 to 3 are the cases where PAFs are the same as AAFs, i.e., $P_1 = 0$ and $P_2 = 0$, which imply that there are no uncertainties. Cases 4 to 12 are uncertain cases where AAFs are different from PAFs, with P_1 and P_2 indicating different levels of uncertainties. For each case, the simulation system randomly generates 100 pairs of PAFs and AAFs for the purpose of Monte Carlo simulations. In the simulation, the operating interval T_{OI} is set as 5 min, and in each of the 12 cases, the RHC is tested with seven different lengths of predictive horizon, i.e., $N = 1, \dots, 7$.

TABLE II
SIMULATION CASES (60 AIRCRAFTS IN THE ARRIVAL FLOW)

	Case 1	Case 4	Case 7	Case 10
$T_{\text{range}} = 50$ (minute)	$P_1 = 0, P_2 = 0$	$P_1 = 0.1,$ $P_2 = 0.1$	$P_1 = 0.2,$ $P_2 = 0.15$	$P_1 = 0.3,$ $P_2 = 0.2$
$T_{\text{range}} = 100$ (minute)	$P_1 = 0, P_2 = 0$	$P_1 = 0.1,$ $P_2 = 0.1$	$P_1 = 0.2,$ $P_2 = 0.15$	$P_1 = 0.3,$ $P_2 = 0.2$
$T_{\text{range}} = 150$ (minute)	$P_1 = 0, P_2 = 0$	$P_1 = 0.1,$ $P_2 = 0.1$	$P_1 = 0.2,$ $P_2 = 0.15$	$P_1 = 0.3,$ $P_2 = 0.2$

Table III gives the result of a single test in Case 1 under both the DOA and the RHC to illustrate how they schedule and sequence arrival traffic flow. Due to space limit, in this test the number of aircraft in the arrival flow is reduced from 60 to 30 and T_{range} changes to 25 min accordingly, so that the degree of congestion is the same as defined in Case 1. No conclusions on the performances of the DOA and the RHC can be made from Table III until Monte Carlo simulation results, which are presented in Figs. 5–7, are analyzed. Each curve in these three figures stands for the mean airborne delay of a single aircraft and the mean online computational time (OCT) of a single run of online optimization over the length of receding horizon, i.e., N , in a certain case. For $N = 1$ to 7, the simulation is conducted under the RHC with a receding horizon of NT_{OI} minutes long; while for $N = \text{“DOA,”}$ the simulation is under the DOA. Figs. 5–7 are related to different degrees of congestion at the airport, respectively. Four line and point styles are used to distinguish different levels of uncertainties.

From the Monte Carlo simulation results shown in Figs. 5–7, some interesting observations can be made as follows on the nature of the ASS problem and the performance of the DOA and the RHC:

- It is evident that the degree of congestion at the airport is a key factor that results in airborne delays. More congested, more delays, no matter which method, the RHC or the DOA, is used. In Fig. 5, the most congested case, the average airborne delay is about 1200 s for each aircraft,

TABLE III
RESULT OF A SINGLE TEST

PAF			Scheduled Traffic Under DOA				Scheduled Traffic Under RHC			
Air . No.	Cat.	PLT (second)	Air . No.	Cat.	ALT (second)	Delay (second)	Air . No.	Cat.	ALT (second)	Delay (second)
1	4	17	1	4	17	0	1	4	17	0
2	2	50	2	2	97	47	2	2	97	47
3	1	117	4	3	167	38	4	3	167	38
4	3	129	7	3	237	20	7	3	237	20
5	4	143	9	3	307	19	9	3	307	19
6	1	174	3	1	379	262	6	1	379	205
7	3	217	12	2	579	74	8	1	475	215
8	1	260	13	2	659	133	3	1	571	454
9	3	288	5	4	769	626	16	3	752	162
10	4	340	16	3	839	249	20	3	822	131
11	4	350	20	3	909	218	22	3	892	130
12	2	505	22	3	979	217	12	2	992	487
13	2	526	8	1	1051	791	13	2	1072	546
14	4	541	18	1	1147	496	19	2	1152	483
15	4	554	21	1	1243	536	24	2	1232	440
16	3	590	26	1	1339	346	27	2	1312	248
17	4	648	6	1	1435	1261	10	4	1422	1082
18	1	651	30	1	1531	130	11	4	1512	1162
19	2	669	19	2	1731	1062	14	4	1602	1061
20	3	691	24	2	1811	1019	15	4	1692	1138
21	1	707	27	2	1891	827	17	4	1782	1134
22	3	762	11	4	2001	1651	23	4	1872	1103
23	4	769	14	4	2091	1550	25	4	1962	1103
24	2	792	15	4	2181	1627	28	4	2052	975
25	4	859	17	4	2271	1623	29	4	2142	935
26	1	993	23	4	2361	1592	5	4	2232	2089
27	2	1064	25	4	2451	1592	18	1	2304	1653
28	4	1077	28	4	2541	1464	21	1	2400	1693
29	4	1207	29	4	2631	1424	30	1	2496	1095
30	1	1401	10	4	2721	2381	26	1	2592	1599

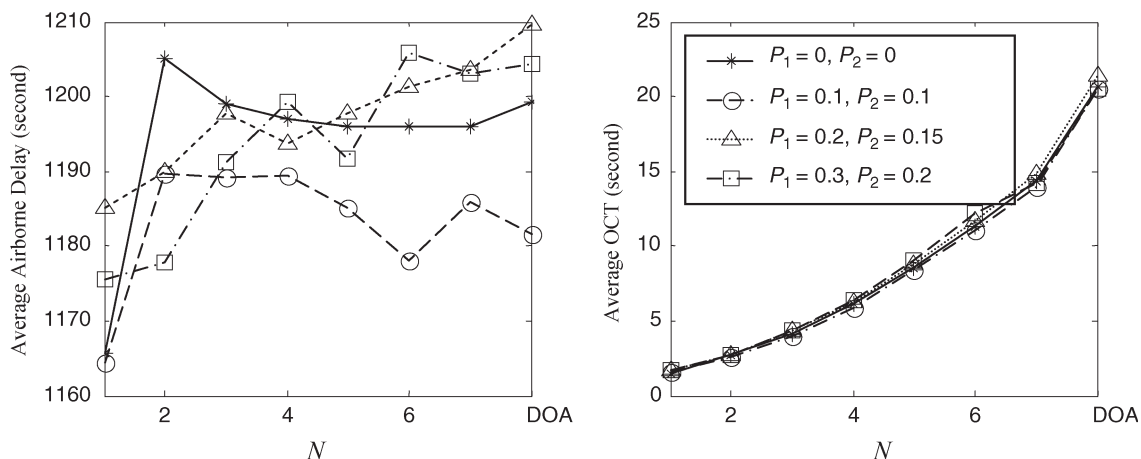


Fig. 5. Monte Carlo simulation results, $T_{range} = 50$ min.

while in Fig. 6, it is about 270 s, and in Fig. 7, the least congested case, it is about 90 s.

- Regarding the performance in terms of airborne delay, generally speaking, the new procedure based on RHC achieves the same performance as or better than the DOA. How much improvement the RHC could bring depends on the degree of congestion at the airport, the level of uncertainties, and the receding horizon. It should be noted that since the ASS is an NP complete problem and the opti-

mization method adopted in this paper is a semisearching method, only local minimum is attained and the degree of optimality the optimizer can achieve depends on many factors such as the size of solution space and the nature of the optimization. This obviously has influence on the performance of the ASS algorithms. The same optimizer is employed for the DOA and the RHC developed in this paper. The airborne delays in Figs. 5–7 are obtained by taking the average of over 100 Monte Carlo simulations.

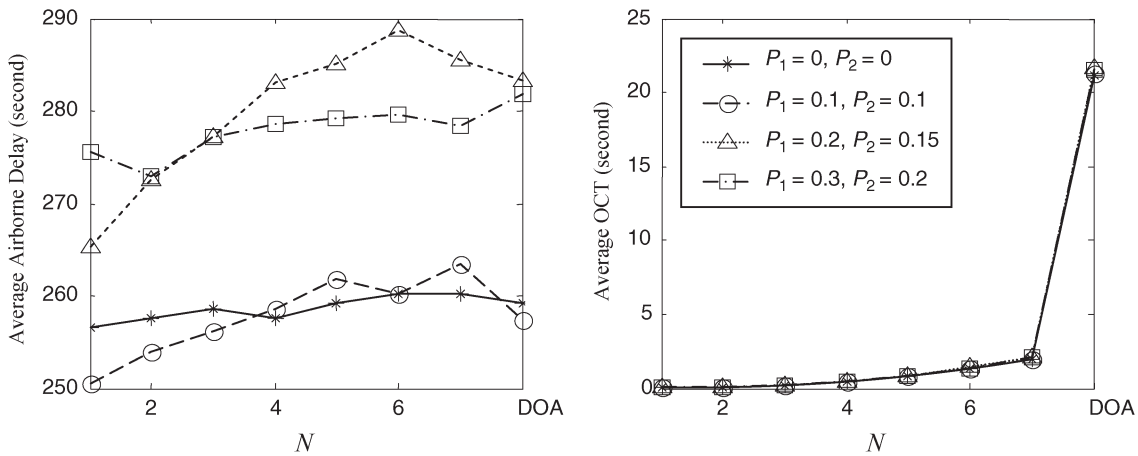


Fig. 6. Monte Carlo simulation results, $T_{range} = 100$ min.

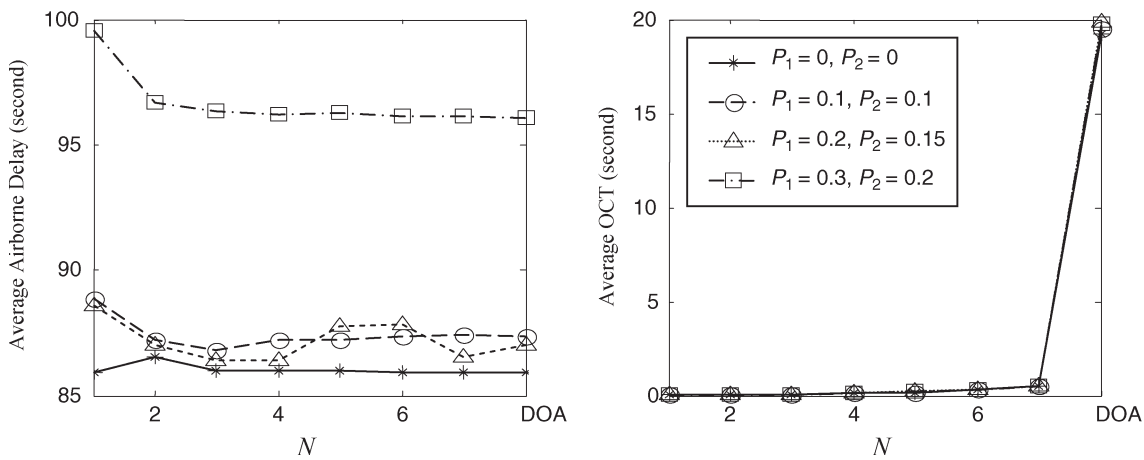


Fig. 7. Monte Carlo simulation results, $T_{range} = 150$ min.

- The influence of the length of receding horizon, i.e., N , is an important parameter in the RHC. In general, a small receding horizon leads to shortsighted behavior and, consequently, poor performance. However, if the receding horizon is too long, the information in the far future that may contain uncertain or inaccurate information is included in the online optimization, and the computational burden is also significantly increased. Figs. 5–7 indicate that the receding horizon with a length of one or two is enough for the ASS. After that, increasing the receding horizon does not significantly increase the performance. This implies that for ASS, one or two receding horizons are sufficient and it is not necessary to solve the ASS problem using existing DOAs as advocated in [7], [8], [10], and [11].
- As expected, in general, the computational time increases with the length of the receding horizon and the DOA takes much more time than all the schemes based on the RHC. When the receding horizon with a length of one or two is employed, the computational time of the new algorithm is only a fraction of that of the existing dynamic optimization approach. Furthermore, the maximum online

computational time under the DOA is observed to be over 100 s, while for most RHC schemes (except those with receding horizon longer than three operating intervals in Fig. 5) it is just under 10 s. In the above Monte Carlo simulation, the number of aircraft in arrival flow is just 60, and only four categories of aircraft are used to define LTIs, as given in Table I. It is expectable that the advantage of the RHC against the DOA, regarding real-time properties, will become more significant when more aircraft and more categories are considered. On the other hand, the high efficiency of the RHC potentially enables some optimization algorithms that have prohibitive computational burdens for current DOAs, such as exhaustive searching algorithm, be integrated as online optimizer for real-time implementations to further improve performance.

V. CONCLUSION

The concept of RHC is introduced in this paper to solve the problem of arrival scheduling and sequencing at a busy hub airport. The performance of the proposed new real-time arrival scheduling and sequencing method is investigated under

different degrees of congestion, different levels of uncertainties, and using different receding horizons for optimization. The results indicate that the new method with a short receding horizon can achieve similar (or better) performance in terms of airborne delay as a DOA that takes into account the whole planning time period. In general, it was found that for the arrival scheduling and sequencing problem, receding horizon with a length of one or two in the RHC-based approach is enough. This significantly reduces computational burden and thus optimization time, and enables to provide decision making support for ASS under dynamic and uncertain ATC environments. Furthermore, since the ASS problem is an NP complete optimization, it is difficult to obtain the global optimum. By significantly reducing the time period for optimization using the RHC concept, the solution space is dramatically reduced. This gives the potential to reduce airborne delay and improve computational efficiency. Some optimization methods such as exhaustive searching might be impossible for the whole ASS planning time period but might be possible in the receding horizon fashion with a length of one or two.

REFERENCES

- [1] M. Pelegrin, "Towards global optimization for air traffic management," AGARD-AG-321, 1994.
- [2] C. C. Gregory, H. Erzberger and F. Neuman, "Delay exchanges in arrival sequencing and scheduling," *J. Aircr.*, vol. 36, no. 5, pp. 785–791, Sep.–Oct. 1999.
- [3] —, "Fast-time study of aircraft-influenced arrival sequencing and scheduling," *J. Guid. Control Dyn.*, vol. 23, no. 3, pp. 526–531, 2000.
- [4] G. Andreatta and G. Romanin-Jacur, "Aircraft flow management under congestion," *Trans. Sci.*, vol. 21, no. 4, pp. 249–253, 1987.
- [5] L. Bianco and M. M. Bielli, "System aspects and optimization models in ATC planning," in *Large Scale Computation and Information Processing in Air Traffic Control*, L. Bianco and A. R. Odoni, Eds. Berlin: Springer-Verlag, 1993, pp. 47–99.
- [6] R. G. Dear, *The Dynamic Scheduling of Aircraft in the Near Terminal Area*, FLT R76.9, Flight Transportation Laboratory. Cambridge, MA: MIT Press, 1976.
- [7] H. N. Psaraftis, *A Dynamic Programming Approach to the Aircraft Sequencing Problem*, FTL R78-4, Flight Transportation Laboratory. Cambridge, MA: MIT Press, 1978.
- [8] —, "A dynamic programming approach for sequencing identical groups of jobs," *Oper. Res.*, vol. 28, no. 6, pp. 1347–1359, 1980.
- [9] C. S. Venkatakrisnan, A. Barnett and A. M. Odoni, "Landings at Logan airport: Describing and increasing airport capacity," *Transp. Sci.*, vol. 27, pp. 211–227, 1993.
- [10] L. Bianco, P. Dell'Olmo and S. Giordani, "Scheduling models and algorithms for TMA traffic management," in *Modelling and Simulation in Air Traffic Management*, L. Bianco, P. Dell'Olmo and A. R. Odoni, Eds. Berlin: Springer-Verlag, 1997, pp. 139–167.
- [11] L. Bianco, G. Rinaldi and A. Sassano, "Scheduling task with sequence-dependent processing times," *Nav. Res. Logist.*, vol. 35, pp. 177–184, 1988.
- [12] D. W. Clarke, *Advances in Model-Based Predictive Control*. London, U.K.: Oxford Univ. Press, 1994.
- [13] B. De Schutter and T. van den Boom, "Model predictive control for max-plus-linear discrete event systems," *Automatica*, vol. 37, no. 7, pp. 1049–1056, Jul. 2001.
- [14] T. Van Den Boom and B. De Schutter, "Properties of MPC for max-plus-linear systems," *Eur. J. Control*, vol. 8, no. 5, pp. 453–462, 2002.
- [15] S. Chand, V. N. Hsu and S. Sethi, "Forecast, solution, and rolling horizons in operations management problems: A classified bibliography," *Manuf. Serv. Oper. Manag.*, vol. 4, no. 1, pp. 25–43, 2002.
- [16] W. H. Chen, T. van den Boom and B. De Schutter, "Discussion on: 'Properties of MPC for max-plus-linear systems' by T. van den Boom and B. De Schutter," *Eur. J. Control*, vol. 8, pp. 463–464, 2002.
- [17] H. Erzberger, "Design principles and algorithms for automated air traffic management," in *AGARD Lecture Series No. 200: Knowledge-Based Functions in Aerospace Systems*, Madrid, Paris, and San Francisco, Nov. 1995.
- [18] F. Neuman and H. Erzberger, "Analysis of delay reducing and fuel saving sequencing and spacing algorithms for arrival traffic," Ames Res. Ctr., Moffett Field, CA, NASA Tech. Memo. 103880, Oct. 1991.
- [19] J. E. Beasley, J. Sonander and P. Havelock, "Scheduling aircraft landings at London Heathrow using a population heuristic," *J. Oper. Res. Soc.*, vol. 52, no. 5, pp. 483–493, May 2001.
- [20] J. E. Beasley, M. Krishnamorthy, Y. M. Sharaiha and D. Abramson, "Displacement problem and dynamically scheduling aircraft landings," *J. Oper. Res. Soc.*, vol. 55, no. 1, pp. 54–64, Jan. 2004.
- [21] F. Schick, "Methods and measurements for the evaluation of ATM tools in real-time simulations and field tests," in *2nd USA/EUROPE Air Traffic Management R&D Seminar*, Orlando, FL, 1998, pp. 1–9.
- [22] J. E. Robinson, III, T. J. Davis and D. R. Issacson, "Fuzzy reasoning-based sequencing of arrival aircraft in the terminal area," in *AIAA Guidance, Navigation and Control Conf.*, New Orleans, LA, Aug. 1997, pp. 422–433.



Xiao-Bing Hu was born in Sichuan Province, China, on June 15, 1975. He received the B.S. degree in aviation electronic engineering from Civil Aviation Institute of China, Tianjin, China, in 1998, and the M.S. degree in automatic control engineering from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2001.

He is currently doing a Ph.D. study at Aeronautical and Automotive Engineering Department, Loughborough University, U.K. His major field of study includes predictive control, artificial intelligence, air traffic management, and flight control.



Wen-Hua Chen (M'00) received the M.Sc. and Ph.D. degrees from Northeast University, China, in 1989 and 1991, respectively.

From 1991 to 1996, he was a Lecturer at the Department of Automatic Control, Nanjing University of Aeronautics and Astronautics, Nanjing, China. He held a research position and then a Lectureship in Control Engineering in Center for Systems and Control, University of Glasgow, U.K., from 1997 to 2000. He holds a Lectureship in Flight Control Systems in Department of Aeronautical and Automotive

Engineering at Loughborough University, U.K. He has published one book and more than 60 papers on journals and conferences. His research interests are the development of advanced control strategies and their applications in aerospace engineering.