JOURNAL OF AIRCRAFT
Vol. 37, No. 3, May–June 2000

Evaluation of Scheduling Methods for Multiple Runways

Michael A. Bolender* and G. L. Slater†
University of Cincinnati, Cincinnati, Ohio 45221-0070

Several scheduling strategies are analyzed to determine the most efficient means of scheduling aircraft when multiple runways are operational and the airport is operating at different utilization rates. Simulation data are compared for two- and three-runway scenarios to results from queuing theory for an *M/D/n* queue. The direction taken, however, is not to do a steady-state, or equilibrium, analysis because this is not the case during a rush period at a typical airport. Instead, a transient analysis of the delay per aircraft is performed. It is shown that the scheduling strategy that reduces the delay depends on the density of the arrival traffic. For light traffic, scheduling aircraft to their preferred runways is sufficient; however, as the arrival rate increases, it becomes more important to separate traffic by weight class. Significant delay reduction is realized when aircraft that belong to the heavy and small weight classes are sent to separate runways with large aircraft put into the best landing slot.

Introduction

THE analysis of aircraft scheduling techniques for airports with multiple runways is becoming more important with the evolution in the design of new airports, such as Denver International, that have the capability to land several aircraft independently on several runways. Therefore, new techniques for scheduling to multiple runways are needed to improve on the traditional first-come-first-serve (FCFS) technique generally employed. New computer-based tools such as the Center TRACON Automation System (CTAS) will give air traffic controllers a tool that gives them accurate aircraft state information and can assist them in their scheduling duties.[1] The intent of this paper is to present and compare several scheduling methods to show the best means to reduce the delay per aircraft.

In a multiple-runway airport, traffic from different directions is assigned a preferred runway based on the geometric relation of the approach geometry to a runway. Previous efforts by Vandevenne and Lippert[2] have shown that significant delay reduction is possible for multiple runways if the aircraft are allowed to cross over without penalty. Using steady-state queuing theory, Vandevenne and Lippert show that the delay should be reduced by a factor of approximately $1/n$ for n runways when compared to a single runway case. It would be expected then that $1 - 1/n\%$ of the aircraft would be switching from their preferred runway. A delay threshold can be added to reduce the number of crossovers. The delay threshold is a lower bound on which the delay on the alternate runway must be reduced for the aircraft to cross to that runway. As a result of that threshold, there is a drop in the number of crossovers and a corresponding increase in the delay.

The approach taken in this paper is to study several different techniques for scheduling aircraft to multiple runways. Numerical simulation is used to determine the effectiveness of several simple runway allocations. These results are compared to results from queuing theory. Because a typical arrival rush at an airport is fairly short, we are interested in looking at the transient state of the queue and how the waiting time or delay evolves during a rush period. This simulates the queuing dynamics during a rush period at a typical airport that is initially operating with light arrival traffic. It is shown that the best method of allocating runways when the airport is operating either near or above capacity is to separate the heavy and small aircraft as much as possible. However, if the traffic is light, it is sufficient to land the aircraft on their preferred runways.

Scheduling Problem

The aircraft scheduling problem can be defined as a procedure that is "to plan automatically the most efficient landing order and to assign optimally spaced landing times to all arrivals, given the times the aircraft are actually arriving at the Air Route Traffic Control Center (ARTCC)."[1] This definition may sound modest, but there are some underlying attributes of the scheduling problem that make it very difficult. The arrival times of the aircraft into the system are random. Over short periods of time, the arrival process can be modeled as a homogeneous Poisson process. However, if one were to observe arrivals at an airport for an entire day, one would see that the number of arrivals varies from hour to hour. There are periods of time where there are very few arrivals and periods of time where the incoming traffic is heavy enough that the airport is operating at its capacity.

An important point regarding the arrival scheduling problem is the classification of aircraft into different weight classes. The interaircraft separation between two aircraft is dependent on the respective weight classes of the aircraft. In practice, we generally deal with three weight classes that we describe as heavy, large, and small. The Federal Aviation Administration (FAA) has specified a separation matrix that gives required minimum distance separations between these classes of aircraft. These separations arise from the consideration of wake vortices, speed differences, etc. The nominal matrix used is shown in Table 1.

The matrix in Table 1 may vary depending on winds, weather, etc. To find the proper separation for a pair of aircraft, one simply goes to the appropriate row for the leading aircraft, then to the column for the weight class of the trailing aircraft.

Analytical Models

To predict the amount of delay that an aircraft can expect for a given traffic mix, arrival rate, and airport capacity, two standard queuing models are considered. The first model has deterministic service times, and the second considers service times that are exponentially distributed. Rather than restricting ourselves to a steady-state analysis, a study of the transient queue dynamics is performed. The motivation for doing a transient analysis is that in actual traffic at hub airports, arrival traffic is concentrated in short periods that last 60–90 min. As such, the system never reaches a steady-state condition. There are times where the peak arrival rate of aircraft will be greater than the runway capacity. The arrival rate preceding the rush period is usually low enough that the aircraft are typically not delayed due to the large interarrival times between them, and so the method of landing aircraft on their preferred runways will be more than adequate.

Received 2 March 1999; revision received 25 October 1999; accepted for publication 30 October 1999.

*Ph.D. Candidate, Department of Aerospace Engineering and Engineering Mechanics. Member AIAA.

†Professor and Department Head, Department of Aerospace Engineering and Engineering Mechanics. Associate Fellow AIAA.

Table 1 Separation matrix in nautical miles

Lead aircraft	Trail aircraft		
	Heavy	Large	Small
Heavy	4	5	6
Large	3	3	4
Small	3	3	3

Deterministic Service Times

In constructing a mathematical model for the scheduling problem, we need to make some simplifying assumptions. The first is that the arrivals are to be modeled according to a homogeneous Poisson process with an arrival rate λ. The Poisson process has a mean number of arrivals in the time period $[t, t + \Delta t]$ equal to $\lambda \Delta t$. Furthermore, the interarrival times of the aircraft have an exponential distribution with a mean of $1/\lambda$. It is further assumed that each server has a constant service time T_s. This queuing system is then said to be $M/D/n$ (Ref. 3), where M denotes that the interarrival times are Markovian or memoryless, D denotes that the service times are deterministic or constant, and n servers are operating in parallel. Also, the servers are fed by a single queue.

The service time may be taken to be constant by probabilistically weighting the required separation times within the separation matrix. We assume that the traffic mix (i.e., the relative proportion of different weight classes) is known and that the probability of an aircraft entering the queue being a member of a particular weight class is given by this relative proportion. By assigning a fixed service time to all aircraft in this manner, it is assumed that any delay results from the randomness of the arrival times. To calculate a service time (and, hence, a runway capacity) from the separation matrix, one only needs to know the traffic mix and the separation matrix. The average service time is $T_s = \boldsymbol{P}_m^T \boldsymbol{S} \boldsymbol{P}_m$, where $\boldsymbol{P}_m = [P_H \;\; P_L \;\; P_S]^T$ is a vector of the probabilities that the aircraft is a heavy, large, or small, respectively, and S is the separation matrix. Let μ represent the runway capacity. The capacity of a single runway is then $\mu = 1/T_s$. For example, if the traffic mix is $\boldsymbol{P}_m = [.2 \;\; .7 \;\; .1]^T$, and the aircraft have a common landing speed of 150 kn, then $\mu = 43.5$ aircraft/h and $T_s = 82.8$ s. For analysis purposes, using a constant T_s simplifies the mathematical model significantly. As will be shown, this assumption does not adversely affect the numerical results. The constant service time used is a function of the assumed traffic mix and the elements of the separation matrix. An alternate approach that utilizes random service times is discussed in the next section.

To analyze the delay buildup during a rush period, one needs to study the transient probabilities of the queuing process. The time-varying equations are taken from Tijms.[4] The derivation of the probabilities is based on the following observations: a customer in service at time t will have left service at time $t + T_s$. The customers in the system at the time $t + T_s$ will be those that entered during the increment T_s as well as those that were in the queue at time t.

Define $A(T_s)$ to be the number of arrivals in the interval $[t, t + T_s]$. Furthermore, let $N(t)$ be the number in the system at time t, and let $P_j(t) = P[N(t) = j]$ denote the probability that j customers are in the system at time t. The event that there are j aircraft in the system is a union of the events that there are j arrivals when either the servers are either full, empty, or less than full and the queue is empty or there are $j - 1$ arrivals when there is a queue of length 1, etc. Using this detail, we have the following expression for the probability of the number of aircraft in the system

$$\begin{aligned} P_j(t + T_s) = {} & P[A(T_s) = j \mid N(t) = 0] P_0(t) \\ & \cup \ldots \cup P[A(T_s) = j \mid N(t) = n] P_n(t) \\ & \cup P[A(T_s) = j - 1 \mid N(t) = n + 1] P_{n+1}(t) \\ & \cup \ldots \cup P[A(T_s) = 0 \mid N(t) = j + n] P_{j+n}(t) \end{aligned} \tag{1}$$

Because the number of arrivals in the interval $[t, t + T_s]$ and the number in the queue are independent events, the conditional probabilities can be written as

$$\begin{aligned} P[A(T_s) = m \mid N(t) = k] &= P[A(T_s) = m] \\ &= e^{-\lambda T_s} \left[(\lambda T_s)^m / m! \right] \end{aligned} \tag{2}$$

The probability given in Eq. (2) is simply the probability that there are m Poisson arrivals in an interval of length T_s. Substituting Eq. (2) into Eq. (1) and simplifying yield

$$\begin{aligned} P_j(t + T_s) = {} & \sum_{k=0}^{n} P_k(t) e^{-\lambda T_s} \frac{(\lambda T_s)^j}{j!} \\ & + \sum_{k=n+1}^{n+j} P_k(t) e^{-\lambda T_s} \frac{(\lambda T_s)^{j-k+n}}{(j - k + n)!}, \qquad j = 0, 1, 2, \ldots \end{aligned} \tag{3}$$

This gives us an infinite set of equations that can be solved at discrete times. Setting $t = kT_s$ and defining the probability vector, $\bar{\boldsymbol{P}}(k) = \bar{\boldsymbol{P}}(kT_s) = [P_0(k) P_1(k) \ldots]^T$, Eq. (3) can be written as the infinite dimensional difference equation:

$$\bar{\boldsymbol{P}}(k) = \boldsymbol{F} \bar{\boldsymbol{P}}((k - 1)) \tag{4}$$

where F is given hereafter for the $n = 2$ case as

$$e^{-\lambda T_s} \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ \lambda T_s & \lambda T_s & \lambda T_s & 1 & 0 \\ \frac{(\lambda T_s)^2}{2!} & \frac{(\lambda T_s)^2}{2!} & \frac{(\lambda T_s)^2}{2!} & \lambda T_s & 1 \\ \vdots & \vdots & \vdots & \ddots & \ddots \\ \frac{(\lambda T_s)^i}{i!} & \frac{(\lambda T_s)^i}{i!} & \frac{(\lambda T_s)^i}{i!} & \frac{(\lambda T_s)^{i-1}}{(i-1)!} & \ddots \end{bmatrix} \tag{5}$$

The solution to this set of equations is

$$\bar{\boldsymbol{P}}(k) = \boldsymbol{F}^k \bar{\boldsymbol{P}}_{(0)} \tag{6}$$

This set can be solved approximately by initializing the time when the queue is empty ($\bar{\boldsymbol{P}}(0) = [1 \;\; 0 \;\; 0 \;\; \cdots]^T$) and choosing a sufficiently large dimension of $\boldsymbol{F}$ such that the significant probabilities of the system are captured.

Once the probabilities are determined, the mean number in the system at any time increment k can be calculated. The mean number in the system is defined as

$$m(k) = \sum_{j=0}^{\infty} j P_j(k)$$

The mean number in the system at any time k may be broken up into two components, those found in service $m_s(k)$ and those in the queue awaiting service $m_Q(k)$. Hence, $m(k) = m_s(k) + m_Q(k)$. The mean number in service can be found in Ref. 3 to be

$$\sum_{j=0}^{n-1} j P_j(k) + n \sum_{j=n}^{\infty} P_j(k) \tag{7}$$

The first summation in Eq. (7) arises because if the number of customers in the system is less than the number of servers, then all customers are being served. The second summation exists because if there are more customers in the system than there are servers, then all servers will be busy. The resulting mean number in the queue is then

$$m_Q(k) = \sum_{j=n}^{\infty} (j - n) P_j(k) \tag{8}$$

After we use Eq. (8) and truncate the upper summation limit to N_F, where N_F is the dimension of $\boldsymbol{F}$ used for calculation, the mean number in the queue can be calculated. The expected waiting time or delay as a function of time can be found by simply applying Little's formula (see Ref. 3). Mathematically, Little's formula is $L = \lambda W$ where L is the length of the queue, λ is the arrival rate of customers into the system, and W is the waiting time in the queue. The waiting time in the queue then becomes

$$W_Q(k) = \frac{1}{\lambda} m_Q(k) = \frac{1}{\lambda} \sum_{j=n}^{N_F} (j - n) P_j(k) \tag{9}$$

Exponential Service Times

A second model that has been used by some authors to analyze the arrival scheduling problem is one where the service times are exponentially distributed with a mean equal to the service time calculated from the separation matrix. A queue that has Poisson arrivals, exponential service times, and n servers is referred to as an $M/M/n$ queue.[3] The resulting model gives rise to the Kolomogrov differential equations, which describe a birth and death process for an n server queue with a constant arrival rate $\lambda_j = \lambda$ for all j and service rates μ_j.

The birth and death differential equations[5] are then

$$\dot{P}_j(t) = \lambda_{j-1} P_{j-1}(t) - (\lambda_j + \mu_j) P_j(t) + \mu_{j+1} P_{j+1}(t) \tag{10}$$

where $P_j(t)$ is the probability of there being j aircraft in the system, $\dot{P}_j(t)$ is the derivative with respect to time of $P_j(t)$, and λ_j and μ_j are the arrival and service rates, respectively. In addition, μ_j is given by

$$\mu_j = \begin{cases} j\mu & j = 0, 1, \ldots, n-1 \\ n\mu & j \geq n \end{cases} \tag{11}$$

Because Eq. (10) results in an infinite set of first-order differential equations, it can be written in the form $\boldsymbol{P}(t) = \boldsymbol{G}\boldsymbol{P}(t)$, with $\boldsymbol{P}(t) = [P_0 \;\; P_1 \;\; \cdots]$. The matrix G in this case is a tri-diagonal matrix of the form (for $n = 2$)

$$\begin{bmatrix} -\lambda & \mu & 0 & 0 & 0 \\ \lambda & -(\lambda + \mu) & 2\mu & 0 & 0 \\ 0 & \lambda & -(\lambda + 2\mu) & 2\mu & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix} \tag{12}$$

Again, we are only able to approximate the infinite set of differential equations by a finite set when solving the system numerically. Hence, one needs to select the dimension of G large enough that the important features of the queuing dynamics are realized.

The solution to the differential equation $\boldsymbol{P}(t) = \boldsymbol{G}\boldsymbol{P}(t)$, $\boldsymbol{P}(0) = \boldsymbol{P}_0$ is

$$\boldsymbol{P}(t) = e^{Gt} \boldsymbol{P}_0 \tag{13}$$

Once the probabilities are found according to Eq. (13), the mean number in the queue and, hence, the mean waiting time in the queue are found using Eq. (9) and replacing k with t. A comparison of the numerical results for the deterministic service time model and the exponential service time model shows that the exponential model yields a significantly greater average delay than that experienced by the deterministic service model. (Note that in Fig. 1, the mean exponential service time is exactly equal to the deterministic service time.) The reason for this difference is the large standard deviation of the exponential distribution. Consider an exponential distribution with rate α. The mean service time is then $1/\alpha$ and the standard deviation is $1/\alpha$. Note that for a purely deterministic service time, the standard deviation is zero. If we consider the service times as determined by the separation matrix, the standard deviation of arrival traffic can be easily computed. After converting the separation matrix from distances to speeds using a common approach speed of 150 kn and the given traffic mix, the average service time is 82.8 s with a standard deviation of 19.3 s. This is compared to 82.8 s for the exponential distribution. The large variance of the exponential distribution introduces a much wider range of service times than that which occurs in practice. The effect of these service times is to cause additional buildup in the queue; therefore, additional delay is introduced into the system that is significantly larger than that observed in simulation. Simulation results presented in the succeeding sections validate the hypothesis that the deterministic service time assumption is a reasonable model for analytical studies, especially in comparison to the exponential model.

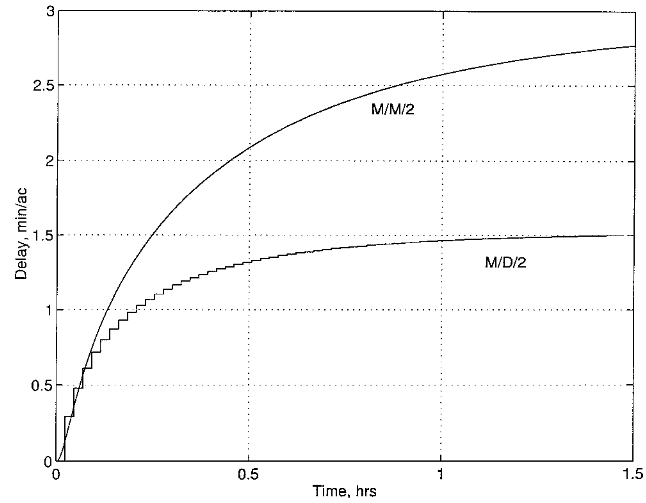

Fig. 1 Mean waiting times for *M/D/*2 and *M/M/*2 queues (λ = 72 aircraft/h).

Comparison of Runway Allocation Strategies

Because of the complex nature of scheduling arrival aircraft, simulation provides a valuable tool to determine the feasibility of a particular scheduling algorithm. In this section, we discuss the merits and drawbacks for several runway allocation methods. The two-runway allocation problem will be discussed, followed by the three-runway problem. Three different traffic densities will be analyzed for each problem: a period of light traffic (two-runway case only), a period of moderately heavy traffic where the airport is operating near, but below capacity, and a period where the traffic is heavy enough that the airport is operating above capacity. The purpose is to show that selection of a given runway allocation method varies with the arrival rate of aircraft into the airport.

Two-Runway Allocation Problem

The two-runway problem is one that is quite common. Runways that operate independently of one another have sufficient separation between their centerlines such that aircraft landing simultaneously can be treated independently.

It is assumed for all scheduling strategies that the aircraft arrive in two different streams. Each arrival stream's estimated times of arrival (ETA) are modeled by a Poisson distribution with a mean of λ aircraft per hour per runway, which gives a total arrival rate of 2λ aircraft/h using the reproductive property of the Poisson process. For each arrival stream, there is a preferred runway that an aircraft desires to land on. Because of the common arrangement of parallel runways, we will nominally call the runways left and right or L and R. The arrival direction and, hence, the preferred runway, was determined by a random draw. The capacity of each runway is approximately 43.5 aircraft/h using the separation matrix and (for simplicity) a common approach speed of 150 kn. The traffic mix is assumed to consist of 70% large aircraft, 20% heavy, and 10% small (this roughly approximates the traffic mix observed in most U.S. traffic.) The performance index to be considered is the average delay of each aircraft because minimizing the average delay per aircraft should result in a maximum throughput in a given time. The delay per aircraft is measured with respect to its estimated time of arrival. It is further assumed that each aircraft can be slowed down as much as needed to meet the minimum spacing requirements of the separation matrix. The flight time to both runways is assumed to be identical. The first aircraft landing on each runway is constrained to land at its nominal time of arrival. Each arrival period consists of a fixed number of aircraft and is on average 90 min long, as this the typical duration of an arrival period at a hub airport. The results presented are the average delays for 5000 replications. This number of replications was chosen to allow the delays to converge and to obtain a 95% confidence interval that was less than 0.1 min.

Light Traffic

For the light-traffic case, the total arrival rate is taken to be 32 aircraft/h (or 16 aircraft/h/runway). Each aircraft, on entry into the system, is placed at the end of the queue. Rearrangement of the queued aircraft was not performed; therefore, the only options are either to assign each aircraft to its preferred runway or to let it land on an alternate runway. We compared three means of allocating runways for the arrival traffic. The first was to land each aircraft on its preferred runway. This is the easiest scheduling algorithm to implement because no decision is made to cross runways. Furthermore, this is a baseline that allows us to later show improvements in delay as compared to this algorithm. By constraining the aircraft to land on their preferred runways, the queue is considered as two separate queues, each feeding a particular server. The second method is to allow an aircraft to switch from its preferred runway whenever the aircraft's delay on the alternate runway is less than its delay on its preferred runway. This plan will be referred to as unconstrained crossovers. Note that this is equivalent to the case of a single queue feeding two servers in parallel. The third allocation strategy is to land the heavy and small aircraft on runways that are designated for each weight class and to place the large aircraft on the runway where the delay for it is the smallest. The fourth method, which is not used in the light-traffic case, attempts to reduce the amount of crossover traffic. Because crossovers increase the workload on the controllers, it is desirable to reduce delay without imposing a significantly higher workload on them. Therefore, this particular algorithm permits the aircraft to cross over to the alternate runway if one of two conditions were satisfied: 1) the aircraft's delay on the alternate runway is less than on the preferred and the sequence was defined to be favorable or 2) the aircraft's delay on the alternate runway is less than that on its preferred runway by some predetermined amount. A favorable sequence is one that tries to group aircraft with the minimal elements of the separation matrix. For example, a favorable sequence would be to land a small aircraft ahead of a heavy aircraft; however, the converse would not be a favorable sequence.

The results of the first three approaches are given in Table 2. The improvements made by allocating runways are 70% better than if the aircraft were constrained to land on their preferred runways. However, from an operational point of view, there is no real advantage for optimizing the landing sequence to reduce the delay per aircraft because the delay is already small. This is because the average separations between arrivals are large, hence, there is little tendency for bunching to occur.

To gain confidence in the simulation, and to compare the validity of the analytical model of the preceding section, we can compare the first two cases above to an $M/D/1$ and an $M/D/2$ queue, respectively. To calculate the expected delay per aircraft over a given time period, the average value of the waiting time [Eq. (9)] is needed. The average value of the expected waiting time curve over N service intervals is then

$$\bar{W} = \frac{1}{N}\sum_{k=0}^{N} W(k) \tag{14}$$

Table 2 Simulation results for light traffic and two runways

Allocation strategy	Average delay, min/aircraft	Standard deviation, min/aircraft
No crossovers	0.3995	0.1919
Unconstrained crossovers	0.1180	0.0848
Separate heavy/small aircraft	0.1647	0.0898

For the allocation strategies used in Table 2, the first two represent cases that can be handled analytically using our earlier results. Using Eq. (14) for an $M/D/1$ queue with an arrival rate of 16 aircraft/h and a service time of 82.8 s, the average delay is found to be 0.3955 min/aircraft, which agrees with the no-crossover case. The unlimited-crossover case shows the same trend. The predicted delay using an $M/D/2$ queue is 0.1174 min/aircraft, whereas the simulation produced a delay of 0.1180 min/aircraft. These differences are small and indicate that averaging the simulation over 5000 trials is adequate for the light-traffic case.

Moderate Traffic

The case where there is moderately heavy traffic allows us to investigate what happens when the airport is operating under a fairly high arrival rate, but is still not at its capacity. This allows for fairly tight bunching to occur, as well as periods where the traffic may be light for several minutes. It is assumed that the total arrival rate is 72 aircraft/h, putting the airport at about 84% capacity. Four different scheduling algorithms are investigated, which can be summarized as follows. The first method is to land each aircraft on its preferred runway, that is, no crossovers allowed. This serves as a baseline strategy and is used to determine the reduction in delay. The second strategy allows an aircraft to cross from its preferred runway to the alternate runway if the aircraft can land at an earlier time. The first two strategies correspond to the analytical models that are considered. The strategies where traffic is separated by weight class and where crossovers are permitted under favorable conditions are also addressed. Results for these four sequencing strategies are summarized in Table 3.

The first scheduling strategy employed was to restrict each incoming aircraft to land on its preferred runway. This is employed as a baseline to find improvements in the runway balance (i.e., are the same amount of aircraft landing on each runway?) and in the delay per aircraft. The aircraft, as stated, entered from the appropriate direction and then were scheduled to the corresponding runway. Using an arrival stream consisting of 108 aircraft and averaging over 5000 replications, the mean delay was 2.56 min/aircraft. For analytical purposes, this is modeled by an $M/D/1$ queue with the arrival rate equal to 36 aircraft/h and a constant service time of 82.8 s. The expected delay curve is shown in Fig. 2. Using Eq. (14), we find

Table 3 Simulation results for moderate traffic and two runways

Allocation strategy	Average delay, min/aircraft	% Crossovers
No crossovers	2.5629	0
Separate heavy/small	1.2371	50
Unconstrained crossovers	1.2581	45
Constrained crossovers	1.3660	23

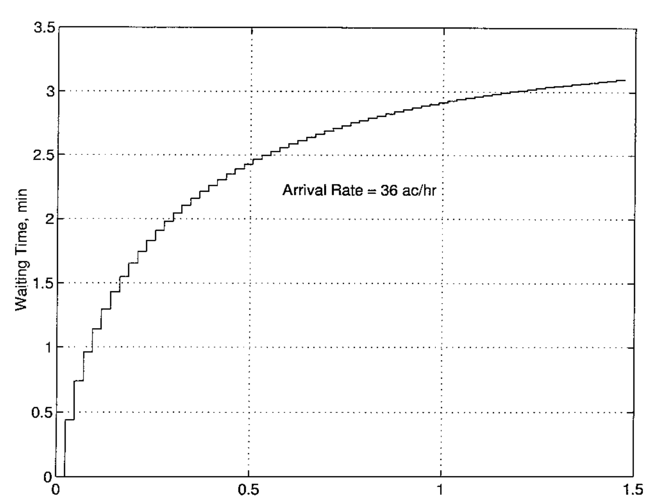

Fig. 2 Mean waiting time for *M/D/1* queue with moderate traffic.

that the average value of the waiting time is 2.53 min. The mean delay obtained by simulation is within 1.2% agreement of the analytical model. However, if we were to use the steady-state delay for the $M/D/1$ queue, the average delay would increase to 3.32 min. In comparison, if we used the exponential service time approximation (an $M/M/1$ queue), its steady-state delay would be 6.64 min. It can be shown for the steady-state case that the delay for a queue with deterministic service times is one half that of a queue with exponentially distributed service times. The exponential service time model is not an appropriate model for modeling the service times for this problem.

The second strategy was to allow the aircraft to switch runways whenever the delay on the alternate runway was less than the delay on the aircraft's preferred runway. This case was studied by Vandevenne and Lippert[2] using traffic statistics from the Dallas–Fort Worth airport. Vandevenne and Lippert studied the reduction in delay relative to the preferred runway case that was discussed earlier. Their analysis looked at the expected waiting time in the steady state for $M/D/n$ queues as compared to an $M/D/1$ queue. They show that the delay for the n runway case is reduced by a factor of approximately $1/n$ as compared to the single runway case. Using this analysis as a starting point, a curve showing how the relative delay evolves as a function of time was generated. Figure 3 shows the delay of an $M/D/2$ queue relative to an $M/D/1$ queue. Calculating the average value of the curve in Fig. 3, an improvement of about 52% would be expected by allowing an aircraft to choose the runway with the lowest delay for it. From Table 3, the delay reduction achieved in the simulation by allowing aircraft to freely switch runways is 49%. The delay per aircraft, averaged over 5000 replications, is 1.26 min/aircraft. With the mathematical model of the $M/D/2$ queue, we expect to see a delay of 1.30 min/aircraft. Vandevenne and Lippert also state that about one-half of the traffic is expected to change runways to reduce its delay. Our simulation shows that this is nearly the case because an average of 45% of the traffic switched runways.

Note that we see a difference between the mathematical model as compared to the simulation for both the $M/D/1$ and $M/D/2$ queues. We attribute the differences in the delay to our input process model. Our simulation uses a fixed number of aircraft entering into the system. This number is fixed to be the expected number of arrivals for a homogeneous Poisson process within a 90-min interval (the expected number of aircraft for the Poisson distribution is $N = \lambda T$). The arrival time distribution for the Nth arrival can be shown to be a gamma distribution with mean N/λ and standard deviation $\sqrt{(N)}/\lambda$. Although we expect the Nth aircraft to arrive at time T, it will most likely arrive within some interval $T \pm \sqrt{(N)}/\lambda$. A result of modeling the arrival process with a fixed number of arrivals means that the length of the arrival period will vary. Hence, there is not a direct correlation between the results generated by the simulation and the analytical model. We believe that fixing the number of arrivals more accurately approximates the actual arrival process as compared to modeling an arrival period with a fixed length where the number of arrivals varies significantly. This is because the number of aircraft in an arrival rush is essentially fixed and the daily variations in traffic and weather cause an expansion or contraction of the duration of the arrival rush. Our simulations confirm that as the number of arrivals increases (either due to increasing the arrival rate or the number of runways), there will be a larger discrepancy with the analytical model. In addition, simulation shows that the fixed-increment Poisson model better approximates the analytical model formulation. Using the unlimited-crossover case with the fixed-length Poisson process, the average delay for 5000 replications is 1.29 min/aircraft. This compares to the 1.30 min/aircraft predicted by the analytical model.

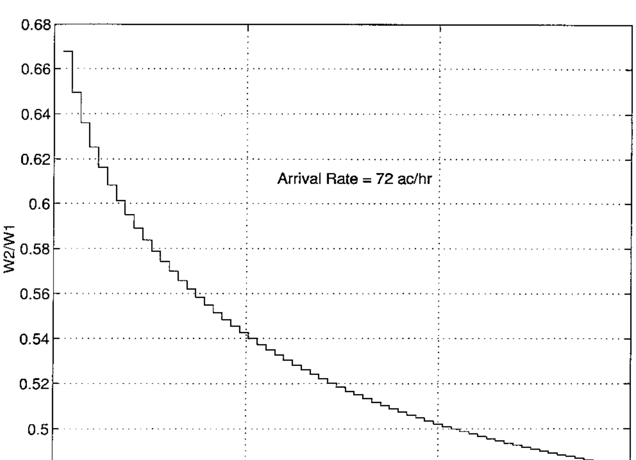

Fig. 3 Two-runway moderate-traffic case: ratio of expected delay for unconstrained runway crossover to no crossover expected delay (time measured from start of traffic).

The next approach that was implemented placed restrictions on when an aircraft could cross over. An aircraft would be allowed to cross over if one of the following logic statements were true: 1) the aircraft had a lower delay on the alternate runway and the aircraft formed a favorable sequence or 2) the scheduled time of arrival (STA) on the alternate runway is less than the STA on the preferred runway by a fixed amount (in this case, taken to be 60 s). A favorable sequence is defined as a sequence that is not one of the following pairs: {heavy, large}, {heavy, small}, or {large, small}. This essentially prohibits the use of the elements in the separation matrix that are above the diagonal. These are the elements that have the largest value and, hence, add the most delay to the landing sequence. The purpose of having the OR logic is that if the improvement is significant enough, it will offset any penalty that may result from an unfavorable sequencing. Simulation showed that the delay per aircraft was 1.37 min/aircraft. The delay is increased by about 6 s/aircraft as compared to the unlimited-crossover case. The increased delays can be because there are fewer crossovers, hence, there are aircraft that are not landing in their optimal slot. Furthermore, there are still instances where the sequencing is not favorable as we have defined it; hence, the larger separations on average will require larger delays. However, the number of runway crossings dropped to 23% of the traffic.

The next scheduling algorithm studied was to assign the heavy and small aircraft to separate runways. The large aircraft in the stream are then scheduled to the runway that minimizes the delay for each particular aircraft. The small aircraft were assigned to land on the left runway and the heavy aircraft were assigned to the right runway. The large aircraft go to the runway where the delay for that particular aircraft is the lowest. If the delay is the same on each runway for an aircraft then it lands on the runway where the sequence is defined as favorable (see preceding text). Here, we are trying to avoid putting the aircraft behind a heavy, when it could be placed behind a small or large, aircraft. However, if there still is no preference after this test (e.g., a large aircraft landed on each runway preceding the current large aircraft), then the aircraft either goes to the runway where there are fewer aircraft or to the runway where the last aircraft was not scheduled. For example, if the preceding aircraft landed on the right runway, then it will land on the left runway. The study of 5000 replications shows that the average delay per aircraft is 1.24 min/aircraft. The improvement in delay is significant as compared to the no-crossover case, and a modest improvement over the unlimited-crossover case. The improvement can be attributed to an increase in the capacity for each of the runways. Because heavy and small aircraft are not in the same stream, the large separations between these weight classes are eliminated, hence, the capacity increases. This method, however, had a large number of crossovers with 50% of the traffic switching runways. The reason for this is simple. Because we know that every aircraft entering the system wants to land on a preferred runway, it stands to reason that there is a 50% probability that the assigned runway for each heavy and small aircraft is its preferred runway. Therefore, one-half of the aircraft that comprise these weight classes have to change runways to land on the appropriate runway.

Table 4 Simulation results for heavy traffic and two runways

Allocation strategy	Average delay, min/aircraft	% Crossovers
No crossovers	8.3157	0
Unconstrained crossovers	6.4525	49
Constrained crossovers	6.5737	26
Separate heavy and small aircraft	5.7951	50

Furthermore, one-half of the large aircraft will switch to reduce delays.

Heavy Traffic

This section addresses the problem of what occurs in the two-runway case when the airport is operating above capacity. An average arrival interval of 90 min is considered, consistent with earlier cases, although rushes with an arrival rate with this intensity typically will last no longer than 15–20 min. The average delays for a numerical simulation consisting of 5000 replications with 216 aircraft arriving during the arrival period (an arrival rate of 144 aircraft/h) are given in Table 4. The scheduling algorithms are the same as considered for the moderate traffic case.

The no-crossover case is again the worst-case scenario to which all other scheduling methods are compared. The mean delay for the $M/D/1$ queue with an arrival rate of 48 aircraft/h and a service time of 82.8 s is 7.4027 min/aircraft. The delay found from numerical simulation was 8.32 min/aircraft. The 95% confidence interval half-width for the 5000 replications was found to be 0.06 min.

The second approach is the unconstrained crossover case whereby the aircraft moves from its preferred runway to its alternate runway if its delay would improve. The simulation returned a result of 6.45 min/aircraft, which is a 22% improvement over the no-crossover case. This is significantly higher than the 6.1517 min/aircraft that is predicted by the $M/D/2$ queuing model. This improvement is less than what was seen for the case of moderate traffic. The reason for seeing less of an improvement is likely due to the decreased spacing between the arrivals. The heavy traffic causes longer queues to form on each runway, resulting in smaller savings when an aircraft changes runways. Note that there is a disagreement between the simulation results and the analytical model that is due to the differences in the arrival process.

The next scheduling approach is the constrained-crossover case discussed in the preceding section. As expected, the average delay is higher than that for the unlimited-crossover case. This strategy had a delay 6.57 min/aircraft as compared to 6.45 min/aircraft for the unlimited crossovers. The number of crossovers as compared to the moderate traffic is also slightly higher. With the increase in traffic, 26% of the aircraft switched runways. This increase is associated with the decreased mean separation in the ETA of the aircraft and the longer queues that will form.

The final allocation process was to separate the heavy and small traffic so that each lands on separate runways. In this case, the delay is reduced to 5.80 min/aircraft. This is significantly less than both the unconstrained- and no-crossover cases. This demonstrates the importance of keeping heavy and small aircraft on separate runways when the traffic is very heavy. The reduction in delay is attributed to the heavy–small sequence being avoided. Even though the large aircraft make up 70% of the total traffic, forcing the separation of the small number of heavy and small aircraft significantly reduces the delay as compared to the unconstrained-crossover case. This is an indication that the scheme for reducing individual delay results in a landing sequence that is a local and not a global optimum.

Three-Runway Allocation Problem

The three-runway case is considered because many larger airports such as Dallas–Fort Worth and Denver International have more than two runways that may be used simultaneously. Only heavy and moderate traffic are considered because only minimal benefits are realized from optimizing runway allocations for light traffic. The most practical means of allocating runways in the light traffic case is to land each aircraft on its preferred runway. The underlying assumptions for the three-runway case are basically the same as for the two-runway case. The three runways are labeled as R, L, and C to denote the right, left, and center runways, respectively. Each aircraft is assigned a runway using a random draw, with each aircraft having an equal probability of being assigned to any of the three runways.

Moderate Traffic

The case of a moderate traffic flow into the airport is discussed first. Three scheduling strategies are examined. The first is the no-crossover case, where each aircraft is assigned to its preferred runway, and the second is the unlimited-crossover case where an aircraft is free to switch runways whenever the delay on one of the alternate runways is lower than the delay on the preferred runway. The third way of scheduling is to separate the aircraft by weight class, landing heavy and small aircraft on separate runways, while assigning the large aircraft to any of the three. This is a direct descendant of the two-runway strategy where the heavy and small aircraft were landed on separate runaways. For the moderate traffic case, it is assumed that the total arrival rate is 108 aircraft/h. Initially there is no queue and the traffic stops arriving after the 162nd arrival. The total runway capacity is 130 aircraft/h for the assumed traffic mix. Results are summarized in Table 5.

The no-crossover case is again compared directly to an $M/D/1$ queue that has an arrival rate of 36 aircraft/h and a service time of 82.8 s. The analytical model predicts an expected delay of 2.5247 min/aircraft over the rush period. Simulation, however, yielded a delay of 2.6089 min/aircraft. The unrestricted crossover case performed as expected. Figure 4 shows the ratio of waiting time for an $M/D/3$ to an $M/D/1$ queue over time. Note that the delay for a single-server queue increases faster than for the three-server queue given the same utilization. It is expected that the delay will be 35% of the no-crossover delay. The delay for the unlimited-crossover case is 0.7520 min/aircraft. The delay that one would expect from the $M/D/3$ queue is 0.8413 min/aircraft. The expected delay from the analytical model is about 12% higher than what the simulation predicts. However, it is 30% of the simulated no-crossover delay. As before, the difference in delay can be attributed to using a fixed

Table 5 Simulation results for three runways and moderate traffic

Allocation strategy	Average delay, min/aircraft	% Crossovers
No crossovers	2.6089	0
Unconstrained crossovers	0.8119	59
Separate heavy and small aircraft	0.8324	67

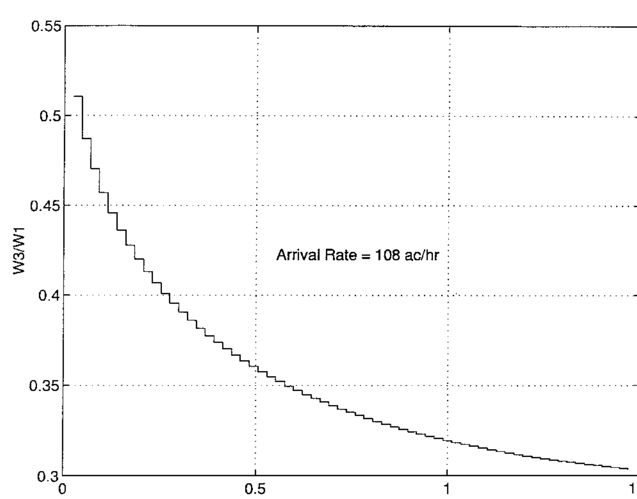

Fig. 4 Three-runway moderate-traffic case: ratio of expected delay for unconstrained runway crossover to no crossover expected delay.

Table 6 Simulation results for heavy traffic and three runways

Allocation strategy	Average delay, min/aircraft	% Crossovers
No crossovers	8.4145	0
Unconstrained crossovers	5.6691	66
Separate heavy and small aircraft	4.5246	67

number of arrivals, rather than a fixed end time. One would expect to see two-thirds of the traffic crossover to an alternate runway because the probability of an aircraft having its preferred runway be the runway that has the lowest delay is one-third. The actual crossover rate was 59%, less than the 67% that would be anticipated. The third allocation method is to land the heavy aircraft and the small aircraft on separate runways. Large aircraft are then assigned to any of the three runways. To be consistent with the allocation strategy for the two-runway case, the large aircraft landed on the runway that minimized the delay for an individual aircraft. Simulation yielded a delay of 0.8324 min/aircraft with 67% of traffic crossing over. This is similar to what was observed on the two-runway case with moderate traffic, but with a very small increase in the delay.

Heavy Traffic

For heavy traffic, the arrival rate was increased to 144 aircraft/h, which gives us 216 aircraft in the rush period. The three strategies employed are the same as for the moderate traffic. Again, comparisons are made to results obtained using queuing theory to predict the delays as well as the improvement in the delay. Table 6 summarizes the results of this section.

For the case of no crossovers, the average delay was found from simulation to be 8.7364 min/aircraft. This compares to the analytical model that has an average delay of 7.1426 min/aircraft. The reason for the discrepancy is discussed in the two-runway/heavy-traffic study. The unlimited-crossover case sees a reduction in the delay as expected. The average delay from the simulation is 5.4203 min/aircraft with 66% of the traffic crossing over. The delay reduction realized by allowing the aircraft to land wherever their delay is minimized is 35%. The $M/D/3$ model predicts an average delay of 5.5834 min/aircraft. Furthermore, from Fig. 5, one would expect the ratio of the no-crossover delay to the unconstrained-crossover delay to be 0.73. The ratio of simulated delays is 0.67, which varies from the analytical model.

The final strategy employed is to land the heavy and small aircraft on separate runways and to land the large aircraft on whichever runway's delay is the smallest. The delay calculated from the simulation is 4.5286 min/aircraft, with 67% of the aircraft switching runways. As with the two-runway setup with heavy traffic, this instance is similar in terms of relative performance. The separation of the weight classes removes some of the components of the separation matrix that result in large delays. As in the two-runway case, this is even more important when the traffic is heavy because bunching is widespread.

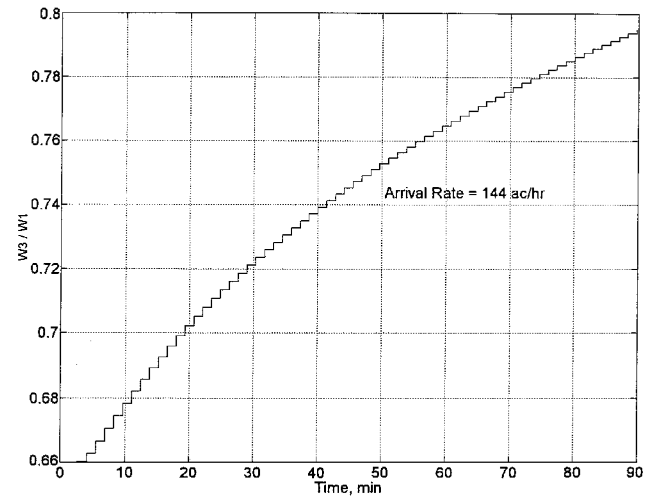

Fig. 5 Three-runway heavy-traffic case: ratio of expected delay for unconstrained runway crossover to no crossover expected delay.

Conclusions

Several methods for scheduling arrival aircraft to multiple runways are studied. We have shown that the transient analysis of an $M/D/n$ queue can give reasonable results in predicting the average delay per aircraft when the runway capacity is known. Furthermore, significant improvements are realizable when one considers the arrival rate in choosing a runway allocation strategy. The greatest reduction in delay for both the two- and three-runway cases for heavy traffic are obtained by separating traffic such that the heavy and small weight classes do not interact. For more moderate traffic, one may either split the traffic by weight class or crossover when there is an improvement in delay. Light traffic simply is scheduled to the preferred runway for the aircraft because the average separation is large enough that most aircraft are likely to be expedited.

Acknowledgment

This research was supported by NASA Ames Research Center Cooperative Agreement NCC2-669.

References

[1]Neuman, F., and Erzberger, H., "Analysis of Delay Reducing and Fuel Saving Sequencing Algorithms for Arrival Traffic," NASA TM 103880, Oct. 1991.

[2]Vandevenne, H., and Lippert, M., "Benefits from an Algorithm for Better Multiple Runway Allocation," 41L-04126, Lincoln Lab., Massachusetts Inst. of Technology, Cambridge, MA, June 1993.

[3]Cooper, R. B., *Introduction to Queueing Theory*, 2nd ed., North-Holland, Amsterdam, 1981, pp. 178–185.

[4]Tijms, H. C., *Stochastic Models: An Algorithmic Approach*, Wiley, New York 1994, pp. 288–292.

[5]Hoel, P. G., Port, S. C., and Stone, C. J., *Introduction to Stochastic Processes*, Waveland Press, Prospect Heights, IL, 1987, pp. 84–110.