NASA-CR-200886

Evaluation of Scheduling Methods for Multiple Runways

Michael A. Bolender   G.L. Slater
Department of Aerospace Engineering and Engineering Mechanics
University of Cincinnati
Cincinnati, OH 45221-0070

April 19, 1996

Evaluation of Scheduling Methods for Multiple Runways

Michael A. Bolender   G.L. Slater
Department of Aerospace Engineering and Engineering Mechanics
University of Cincinnati
Cincinnati, OH 45221-0070

April 19, 1996

Abstract

Several scheduling strategies are analyzed in order to determine the most efficient means of scheduling aircraft when multiple runways are operational and the airport is operating at different utilization rates. The study compares simulation data 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 since 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 upon 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.

1 Introduction

The analysis of aircraft scheduling techniques for airports with multiple runways are becoming more important with the evolution of new airport designs, such as Denver International (DIA), that have the capability to land several aircraft independently on several runways. Therefore, new techniques for scheduling to multiple runways are needed in order to improve upon the traditional First Come First Serve (FCFS) technique generally employed. With the advent of the Center TRACON Automation System (CTAS), air traffic controllers will have a tool that gives them accurate aircraft state information that will assist them in their scheduling duties [1]. The intent of this paper is to present and compare several scheduling methods in order 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 upon the geometric relation of the approach geometry to a runway. Previous efforts by Vandevanne [2] have shown that significant delay reduction is possible for multiple runways if the aircraft are allowed to crossover without penalty. Using steady-state queuing theory, one should realize an improvement in delay by a factor of approximately $1/n$ for n runways as compared to a single runway case with $1 - 1/n$ percent of the aircraft switching from their preferred runway. A "delay threshold" can be added in order to reduce the number of crossovers. The delay threshold is a lower bound upon which the delay on the alternate runway must be reduced in order for the aircraft to cross to that runway. As a result, 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 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 the 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 builds-up during a rush period. This simulates the queuing dynamics during a rush period at a typical airport which 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.

2 The Scheduling Problem

The aircraft scheduling problem can be defined as a procedure which 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. First is that the arrival times of the aircraft into the system are random. Theoretically and practically, the arrival times are modelled as a Poisson process. In practice, if one were to observe arrivals at an airport for a day, one would see that the number of arrivals varies throughout the day. There are periods of time where the arrival traffic is "light" and periods where the incoming traffic is so heavy that the airport is operating near or above capacity. The arrivals are still consistent with the Poisson process, but with a time varying arrival rate.

A practical factor which is of extreme importance in scheduling is classification of aircraft into different weight classes, and the minimum separation between them. In practice we generally, deal with three weight classes which we describe as heavy, small, and large. The Federal Aviation Administration (FAA) has specified a "separation matrix" which 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 given below (with distances in n. mi.).

$$
\begin{array}{c|ccc}
 & H & L & S \\
H & 4 & 5 & 6 \\
L & 3 & 3 & 4 \\
S & 3 & 3 & 3 \\
\end{array}
\tag{1}
$$

This matrix changes depending upon 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. One converts the distances to times using the approach speeds of the aircraft.

3 Analytical Models

In order 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 the peak arrival rates may be short compared to the time required for the system to reach steady- state. The real benefits of an efficient scheduling technique are realized when the arrival traffic is heavy, and there are times where the peak arrival rate of aircraft is greater than the number that are able to land in a given time period. Secondly, the arrival rate preceding the rush period is usually low enough that aircraft are sufficiently spaced, and the method of landing aircraft on their preferred runways will be more than adequate since aircraft are typically not delayed due to the large inter- arrival times between aircraft.

3.1 Deterministic Service Times

In constructing a mathematical model for the scheduling problem, one needs to make some simplifying assumptions. The first is that the arrivals are to be modelled 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 inter-arrival 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 [3], where the "M" denotes that the inter-arrival times are "Markovian" or "memoryless," the "D" denotes that the service times are "deterministic" or constant, and n servers are operating in parallel. All aircraft will share a common queue, unless specified otherwise.

The service time may be taken to be constant by averaging the actual separation times within the separation matrix. This may be done since the traffic mix and the separation matrix are known quantities. 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. In order 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 $T_S = P_m^T S P_m$, $P_m = [P_H \ P_L \ P_S]^T$ is the traffic mix (P_H, P_L, and P_S are the probabilities that the aircraft is a heavy, large, or small respectively), 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 $P_m = [.2 \ .7 \ .1]^T$, and the aircraft have a common landing speed of 150 knots, then $\mu = 43.5$ ac/hr and $T_s = 82.8$ sec. For analysis purposes, using

this constant T_s allows us to preserve the effects of different traffic mixes upon the delay in the system while still using the simplifying assumption of a constant deterministic service. An alternate approach which utilizes random service times is discussed in the next section. Whereas capacity is affected by the order of arrivals of various weight classes, one can safely assume that no major re-ordering of the landing times is possible (or desired.) To understand how the landing order affects capacity, consider landing all small aircraft, followed by the large and then the heavy aircraft. This will maximize the capacity, but will likely result in large delays for a large percentage of the aircraft in the stream [4].

In order to analyze the delay build-up during a rush period, one needs to study the transient probabilities of the queuing process. The time-varying equations are taken from Tijms [5]. They are based upon the following observation: 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 $[t, t+T_s]$ (because we consider a Poisson process with a constant rate, there is no time dependency). We write the number of arrivals as a function of the length of the interval since the Poisson process has the following property: for $0 \leq s \leq t$ the random variable A(t)-A(s) is the number of arrivals in the interval $[s, t]$, which may be written as $A(t - s)$ [6]. Furthermore, let $N(t)$ be the number in the system at time t, and $P_j(t) = P(N(t) = j)$ denote the probability that j customers are in the system at time t. We will condition on the number 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 number of aircraft in the system

$$P_j(t + T_s) = P(A(T_s) = j | N(t) = 0)P_n(t) \cup \ldots \cup P(A(T_s) = j | N(t) = n)P_n(t) \cup \quad (2)$$
$$P(A(T_s) = j - 1 | N(t) = n + 1)P_{n+1}(t) \cup \ldots \cup$$
$$P(A(T_s) = 0 | N(t) = j + n)P_{j+n}(t)$$

Also, note that the number of arrivals in the interval $[t, t + T_s]$ and the number in the queue are independent events. Thus, for any m and k, the conditional probability above becomes

$$P(A(T_s) = m | N(t) = k) = \frac{P(A(T_s) = m, N(t) = k)}{P(N(t) = k)}$$
$$= \frac{P(A(T_s) = m)P(N(t) = k)}{P(N(t) = k)} = P(A(T_s) = m)$$
$$= e^{-\lambda T_s} \frac{(\lambda T_s)^m}{m!} \quad (3)$$

The probability given in Equation 3 is simply the probability that there are m Poisson arrivals in an interval of length T_s. Substituting Equation 3 into Equation 3 and simplifying

yields

$$P_j(t + T_s) = \sum_{k=0}^{n} P_k(t)e^{-\lambda T_s}\frac{(\lambda T_s)^j}{j!} + \sum_{k=n+1}^{n+j} P_k(t)e^{-\lambda T_s}\frac{(\lambda T_s)^{j-k+n}}{(j-k+n)!}, \quad j = 0, 1, 2, \ldots \quad (4)$$

This gives us an infinite set of equations that can be solved at discrete times. Assuming that the queue is initially empty, this set of equations can then be re-written in the matrix-vector form $\bar{P}(t + T_s) = F\bar{P}(t)$. The vector $\bar{P}$ is the probability vector, where the jth element is the probability that $j - 1$ customers are in the system. This equation can be solved by setting $t = kT_s$, and using the initial condition $\bar{P}_0 = \bar{P}(0) = [1\,0\,0\,0:\ldots]^T$. Re-writing the probability vector, we get $\bar{P}(k) = \bar{P}(kT_s) = [P_0(k)\,P_1(k)\ldots]$. Hence, Eq. 4 can be written as the infinite dimensional difference equation,

$$\bar{P}(kT_s) = F\bar{P}((k-1)T_s) \quad (5)$$

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

$$F = e^{-\lambda T_s}\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ \lambda T_s & \lambda T_s & \lambda T_s & 1 & 0 \\ \frac{(\lambda T_s)^2}{2!} & \frac{(\lambda T_s)^2}{2!} & \frac{(\lambda T_s)^2}{2!} & \lambda T_s & 1 \\ \vdots & \vdots & \vdots & \ddots & \ddots \\ \frac{(\lambda T_s)^i}{i!} & \frac{(\lambda T_s)^i}{i!} & \frac{(\lambda T_s)^i}{i!} & \frac{(\lambda T_s)^{i-1}}{(i-1)!} & \ddots \end{bmatrix} \quad (6)$$

The solution to this set of equations is

$$\bar{P}(kT_s) = F^k\bar{P}_0 \quad (7)$$

This set is solved approximately by choosing a sufficiently large dimension of F such that the significant probabilities of the system are captured.

After solving for the time-varying probabilities, the mean number in the system at any time increment k is calculated. The mean number in the system is defined as $m(k) = \sum_{j=0}^{\infty} jP_j(k)$. The mean number in the system may be broken up into two components, those found in service at time-increment k, m_s, and those in the queue awaiting service, m_Q. Hence, $m(k) = m_s(k) + m_Q(k)$, where $m_Q(k)$ is the mean number in the queue and $m_s(k)$ is the mean number in service. The mean number in service can be found in Cooper [3] to be

$$\sum_{j=0}^{n-1} jP_j(k) + n\sum_{j=n}^{\infty} P_j(k) \quad (8)$$

The first summation in Eq. 8 arises from the recognition that 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 due to the realization that 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

$$m_Q(k) = \sum_{j=n}^{\infty}(j - n)P_j(k) \quad (9)$$

5

Using Equation 9 and replacing ∞ by N_F, where N_F is the dimension of 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 [3]. Little's Formula provides a simple relation between the waiting time in the queue and the number in the queue. 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 the waiting time in the queue. The waiting time in the queue then becomes

$$W_Q(k) = \frac{1}{\lambda} m_Q(k) = \frac{1}{\lambda} \sum_{j=n}^{N_F} (j - n) P_j(k) \tag{10}$$

3.2 Exponential Service Times

A second model that has been applied to analyze the aircraft 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 Kolomogrov differential-difference equations, which describe a birth and death process, were numerically solved to get the probabilities for an n server queue with a constant arrival rate $\lambda_j = \lambda$ for all j and service rates

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

The service rate denotes how quickly customers would complete service in a specified period of time. The birth and death differential-difference equations are then

$$\dot{P}_j(t) = \lambda_{j-1} P_{j-1}(t) - (\lambda_j + \mu_j) P_j(t) + \mu_{j+1} P_{j+1}(t) \tag{12}$$

Since this is an infinite set of first-order differential equations, we can write this in the form $\dot{P}(t) = G\bar{P}(t)$, $\bar{P}(0) = \bar{P}_0$. The matrix G in this case is a tri-diagonal matrix of the form (for $n = 2$)

$$G = \begin{bmatrix} -\lambda & \mu & 0 & 0 & 0 \\ \lambda & -(\lambda + \mu) & 2\mu & 0 & 0 \\ 0 & \lambda & -(\lambda + 2\mu) & 2\mu & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix} \tag{13}$$

The elements $P_j(t)$ of the vector $\bar{P}(t)$ are simply the probabilities that $j - 1$ customers are in the system at time t. Again, we are only able to approximate the infinite set of differential-difference 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 $\dot{P}(t) = G\bar{P}(t)$, $\bar{P}(0) = \bar{P}_0$ is

$$\bar{P}(t) = e^{Gt} \bar{P}_0 \tag{14}$$

6

Once the probabilities are found according to Eq. 14, the mean number in the queue and hence the mean waiting time in the queue are found using Eq. 10 and replacing k by t. One can see in Figure 1 that the waiting time in the queue for the M/M/2 queue is almost twice that when compared to the M/D/2 queue for the same service time, arrival rate, and number of servers. The reason for this difference is attributed to the large standard deviation of the exponential distribution. Consider an exponential distribution with rate α. The mean is then $1/\alpha$ and the variance is $1/\alpha^2$. This results in a large 1σ deviation, where we would expect to see the service times between 0 and $2/\alpha$. Note for a purely deterministic service time, the variance is zero. If we consider the service times to determined by the separation matrix, the variance in arrival traffic can be easily computed. After converting the separation matrix from distances to speeds using a common approach speed of 150 kts and the traffic mix above, the standard deviation is found to be 19.3 seconds, compared to 82.8 seconds for the exponential distribution. The large variance of the exponential distribution introduces a much wider range of service times than what occurs in practice. The effect of these service times is to introduce additional delay into the system that is not present. Hence, the deterministic service time queue better suits our results.

4 Comparison of Runway Allocation Strategies

Due to 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. First, 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.

4.1 Two Runway Allocation Problem

The two runway problem is one that is quite common at many airports which operate at least two independent runways. Runways that operate independently of one another have sufficient separation between their center lines such that aircraft landing simultaneously do not have to be "staggered."

It is assumed for all scheduling strategies that the aircraft arrive from two different directions. Each arrival stream's estimated times of arrival (ETAs) are modelled by a
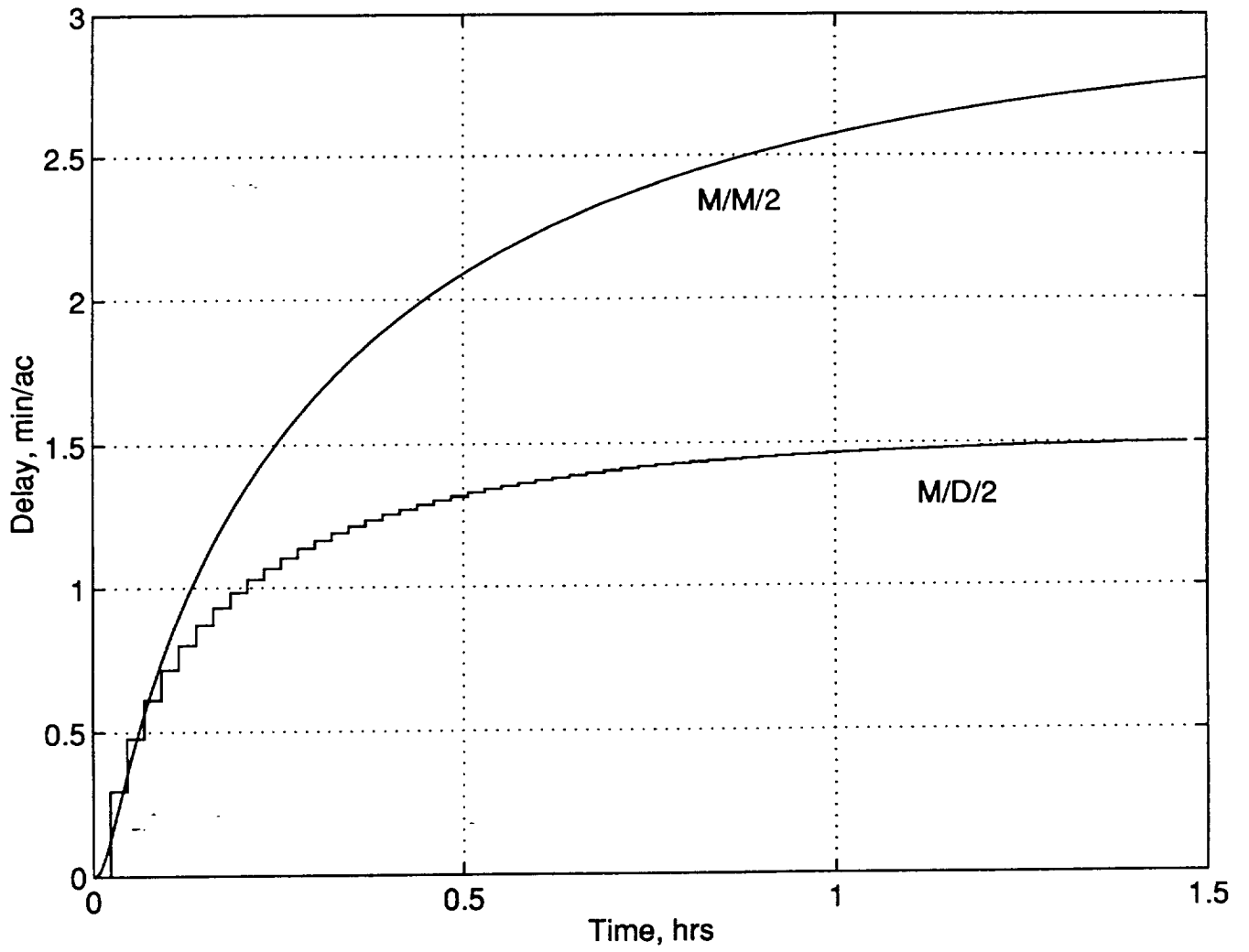
Figure 1: Mean Waiting Times for M/D/2 and M/M/2 Queues

Poisson distribution with a mean of λ aircraft per hour per runway, which gives a total arrival rate of 2λ ac/hr using the reproductive property of the Poisson process. Each arrival direction has a "preferred" runway that an aircraft desires to land on. Due to 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 from a standard normal distribution. The capacity of each runway is approximately 43.5 aircraft per hour using the separation matrix and (for simplicity) a common approach speed of 150 knots calibrated airspeed. The traffic mix is assumed to consist of 70% large aircraft, 20% heavy, and 10% small. The performance index to be considered is the average delay of each aircraft, since minimizing the delay per aircraft results in a maximum throughput. The delay per aircraft is measured with respect to an earliest estimated time of arrival (ETA), such that an aircraft that arrives at its "fast" ETA has zero delay. It is further assumed that each aircraft can be expedited by 60 seconds (i.e. $ETA_{fast} = ETA - 60$) and 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. Furthermore, the first aircraft landing on each runway is constrained to land at its nominal time of arrival in order to prevent negative landing times. The results presented are the average for "batch" runs comprised of 500 different streams that are each 90 minutes long. All scheduling strategies for a given arrival rate use the same traffic.

4.1.1 Light Traffic

For the light traffic case, the total arrival rate is taken to be 32 ac/hr (or 16 ac/hr/runway). In queuing theory, the "utilization rate" is used to demonstrate the how "busy" a system is [3]. The utilization rate is defined to be $\rho = \frac{\lambda}{n\mu}$, where ρ is the utilization rate, λ is the arrival rate, n is the number of servers, and μ is the service rate. If we substitute the service time for the service rate, the utilization becomes $\rho = \frac{\lambda T_s}{n}$. Therefore, for this case of light traffic, $\rho = .37$, which means that the runways will be occupied 37% of the time. 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, since 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 service. This is identical to a supermarket with two checkouts, each with its own line, where the customers, upon entering the line, cannot go to another register. The second 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 queue is unlike the one above in that there is only one line, but the customer chooses the server that becomes open the soonest (i.e. the baggage check-in counter at the airport.) This plan will be referred to as unconstrained crossovers. The final allocation strategy is to land the heavy and small aircraft on runways which are designated for this weight class, and to place the large aircraft on the runway where the delay for it is the smallest. The

9

Allocation Strategy	Ave Delay, min/ac	Std Dev. min/ac
No Crossovers	0.4725	0.2022
Unconstrained Crossovers	0.1847	0.0963
Separate Heavies and Smalls	0.2359	0.1080

Table 1: Light Traffic Comparison for 2 Runways

results of these three approaches are given in Table 1. From Table 1 we conclude that an aircraft is likely to be expedited, even in the case where the airplane cannot cross runways (recall that 60 seconds of delay corresponds to the aircraft arriving at its nominal ETA). The improvements made by allocating runways are 50% to 65% percent better. However, from an operational point of view, there is no real advantage for optimizing the landing sequence in order to reduce the delay per aircraft since the delay is already small. This is due to the fact that the average separations between arrivals are large, hence there is little tendency for bunching to occur.

In order to get a feel of how accurate the numbers for the no crossover and the unlimited crossover cases, we can compare them to an M/D/1 and an M/D/2 queue respectively. In order to calculate the expected delay per aircraft over a given time period, the "average" value of Equation 10 is needed. To calculate this, note that the waiting time is constant over a service period. Therefore, the average value of the expected waiting time curve is then

$$\bar{W} = \frac{1}{T} \int_0^T W(t)dt = \frac{1}{N} \sum_{k=0}^{N} W(k) \tag{15}$$

Using Equation 15 for an M/D/1 queue with an arrival rate of 16 ac/hour and a service time of 82.8 sec, the average delay is found to be 0.3955 min/ac, which agrees well with the no crossover case in Table 1. The unlimited crossover case shows the same trend. The predicted delay using an M/D/2 queue is 0.1174 min/ac while the simulation produced a delay of 0.1847 min/ac. Since the differences between the simulation and the predicted results are nearly identical, these quantities appear to give a suitable representation of what can be expected when the traffic is light.

4.1.2 Moderate Traffic

The case where there is moderately heavy traffic allows us to investigate into 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 ac/hr, putting the airport at about 84% capacity. Results are summarized in Table 2.

Allocation Strategy	Ave Delay, min/ac	% Crossovers
No Crossovers	2.5161	0
Unconstrained Crossovers	1.3402	44.85
Constrained Crossovers	1.4712	23.12
Separate Heavies and Smalls	1.3140	50.17

Table 2: Moderate Traffic Comparison for 2 Runways

Four different scheduling algorithms were investigated. The first method is to again land each aircraft on its preferred runway (i.e. no crossovers allowed). This serves as a baseline strategy used determine how much improvement in delay can be obtained. The second strategy allowed an aircraft to cross from its preferred runway to the alternate runway if the aircraft could land at an earlier time on the alternate. The first two strategies correspond to the analytical models that are considered. Two additional algorithms also are considered. One of the methods attempts to reduce the number of crossovers. Because crossovers increase the workload of the controllers, one wants to be able to reduce delay without imposing a higher workload on them. Therefore, this particular algorithm permitted the aircraft to crossover to the alternate runway if one of two conditions were satisfied: a) the aircraft's delay on the alternate runway was less than on the preferred **and** the sequence was defined to be "favorable" or b) the aircraft's delay on the alternate runway was less than that on its preferred runway by some predetermined amount. The second alternative method is to see what improvements in delay may be realized by separating some of the traffic so it does not interact. Upon inspection of the separation matrix, it is evident that the element with the largest value is the case where a small aircraft trails a heavy. The goal is to then eliminate this sequence of aircraft. Thus, the strategy is to send the heavy aircraft and small aircraft to separate runways and to then schedule the large aircraft to the runway where its delay was lowest.

The first scheduling strategy employed was to restrict each incoming aircraft to land on its preferred runway. This is employed as a baseline in order 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 above, entered from the appropriate direction, and then were scheduled to the corresponding runway. For 500 runs of 108 aircraft, the mean delay was 2.59 min/ac. The average number of aircraft landing on each runway was 54.27 and 53.73 on the left and right respectively. Note that the amount of traffic is nearly evenly split between the runways. This is expected since the runway assignment is based upon the sign of a draw from a normal distribution. For analytical purposes, this is modelled by an $M/D/1$ queue with the arrival rate equal to 36 ac/hr and a constant service time of about 83 seconds. The expected delay curve is shown in Figure 2. Using Eq. 15, we find that the average value of the waiting time is 2.53 min. This is in agreement with the results of the simulation.

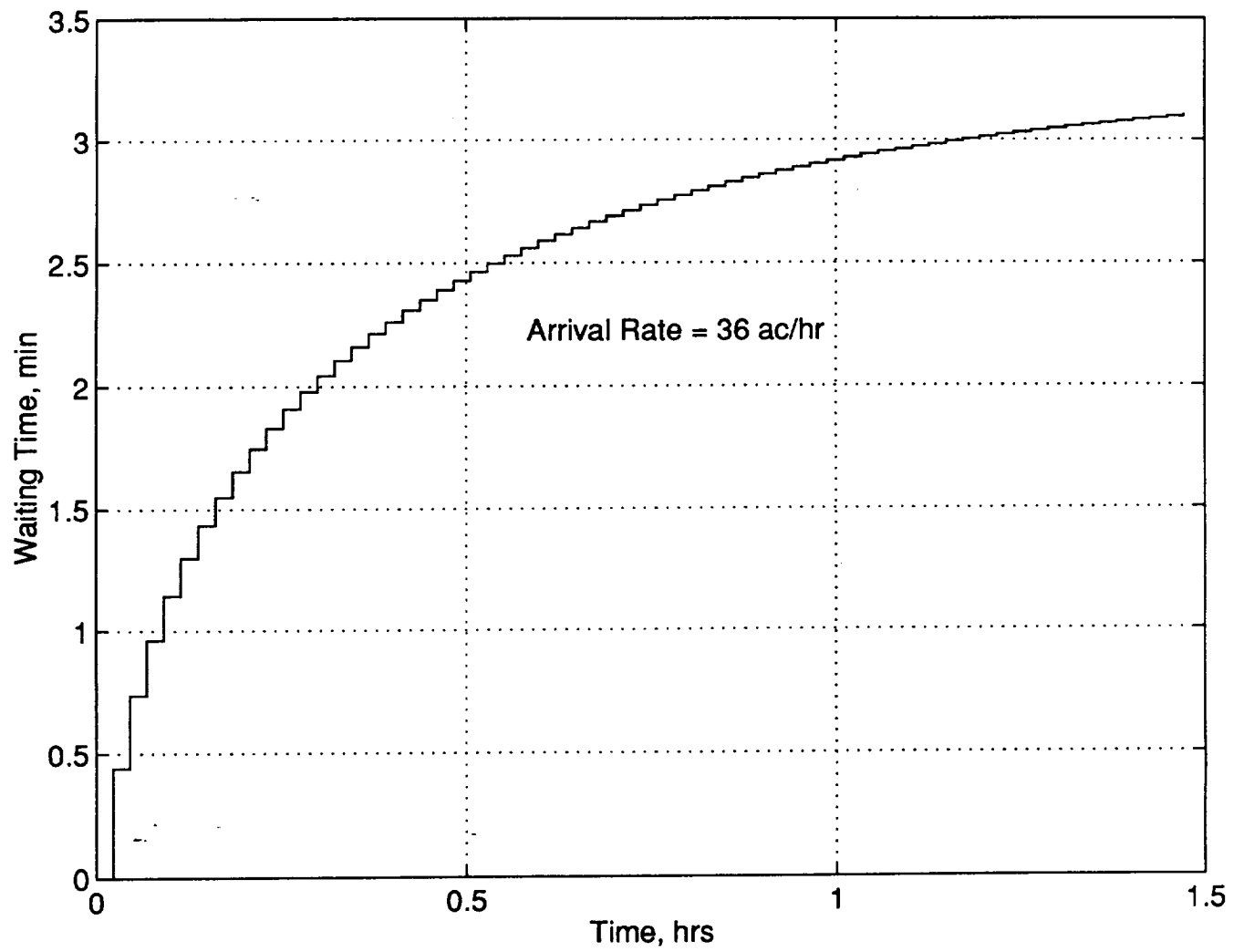The second strategy was to allow the aircraft to crossover when the delay on the

Figure 2: Mean Waiting Time for M/D/1 Queue with Moderate Traffic

alternate runway was less than on the preferred. This case was studied by Vandevanne [2] using traffic statistics from the Dallas-Fort Worth airport. Vandevanne studied the reduction in delay relative to the preferred runway case that was discussed above. His analysis looked at the expected waiting time in the steady state for $M/D/n$ queues as compared to an $M/D/1$ queues. He shows 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 that the delay of an $M/D/2$ queue relative to an $M/D/1$ queue with the same utilization. The arrival rate used is the same in this section for both queue types. Calculating the average value of the curve in Figure 3, an improvement of about 52% would be expected by allowing an aircraft to choose the runway with the lowest delay for it.

Simulation of a batch run of 500 streams shows that the average delay is halved as compared to the no crossover case. The delay per aircraft turns out to be 1.34 min/ac with an average of 53.90 aircraft landing on the left runway and 54.10 aircraft landing on the right. This shows that the effect of crossing runways in order to minimize the individual delay also will tend to balance the runway throughput. The average delay that one expects to see can be found using Eq. 15 derived above. Using this we find that the average delay should be about 1.31 min/ac. Furthermore, the simulated delay for the unlimited crossover case is 52% that of the no crossover case. Vandevanne also states that about one- half of the traffic will cross runways in order to reduce their delay. Our simulation shows that this is nearly the case, as 44.85% of the traffic switched runways in a single stream of 108 aircraft. The reason for the large number of crossovers can be attributed to the fact that an aircraft has a 50 percent probability of having its preferred runway be the one for which its delay is minimized.

The next approach that was implemented placed restrictions on when an aircraft could crossover. An aircraft was allowed to crossover if one of the following logic statements were true: a) the aircraft had a lower delay on the alternate runway and the aircraft formed a "favorable" sequence or b) the scheduled time of arrival (STA) on the alternate runway is less than the STA the on the preferred runway by a fixed amount (taken to be 60 seconds). 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, hence adding 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 1.47 min./ac and each runway landed an average of 54 aircraft. Similar to the unlimited crossover case, the runways are balanced, but the delay is increased by about 8 seconds per aircraft on each runway. The increased delays can be attributed to the fact that 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
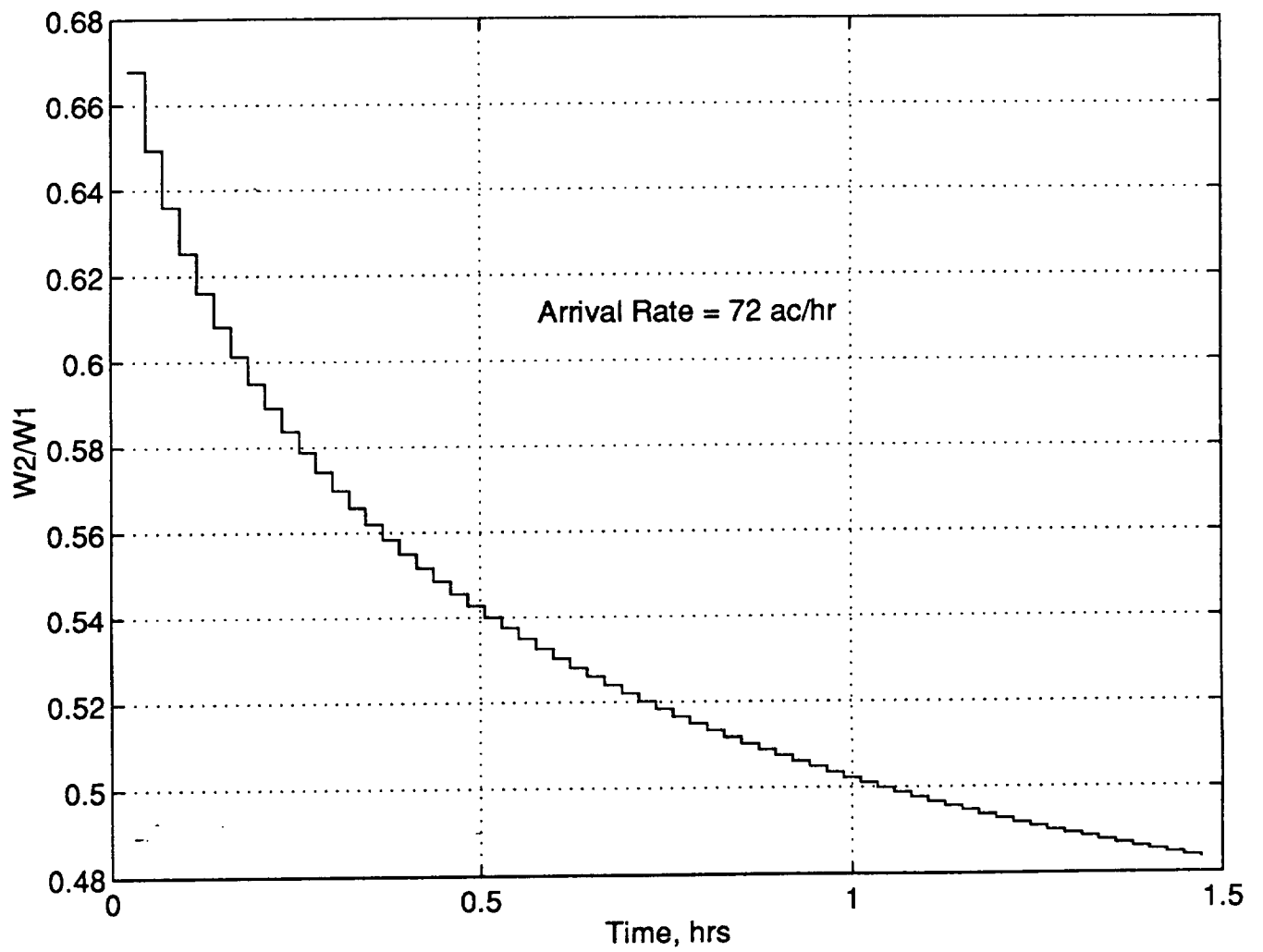
Figure 3: Ratio of 2 Runway Delay to 1 Runway Delay for Moderate Traffic

14

favorable as we have defined it, hence the larger separations on average will require larger delays. However, the number of runway crossings dropped to 23.12% of the traffic. The next scheduling algorithm studied was to split the heavy and small aircraft and to schedule them to separate runways. The large aircraft into the stream are subsequentially scheduled to the runway which minimizes the delay for the particular aircraft. The small aircraft were sent to the left runway and the heavies were sent to land on 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. 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 (e.g. if the previous aircraft landed on the right runway, then land on the left runway). The study of 500 runs shows that the average delay per aircraft is 1.3140 min/ac The average runway throughputs are 55.85 aircraft landing on the left and 52.15 on the right. In both cases, 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. Since heavy and small aircraft are not in the same stream, the large separations between these weight classes are eliminated, hence the capacity increase. This method, however, had the largest number of crossovers with about 50.17% of the traffic switching runways. The reason for this is simple. Since 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. Furthermore, one-half of the large aircraft will switch in order to reduce delays based upon the argument given in the section above.

4.1.3 Heavy Traffic

This section addresses the problem of what occurs in the two runway case when the airport is operating above capacity. An interval of 90 minutes is being considered, although in practice an airport never operates under such conditions for periods this long. The reason for choosing such a long interval is to keep continuity with the light and moderate traffic densities discussed above. The arrival rate is 96 ac/hr, and the average delays are for 500 runs of 144 aircraft. The scheduling algorithms are the same as considered for the moderate traffic. The results for the heavy traffic case are summarized in Table 3.

The no crossover case is again the worst case scenario to which all other scheduling methods are compared. The average delay expected from an M/D/1 queue with an arrival

Allocation Strategy	Ave Delay, min/ac	Crossovers %
No Crossovers	8.5629	0
Unconstrained Crossovers	6.8800	49.24
Constrained Crossovers	7.0772	26.04
Separate Heavies and Smalls	6.1792	49.94

Table 3: Heavy Traffic Comparison for 2 Runways

rate of 48 ac/hr and an service time of 82.8 sec is 7.4027 min/ac. The simulation had an average delay of 8.5629 min/ac. The difference between the expected and the simulated delay is attributed to the approximation of the infinite-dimensional system by one that is finite. To begin with, since the arrival rate is larger than the service rate, the system will never reach steady-state. This means that if the system were to run for an infinite amount of time, the queue length would become infinite. Therefore, by estimating the infinite-dimensional system with one that is finite, large errors have been introduced in the expected waiting time calculations. However, what we are really looking for here is the improvement in delay relative to the M/D/1 case; therefore, based on our previous experience, we would expect to see the same relative improvement below.

In the unconstrained crossover case, an improvement of 18% in the delay should be realized when compared to the no crossover case. The expected delay found using the M/D/2 queue is 6.1517 min/ac. The simulation returned a result of 6.8800 min/ac, which is almost a 20% improvement over the no crossover case. Figure 4 shows how delay is reduced relative to the single runway case. Although the computed delays are not close to the expected delay from queuing theory, the relative improvement is reasonably close.

The next scheduling approach is the constrained crossover case discussed in the section above. As expected, the average delay is higher than that for the unlimited crossover case. This strategy had a delay 7.0772 min/ac as compared to 6.8800 min/ac 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's of the aircraft.

The final allocation process was to separate the heavy and small traffic so that each lands on separate runways. The delay was found to be 6.1792 min/ac. This is significantly less than the no crossover case. In fact, 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 fact that the heavy-small sequence is avoided. Furthermore, the large aircraft, which make up 70% of the total traffic, land wherever the delay is minimized, hence this likely accounts for a part of the reduction in the delay. Approximately 49% of the traffic switched runways.
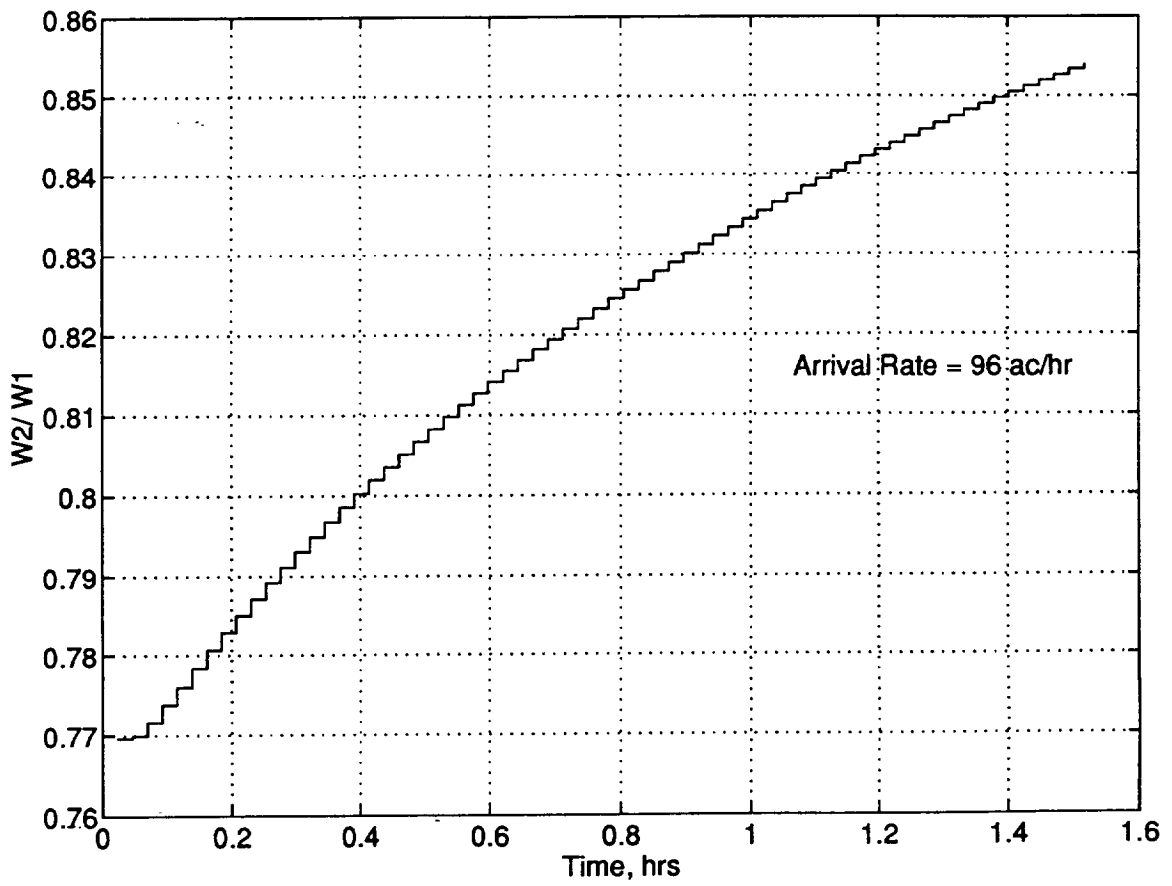
Figure 4: Ratio of 2 Runway Delay Relative to Single Runway for Heavy Traffic

Allocation Strategy	Ave Delay, min/ac	% Crossovers
No Crossovers	2.7008	0.0
Unconstrained Crossovers	0.9249	59.38
Separate Heavies and Smalls	0.9399	66.59

Table 4: Moderate Traffic Comparison for 3 Runways

4.2 Three Runway Allocation Problem

The three runway case is considered since 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 as 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 labelled as "R", "L", and "C" to denote the right, left and center runways respectively. The preferred runway is chosen from a uniform distribution instead of a normal distribution as in the two runway case. This is done to take advantage of the symmetry of the uniform distribution.

4.2.1 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 unlimited crossover case where an aircraft is free to switch runways whenever its delay is lower on the alternate runway than the delay on the preferred runway. The third way of scheduling is to land heavies and smalls 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. Furthermore, it is assumed that the total arrival rate is 108 ac/hr and the runway capacity is 130 ac/hr. Results are summarized below in Table 4.

The no crossover case is again compared directly to an M/D/1 queue that has an arrival rate of 36 ac/hr and a service time of 82.8 sec. As such, we expect a delay of 2.5247 min/ac. Simulation, however, yielded a delay of 2.7008 min/ac. The difference is attributed to an uneven distribution of aircraft on each runway as well as not enough data in the sample space to get adequate convergence. The unrestricted crossover case performed as expected. Figure 5 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
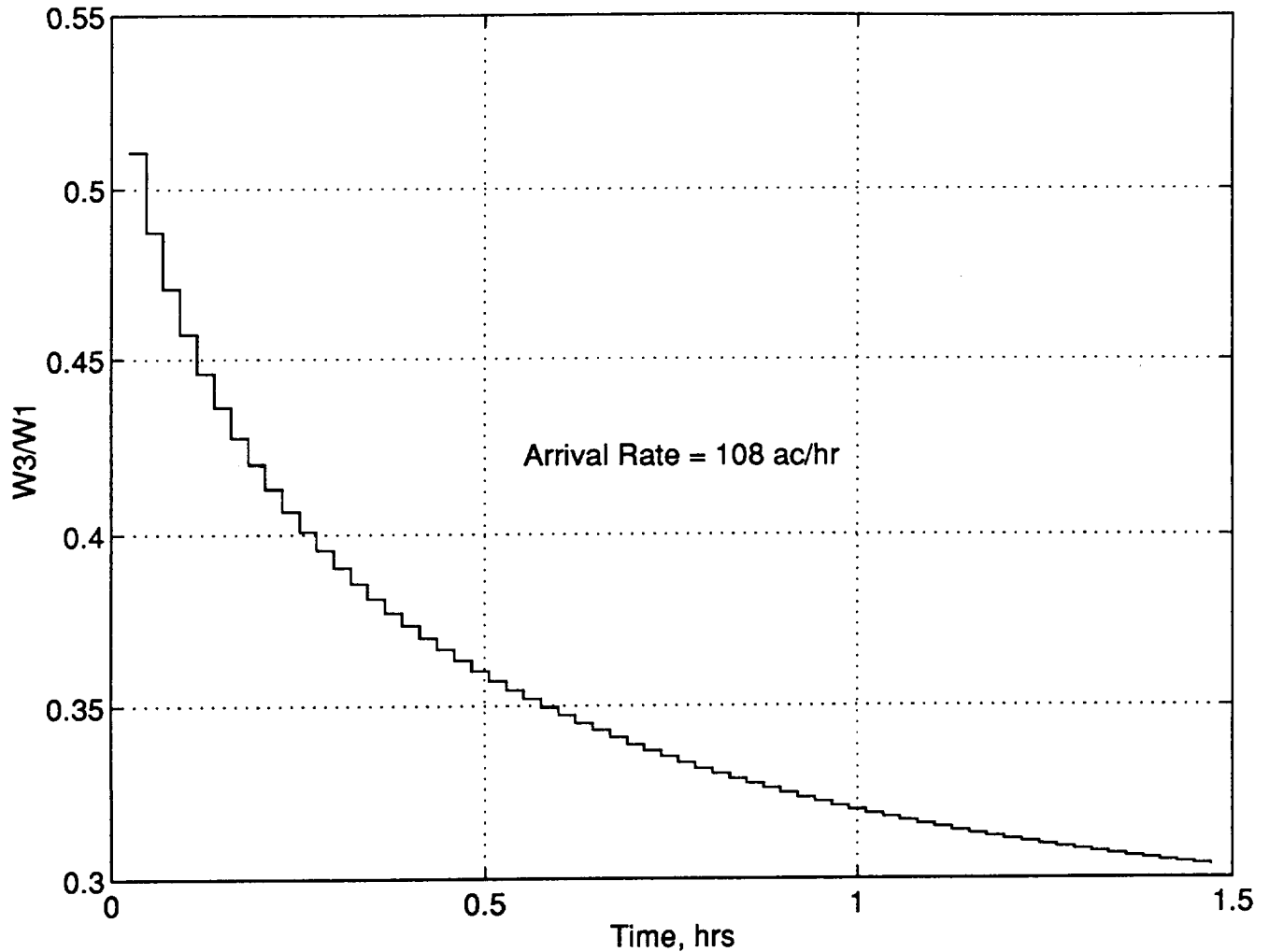
Figure 5: Ratio of 3 Runway Delay to 1 Runway Delay for Moderate Traffic

for 90 minutes of traffic. The delay for the unlimited crossover case is 0.9249 min/ac. The delay that one would expect from the M/D/3 queue is 0.8544 min/ac. Again, the delay is higher than what is expected, but still is 34% of the no crossover delay. One would expect to see 2/3 of the traffic crossover to an alternate runway since the probability of an aircraft of having its preferred runway be the runway which has the lowest delay is 1/3. The actual crossover rate was 59.4%, less than the 67% that would be anticipated. Yet, this is also consistent with what was observed for the two runway/moderate traffic case done above. The next allocation method is to land the heavy aircraft and the small aircraft on their own runways. Then the large aircraft are assigned to any of the 3 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 0.9399 min/ac with 66.6% 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.

19

Allocation Strategy	Ave Delay, min/ac	% Crossovers
No Crossovers	8.7364	0.0
Unconstrained Crossovers	6.4254	65.69
Separate Heavies and Smalls	4.9582	66.79

Table 5: Heavy Traffic Comparison for 3 Runways

4.2.2 Heavy Traffic

For heavy traffic, the arrival rate was increased to 144 ac/hr for a duration of 90 minutes. The three strategies employed are the same as for the moderate traffic. Again, comparisons are made to results obtained using queuing theory in order to predict the delays as well as the improvement in the delay. Table 5 summarizes the results of this section.

For the case of no crossovers, the expected delay is 7.4027 min/ac. However, simulation once again had a higher delay, found to be 8.7364 min/ac. The reason for the discrepancy is as discussed above 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 6.4245 min/ac with 65.7% of the traffic crossing over. Yet, the expected delay is 5.5834 min/ac. Furthermore, one would expect the ratio of the delays to be about 0.7392 (Figure 6). Note the behavior of the curve in Figure 6. The relative delay is increasing for most of the rush period, before reaching a maximum, then beginning to decrease. This implies that the delay for the three server queue is growing faster than the delay for a single server queue after start-up. The delays in the single-server queue then begin to grow faster than for the three-server queue. The ratio of the simulated delays is 0.7354, and our simulations agree with this value. A 26% delay reduction is realized by allowing the aircraft to land on the first runway that becomes available for it.

The final strategy employed is to land the heavy and small aircraft on separate runways and to land the large aircraft on whichever runway its delay is the smallest. The delay calculated from the simulation is 4.9582 min/ac, with 66.78% of the aircraft switching runways. As with the two runway set-up 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. This is even more important when the traffic is heavy, since bunching is widespread.
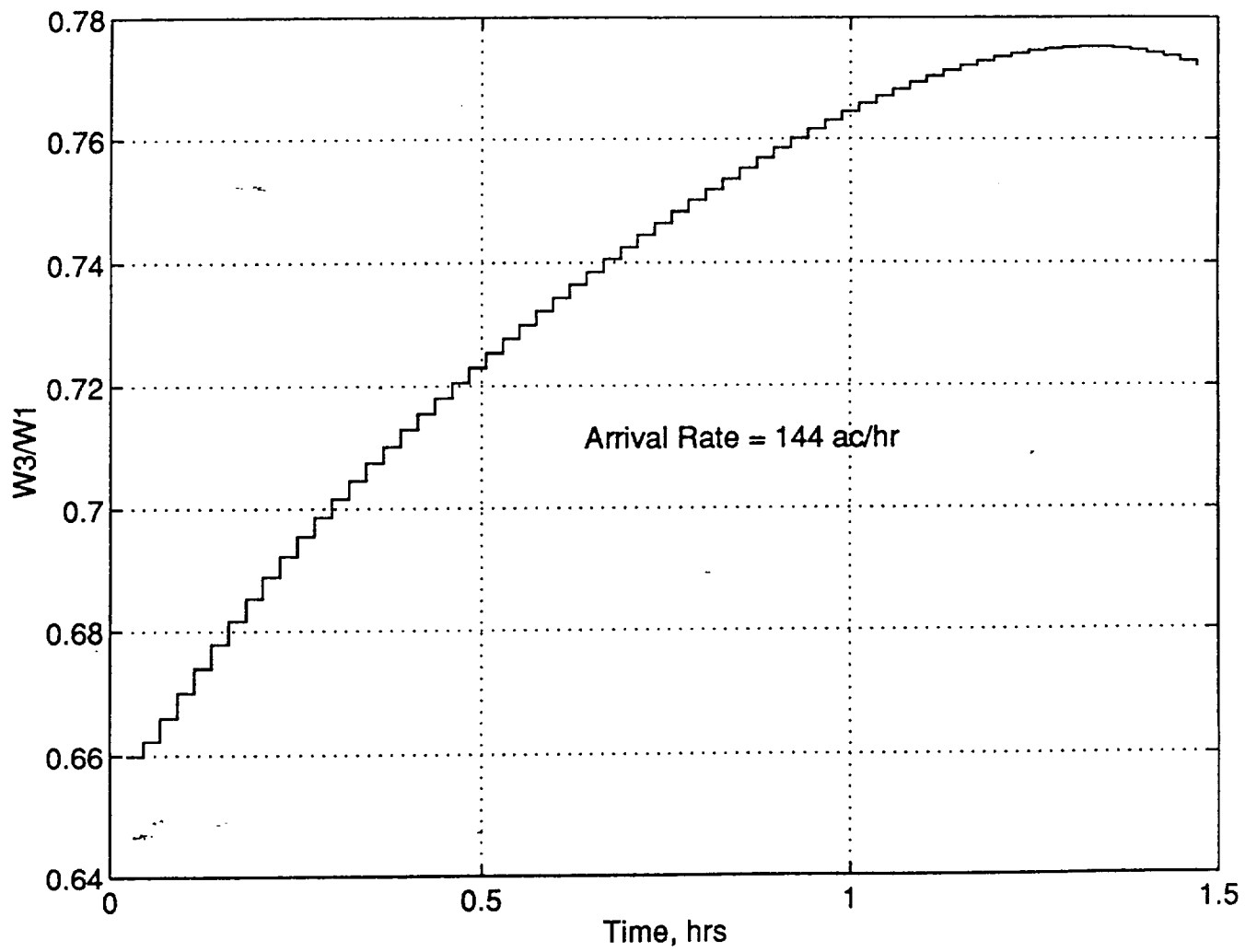
20

Figure 6: Ratio of 3 Runway Delay to 1 Runway Delay for Heavy Traffic

5 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 is accurate 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 since the average separation is large enough that most aircraft are likely to be expedited.

6 Acknowledgement

References

[1] Frank Neuman and Heinz Erzberger. Analysis of delay reducing and fuel saving sequencing algorithms for arrival traffic. Technical Report 103880, NASA Ames, October 1991.

[2] H.F Vandevanne and M.A. Lippert. Benefits from an algorithm for better multiple runway allocation. Technical Report 41L-04126, M.I.T. Lincoln Lab, June 1993.

[3] Robert B. Cooper. *Introduction to Queueing Theory*. North Holland, second edition, 1981.

[4] H.F Vandevanne and M.A. Lippert. The best, the worst, and fcfs sequence. comparison of capacity. Technical Report 41L-0378, M.I.T. Lincoln Lab, January 1991.

[5] Henk C. Tijms. *Stochastic Models: An Algortihmic Approach*. Wiley and Sons, 1994.

[6] Paul G. Hoel, Sidney C. Port, and Charles J. Stone. *Introduction to Stochastic Processes*. Waveland Press, Prospect Heights, Illinois, 1987.