A Dynamic Programming Algorithm for Robust Runway Scheduling

Bala Chandran and Hamsa Balakrishnan

***Abstract*— An algorithm for generating schedules of airport runway operations that are robust to perturbations caused by system uncertainty is presented. The algorithm computes a tradeoff curve between runway throughput and the probability that random deviations of aircraft from the schedule violate system constraints and require intervention from air traffic controllers. The algorithm accommodates various operational constraints imposed by the terminal-area system such as minimum separation requirements between successive aircraft, earliest and latest times for each aircraft, precedence constraints among aircraft and the limited flexibility in deviating from the First-Come-First-Served (FCFS) order afforded to air traffic controllers (a concept known as Constrained Position Shifting). When the maximum allowable number of position shifts from the FCFS order is bounded by a constant, the complexity of the algorithm is $O(n(L/\epsilon)^3)$, where n is the number of aircraft, L is largest difference between the latest and earliest arrival time over all aircraft, and ϵ is the desired output accuracy.**

I. Introduction

The safe and efficient planning of airport operations are an important part of the responsibility borne by the Air Traffic Control (ATC) system. As aircraft arrive at the boundaries of the Air Route Traffic Control Centers (ARTCCs or Centers), air traffic controllers have a short period of time (about 45 minutes) to determine the landing times and positions of aircraft in the landing sequence, and to also issue the appropriate control actions necessary to obtain the sequence [1,2]. Similar challenges are also faced by controllers who are responsible for scheduling departure runways at airports.

Controllers also have to contend with various forms of uncertainty in the system caused by weather effects such as winds, the limitations imposed by the precision of onboard equipment, as well as the uncertainty in pushback times and taxi times for departing aircraft. The presence of uncertainty in the system motivates the development of robust schedules for runway operations. The notion of robustness is one that can be defined in several ways. In the context of aircraft arrival and departure sequences, the uncertainty in the system could result in the aircraft violating important safety constraints, thereby necessitating re-sequencing on the part of the air traffic controllers. For this reason, we consider a runway sequence robust if there is a sufficiently high probability that an air traffic controller does not have to intervene once the schedule has been determined.

Runway schedules must satisfy the operational constraints that are imposed by the system. In this paper, we present a technique to determine robust arrival and departure schedules that can potentially improve runway productivity, while still satisfying the various constraints required of any practical solution. We show that the proposed method is computationally efficient, with complexity that scales linearly with the number of aircraft, and as the cube of the largest difference between the latest and earliest arrival time over all aircraft.

The algorithm proposed in this paper is based on dynamic programming, using concepts from the algorithm for scheduling arrival flows that was proposed in our earlier work [3] for deterministic environments. However, the output of the algorithm is not a single schedule, but a tradeoff between the likelihood of controller intervention (robustness) and the time to complete runway operations for the given set of aircraft (makespan of the sequence). The technique gives system designers the ability to set the appropriate threshold that determines the tradeoff between robustness and efficiency. The proposed algorithm can also be used to assess broader policy measures such as the benefit (in terms of throughput and safety) of introducing onboard or ground-based systems to decrease the uncertainty in the system.

II. Problem Description

The goal of algorithms for scheduling runway operations is to increase the throughput of the runway system while still satisfying the various safety and operational constraints of the system. In this paper, we primarily use the example of scheduling arrivals at a runway, but the techniques described can also be utilized for departure runway scheduling.

A. Constraints

1) Minimum separation requirements: The primary constraint that air traffic controllers need to ensure in an arrival sequence is that the inter-arrival spacings equal or exceed the minimum requirements specified by the Federal Aviation Administration (FAA). For reasons of safety, it is necessary that an arriving aircraft does not face interference from the wake-vortex of the aircraft landing in front of it. The risk posed by the wake vortex depends on the sizes of both the leading and trailing aircraft; therefore, the required time interval between two landings depends on the sizes of the two aircraft. Similarly, separation is required between departures, and between arrival and departure operations [4].

The most common approach to sequencing aircraft has been to maintain the First-Come-First-Served (FCFS) order [4], under which aircraft utilize the runway in order of their estimated arrival times at the runway, and air traffic controllers only enforce the minimum separation requirements. The FCFS schedule is easy to implement, reduces controller

Bala Chandran is with the Department of Industrial Engineering and Operations Research, University of California at Berkeley. chandran@ieor.berkeley.edu

Hamsa Balakrishnan is with the Department of Aeronautics and Astronautics, and Engineering Systems Division, Massachusetts Institute of Technology. hamsa@mit.edu

workload, and maintains a sense of fairness but may lead to reduced runway throughput due to large spacing requirements. This motivates deviating from the FCFS sequence to achieve schedules that increase runway throughput.

2) Limited flexibility: The terminal area is an extremely dynamic environment, and re-sequencing aircraft increases the workload of controllers. Due to limited flexibility, it might not be possible for air traffic controllers to implement an efficient sequence that deviates significantly from the FCFS order. This is the basic motivation for Constrained Position Shifting (CPS) methods. CPS, first proposed by Dear [5], stipulates that an aircraft may be moved up to a specified maximum number of positions from its FCFS order. We denote the maximum number of position shifts allowed as k ($k \leq 3$ for most runway systems), and the resulting environment as a k-CPS scenario. For example, in 2-CPS, an aircraft that is in the 8^{th} position in the FCFS order can be placed at the 6^{th}, 7^{th}, 8^{th}, 9^{th}, or 10^{th} position in the new order. The restricted deviation from the FCFS order helps maintain equity among aircraft operators, and also increases the predictability of landing times. A detailed description of CPS can be found in our prior work [3].

3) Time-windows: While determining a schedule for the runway, controllers need to account for the possible times that an aircraft can utilize the runway. In the case of the scheduling of aircraft landings, these times will be the possible arrival times at the runway, corresponding to different controller requests to the aircraft. There is typically an earliest time at which the aircraft can reach the runway, as well as a latest time [1]. In the case of departure runway scheduling, these could be the result of traffic flow management strategies, such as Ground Delay Programs at destination airports, during which aircraft at origin airports are assigned departure time windows [6]. In general, an aircraft's runway time of arrival could lie in any one of a number of disjoint time-intervals [3].

4) Precedence relations: It is also necessary to consider precedence constraints, which are those of the form "Aircraft i must land before aircraft j". Such constraints are important because in current ATC automation systems overtaking is limited [1]; in addition, airlines themselves may have precedence preferences, arising from their banking strategies [7].

B. Uncertainty

In prior work [3], we presented an algorithm to compute the optimal sequence of runway operations, subject to the constraints outlined above, in a deterministic environment. However, the presence of uncertainty results in perturbed schedules, with the aircraft no longer landing at the intended landing times. This lack of precision can lead to the violation of the minimum separation requirements between aircraft, and require intervention by air traffic controllers to enforce the safety minimums. The degree to which an aircraft is likely to be perturbed from its scheduled arrival time at the runway depends on the equipage of the aircraft. For example, aircraft with precise Flight Management Systems (FMS) are likely to be more accurate in meeting their scheduled times than aircraft which are less equipped [8,9,10]. Estimates of runway arrival time accuracy when predicted at the metering fix have also been measured for different aircraft types and levels of equipage [9,11].

The benefits of improved accuracy of arrivals were studied by Meyn and Erzberger [9], who used stochastic simulations for FCFS sequencing with parallel runway reassignments. The likelihood of spacings being violated due to uncertainty was reduced by adding a buffer to the minimum inter-arrival separation requirements, and then solving the deterministic problem. This form of buffering is useful if all aircraft separations were buffered by some fixed fraction. However, we would like to solve the more difficult case in which all aircraft are not equally equipped (mixed equipage), and the uncertainty associated with meeting the scheduled times of arrival is not the same for all of them. In such situations, buffering all aircraft could lead to sub-optimal solutions.

Most prior research on the accuracy of aircraft arrivals make the simplifying assumption that the inter-arrival spacings in a sequence of aircraft landings are independent of each other [9,12]. However, this assumption does not hold true in practice, and the exact distribution of the inter-arrival spacing between two aircraft is dependent on the inter-arrival spacing between all pairs of aircraft that preceded them. However, it can be shown that when there is a substantial difference in the accuracy of equipped and non-equipped aircraft, it is sufficient to consider all preceding spacings until the closest equipped aircraft in the sequence. In this paper, we assume that the inter-arrival spacing depends only on the immediately preceding inter-arrival time. We believe that this will be a reasonable approximation in the future system, where incentives to equip will result in at least 50% of aircraft being equipped, especially in congested areas.

There is also uncertainty associated with the aircraft being able to be present at the runway at a particular time. This implies that instead of time-windows representing the times when an aircraft can utilize the runway, there is a distribution representing the probability that an aircraft can use the runway at a particular time (for example, the probability that an aircraft can land at a particular time, in the case of arrival flows into an airport). This would reflect the impact of weather on traffic and pilot behavior [13].

C. Robustness and reliability

There are several possible definitions of the robustness or reliability of a schedule for runway operations. For example, airlines schedule their flights in major hubs such that passengers from a bank of arriving flights connect to (one or more) departing flights. In such situations, airlines prioritize their flights, and reliability is measured by the degree to which aircraft maintain their order with respect to other aircraft in the same bank, and not on the landing times [7]. Precedence relations can account for this form of airline prioritization. The more adverse effect of uncertainty from an air traffic control perspective is the violation of minimum separation requirements [9]. The violation of these spacing constraints means that an air traffic controller has to intervene to enforce spacing between the two aircraft involved. This in turn may affect the schedule of all the aircraft that follow, requiring interventions to readjust the scheduled landing times of subsequent aircraft in the sequence. Given a sequence of aircraft, the reliability of a schedule can be measured in terms of the probability that none of the inter-aircraft spacing constraints will be violated.

Let "$t_i \leftrightarrow t_j$" represent the event that the minimum spacing between two aircraft i and j (denoted δ_{ij}) will *not* be violated given that i is scheduled to land at t_i and j is scheduled to land at t_j. If the scheduled arrival times are denoted $s(\cdot)$ and the actual landing times are denoted $a(\cdot)$, then
$t_i \leftrightarrow t_j \Rightarrow \{a(j) \geq a(i) + \delta_{ij} | \, s(i) = t_i \wedge s(j) = t_j\}$.

Given a sequence of aircraft $\{i_1, \ldots, i_n\}$ with corresponding scheduled arrival times $\{t_{i_1}, \ldots, t_{i_n}\}$, we define the *reliability* of the schedule, denoted by $\mathrm{R}(t_{i_1}, \ldots, t_{i_n})$, as the probability that none of the spacing requirements is violated.

$$\begin{aligned}\mathrm{R}(t_{i_1}, \cdots, t_{i_n}) &= \Pr\{t_{i_1} \leftrightarrow t_{i_2} \wedge t_{i_2} \leftrightarrow t_{i_3} \wedge \cdots \wedge t_{i_{n-1}} \wedge t_{i_n}\} \\ &= \Pr\{t_{i_{n-1}} \leftrightarrow t_{i_n} | \, t_{i_1} \leftrightarrow t_{i_2} \wedge \cdots \wedge t_{i_{n-2}} \leftrightarrow t_{i_{n-1}}\} \\ &\quad \times \Pr\{t_{i_1} \leftrightarrow t_{i_2} \wedge \cdots \wedge t_{i_{n-2}} \leftrightarrow t_{i_{n-1}}\}.\end{aligned}$$

As explained in Section II-B, we assume that the inter-arrival spacing between any pair of aircraft is conditionally independent of the past history of arrivals, given the inter-arrival spacing of the immediately preceding pair. In other words, $\Pr\{t_{i_{n-1}} \leftrightarrow t_{i_n} | t_{i_1} \leftrightarrow t_{i_2} \wedge \cdots \wedge t_{i_{n-2}} \leftrightarrow t_{i_{n-1}}\} = \Pr\{t_{i_{n-1}} \leftrightarrow t_{i_n} | t_{i_{n-2}} \leftrightarrow t_{i_{n-1}}\}$. This can be used to show that the reliability of a sequence can be expressed as follows.

$$\begin{aligned}\mathrm{R}(t_{i_1}, \cdots, t_{i_n}) &= \Pr\{t_1 \leftrightarrow t_2\} \times \Pr\{t_2 \leftrightarrow t_3 \mid t_1 \leftrightarrow t_2\} \times \cdots \\ &\quad \cdots \times \Pr\{t_{i_{n-1}} \leftrightarrow t_{i_n} \mid t_{i_{n-2}} \leftrightarrow t_{i_{n-1}}\} \qquad (1)\end{aligned}$$

The two objectives of increasing throughput (or minimizing makespan) and increasing reliability are conflicting: it is possible to propose a sequence with very large buffers in inter-aircraft separations to obtain a runway schedule that was very robust but would take a long time to complete; similarly, the most efficient (deterministic) schedule would maintain inter-aircraft spacings as close to the minimums as possible, but would be very sensitive to uncertainty. The technique we propose helps us determine the tradeoff contours between reliability and throughput for the runway operations scheduling problem. In the context of a constrained optimization problem with uncertain inputs, a robust solution is defined as one that has low likelihood of violating the constraints while being acceptably close to optimal [14]. In our case, given an upper bound on the makespan, a robust schedule is one that maximizes reliability.

D. Problem statement

We define the minimum time-separation matrix by Δ, where the element δ_{ij} is the minimum required time between runway operations, if aircraft i lands before aircraft j. Currently, these classes are defined based on the maximum take-off weight for scheduling runway operations, but could be generalized to other classifications as well. In this paper, we assume that the separations satisfy the triangle inequality, that is, $\delta_{ik} \leq \delta_{ij} + \delta_{jk} \; \forall i, j, k$. This condition is satisfied by current separation minimums [4].

We represent precedence relations by an $n \times n$ matrix $\{m_{ij}\}$, such that element $m_{ij} = 1$ if aircraft i *must* land before aircraft j, and $m_{ij} = 0$ otherwise.

We identify two different forms of uncertainty:

1) For every aircraft i, the probability $\mathrm{Pr}_i(t)$ represents the likelihood that i can utilize the runway at time t.
2) For every aircraft i, we also consider the distribution $\mathrm{Pr}_i(t|t_i)$, which is the probability that aircraft i lands at time t given that it was scheduled to land at time t_i. This distribution reflects the accuracy of the aircraft navigation system, and the effect of uncertainty on an aircraft's schedule. We denote the probability density function (p.d.f.) of this distribution as $f_i(t|t_i)$.

Consolidating our objective and constraints, we can pose the following problem:
Given n aircraft indexed $1, \cdots, n$, probability distribution $\mathrm{Pr}_i(t)$ over the times at which aircraft i can land, separation matrix Δ, precedence matrix $\{p_{ij}\}$, the maximum number of position shifts k, and the p.d.f. $f_i(t|t_i)$ for the delivery accuracy of the aircraft at the runway, compute the k-CPS sequence and corresponding times of runway utilization that minimize the makespan of the sequence, while satisfying the minimum level of reliability. Alternatively, compute the runway utilization schedule that maximizes the level of reliability, while possessing a makespan that is less than a specified maximum value. The solution to this problem allows us to determine the tradeoff between reliability and throughput for the system.

For simplicity, we assume that the aircraft are labeled $(1, 2, \cdots, n)$, according to their position in the FCFS sequence. We also note that given any three consecutive aircraft in the sequence (a–b–c), and their arrival time error distributions $f_a(t|t_a)$, $f_b(t|t_b)$ and $f_c(t|t_c)$, it is possible to compute the probability distributions for $\Pr\{t_a \leftrightarrow t_b\}$, $\Pr\{t_b \leftrightarrow t_c\}$ and $\Pr\{t_b \leftrightarrow t_c \mid t_a \leftrightarrow t_b\}$. Due to limitations of space, this paper is restricted to the case where $\mathrm{Pr}_i(t) = 1$ for all $t \in \mathcal{I}(i)$, the set of times during which aircraft i is allowed to land. The proposed technique can be extended quite easily to more general distributions for $\mathrm{Pr}_i(t)$.

III. Dynamic programming algorithm

In prior work [3], we demonstrated that every k-CPS sequence can be represented as a path in a directed graph whose size is polynomially bounded in n and k. We briefly describe the structure of this network and its properties.

A. The CPS network

The network consists of n *stages* $\{1, \cdots, n\}$, where each stage corresponds to an aircraft position in the final sequence. A node in stage p of the network represents a subsequence of aircraft of length $\min\{2k+1, p\}$ where k is the maximum position shift. For example, for $n = 5$ and $k = 1$, the nodes in stages $3, \cdots, 5$ represent all possible sequences of length $2k + 1 = 3$ ending at that stage. Stage 2 contains a node for every possible aircraft sequence of length 2 ending at position 2, while stage 1 contains a node for every possible sequence of length 1 starting at position 1. This network, shown in Figure 1, is obtained using all possible aircraft assignments to each position in the sequence (given below).

Position	1	2	3	4	5	6
Possible aircraft assignments	1	1	2	3	4	5
	2	2	3	4	5	6
		3	4	5	6	

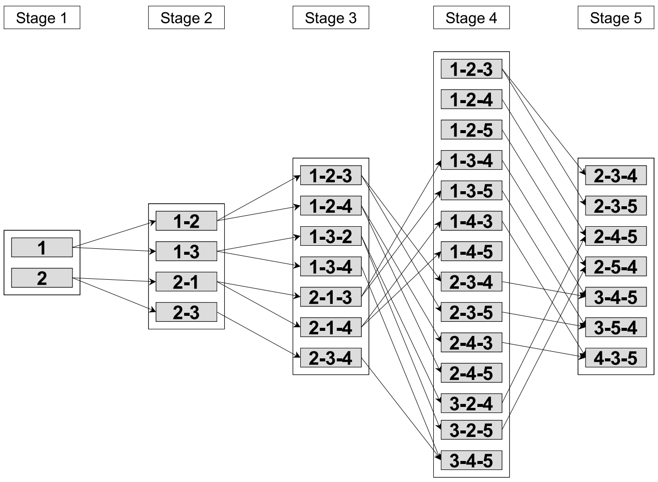

Fig. 1. Network for $n = 5$, $k = 1$.

For convenience, we refer to the last aircraft in a node's sequence as the *final aircraft* of that node. For each node in stage p, we draw directed arcs to all the nodes in stage $p+1$ that can follow it. For example, a sequence (1–2–3) in stage 3 can be followed by the sequences (2–3–4) or (2–3–5) in stage 4. This results in a network where every directed path from a node in stage 1 to one in stage n represents a possible k-CPS sequence. For example, the path (2)→(2–1)→(2–1–3)→(1–3–5)→(3–5–4) represents the sequence 2–1–3–5–4.

Nodes such as (1–2–4) in stage 4 that cannot belong to a path from stage 1 to stage n are removed from the network. Finally, nodes that violate precedence constraints are also eliminated to generate a "pruned" network that may be significantly smaller than the original network. The key properties of this network, as shown in [3], but stated somewhat differently here are as follows.

(i) Every possible k-CPS subsequence of length $2k+1$ or less is contained in some node of the network.

(ii) Every feasible sequence (one that satisfies maximum position shift constraints and precedence constraints) can be represented by a path in the network from a node in stage 1 to a node in stage n.

(iii) Every path in the network from a node in stage 1 to a node in stage n represents a feasible k-CPS sequence.

B. Dynamic programming recursion

We use the following notation.

- $\ell(x)$ The last (final) aircraft of node x.
- $\ell'(x)$ The second from last aircraft of node x.
- $P(x)$ Set of nodes that precede x. (A node w is said to precede x if arc (w,x) exists).
- $\mathcal{I}(j)$ Set of times during which aircraft j can land.

Let $J_x(t_1, t_2)$ be the maximum reliability of a sequence starting in stage 1 and ending in node x, given that $\ell(x)$ is scheduled to land at time t_2 and $\ell'(x)$ is scheduled to land at time t_1. The reliability of the sequence is as defined in Equation 1. We would like to compute the value of $J(\cdot)$ for all nodes in stage n.

Lemma 1: The values of $J(\cdot)$ are correctly computed by the following recursion:

$$J_y(t_{\ell'(y)}, t_{\ell(y)}) = \max_{x \in P(y)} \max_{t_{\ell'(x)} \in \mathcal{I}(\ell'(x))} \big\{ J_x(t_{\ell'(x)}, t_{\ell(x)}) \times \Pr\{t_{\ell(x)} \leftrightarrow t_{\ell(y)} \mid t_{\ell'(x)} \leftrightarrow t_{\ell(x)}\}\big\},$$
$$\forall\ t_{\ell(y)} \in \mathcal{I}(\ell(y)) : t_{\ell(y)} \geq t_{\ell(x)} + \delta_{\ell(x),\ell(y)}.$$

Proof: The proof follows standard techniques for proving the validity of dynamic programming recursions, and is presented in the appendix for completeness. ■

We can now compute the value if $J_y(\cdot)$ for each node in stage n by unrolling the recursion using the boundary condition $J_x(t_{\ell'(x)}, t_{\ell(x)}) = \Pr\{\ell'(x) \leftrightarrow \ell(x)\}$ for every node x in stage 2 and for all $t_{\ell'(x)} \in \mathcal{I}(\ell'(x))$ and $t_{\ell(x)} \in \mathcal{I}(\ell(x))$.

C. Algorithm

Since the state space for $J(\cdot)$ is infinite, the recursion as such is computationally not practical. Therefore, we discretize all times into periods of length ϵ. In practice, the accuracy of measurements in the airspace is of the order of seconds, so an ϵ value between 1 and 10 sec is reasonable.

At the end of this procedure, the values of J for all nodes in stage n are obtained for all feasible time periods. The maximum reliability sequence for a given makespan t is the maximum over all $J_x(t_{\ell'(x)}, t_{\ell(x)})$ for $t_{\ell(x)} = t$. This value can be computed for all periods of interest to generate a curve that trades off makespan against reliability. The corresponding schedule can be recovered by keeping track of the argument of the maximization during the algorithm.

D. Complexity

We had shown in [3] that the number of nodes in the network is $O(n(2k+1)^{(2k+1)})$, and the number of arcs is $O(n(2k+1)^{(2k+2)})$.

The algorithm loops through 3 time intervals (corresponding to three aircraft) for each arc in the network. Given a period length of ϵ, the total work done throughout the algorithm is $O((L/\epsilon)^3)$ per arc where L is the length of the largest interval $\mathcal{I}(\cdot)$ among all aircraft. In practice, the value of the maximum position shift parameter k is usually 1, 2, or 3, so the terms in k can be regarded as a constant.

Lemma 2: The complexity of the proposed dynamic programming algorithm is $O(n(L/\epsilon)^3)$, where n is the number of aircraft, L is the largest difference between the latest and earliest arrival times over all aircraft, and ϵ is the desired output accuracy.

Since there are relatively few types of aircraft, the probabilities $\Pr\{\ell'(x) \leftrightarrow \ell(x)\}$ and $\Pr\{t_{\ell(x)} \leftrightarrow t_{\ell(y)} \mid t_{\ell'(x)} \leftrightarrow t_{\ell(w)}\}$ can be computed (either through a simulation or analytically depending on the distribution) and stored offline. The work done to compute these probabilities needs to be done only once, and hence is not part of the complexity expression.

IV. Examples

We consider the example of scheduling aircraft landings on a single runway. The times at which aircraft cross the Center boundaries are generated using a Poisson distribution. Jet routes are assigned based on traffic flow statistics and determine the precedence relations, since aircraft along the same jet route are not allowed to overtake each other. Since the runway schedules are determined when the aircraft cross the Center boundary using a nominal trajectory, there is considerable inaccuracy in an aircraft meeting its scheduled landing time. We model the distribution of the error (that is, the difference between actual landing time and the scheduled landing time) as a triangular distribution, with a range of $\pm 300\,\mathrm{sec}$ for aircraft not equipped with an FMS, and $\pm 150\,\mathrm{sec}$ for equipped aircraft. Fuel considerations make speed-ups of more than a minute inefficient, therefore the earliest possible scheduled time of arrival is a minute before the estimated time of arrival (ETA). The latest possible scheduled time of arrival is set to one hour after the ETA.

The aircraft belong to one of three categories based on their Maximum Takeoff Weight (MTOW): Small, Large or Heavy, and can be either equipped with FMS or be controlled by pilots. A representative matrix of minimum time separations in seconds is given in the table below [4].

	Trailing Aircraft		
Leading Aircraft	Heavy	Large	Small
Heavy	96	157	196
Large	60	69	131
Small	60	69	82

One approach to accommodating the uncertainty in arrival times is to buffer the required separation requirements. The size of this buffer is set to $12\,\mathrm{sec}$ if both the leading and trailing aircraft are equipped with an FMS, and $24\,\mathrm{sec}$ if at least one of them is not equipped. The FCFS sequence is then determined by maintaining the order of the estimated runway times of arrival, but by enforcing the minimum spacing requirements with the appropriate buffering. We use this FCFS order with buffering as the baseline makespan with which to determine improved schedules. This is the minimum acceptable makespan. The probability of this FCFS sequence being feasible (i.e., none of the separation requirements is violated) is the baseline value of the robustness. We use the ratio of the probability of a schedule being feasible to the probability of FCFS sequence as the measure of reliability or robustness of the schedule. We represent the throughput of the schedule as the number of aircraft divided by the time taken to complete the schedule (the makespan). We can then use the dynamic programming algorithm to compute the tradeoff curve between throughput and robustness.

We present an example that illustrates the potential of the proposed technique to produce robust schedules. We consider a sequence of 20 aircraft landing on a single runway, generated using a Poisson distribution at the rate of 45 aircraft an hour. The sequence of aircraft along with their weight classes, equipage and arrival times in the FCFS schedule with buffering are presented in Table I. The landing times in the FCFS schedule satisfy the separation requirements, but do not necessarily form the most robust schedule, even for the FCFS landing order. We can compute the tradeoff curve between the throughput and reliability to determine a more robust FCFS sequence. Similarly, we compute the tradeoff between reliability and throughput for $k = 1$ and $k = 2$. The results are plotted in Figure 2, and are representative of the type of output we would like to produce using the proposed algorithm. We note that the FCFS makespan can be achieved with a substantially higher level of reliability, and a greater throughput can be achieved with the same level of reliability. We also note that the tradeoff improves as we move from FCFS to 1-CPS, and as we proceed to 2-CPS. The schedules (with landing times) are presented in Table I for sequences which have the same makespan as the baseline FCFS sequence with buffering.

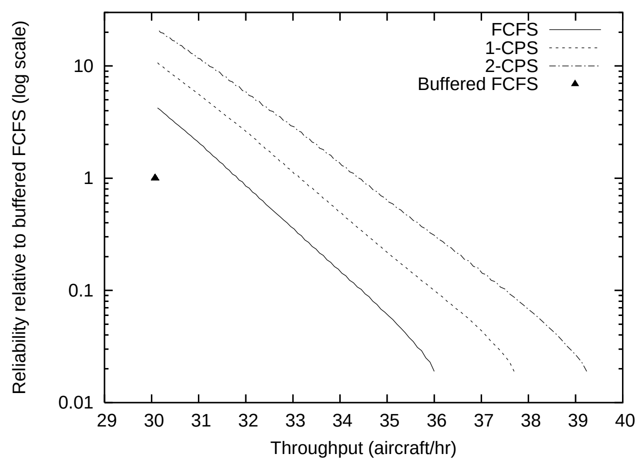

Fig. 2. (Top) Reliability-throughput tradeoff contours in a semi-log scale.

The method proposed in this paper is amenable to real time implementation since the computation time (once internal data structures have been created) for 1- or 2-CPS is less than $1\,\mathrm{sec}$ for a 30-min time horizon for up to 50 aircraft.

V. Conclusions

We have presented an approach for determining the tradeoff between robustness and throughput, while scheduling single runway operations under Constrained Position Shifting. The approach we present can handle precedence constraints that could arise from operational constraints or airline preferences, and take into account restrictions on possible arrival times of aircraft. The proposed Dynamic Programming approach can accommodate several sources of uncertainty, and is computationally efficient enough for a real-time application. We believe that this technique will be valuable both in assessing the benefits of equipping aircraft with advanced Flight Management Systems, and in determining robust schedules for runway operations.

References

[1] F. Neuman and H. Erzberger, “Analysis of delay reducing and fuel saving sequencing and spacing algorithms for arrival spacing,” NASA Technical Memorandum 103880, October, 1991.

ID (Type, Equipage)	FCFS with buffering		FCFS*		1-CPS*		2-CPS*	
	Scheduled arrival time (in seconds) and sequence position at runway							
	Position	Time	Position	Time	Position	Time	Position	Time
Aircraft 1 (S, Equipped)	1	50	1	50	1	50	1	50
Aircraft 2 (S, Not Equipped)	2	156	2	210	2	220	2	220
Aircraft 3 (H, Equipped)	3	240	3	280	3	300	3	330
Aircraft 4 (H, Equipped)	4	348	4	380	4	420	4	450
Aircraft 5 (L, Not Equipped)	5	529	5	580	5	630	5	660
Aircraft 6 (L, Not Equipped)	6	622	6	680	6	730	6	760
Aircraft 7 (S, Not Equipped)	7	777	7	820	7	870	8	970
Aircraft 8 (S, Not Equipped)	8	883	8	910	8	970	9	1120
Aircraft 9 (L, Equipped)	9	976	9	980	9	1070	7	830
Aircraft 10 (H, Not Equipped)	10	1060	10	1040	11	1240	12	1330
Aircraft 11 (L, Not Equipped)	11	1241	11	1230	10	1140	10	1200
Aircraft 12 (L, Equipped)	12	1334	12	1310	12	1400	11	1270
Aircraft 13 (H, Not Equipped)	13	1418	13	1380	13	1470	13	1480
Aircraft 14 (S, Not Equipped)	14	1638	14	1620	14	1710	16	1940
Aircraft 15 (S, Equipped)	15	1744	15	1710	15	1800	17	2090
Aircraft 16 (L, Not Equipped)	16	1837	16	1790	16	1880	14	1640
Aircraft 17 (H, Equipped)	17	1921	17	1890	17	1980	15	1740
Aircraft 18 (S, Equipped)	18	2129	18	2100	18	2190	18	2180
Aircraft 19 (H, Not Equipped)	19	2213	19	2230	20	2390	20	2390
Aircraft 20 (L, Equipped)	20	2394	20	2390	19	2330	19	2330

TABLE I

AIRCRAFT TYPES, EQUIPAGE, SEQUENCE POSITION, AND ARRIVAL TIMES FOR FCFS, "ROBUST" FCFS AND CPS SEQUENCES WITH THE SAME MAKESPAN. THE ASTERISKS DENOTE THE OUTPUT OF THE ROBUST SCHEDULING ALGORITHM.

[2] G. L. Wong, "The Dynamic Planner: The Sequencer, Scheduler, and Runway Allocator for Air Traffic Control Automation," NASA/TM-2000-209586, April, 2000.

[3] H. Balakrishnan and B. Chandran, "Scheduling aircraft landings under Constrained Position Shifting," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, Keystone, CO, 2006.

[4] R. de Neufville and A. Odoni, *Airport Systems: Planning, Design and Management*. McGraw-Hill, 2003.

[5] R. G. Dear, "The dynamic scheduling of aircraft in the near terminal area," Ph.D. dissertation, Massachusetts Institute of Technology, 1976, also Flight Transportation Laboratory Report R76-9.

[6] F. R. Carr, "Robust decision-support tools for airport surface traffic," Ph.D. dissertation, Massachusetts Institute of Technology, 2004.

[7] G. C. Carr, H. Erzberger, and F. Neuman, "Airline arrival prioritization in sequencing and scheduling," in *USA/Europe Air Traffic Management R&D Seminar*, 1998.

[8] H. N. Swenson, T. Hoang, S. Engelland, D. Vincent, T. Sanders, B. Sanford, and K. Heere, "Design and operational evaluation of the traffic management advisor at the fort worth air route traffic control center," in *USA/Europe Air Traffic Management R&D Seminar*, 1997.

[9] L. A. Meyn and H. Erzberger, "Airport arrival capacity benefits due to improved scheduling accuracy," in *AIAA Aviation, Technology, Integration and Operations (ATIO) Conference*, Arlington, VA, 2005.

[10] J. Scharl, A. Haraldsdottir, and E. Schoemig, "A trajectory modeling environment for the study of arrival traffic delivery accuracy," in *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Keystone, CO, 2006.

[11] S. M. Green and R. Vivona, "Field evaluation of Descent Advisor trajectory prediction accuracy," in *AIAA Guidance, Navigation and Control Conference*, San Diego, CA, July 1996, AIAA 96-3764.

[12] L. Credeur, W. R. Capron, G. W. Lohr, D. J. Crawford, D. A. Tang, and J. W. G. Rodgers, "Final-Approach Spacing Aids (FASA) Evaluation for Terminal- Area, Time-Based Air Traffic Control," NASA-TP-3399, December, 1993.

[13] D. A. Rhoda and M. L. Pawlak, "An assessment of thunderstorm penetrations and deviations by commercial aircraft in the terminal area," MIT Lincoln Laboratory Technical Report NASA/A-2, 1999.

[14] D. Bertsimas and M. Sim, "The price of robustness," *Operations Research*, vol. 52, no. 1, pp. 35–53, 2004.

APPENDIX

Proof: [Lemma 1] We first observe that, by construction, $\ell(x) = \ell'(y)$ for $x \in P(y)$. Therefore, $J_y(t_{\ell'(y)}, t_{\ell(y)}) = J_y(t_{\ell(x)}, t_{\ell(y)})$.

Since $J_y(t_{\ell(x)}, t_{\ell(y)})$ is the maximum value of reliability over all paths leading to node y,

$$J_y(t_{\ell(x)}, t_{\ell(y)}) \geq J_x(t_{\ell'(x)}, t_{\ell(x)}) \times \Pr\{t_{\ell(x)} \leftrightarrow t_{\ell(y)} | t_{\ell'(x)} \leftrightarrow t_{\ell(x)}\}$$
$$\forall\ x \in P(y), t_{\ell'(x)} \in \mathcal{I}(\ell'(x)), t_{\ell(x)} \in \mathcal{I}(\ell(x)), t_{\ell(y)} \in \mathcal{I}(\ell(y)),$$

where $t_{\ell(x)} - t_{\ell'(x)} \geq \delta_{\ell'(x),\ell(x)}$ and $t_{\ell(y)} - t_{\ell'(y)} \geq \delta_{\ell'(y),\ell(y)}$.

This means that, in particular,

$$J_y(t_{\ell'(y)}, t_{\ell(y)}) \geq \max_{x \in P(y)} \max_{t_{\ell'(x)} \in \mathcal{I}(\ell'(x))} \left\{ J_x(t_{\ell'(x)}, t_{\ell(x)}) \times \Pr\{t_{\ell(x)} \leftrightarrow t_{\ell(y)} \mid t_{\ell'(x)} \leftrightarrow t_{\ell(x)}\} \right\},$$
$$\forall\ t_{\ell(y)} \in \mathcal{I}(\ell(y)) : t_{\ell(y)} \geq t_{\ell(x)} + \delta_{\ell(x),\ell(y)}$$

To complete the proof, we only need to show that the above relationship can never hold as a strict inequality. Suppose (for contradiction) that

$$J_y(t_{\ell(x)}, t_{\ell(y)}) > J_x(t_{\ell'(x)}, t_{\ell(x)}) \times \Pr\{t_{\ell(x)} \leftrightarrow t_{\ell(y)} | t_{\ell'(x)} \leftrightarrow t_{\ell(x)}\}$$
$$\forall\ x \in P(y), t_{\ell'(x)} \in \mathcal{I}(\ell'(x)), t_{\ell(x)} \in \mathcal{I}(\ell(x)), t_{\ell(y)} \in \mathcal{I}(\ell(y))$$

Given that the times are feasible and that all spacings satisfy at least the minimum separation requirement, $\Pr\{t_{\ell(x)} \leftrightarrow t_{\ell(y)} \mid t_{\ell'(x)} \leftrightarrow t_{\ell(x)}\} > 0$. Dividing by this probability, we get

$$\frac{J_y(t_{\ell(x)}, t_{\ell(y)})}{\Pr\{t_{\ell(x)} \leftrightarrow t_{\ell(y)} \mid t_{\ell'(x)} \leftrightarrow t_{\ell(x)}\}} > J_x(t_{\ell'(x)}, t_{\ell(x)}),$$
$$\forall\ x \in P(y), t_{\ell'(x)} \in \mathcal{I}(\ell'(x)), t_{\ell(x)} \in \mathcal{I}(\ell(x)), t_{\ell(y)} \in \mathcal{I}(\ell(y)).$$

This implies that

$$\max_{\substack{w \in P(y) \\ t_{\ell'(w)} \in \mathcal{I}(\ell'(w)) \\ t_{\ell(w)} \in \mathcal{I}(\ell(w)) \\ t_{\ell(y)} \in \mathcal{I}(\ell(y))}} \frac{J_y(t_{\ell(w)}, t_{\ell(y)})}{\Pr\{t_{\ell(w)} \leftrightarrow t_{\ell(y)} | t_{\ell'(w)} \leftrightarrow t_{\ell(w)}\}} > J_x(t_{\ell'(x)}, t_{\ell(x)})$$
$$\forall\ x \in P(y), t_{\ell'(x)} \in \mathcal{I}(\ell'(x)), t_{\ell(x)} \in \mathcal{I}(\ell(x)).$$

However, $\frac{J_y(t_{\ell(w)}, t_{\ell(y)})}{\Pr\{t_{\ell(w)} \leftrightarrow t_{\ell(y)} \mid t_{\ell'(w)} \leftrightarrow t_{\ell(w)}\}}$ is the reliability of the subsequence of $J_y(t_{\ell(w)}, t_{\ell(y)})$ that ends at node w and time $\ell'(w)$ and $\ell(w)$. This contradicts the maximality of $J_x(t_{\ell'(x)}, t_{\ell(x)})$ for $x = w$. ■