Some Queueing Models of Airport Delays

by

Basil R. Horangic

S.B. in Computer Science
Massachusetts Institute of Technology (1988)

S.B. in Economics
Massachusetts Institute of Technology (1988)

Submitted in Partial Fulfillment
of the Requirements of the Degree of

Master of Science in Operations Research

at the

Massachusetts Institute of Technology

February 1990

Signature of Author______________________________
Interdepartmental Program in Operations Research
February 2, 1990

Certified by______________________________
Amedeo R. Odoni
Thesis Supervisor

Accepted by______________________________
Amedeo R. Odoni
Co-Director Operations Research Center

Some Queueing Models of Airport Delays

by

Basil R. Horangic

Submitted to the Department of Electrical Engineering and Computer Science
on February 2, 1990 in partial fulfillment of the
requirements for the Degree of Master of Science in
Operations Research

Abstract

Air traffic delays presently cost the nation over 3 billion dollars per year. Models of the behavior of delays around airports can be used to direct improvements to areas that will have the greatest positive effect. They can also help in avoiding the creation of new or more restrictive bottlenecks in the sensitive areas of the air traffic system. Models can predict the effect on delays of changes in air traffic control regulations, airport equipment and facilities, and landing procedures.

A number of transient queuing models are investigated, including the fluid flow, equilibrium, and difference equation models, an interpolation model, and the Kivestu model. The interpolation model was developed as part of this thesis, and the Kivestu model as part of a previous thesis. The models are characterized by their computational cost, accuracy, and applicability to the transient modeling of airport delays. Both the Kivestu model and the interpolation model are found to be desirable alternatives to the other models.

The models are implemented and used in the analysis of the delays at Logan airport in Boston. The sensitivity of the system to changes in demand and service levels, as well as service time variance, are explored. Delays are found to be particularly sensitive to service time variance when the system is underutilized, and to be less sensitive when the system is highly saturated. The accuracy of the time varying Poisson assumption for arrivals with respect to demand at Logan is also investigated. It is concluded that this assumption may be of questionable validity under some circumstances.

Thesis Supervisor:
Amedeo R. Odoni, Professor of Aeronautics and Astronautics
Co-Director of the Operations Research Center

Acknowledgements

I would like to thank Prof. Amedeo Odoni for his help and patient understanding in the pursuit of this research and the production of this thesis. Some of the research presented in this thesis was done with funding from the FAA through M.I.T. Lincoln Labs.

This thesis is dedicated to my father, who died five and one half years ago on my first day at M.I.T.. I wish he could have seen me, my brothers and my sisters graduate.

Table of Contents

1. Introduction

Air travel delays. When we encounter them it always seems to be at the end of a hard business trip or at the start of a well deserved vacation. We spend them caged in a terminal or plane, usually unable to see the cause of the delay, or worse, able to see that our destination is within reach, yet unable to reach it. For airlines themselves the delays are just as maddening, as they watch fuel being burned up in holding patterns and ticket sales disappearing with travelers who choose to drive instead. Society as a whole pays a price also, in the loss of energy resources, manpower, and safety. On average, U.S. flights encounter over 2,300 hours of delay per day. [NYT 88] Multiply this figure by a typical average of $30/min in direct operating costs required to keep a plane holding and the cost in time to the hundreds of passengers on each plane. This begins to approximate the estimated 3 billion dollar annual cost of air travel delay. [ANDR 89]

The obvious but naive solution is to add more capacity to the system; more airports, more runways, and more air traffic control ability. This is, of course, not generally feasible. We face massive limitations on available land, of which airports need a great deal, noise, of which airports create an excessive amount, and capital, of which airports use a lot. There have been no new airports added to the U.S. national system since the opening in 1974 of the Dallas Fort Worth Airport. The next new airport is not scheduled to open until the mid 1990's in Denver. It may never open, due to strong opposition from airlines and some local residents, and its enormous cost. [NYT 88] The only alternative to large scale expansion of the system is to optimize the use of the facilities we now have. This requires finding smaller scale, more feasible changes that have large positive effects on performance of the facilities.

As is usual in the real world, it is not possible to experiment with many alternatives to the present system in order to determine the best way of changing it. While we can conceive of many possible changes, we must be able to test their usefulness in some way before choosing which to implement. For this reason we create models of the system which allow us to predict the approximate effects of changes without incurring the costs and risks of physical experimentation. Models, in order to be useable, must assume away many of the seemingly unimportant factors in the system of interest. They must concentrate on the important aspects in a limited framework such that analysis can be conducted efficiently, yet the results must be applicable to the actual system. Such models are the subject of this thesis.

In the air travel system, the bulk of delays are caused by excessive demand on limited facilities, causing queues to form for service, and forcing those who must wait in the queues to incur delay costs. The demand comes from arriving and departing planes, and the service they are demanding is usage of the airport runways, terminals, and other facilities. The demand level is often uncertain due to the unanticipated delays encountered by scheduled flights and the even more unpredictable arrival and departure of unscheduled general aviation flights. The capacity of the service facility (i.e. runways and airspace) is also often uncertain due to weather conditions, non optimal controller behavior, and equipment failure. A model of air travel delays will incorporate the behavior and uncertainty of these two components in some framework that allows the experimenter to investigate their interaction and the resulting behavior of the whole system.

There are two branches of modeling theory that can be useful in approaching this problem, Queueing theory and Simulation. Queueing theory is concerned with the mathematical analysis of highly abstracted queueing systems. Simulation is concerned with analyzing systems of arbitrary complexity by generating repeated random trials over many different scenarios. In general terms, queueing theory permits deeper analysis at less cost with more restrictive assumptions. Simulation permits shallower analysis at greater cost with less restrictive assumptions. The restrictiveness of the assumptions is inversely related to the applicability of the model to the true system. The goal of modeling is to balance the depth of analysis, its cost, and its applicability. The goal of this thesis is to explore some of the models that these two disciplines provide that might be useful in understanding airport delays. The characteristics of the models with respect to depth of analysis, cost, and applicability will be used to determine their desirability.

The genesis of this exploration was in the need to analyze the delay characteristics of Boston's Logan airport. This is part of a larger project to develop a new prototype air traffic control system for Logan. [ANDR 89] The investigation of the models and their tradeoffs is conducted with an eye to their applicability to the particular situation at Logan. This is not a severe restriction of scope. The results will be applicable to the modeling of most busy urban airports, among which Logan is counted. An actual analysis of the present situation at Logan using the models is also included in the thesis.

The background section (Section 2) contains an introduction to the topics of airport operations, measurement of delay costs, and the structure and operation of Logan airport. It also presents a quick summary of the notation and results of queueing theory. This section can be skipped by those with knowledge in these areas. The modeling section (Section 3) presents an analysis of the characteristics that a useful model of airport delays must have, given the state of modern airports, and some of the methods by which such models could be analyzed. The implementation section (Section 4) explores the performance of a number of implementations of the models described in the modeling section. The models are evaluated based on depth of analysis, cost, and applicability. The Logan analysis section (Section 5) contains an analysis of the present situation at Logan airport performed using the models described previously in the thesis. It also contains further evaluation of the applicability of the models to realistic scenarios. The conclusion (Section 6) summarizes the key revelations of this investigation.

2. Background

2.1 The Structure of Airports

Airports connect earthbound travelers to the much faster air travel system composed of commercial airliners and private and corporate planes. The U.S. air travel system handles well over one million passengers per day, making airports among the most intensely used service facilities. [ANDR 89] The services that airports provide and how they do so determines their capacity. Airport capacity in turn determines the capacity of the air travel system. This section describes how airports provide their services, especially those with the potential to delay travellers if not available immediately. The focus is then narrowed to the specific services that will be investigated in this thesis.

The essential task of the airport is to act as an interface that allows one to pass from the land travel system to the air travel system, and vice versa. An obvious distinction can be made, then, between its land side operations and its air side operations. The land side operations encompass tasks such as bringing departing travelers to the airport facility along with their well-wishers and baggage, processing them through the airline facilities, and getting them on the correct flight. This sequence must also operate in reverse for arriving passengers. The air side operations encompass getting planes to the airport facility, maneuvering them around the terminal airspace, landing them, permitting them to take off, and guiding them out of the terminal airspace. All of these airport operations have the potential for introducing delays into the system.

The land side of the airport is typical of public transportations facilities that need to move people through a ticketing process and onto different routes. Airport services can be divided into passenger processing and enplaning. Passenger processing includes rental car return, ticket purchase, check in, and baggage check in. Enplaning includes getting the passengers to the terminal, doing safety checks, checking boarding passes, and moving passengers on to the plane. [BLUM 1976] Of course these services must be provided for arriving passengers in the reverse order.

While individual passengers may be delayed for short periods or even miss flights due to the wait encountered for airport land side services, this is a rare occurrence. The bulk of delays are encountered on the air side of the system, where passengers are delayed as a group in planes or terminals, often for an extended period. The air side system has two 'modes' of operation, one for when weather and visibility are good, and one for poor weather conditions when instruments are necessary for navigation and landing. In visual flying conditions air traffic controllers may permit pilots to fly using Visual Flight Rules (VFR). In instrument flying conditions, however, pilots are required to use Instrument Flight Rules (IFR). In visual flying conditions pilots can see most other planes and, with simple instructions from the controllers, execute their landing and takeoff operations with a high level of efficiency and safety. Thus, controllers sometimes use this mode of operation in good conditions. Under instrument flying conditions the pilots must rely on the controllers for most of their direction, and additional separations and delays are mandated by law for safety reasons. Thus, controllers use IFR in poor conditions and when safety demands. Most airports are scheduled to accept a number of takeoffs and landings which is close to their maximum capacity on a normal day in VFR. Almost as many aircraft arrive on IFR days, though, since the airlines have schedules that are independent of weather. These are the days on which delays are most likely to occur. The operation of the facility at its optimum possible level is vital in these conditions. Therefore the following description of the air side, and the models presented later in the thesis, will be biased towards IFR operations.

The air side services are best understood from the sequence of controllers who direct planes through the stages of arrival and the delays that may be encountered at each stage. As will be explained later, services to departing planes are much simpler and are less significant in generating delays. Aircraft traveling around the country are controlled by a network of enroute controllers. Each enroute controller watches a sector of airspace over the U.S.. Radar, voice communications, and radio beacons on the ground, called fixes, are used to monitor and direct the aircraft passing through each sector. Enroute controllers redirect planes from their current flight paths if they are in danger of coming too close to another plane in the same flight path, or if they may pass too close to a plane in an intersecting flight path. These redirections can include slowing down the plane, having it move off the original flight path to go around a slower plane ahead of it, or moving the plane to a slightly different course to avoid an intersection. The controllers will also change flight paths to avoid hazardous weather patterns. Each enroute controller 'hands off' the planes leaving his sector to the controller of the adjacent sector they are moving into. In the case of sectors containing airports, this is the airport arrival controller.

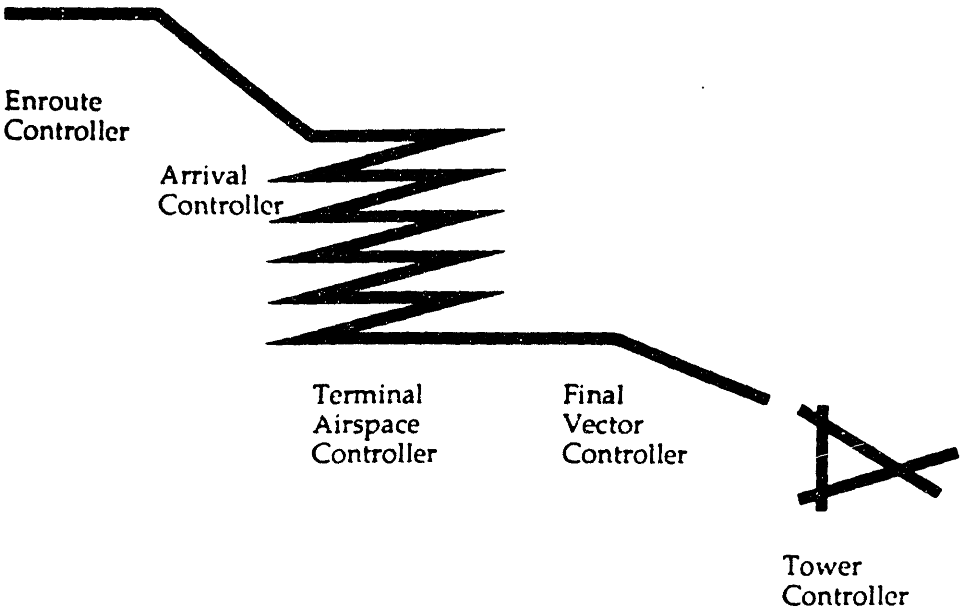


FIGURE 2.1

The airport arrival controller admits planes into the terminal airspace. The terminal airspace extends out in a 20 to 30 mile radius around the airport. In order to maintain an orderly progression of planes, the arrival controller only admits planes into the airspace over a few select fixes (radio beacons located on the ground). Planes not located near a fix must travel to it to enter the terminal airspace. He also predicts the number of planes that will be able to land according to the capacity conditions of the airport, and in case the airport is heavily overloaded or closed will redirect planes to other airports. The closing of an airport and redirection of aircraft are very rare occurrences, though. Such a delay is not specific to this stage but results from overloaded capacity in the later stages of arrival.

The terminal airspace controller directs planes admitted to the airspace to proceed to the airport to land or, in the case of congestion, to delay their landing. The delay can take two forms. A slight delay might be introduced by having the plane reduce its speed or fly a wide arc. More substantial delays are introduced by placing the aircraft in holding stacks. Holding stacks are areas of the terminal airspace where aircraft fly the same oblong flight paths, separated by 1000 feet of altitude, around a ground fix. Up to seven or eight aircraft can be placed in a holding stack and multiple holding stacks can be used for temporarily 'storing' aircraft. Planes are taken off the stack from the bottom, and then the planes above them move down in sequence. Holding stacks in the terminal airspace and delay actions of the terminal airspace controllers are generated in response to congestion in succeeding stages of the arrival sequence (closer to the airport), and not by conditions particular to this stage. However, some arrival controller actions, and possibly mistakes, can introduce small but significant delays at this stage.

The final vector controller takes aircraft from the terminal airspace controller, or the holding stacks in the case of congestion, and directs the aircraft to the beginning of its final approach. The final approach consists of a 'funnel' area that narrows down to a final marker approximately 5 miles from the end of the runway. From this outer marker onward all aircraft must fly the same path, called the common approach path, at the same altitude. All planes in flight must be separated by distances that depend on their size and whether VFR or IFR rules are in effect. Usually this separation can be easily maintained by keeping planes at different altitudes, as in the case of holding stacks. On the final approach path, though, all planes are at roughly the same altitude, so horizontal separation requirements must be maintained. These restrictions are far more severe than vertical separation requirements since it is much easier for aircraft to implement a 1000 ft. vertical separation than a 3 mile horizontal separation. In addition, as they progress down the path the aircraft travel at different speeds and the horizontal separations between them become larger or smaller, possibly introducing violations of the separation requirements.

As the final vector controller brings planes to the final approach path's outer marker he may introduce delays by slowing planes or making slight alterations in their flight paths. The incoming aircraft must be separated by sufficient time interval such that the separation requirements will not be violated on the final descent path. Various strategies exist for merging planes from different entry fixes and holding patterns, flying at different speeds and with different separation criteria, into an efficient progression down the final descent path with the proper spacing. [SIMP 88] [SIMP 89] This stage of arrival introduces most of the delays that spill back to the holding patterns and even to the arrival and enroute controllers. It is the primary bottleneck of the system.

Once the planes are on the final approach path the tower controller takes over. In rare cases he might request minor speed adjustments to maintain separations while planes are on the path. Once the planes are on the ground they are directed off the runway onto taxiways and to their terminals. This movement of planes also causes occasional delays as the taxiways and runways become congested.

In general, if the same runway is being used for both arrivals and departures, the tower controller only allows takeoffs to occur during gaps in the arrival sequence. Thus departing planes are often delayed while waiting on the taxiways or at terminals for takeoff clearance. These planes can often make up much of this delay time enroute by burning slightly more fuel. On some occasions the progression of landings will be halted for the planes on the ground to take off, but this is rare. Once planes take off they require little controller attention and typically depart the terminal airspace without additional terminal airspace delay. [ODON 69]

2.2 Capacity Limitations and Delay Costs

Delays can be introduced into the system from any overloaded or poorly performing service component. The stages of service that constitute an airport are arranged in a network. This network can be thought of as a single macroscopic server. We are concerned with the progression of traffic through this server as a whole. The particular arrangement of component servers and their interconnection in a network is often limited in total capacity by only a few key components. These are the bottlenecks in the system. In pursuit of improvements in the capacity of the server as a whole, the primary bottlenecks of the system are the first parts that should be investigated.

In a typical airport, the primary bottleneck is always the runway system. This capacity constraint manifests itself through the rate at which the final vector controller brings planes from the terminal airspace to the outer marker of the final descent path. The variable and often relatively high demand, and the uncertain service capacity due to weather conditions, make it obvious that overloaded runway systems can explain a majority of the delays encountered at airports. For the purpose of this thesis the runway system will be considered the primary bottleneck that generates airport delays.

Bottlenecks force those that need to use a certain service to wait. This wait is unwanted because time is valuable. We can quantify how unwanted a wait is by expressing it in terms of costs to those waiting for service. An important reason for considering the runway system to be a very significant bottleneck is because its delay costs are so high. It costs approximately twice as much to operate a plane in the air than on the ground. In addition, all passengers on a plane are kept waiting, as a group, until it lands. Bottlenecks on the land side do not delay such large groups of passengers.

These are some estimates of waiting costs:

Passenger Time:	$25/hour
Airplane holding time in the air.	
Commercial Jets:	$40/min
Commuter Aircraft:	$15/min
Private Aviation:	$5/min
Airplane holding time on the ground.	
Commercial Jets:	$25/min
Commuter Aircraft:	$10/min
Private Aviation:	$2/min

With the present system, the yearly cost of air delays to society are staggering. The total delays to airlines and passengers is estimated to be over one million hours annually, costing more than 3 billion dollars. Yet the costs of increasing capacity through new construction would be even higher. These costs underline the necessity of improving air side service efficiency. [ANDR 89]

2.3 Example: Logan Airport

As an introduction to the airport on which this analysis is focused, this section will profile the aspects of Logan airport that are of interest with respect to air side delays. Logan is one of the busiest U.S. airports since it handles a very large number of operations, that is, takeoffs and landings. Logan typically handles 100 operations per hour in good weather. With certain runway configurations Logan can handle up to 120 operations per hour. In

1988, Logan had 434,272 operations. By this measure it is ranked as twelfth in the United States in number of operations. [ANDR 89]

Below is a diagram of the number of scheduled arrivals over the course of a typical weekday. The total arrivals can be split into approximately 60% jets, 30% commuter aircraft, and 10% general aviation. Since general aviation flights are not scheduled, they would have to be added at random to the profile below. [OAG 89]

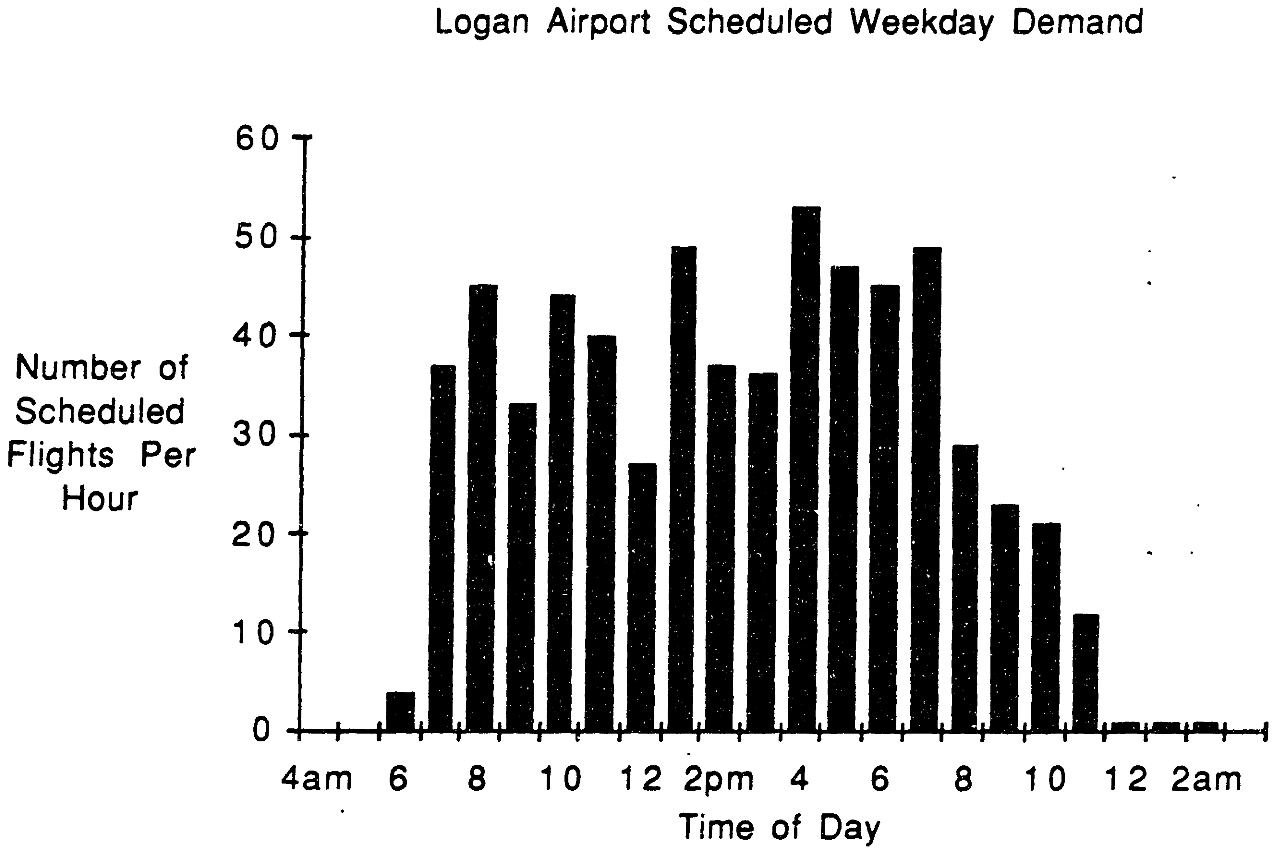


FIGURE 2.2

The Logan runway system, shown below, is the prime determinant of its capacity to handle aircraft and the primary source of delay. The configuration of runways in use at any particular time is determined primarily by the wind direction, and secondarily by noise abatement procedures, traffic demand, and weather conditions in general.

The taxiways, which connect terminals and runways, are shown on the diagram as unhighlighted paths. Congestion rarely forms on the taxiways, since crowding there would be alleviated by controllers keeping aircraft at their terminal. Never would a taxiway bottleneck leave a runway out of use.

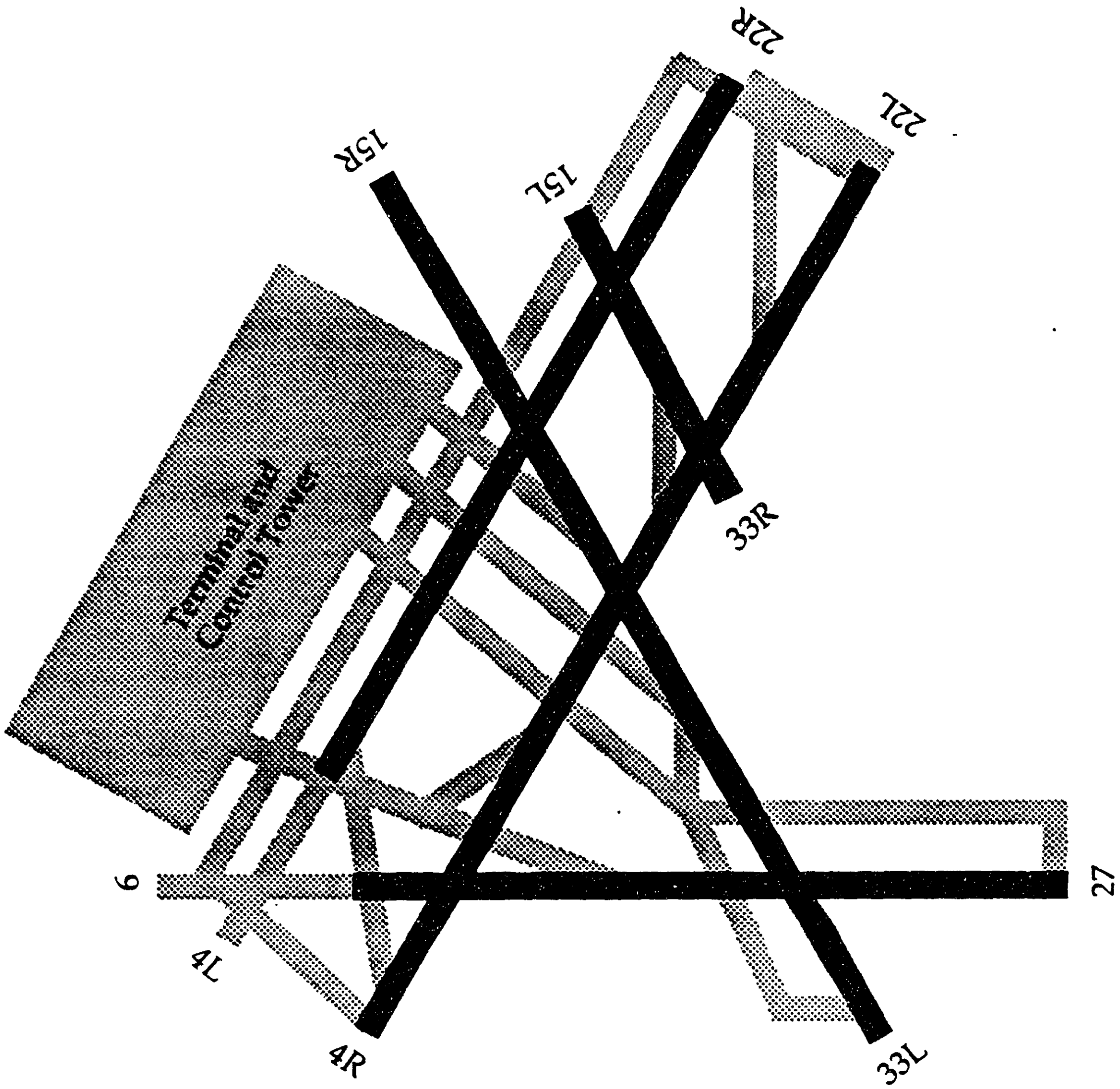


FIGURE 2.3

There are six major runway configurations used by the controllers at Logan. The configuration used is primarily determined by wind conditions. It is always safer to have planes land and take off into the wind. Also, only a few of the runways have equipment to land planes in severe weather conditions. This can further restrict the choice of runway in bad weather. If weather is not a factor, then noise abatement regulations stipulating that the amount of time planes are flown over surrounding areas be distributed fairly can come into play in determining runway configuration.

Primary Runway Configurations at Logan

1. 4L,4R,9
2. 22L,22R
3. 22L,22R,27
4. 22L,22R,15
5. 33L,33R,27
6. 15L,15R,9

Each of the six runway configurations has a corresponding maximum capacity. The capacity is reduced in severe weather conditions. Weather conditions are divided by the controllers into five categories. The primary distinction, mandated by the FAA, is between good weather when visual flight rules are in effect, called VFR, and bad weather when instrument flight rules are in effect, called IFR. The controllers further subdivide IFR into four categories of severity, IFR-1, IFR-2, IFR-3, and IFR-4. The type of weather is determined by cloud ceiling and wind speed, using the diagram below. The percentage occurrence of each type of weather is noted in the chart.

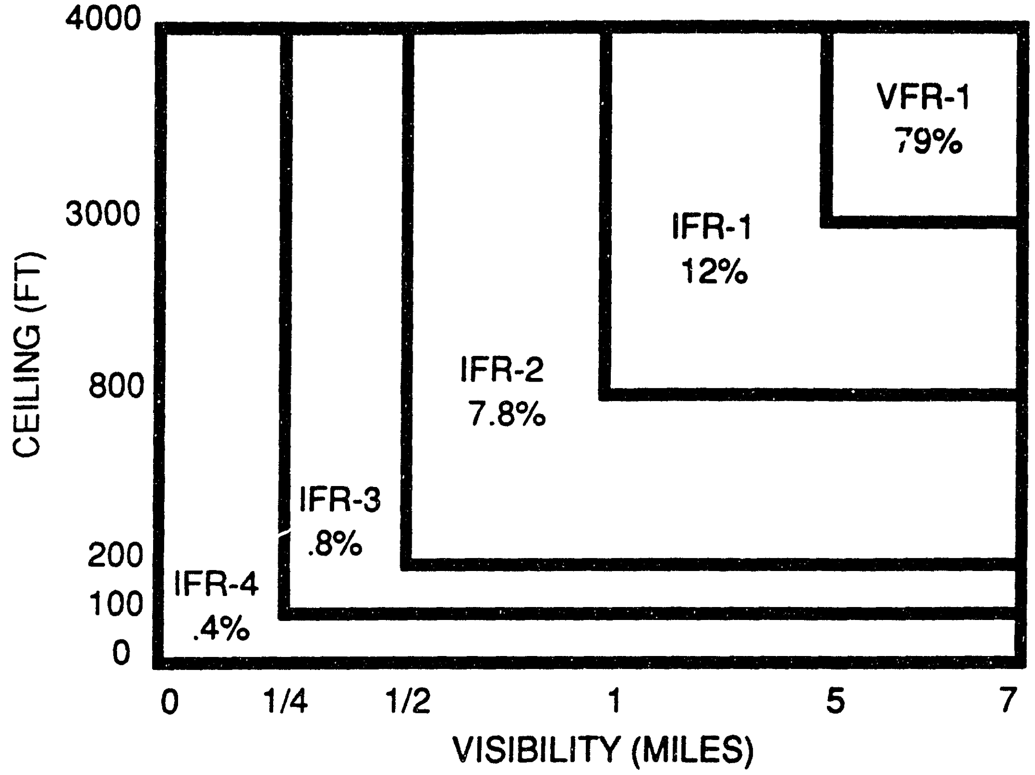


DIAGRAM 2.4

The approximate total capacities in different conditions and runway configurations at Logan are shown below, assuming 50% takeoffs and 50% landings. The figures are in terms of number of operations per hour.

	VFR-1	IFR-1	IFR-2	IFR-3
4L,4R,9	111	64	58	54
22L,22R	107	67	58	NA
22L,22R,27	110	95	NA	NA
22L,22R,15	NA	NA	58	NA
33L,33R,27	76	55	48	NA
15L,15R,9	70	57	54	NA

In the worst weather conditions, low capacity sometimes causes delays to rise to an average of 60 minutes or more per plane. In the summer increased flights from Cape Cod and the high frequency of thunderstorms can lead to even higher average delays.

If the airport does backup due to inadequate runway capacity, the Boston TRACON (Terminal Radar Approach Control Facility), from which the arrival and terminal airspace controllers control the airspace around Logan, keep the aircraft in holding patterns until the Logan final approach controller is ready for them. A diagram of the Boston TRACON area is shown below, outlined in heavy black. Depending on the runway configuration in use, this area is divided into sectors for use as holding, approach, and overflight areas. Overflight paths are obviously the ones shown that do not stop at Logan. Aircraft are accepted from enroute controllers into the TRACON airspace through only three fixes, Providence, BRONC, and SCUPP, which are shown on the diagram. Note also the three holding stack areas around markers LOBBY, SCUPP, and EXALT. The stack areas are shown as small oblong loops. Few delays are generated by service constraints in this stage of the arrival process; it simply serves as a queueing area for delays generated in the final approach stage.

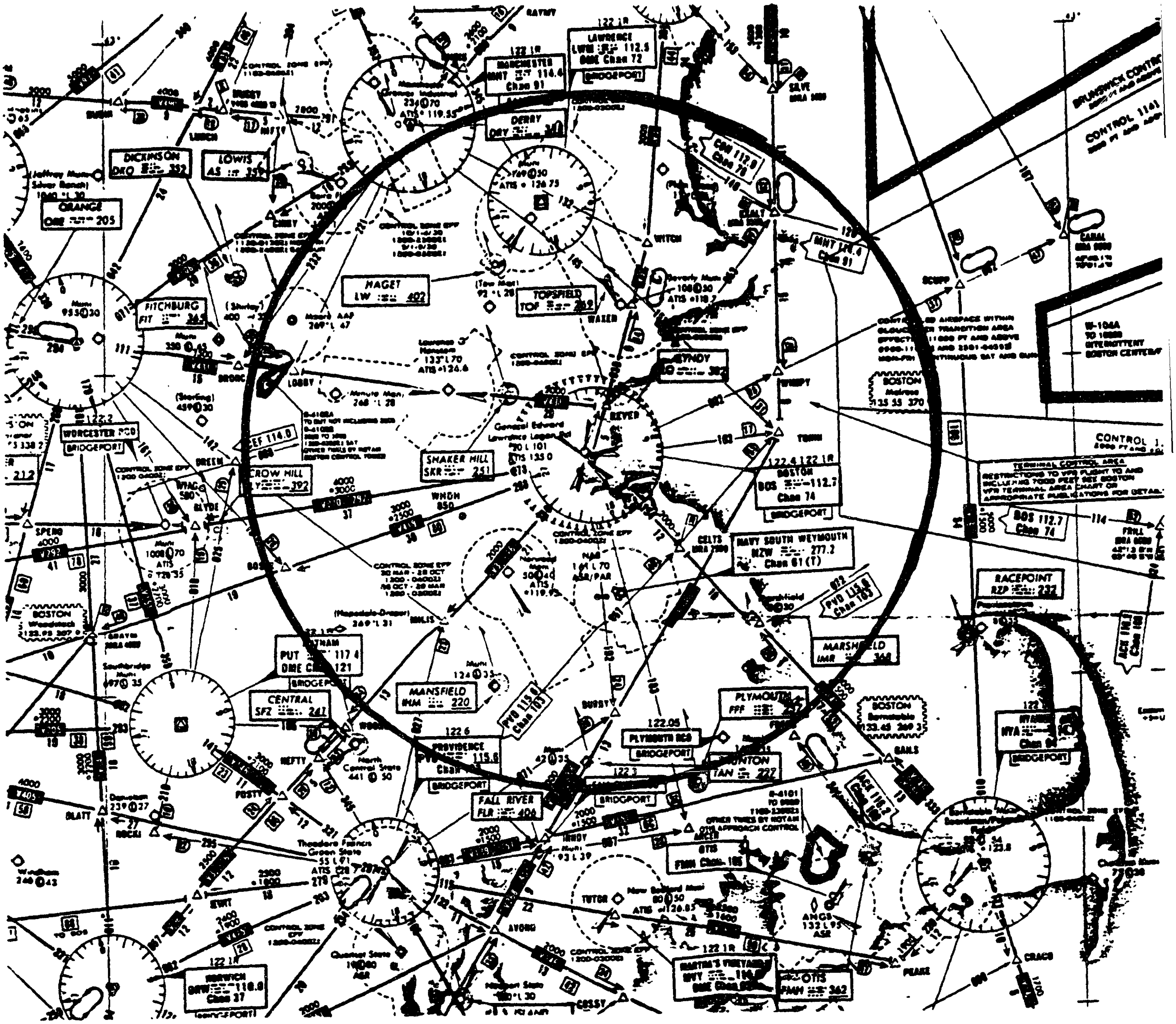


DIAGRAM 2.5

If the Logan final approach and the holding patterns back up, or the TRACON airspace becomes too crowded, the arrival controller, along with the enroute controllers directing aircraft to Logan, decide how to limit the acceptance rate into the TRACON airspace. Aircraft delayed by the enroute controller may be sent to another airport or kept on the ground at their origin.

For the purpose of studying delays at Logan, we abstract from the airport as a whole only those services directly connected with the primary bottleneck, the runway system. This includes the physical runways and taxiways, the final approach vectoring space, the TRACON airspace and holding patterns, and finally the enroute controller airspace outside of the Logan TRACON. Each section both provides a service, and can be used, up to a certain capacity, as a queueing area for aircraft bottlenecked further down the sequence towards the runways.

2.4 Focus of Analysis

Modeling the airport as a landing and takeoff server, and the subsequent analysis that can be performed, can show the best course of action to take in reducing delays. The least expensive sources of increased efficiency are small adjustments to the system as it operates today. For instance, analysis can tell us how much increased landing rates reduce delays. This benefit can then be weighed against the costs of this improvement, in terms of safety, workload, new equipment needed, etc.. Analysis can also tell us how much delays will be reduced if the landing or takeoff time for each plane is made less variable. This benefit can also be weighed against the cost of the new equipment and personnel required to reduce the variance of landing times. Another application is in determining the value, in terms of delays, of adding runway capacity by lengthening or adding runways, or by adding equipment so that more aircraft can use the runways in inclement weather.

Our particular application of this analysis is in quantifying the benefits of a new, more efficient, air traffic control system. Such a system could increase runway capacity, reduce landing time variance, increase the holding capacity and efficiency, and help in setting acceptance rates into the TRACON airspace. The benefits of these changes in terms of reduced delays can be analyzed by models such as the ones developed in this thesis. This is our primary goal in developing models of delays at Logan.

Another use of this type of analysis is in the design of completely new systems, such as new air traffic control strategies and even new airports. The design of new airports requires deciding the type, length, and configuration of runways. Primary in determining this will be the space constraints on the ground. Also important are the frequency and direction of winds in the area, the types and frequency of weather, and obstacles in the airways, such as high structures or other airports' approach airspace. The effects of each of these considerations can also be quantified by a model of airport delays.

Government policy-making is another area where this type of analysis is useful. Policy-making is necessary to enforce efficient and fair use of society's resources. An area of government regulation that is of concern to many ordinary citizens is noise abatement regulations. These regulations specify the number of low flying airplanes that may pass over certain areas, in an attempt to limit the total noise encountered by residents, and to distribute the noise more fairly. Obviously noise abatement regulations restrict airport capacity by forcing the use of suboptimal runway configurations when they are not necessitated by weather conditions.

A more sensitive area of government regulation is in setting user fees for airports. An arriving plane at an airport generates two sources of delay costs. The most obvious is called internal delay, that is, the cost of the passengers' time and of aircraft operation while the plane waits to land. If this cost is too high, planes will choose not to come to Logan. Another cost, called external cost, comes from the added delay the new arrival adds to other planes that arrive after it and must wait to land. To be fair, each plane should be charged both its external and i⁻ternal cost, the external in the form of landing fees. Small planes, in particular, have low internal cost but create large external costs by delaying large numbers of passengers in big, expensive to operate jets.

In 1988 Logan instituted increased landing fees for small planes. This was an attempt to distribute more fairly delay costs and influence planes to land during the low demand portions of the day. This policy was overturned in December 1988 through the efforts of small craft operators, and is being re-evaluated by Logan. A model of delays can demonstrate the need for and effect of government regulation of airport usage. [MOOR 89]

2.5 Queueing Theory

This section is meant as a short summary of the notation and results of queueing theory. Queueing theory is oriented toward analytical investigation of service systems, their demand characteristics, and the delays they produce. In order to provide analytical results, simplifying assumptions are often made with regard to the important aspects of the system. These assumptions can limit the applicability of the results.

All simple, non-network queueing systems can be abstracted to three components: a demand generator; a queuing area for holding customers that are being delayed; and a server. Each component can be simple or complex. For instance, demand can be generated by a simple memoryless Poisson distribution, or can be a complex distribution possibly dependent on the state of the system. The queue can be a simple infinitely long FIFO line, or contain different priority customers, have limited capacity, or complex queue disciplines. The server can be a single unit with memoryless service time, or multiple units with complex and possibly state dependent service time. The exact characteristics of each part of the system must be specified in order to conduct an analysis.

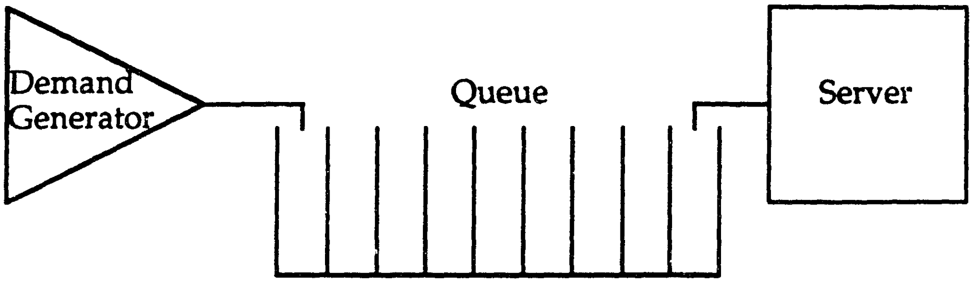


FIGURE 2.6

A shorthand has been developed for specifying common queueing systems. It consists of a letter representing the demand distribution, a letter representing the service distribution, and a number signifying the number of servers. The meanings of the letters are as follows.

M	Poisson	M(t)	Time-varying Poisson
D	Deterministic	D(t)	Time-varying Deterministic
E_k	Erlang of order k	$E_k(t)$	Time-varying Erlang of order k
G	General Distribution	G(t)	Time-varying General Distribution

The first three distributions have special qualities in that they allow the queue length to be represented as a Markov system, and so they are assigned special letters. All distributions, including M, D, and E, are lumped under general (G). The specifications for a queueing system are written using the format of demand distribution, service distribution, and number of servers separated by slashes. For example M/M/k, M/D/k, M(t)/G/k. The last specification would signify a system with time varying Poisson arrivals, a general service distribution, and k servers.

The principal aspect of queueing systems that investigators wish to determine is the waiting time for customers. Some commonly sought after parameters associated with this are the average waiting time, the average queue length, the number of customers who are turned away or who leave because of too long a wait, and the probability of being delayed past certain time limits. All statistics of interest can be derived from the probabilistic behavior of queue length over time, if this can be determined. It is often not possible to fully specify queue length behavior, but there are alternate ways of calculating the statistics of interest without going through this intermediate step. Not all statistics can be derived analytically for all queueing systems, though.

There are two types of fundamental results in queueing theory, those which are valid in the steady state and those valid at any time. The steady state results are applicable if the characteristics of the demand and service distributions do not change and the queue has a very long time to adjust from its initial state. Results valid at any time, especially during adjustment to new conditions, are called transient results. If a queue's characteristics do not change for a long time the transient result will converge to the steady state result. For this reason steady state results can be used as an approximation to transient results if the change causing the transient is not large and the queue has a long time to adjust. This assumption is often drawn upon since very few analytical solutions for the transient behavior of queueing systems exist but many steady state solutions do.

Systems with demand and service rates determined by Poisson, Erlang, and deterministic components have some analytical results for the discrete distribution of queue lengths in both steady state and transient states due to the fact that.they can be represented as Markov processes. From this distribution all other values of interest can be calculated. Systems with service and demand rates determined by general distributions have only steady state analytical results at best, although some transient approximations exist. [BERT 89] For systems lacking any analytical results at all, only simulation or approximation by an analytical system can be used to obtain queue characteristics. Obviously, for our modeling purposes we are more interested in transient analysis since conditions at airports are constantly changing. [LARS 81]

3. Queueing Models of the Terminal Area

3.1 Queueing Models to Characterize Delays in Terminal Areas

The terminal area is composed of the airport passenger facilities, the runway system and the surrounding airspace. The airspace extends up to a 20 to 30 mile radius around the airport and 15 or 20 thousand feet above. Our goal is to model flight operations within this area and the delays incurred by aircraft performing them. Aircraft within the terminal area have one of three goals, to land, to take off, or to pass through. They require the services of the airport controllers and use of the runway facilities and the airspace in order to fulfill these goals. A terminal area system that operates well will fulfill these goals efficiently and without endangering the safety of the aircraft or its passengers.

One can view the terminal area macroscopically as a service facility which provides a complex mix of services which impose demands on controllers, the runways and airspace of the area. As with any service facility, when the demand for service outstrips the facility's capacity to provide it, the users of the facility are delayed. If more than one user is delayed, a queue forms. If the disparity in demand and service capacity is large, the queue becomes large and the users are forced to wait for extended periods for service. This can incur costs in terms of time, money, or opportunity, depending on the type of system and type of delay. A facility with large capacity causes less wait and fewer delay costs, and thus is of more value. The value of increasing the capacity of a server, then, can be measured by the cost of delay that would have been incurred by users and which is eliminated by the change in capacity.

The terminal area service facilities associated with large urban airports are often overloaded, and planes wishing to land, take off, or pass through the terminal area are often denied immediate service and forced to wait. In order to improve the performance of the terminal area facilities, that is, increase their capacity and reduce delays, we must understand the behavior of the system with respect to how delays are generated. Modeling the queueing aspect of the system (the phenomenon of users lining up to wait for service) is the obvious solution. The only feasible improvements to terminal area capacity are small scale changes in the way the system operates. Modeling and understanding the behavior of terminal facilities with respect to the generation of delays is essential for making informed decisions about the changes and improvements that are worthwhile to implement.

Basic queueing models assume a simple demand behavior and a simple service behavior at the facility, and investigate the resulting queues and the delays encountered by customers in the queues. Applied to the scenario under consideration here, the requests for service by aircraft in the terminal area constitute the demand, the ability of the terminal to fulfill their requests constitutes the service behavior, and differences between actual and expected service completion times constitute the delays.

The terminal area can also be thought of as a number of interconnected servers each representing different aspects or components of service. For instance, the arrival metering process, holding stacks, final approach, and runways can be thought of as four servers arranged in a sequence. Together they constitute the terminal area server. 'Network' queueing models such as these tend to be far more complex than basic single demand/single server models. The different components often have varying capacities and alter their behavior depending on the condition of the other components. Also, queues generated by one server can back up into preceding servers. There are many models of varying complexity that can be constructed. The choice of model depends on what type of behavior we wish to model, to what accuracy, and at what cost.

In this chapter we discuss the types of delay related issues we wish to understand using a terminal area model. Then various conceptual models are presented which might incorporate these types of behavior to some degree. Finally, these conceptual models are related to various practical models which have similar behavior and can be or have been investigated in depth.

3.2 Goals of a Queuing Model of the Terminal Area

The overall goal of this modeling exercise is to determine the relationship between delays incurred by aircraft and the characteristics of the terminal area server. What is meant by 'delays incurred' is open to many interpretations. Are we concerned with average delay or maximum possible delay? Are all aircraft delayed equally, or are some treated differently than others? The characteristics of the terminal area server that are of interest with respect to their effect on delays must also be chosen from a myriad of interconnected components that constitute the server as a whole. Examples of possible characteristics include the accuracy of the final vector controllers directives, the local weather patterns and their effect on runway capacity, or the metering rate set by the arrival controller. In addition, the specific aspects of the relationship between delays and server characteristics that are of interest must be specified. Are we interested in the long run, steady state relationship or the transient effects? The aspects of delay that are of interest, the characteristics of the server that are of interest, and the aspects of the relationship between them that are of interest are detailed below. Incorporating these is the goal of our terminal area model.

In the general sense we are interested in the delays incurred by all aircraft requesting some form of service. All delays cost time and money. However, some delays cost more than others. We will be concerned primarily with the delays incurred by planes requesting use of the runways to land. The reason for this is that landings take more time on the runways than takeoffs. In addition, planes delayed at their origin can often make up substantial amounts of time enroute at the cost of excess fuel burn, thus reducing the effect of takeoff delay on their total delay. It also costs far more to the operators to keep planes waiting in the air than on the ground, usually double the amount of money, and it requires more controller ability to keep the numerous planes safely separated in a congested airspace than it does for planes standing on a taxiway separated. Finally, landings are more dangerous, especially in poor weather. The usual policy of controllers is to allow planes to take off only when there is a break in the stream of arrivals. This decision signals the relative importance of landings and takeoffs to those responsible for directing them. Departing aircraft will be of interest only for their effect on the delays incurred by arriving aircraft.

The most important aspect of the delays incurred by landing aircraft will be the average delay across all aircraft. If possible, we are also interested in further characterizing the distribution of delays experienced by aircraft, especially by investigating its variance and functional form. An important instance of the effect of variance and functional form is in determining the maximum delay encountered with some fixed probability, or the probability of an aircraft encountering a delay in excess of some fixed value. Other characteristics which might further refine the distribution of delays among aircraft are also significant. Planes arriving from certain directions or of a certain type which encounter delays systematically different from the average aircraft are an example.

Determining which characteristics of the terminal area are of interest is much more complex than clarifying the characteristics of delay that are of interest. In the general sense we are most interested in the characteristics that significantly affect delays incurred by landing aircraft. More particularly, we are interested in those that have the potential to lead to large reductions in these delays, although knowing which characteristics have the potential to increase delays is also important.

The runway is the primary bottleneck of the server and the aspects that determine its capacity will be of interest in this respect. These aspects can be divided into four categories, the way air traffic control is performed, the characteristics of the planes using the runway, the environmental conditions in the terminal vicinity, and the structure of the runway system. [ASHF 79]

Air traffic control procedures determine the patterns in which aircraft may fly and how much space must be maintained between them. Of primary importance are the horizontal separation requirements of 2 to 6 miles, depending on the types of planes and weather conditions. Since planes must be separated by this amount until they land, this limits the capacity of the runways and of the server as a whole. The stipulation that only one plane occupy the runway at a time also restricts the capacity. Since takeoffs require less time and separation than landings, it is possible to optimize runway usage by inserting departures between arrivals. These techniques are of interest with respect to their effect on the delays encountered by arriving planes. As was stated, departures are normally queued until a break in the arrival stream, so air traffic control procedures applying to takeoffs are only significant in high demand situations where these breaks do not appear naturally and the progress of landings must be interrupted to allow takeoffs.

The runway in effect extends about five miles from its end to the outer marker of the final approach. This is because between the outer marker and the runway all planes must fly the exact same path. Planes traveling at different speeds along this path will close some of the gaps that exist between them. Controllers must consider this when directing planes to the outer marker to begin their final approach, and they must increase the separation more when slower planes are followed by faster ones in order to adhere to air traffic control separation requirements. This problem also leads to optimization techniques where planes of the same speed are grouped together or faster planes are sent to the runway before slower ones. The effect of these interarrival gaps and the techniques used to optimize them are of interest with respect to their effect on delays.

Of great significance to the performance of air traffic control is the accuracy of monitoring aircraft speeds and locations. Better planning and complex runway usage optimization techniques are less useful when the controllers lack the ability to accurately monitor and direct the aircraft. If the controllers have good capability in this respect, then the effectiveness of the sequencing and spacing system and the techniques it uses become more significant in increasing runway capacity and reducing delays.

Noise abatement procedures are regulations that must be fulfilled by controllers which attempt to fairly distribute the noise generated by the airport. Sectors over which planes fly at lower altitudes are assigned restrictions on the annual percentage of flights that may pass over them. If the controllers have a choice of runway configurations they must consider noise abatement requirements before automatically choosing the runway configuration with the highest capacity.

Environmental factors that affect runway capacity include winds and visibility, surface conditions, and noise abatement procedures. Winds are the primary determinant of the runway configuration in use since planes land and take off much more safely into the wind. Cross winds are particularly dangerous when landing. Visibility determines if increased separations are required and may limit landing only to those runways which are equipped for instrument approaches. Surface conditions can cause longer braking times and thus, increase occupancy time. In the case of snow and icy surface conditions runways may even be forced to close.

The design of the runways has a large effect on their capacity. Length of runways has been mentioned in connection with their ability to land aircraft of different types. Also of importance is whether runways are parallel and by how much they are separated. Runways separated by more than 4300 feet can both be used simultaneously and independently for landings. If parallel runways are separated by a smaller distance only one can be used for landings, but the other may be used for takeoffs. Again, optimization techniques exist for alternating between usage, and these are of interest. The taxiways connecting the runways and the passenger terminals are also significant. Few or poorly placed taxiways cause planes to stay on the runway for longer periods of time. In addition, runways that crisscross have complicated operating procedures that limit the capacity they might have if they did not intersect. [ASHF 79]

While the runways are the primary bottleneck, the characteristics of the stages of approach before the runways are significant due to the delays they create. When the capacity of the final approach and runway is overloaded the queued planes are held in holding stacks or other patterns in the terminal airspace. The location and operation of holding stacks is significant in generating delays. Non-optimal holding patterns waste time as the aircraft do not reach their destinations accurately when released. They waste fuel if the aircraft are required to accelerate or climb. The accuracy of the monitoring system for aircraft in the airspace is also significant for similar reasons. Inaccurate controller direction or pilot reaction can cause delays due to positioning mistakes, costs in excess fuel burn, and decrease safety margins.

There is also a feedback effect between the holding stacks and the runways. Depending on the runway configuration in use, the holding patterns, and thus their characteristics, will change. Also, expectations of weather changes or demand changes affect the configuration of holding patterns. A characteristic of approach that also affects congestion and delays is pilot ability and willingness to execute controller instructions accurately. Large mistakes can cause missed approaches or increased delays for planes queued up behind the aberrant one.

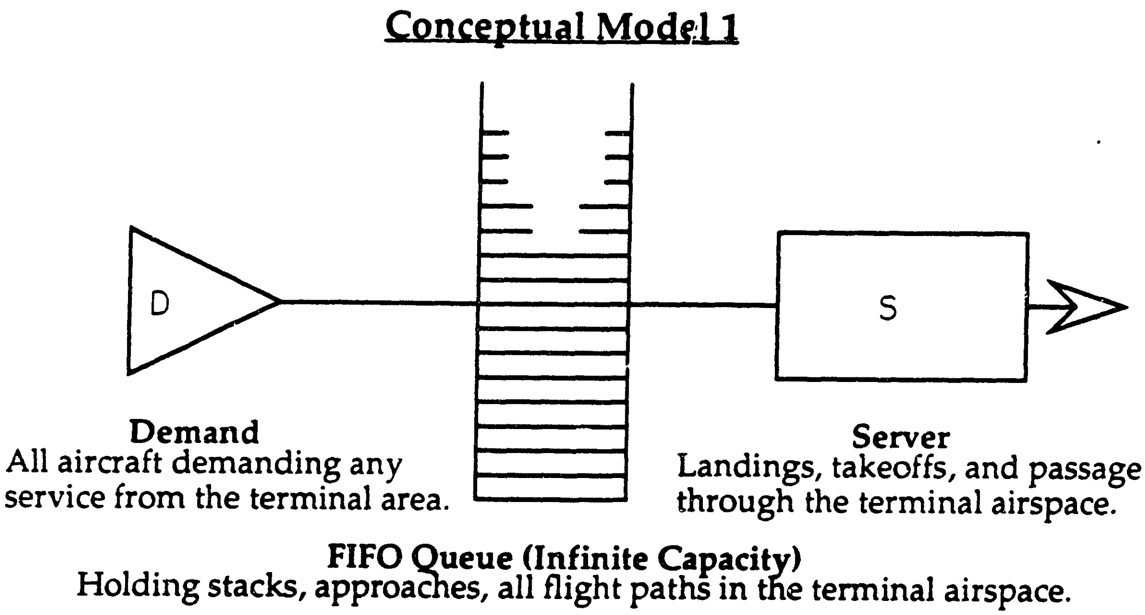
At the border of the terminal airspace, the metering of arriving flights into the airspace by the arrival controller is significant with respect to delays. The acceptance rate is determined with reference to the condition of the runways and holding stacks. The speed of obtaining and using this feedback information is important in determining the congestion encountered by succeeding flights. In times of excessive demand, some aircraft may be kept on the ground at their origin to avoid congestion and reduce fuel consumption. Some may be redirected to other airports, especially in the case of the temporary closing of the destination airport.

In summary, the characteristics of interest with respect to their effect on delays fall into the areas of air traffic control, structure of the facility, weather conditions, demand characteristics, and pilot capability. The category of air traffic control can be subdivided into the effects of (1) air traffic control regulations, (2) other government regulations that must be fulfilled by controllers, (3) planning capability of the air traffic controllers, (4) degree of accuracy in directing aircraft, (5) degree and accuracy of feedback between controllers of the different aspects of the system, and (6) predictability of changes in weather or demand and how this information is used. The category of facility structure can be subdivided into the effects of both (1) the structure of the runways and airspace and (2) the degree of accuracy in monitoring aircraft position allowed by the monitoring equipment. The goal of this modeling exercise is to determine the behavior of delays incurred by landing aircraft with respect to these characteristics of the terminal server.

3.3 Conceptual Models of the Terminal Area

The most basic model of the behavior of queues and delays given some demand and service characteristics is the simple demand, single server queueing system. This type of system has been extensively investigated by queueing theorists. In this model, isomorphic (single class) arrivals are generated by some stochastic process. The time required to service each arrival is controlled by another stochastic process. Delayed aircraft are assumed to wait in a queue which has infinite capacity and operates using a first in first out (FIFO) discipline. The behavior of many particular specifications of this type of queueing system can be determined analytically, at least in the steady state.

In applying this model to the terminal area, one would consider the whole terminal area including runways, taxiways, holding stacks, approach space and terminal airspace as one macroscopic server. Whether a plane wishes to enter the terminal airspace to land, leave the airport by taking off, or pass through the airspace, it is requesting the services of the controllers and will use some part of the terminal area. If the resources required to perform the requested service are busy, the aircraft will have to wait in the queue with the other aircraft requesting service. The aircraft is considered to have finished its service when it no longer requires the resources of the server. The time required to complete its service is considered to be the time it precludes other aircraft from being serviced, either by consuming controller attention or space in the server area.



Conceptual Model 1

Demand
All aircraft demanding any service from the terminal area.

Server
Landings, takeoffs, and passage through the terminal airspace.

FIFO Queue (Infinite Capacity)
Holding stacks, approaches, all flight paths in the terminal airspace.

FIGURE 3.1

This type of model is relatively simple to analyze, and if the processes representing demand generation and service times are chosen carefully, analytical results exist for the steady state characteristics of the resulting delays. It is also relatively easy to generate descriptions of the transient, or real time, delay behavior. The problem with the model, though, is that we only learn about the behavior of delays with respect to the overall demand for service and the most general characteristics of the terminal area server. No distinction is made between the types of service requested, the category of user that is making the request, the different characteristics of the many types of services, and the interdependencies between different components of the server. This model has little practical use except in roughly measuring overall delay and controller workload with respect to simple demand changes and simple service capacity changes.

This model can be made more useful though by narrowing the definition of the components to represent the more important aspects of the system. A modified model might incorporate only requests for service to land as the demand component. The effect on delays of requests for takeoff or transition through the terminal airspace can be incorporated into the service process by increasing the time it takes for just landing requests to obtain service. Note that the demand still does not distinguish between categories of customers. The service process thus represents the landing time intervals without regard to the landing airplane type. It is also less accurate in generating these intervals because the effect of takeoffs and transitional aircraft on delays are incorporated only in the long run sense. This model does, however, allow us to investigate the behavior of the important class of delays associated with landing aircraft with respect to a general process generating requests for service and a process generating service times. It definitely fulfills the goals of the previous section to a larger extent than the last model.

Conceptual Model 2

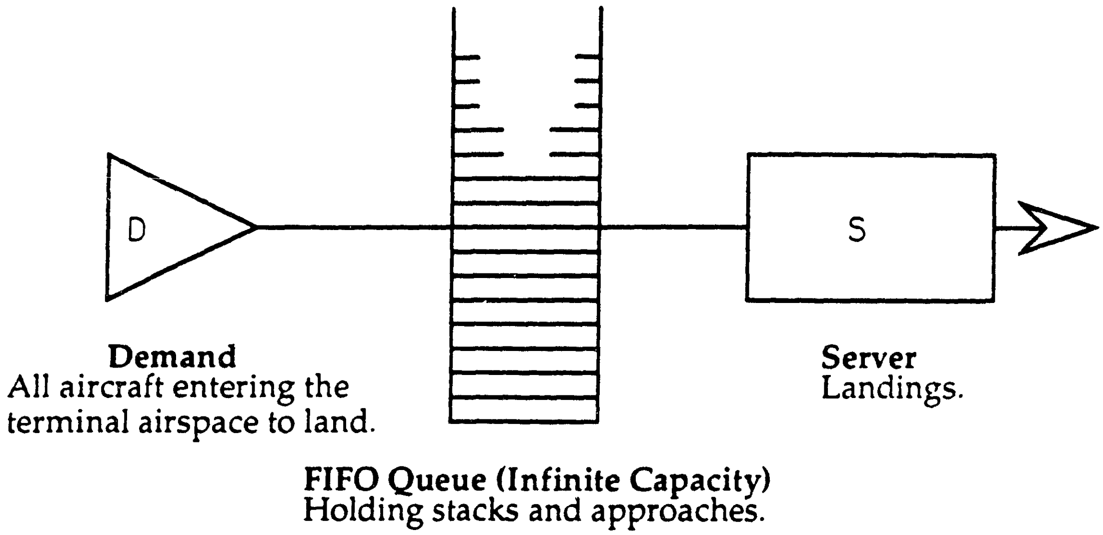

D
S
Demand
All aircraft entering the terminal airspace to land.
Server
Landings.
FIFO Queue (Infinite Capacity)
Holding stacks and approaches.

FIGURE 3.2

Further modifications to this model can make it conform more to the true behavior of an airport. First, the process generating demand and service times can depend on the time of day. This allows analysis of both delays in the long run and instantaneous delays during the day. Second, the queue can be limited in capacity to represent the limited capacity of the terminal airspace and the possibility of redirecting flights to other airport in the case of heavy congestion. This is equivalent to making the demand and service processes depend on the number of aircraft in the queue. Introduction of this dependency also allow the implementation of certain other queueing disciplines and effects such as balking, communication slow down due to system overload, and some effects of different server components. This more complex model can still be analyzed by the analytical methods of queueing theory, and it permits analysis of more specific aspects of delay with respect to both the time dependency and queue size dependencies of the system.

Conceptual Model 3

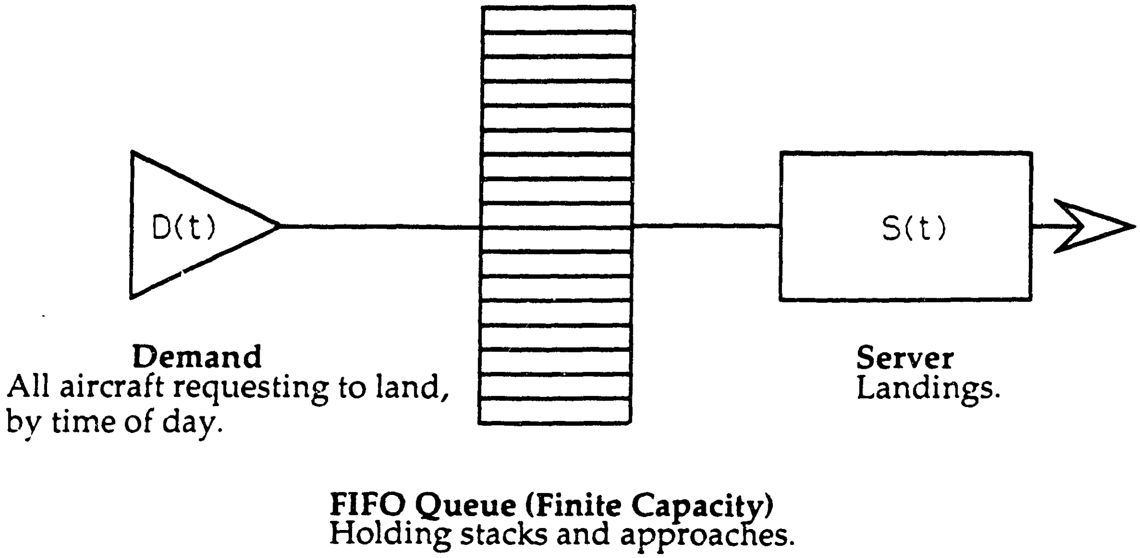

D(t)
S(t)
Demand
All aircraft requesting to land,
by time of day.
Server
Landings.
FIFO Queue (Finite Capacity)
Holding stacks and approaches.

FIGURE 3.3

Additional modifications to the demand components can improve the fit of the model to our goals even further. Instead of a single time varying process representing demand, different processes for each class of user can be introduced. Thus we can separate commercial aviation demand from general aviation, or the different classes of aircraft from one another, and even distinguish between the direction from which the arriving aircraft enter the terminal airspace by creating different demand generators for each category. In addition, we can make the service times dependent on the class of user. This allows one to more accurately represent the service requirements of each class of user and also to introduce different queue disciplines based on user class and arrival sequences. In addition, we can create multiple servers that would represent multiple runways in operation. This improves the fit of the model in low demand scenarios where runway usage is intermittent and not continuous.

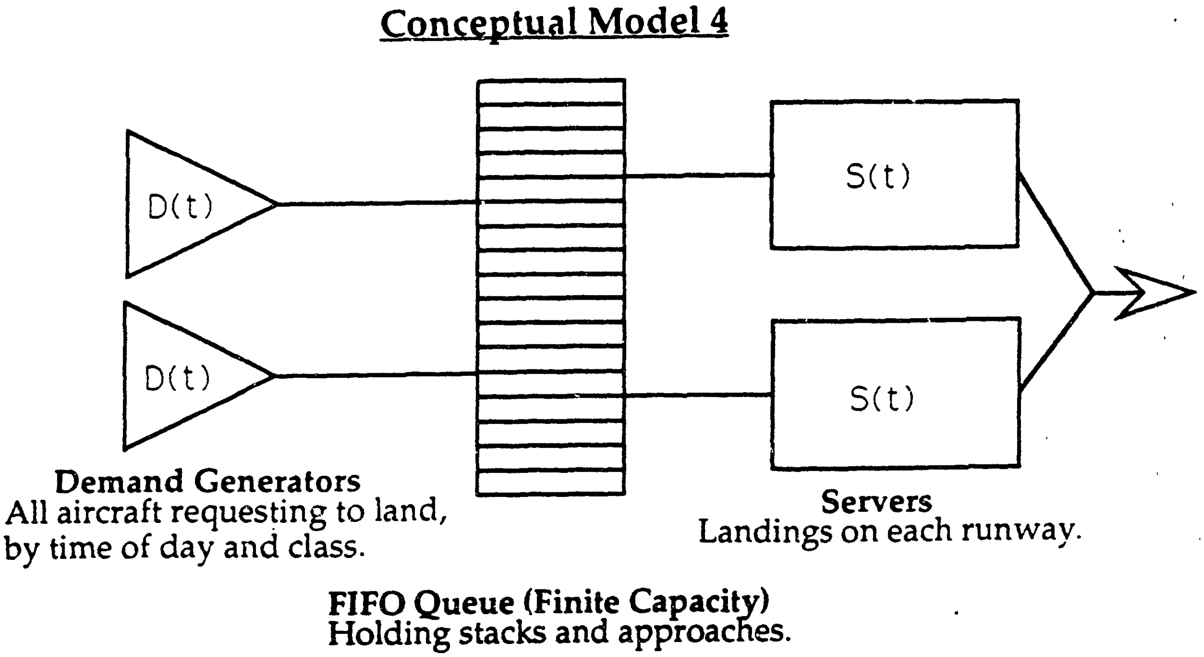

Conceptual Model 4
D(t)
D(t)
S(t)
S(t)
Demand Generators
All aircraft requesting to land, by time of day and class.
Servers
Landings on each runway.
FIFO Queue (Finite Capacity)
Holding stacks and approaches.

FIGURE 3.4

A model of this type is about as complex as the results of standard queueing theory will allow us to describe analytically. It would be possible to add demand and service dependencies on some external factors such as weather, assuming some appropriate model of weather in the terminal vicinity. This model is quite satisfactory for investigating arrival delays with respect to time of day, aircraft characteristics, and different queue disciplines (which can be made to roughly model different air traffic control procedures). Each of the target characteristics from the previous section can be investigated to some extent. For instance, the effect on delays of the controller's ability to know aircraft position accurately can be tested by changing the variance of service times. The effect of better planning, more lax government regulations, or different runway configurations can also be tested by suitable modifications to the service process.

This model is unsatisfactory, though, in the sense that it assumes a single server and a single queueing area. In reality the terminal area is composed of a number of interconnected components that both perform services and have some capacity to enqueue arriving aircraft. Also, these components interact and change behavior depending on the state of other components. A model that recognizes such a structure is no longer in the realm of simple queueing theory, but falls in the realm of queueing networks.

It is much harder to analyze queueing networks than simple queueing systems, even with very severe assumptions about demand and service processes. Queueing networks can be analyzed by simulation, though, but this is usually computationally expensive. The benefit of a network model is that it not only recognizes the network structure of the problem and therefore gives more accurate results, but it also allows us to test the effects of changes in individual components of the system and changes in how these components interact.

The most obvious division of services into networked components is between the runway / final approach section and the rest of the terminal airspace. The runway and final approach have severely limited queueing capacity, severely limited service capacity, and consist of a single flight path along which all aircraft travel. The rest of the terminal airspace (called the vectoring space) has much greater queueing capacity in the form of holding stacks and space to fly other patterns. It also has a larger service capacity in that it can process planes just about as fast as the runway and final approach can accept them, with a few exceptions. The controllers of these two areas are in contact with each other and interact, so that the service characteristics of each section is dependent on the state of the other section.

Conceptual Model 5

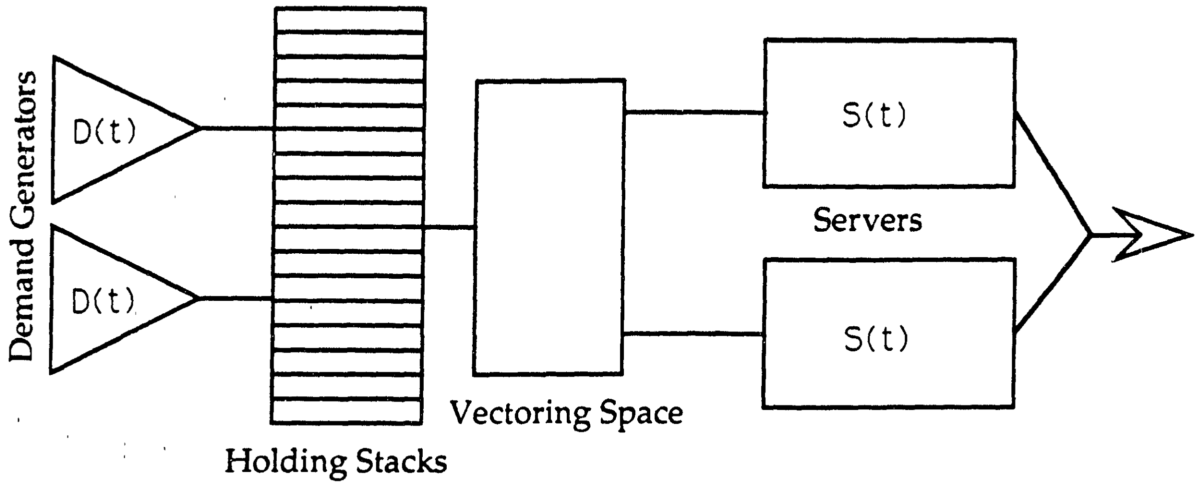


FIGURE 3.5

The terminal server can be further subdivided into arbitrarily small components. For instance the first server above can be divided into an arrival metering stage, a holding stack stage, and a final vectoring stage. The second server above could be split into a final approach stage and a runway stage. This allows representation of aborted landings, different holding stack configurations, and different vectoring scenarios. For instance, an aborted landing would take the user back to the transit stage from the final approach stage.

Conceptual Model 6

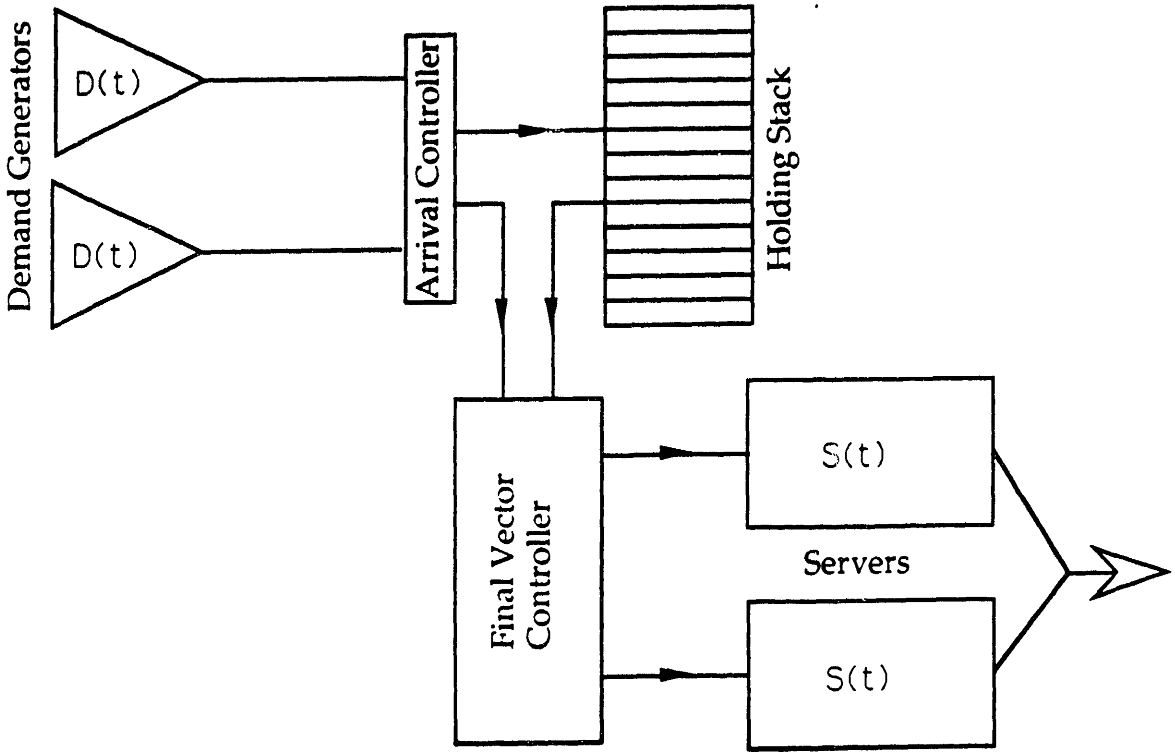


FIGURE 3.6

As the model becomes more complex the analysis becomes increasingly harder and must be conducted at a much shallower level. In addition, the types of scenarios under which any analysis might be valid become much more constrained. It is possible to create finely subdivided abstractions of every piece of terminal area space and every controller action. At some point, however, we must decide that increased complexity will not add much more accuracy to our analysis, or at least that the increased accuracy is not worth the additional cost.

The model chosen should be the one best suited for addressing the issues at hand efficiently and accurately. In fact, for a single problem different models may be used and their results viewed with respect to each model's limitations and advantages. One model might be used for long term behavior and another for short term. If the sensitivity of delays to only a few components is required, a model representing only those components might be constructed.

3.4 Practical Models of the Terminal Area

The conceptual models described above must be made more concrete if one is to apply the available analysis techniques to them. Making the models concrete means specifying the stochastic processes that generate demand and service times and describing exactly how the queue and server operate. The specification of these items will determine the types and depth of analysis that can be performed, if at all, and at what cost. Assuming specifications that allow more in depth analysis may reduce the applicability of the results.

There are three methods by which we can perform analysis of queueing systems. Steady state analysis can often lead to closed form analytical expressions if the demand process is not exponentially based. Transient analysis can be achieved through difference equation approximations if the queue length can be represented by a Markov system, that is, the probabilistic distribution of successive states depends only on the previous state. This is possible when the process for both demand and service are exponentially based. If the model is not representable as a Markov system, it can be simulated. Using simulation both steady state and transient results can be approximated, but at much higher computational cost and much lower accuracy than possible through the difference equation approximations. Each of our implementations of the conceptual models falls into one of these three categories.

The first two conceptual models of the previous section were not time dependent and thus only have steady state solutions. These models can be implemented as M/M/1, M/D/1, M/E_r/1, or M/G/1 queueing systems and analyzed using the appropriate analytical formulas for steady state solutions. M/M/1 has time invariant demand generated by a Poisson process, time invariant service rates generated by some other Poisson process, and one server. The M/D/1 has deterministic service time, the M/E_r/1 has Erlangian service time, and the M/G/1 has a general service times characterized by a mean and variance. Analytical steady state solutions exist for all four of these models. If other time invariant models are specified that do not have analytical steady state solutions, simulation can be used to approximate these solutions.

Such models might be used to determine the long range (i.e. yearly) capacity of airport facilities, and for this purpose they give good approximations. They are essentially useless for more detailed analysis, though. These same systems may be used to analyze conceptual models three and four above, but again the steady state results would only be applicable in long term analysis.

The third and fourth conceptual models described in the previous section correspond to time varying processes and thus have transient solutions. With proper assumptions about the nature of the service and demand processes they can be represented as a Markov state system and analyzed by the first method. An M(t)/ M(t)/k/n_{max} system is one such system that can be analyzed this way. The M(t)/ M(t)/k/n_{max} system has time varying Poisson arrivals, time varying exponential service times, k servers, and a queue capacity of n_{max}. An M(t)/ D(t)/k/n_{max} is similar except with deterministic service times. These two models were first explored by

Koopman using finite difference techniques to arrive at transient solutions. [KOOP 72] The models were expanded by Odoni, Hengsbach, and Roth to include multiple servers and different queue disciplines. [HENG 75] [ROTH 79] [ODON 83]

A compromise between Poisson M(t) and deterministic D(t) processes is the Erlang process $E_r(t)$. A first order Erlang $E_1(t)$ is the same as an exponential M(t). As the order r increases the distribution moves to a skewed normal shape and then to an impulse D(t) as r goes to infinity. The distributions in between are attractive since choosing them allows one to control the variance. $M(t)/E_r(t)/k/n$ and $E_r(t)/E_r(t)/k/n$ are possible models which use Erlang distributions to control variance. Such models are also representable as Markov state systems and can be analyzed by solving a system of differential equations.

Another method of controlling the variance of the stochastic processes is to use M(t) as the upper bound on variance and D(t) as the lower bound on variance and use an interpolation of the results of each as an approximation of an Erlang system. Such a model is developed later in this thesis. A method of approximating the Erlang system by modifying the way the system of difference equations for the $M(t)/ D(t)/k/n_{max}$ queue is solved was developed by Kivestu and is also implemented as part of this thesis. [KIVE 76]

A D(t)/D(t)/k/n system is the trivial case of non stochastic flow. This model can be easily analyzed numerically. It is also implemented as part of this thesis. Such models have been used to determine airport delays by Oliver. [OLIV 64]

If the distribution chosen to represent service times and demand interarrival times are not compatible with representation as a Markovian system, then simulation may be used to generate transient as well as steady state solutions. In some cases it may still be possible to generate such solutions analytically through a complex decomposition into the space of Bessel functions. [BERT 89] The the algorithm for doing so is far more complex than that for simulation, though.

Conceptual models five and six above are more complicated network queue models and cannot usually be analyzed by representation as a Markovian system. Such models would typically be analyzed through simulation. Work on such detailed simulations has been done by Brown and Nordin, as well as by the FAA. [BROW 76] [NORD 78] The statistical and other problems associated with analysis of simulation models have been extensively documented and solved, thus allowing accurate analysis. The primary arguments against simulation, though, are its high cost and low accuracy.

The implementation of some of the models mentioned above will be discussed in detail in the next section. Even though the models are fully specified there are often alternative ways of generating solutions. Each solution method has different computational costs and different accuracy.

4. Implementations of the Model

4.1 Conceptual Model

We are primarily interested in analyzing delays to landing aircraft at Boston's Logan airport, and especially in analyzing the effects that an improved air traffic control system might have on delays. A secondary goal is to generate accurate predictions of delays for possible use in the planning and metering aspects of a new air traffic control system. The time frame of interest is intra-day. That is, we are interested in the behavior of delays over hourly or smaller periods.

Transient analysis is necessary since we are interested in intra-day behavior and not long term trends. Varying demand and service levels during the day require that the demand and service rates be time dependent. We also wish to evaluate the effect of changes other than the average rate that might affect delays, especially the introduction of an improved air traffic control system. Thus we require flexibility in altering the stochastic process describing service times. We are not interested in the effects of multiple servers, nor are we interested in multiple classes of arrivals to the system. The ability to represent sub-components of the terminal server is also not required. A model fulfilling the requirements specified above is diagramed below.

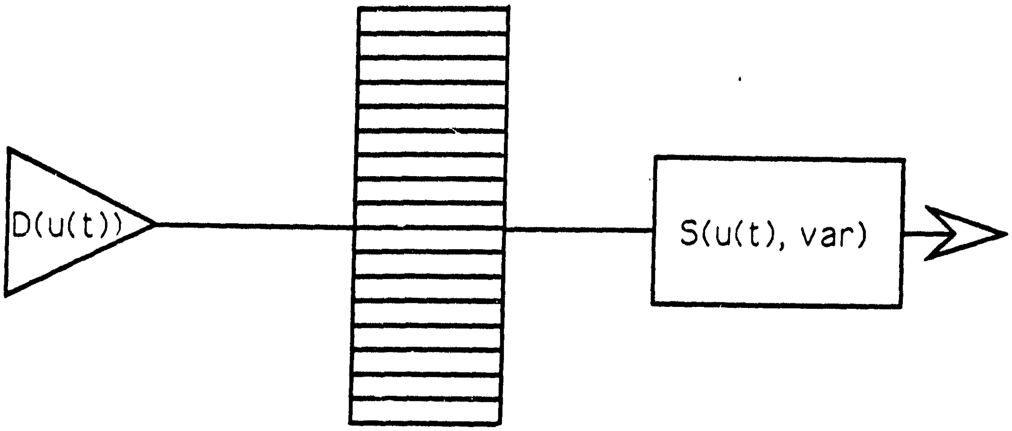


FIGURE 4.1

The demand consists of aircraft that need to land at the airport. The demand rate can vary during the day. There is no provision for modifying other parameters of the demand process, such as the variance. No distinction is made between aircraft with different characteristics. All arrivals are isomorphic, they are treated the same way by the queue discipline and have exactly the same service characteristics. The queue is a finite capacity FIFO queue. The server has a time varying rate, and the variance of the process governing it can be altered as well.

Different implementations of this model are discussed in the next section. Each implementation requires different assumptions and different computational costs. Each also permits varying levels and accuracy of analysis, and is most applicable under certain scenarios.

4.2 Implementations

4.2.1 Fluid Flow Model ($D_{(t)}/D_{(t)}/1$)

This model is discussed as a base case. It has no stochastic components. It is equivalent to a reservoir system into which an incompressible fluid flows at a predetermined, time varying rate and out of which it flows at some predetermined rate.

The model has a time varying demand rate but the process governing the interarrival times of users demanding service is deterministic, that is, it has no random component. This assumption makes this model more oriented to scenarios where all flights are scheduled and they arrive very close to the scheduled time. It is less applicable where unscheduled, general aviation flights make up a segment of demand, or when the distribution of scheduled flight arrivals around their scheduled arrival time has significant variance.

The server also has a time varying rate and is a deterministic process. This is more applicable to scenarios where service times have insignificant variance. It is less applicable to situations where service times are random, determined by multiple components, or dependent on external factors. The deterministic service process limits our ability to analyze changes in server characteristics other than service rate.

The computational costs of this model are minimal since an exact result for the transient behavior of the queue length and waiting times can be generated by simply tracking each arrival and service. That is, the arrivals are simply counted since it is known exactly when they arrive. The deterministic service time is then applied to each of the arrivals. The server only handles one aircraft at a time, so aircraft that arrive during another aircraft's service join a queue. They begin service when the aircraft ahead of them in the queue have been served. The difference between the time of arrival and the beginning of service is the queueing delay experienced by each aircraft.

This type of model was used for predicting airport delay by Oliver. [OLIV 64] It is still used as a first cut approximation in determining runway delay, but is severely limited in its accuracy unless its restrictive assumptions are satisfied. Since the behavior of demand and service at most airports does not follow these assumptions, it is also of minimal use in analyzing the questions in which we are interested.

4.2.2 Steady State Approximation Model (M/G/1)

An alternative model that is more applicable to the situation we are investigating, and requires approximately the same level of computation as the previous model, is a steady state model. The basic assumption of this model is that the transient behavior of the queue is closely approximated by its steady state behavior. Our version assumes time varying Poisson demand and a time varying service time that may be any general stochastic process.

A well known result of queueing theory states that average delay and average queue length of a M/G/1 queueing system in the steady state depends only on the arrival rate and the mean and variance of the service time distribution, and can be calculated analytically. [LARS 81] We propose to approximate the transient solution of a M(t)/G(t)/1 system by using the analytical formula for steady state delay of M/G/1 systems given these assumptions. First, the changes in the demand rate and service rate and service variance are discretized so that they are constant during intervals between discrete changes. During these intervals the average delay is calculated using the analytical formula. This type of analysis is sometimes called equilibrium analysis.

The steady state result will be a good approximation for the actual transient result only when the demand and service rates change slowly, and the utilization ratio of the queueing system is low. Thus, making the step intervals smaller to more closely follow the actual time varying rate quickly becomes futile. The lack of accuracy of the steady state approximation usually outweighs the benefits of a more accurate representation of the demand and service rates. The steady state solution will tend to underestimate the transient solution when the utilization ratio of the server (the demand rate with respect to the service rate) is falling, and will overestimate the transient solution when the utilization ratio is rising.

This model is certainly more accurate than the simple fluid flow model, and it requires no more computational power. The service process can assume any mean and variance values, and includes the cases M(t)/M(t)/1 and M(t)/D(t)/1. This allows us to test the effect on delays from modifications that affect mean service time and/or its variance. The assumption of a time varying Poisson process for aircraft arrivals is not very restrictive. Even for demand composed mostly of scheduled flights, if there is a significant chance of the aircraft being off schedule the assumption of a Poisson demand process may not be far from what will be observed empirically. The validity of this assumption is tested later in this thesis with respect to scheduled arrivals at Logan airport.

The primary problem with this model is the inaccuracy of the steady state approximation. For scenarios where the demand and service characteristics change rapidly, which is quite often the case at many airports, this model will be only marginally applicable. It would be applicable to heavily saturated airports which operate at approximately a constant level of demand and close to full capacity all day.

4.2.3 Difference Equation Models ($M(t)/M(t)/1$, $M(t)/D(t)/1$, $M(t)/E_r(t)/1$)

If the demand and service processes are limited to those processes whose inter-event times are exponential or sums of the same exponential, the state of the queue can be represented as a Markov system. This allows the transient solution to be approximated to an arbitrary level of accuracy by solving the time varying set of difference equations that describe the evolution of the Markov system over time.

The key element that permits the evolution of these queues to be described by a system of difference equations is that for small time increments the stochastic processes fulfills the Markov condition that the probability of transition to subsequent states depends only on the state of the system at the present time. This is another way of saying that the inter-event times are exponential or sums of exponentials. Examples of such processes are the Poisson, with a variance which is inversely proportional to the square of its rate, and the Erlang, with a variance inversely proportional to its order and to the square of its rate.

The Erlang process allows us to choose from a family of distributions for interarrival times with variances ranging from zero to that of the Poisson. The first order Erlang process is equivalent to a Poisson, and higher order Erlang processes asymptotically approach a deterministic process. The figure below shows a number of the members of the Erlang family.

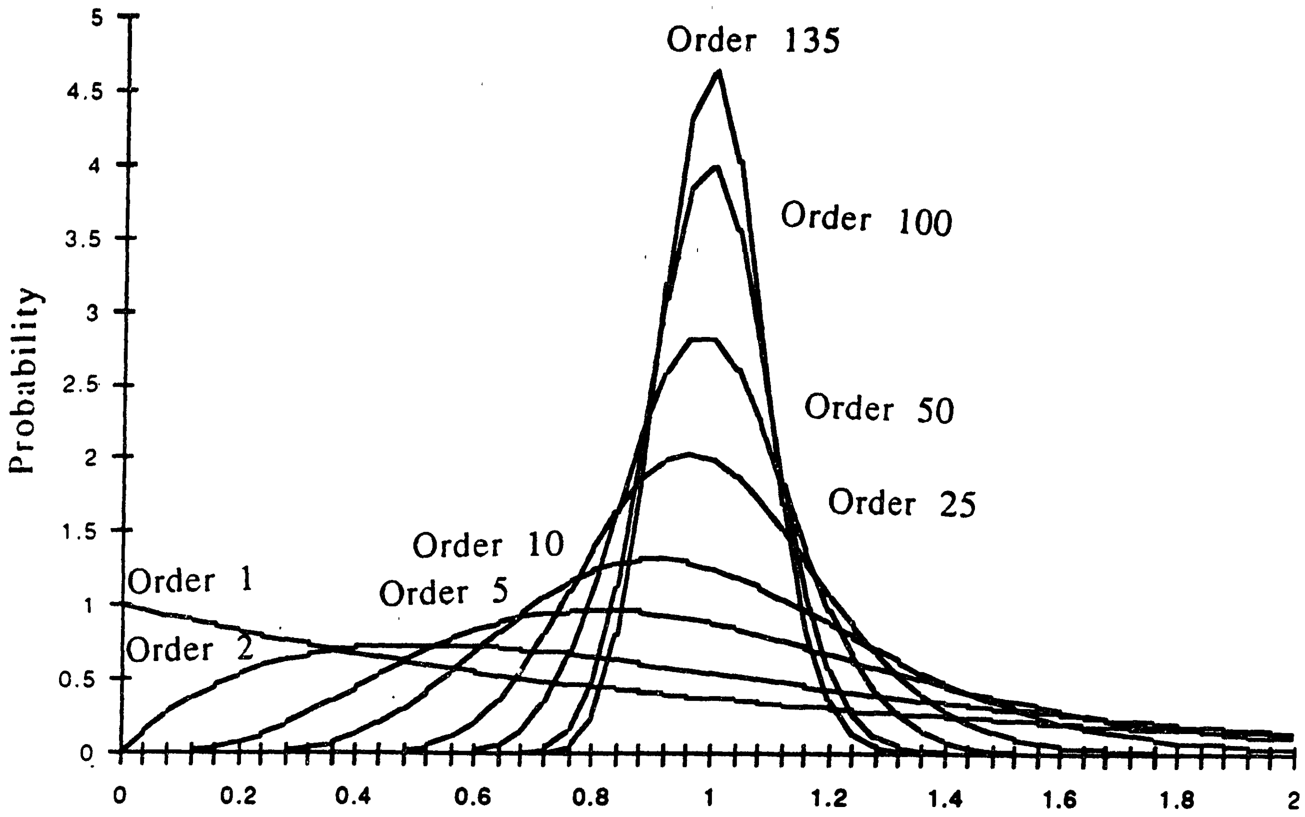
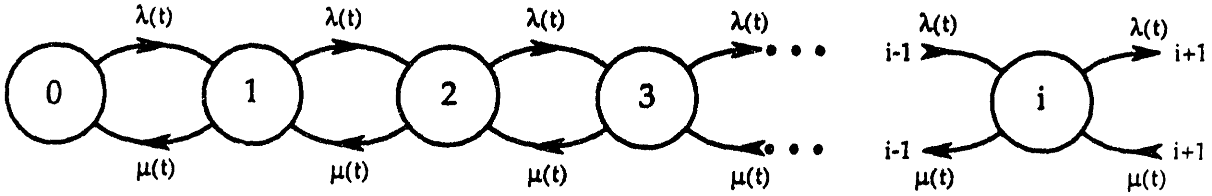


FIGURE 4.2

For our model, we assume a Poisson arrival process, as justified in the previous section. The service process is assumed to be governed by a time varying Erlang process with a variance (resulting from the order chosen) that approximates the true service variance. This allows detailed analysis of the transient effects of changes in the variance and rate of the service process. The queue is a finite FIFO queue.

The transient solution is generated by representing the queue length as a Markov system of discrete states. The transition rates between the states are determined by the rate of demand and service. Demand rates apply to the up transitions, that is, arrivals. Service rates apply to the down transitions. The continuous time Markov system corresponding to a queue witł Poisson service time is shown below. The zero state is absorbing, since it is not possible to reduce the length of the queue below zero.



Markov State Space Representing the Queue Length of the M/M/1 Queue

FIGURE 4.3

We can write equations relating the rate of flow of probability between the states in the Markov system since the inflow to each state must equal its outflow. These are called the balance equations. In addition, since the system represents a discrete distribution of queue lengths, the total probability of all the states must sum to one.

$$\lambda(t)P_0 = \mu(t)P_1$$

$$(\lambda(t)+\mu(t))P_1 = \lambda(t)P_0 + \mu(t)P_2$$

$$(\lambda(t)+\mu(t))P_2 = \lambda(t)P_1 + \mu(t)P_3$$

$$(\lambda(t)+\mu(t))P_3 = \lambda(t)P_2 + \mu(t)P_4$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$(\lambda(t)+\mu(t))P_i = \lambda(t)P_{i-1} + \mu(t)P_{i+1}$$

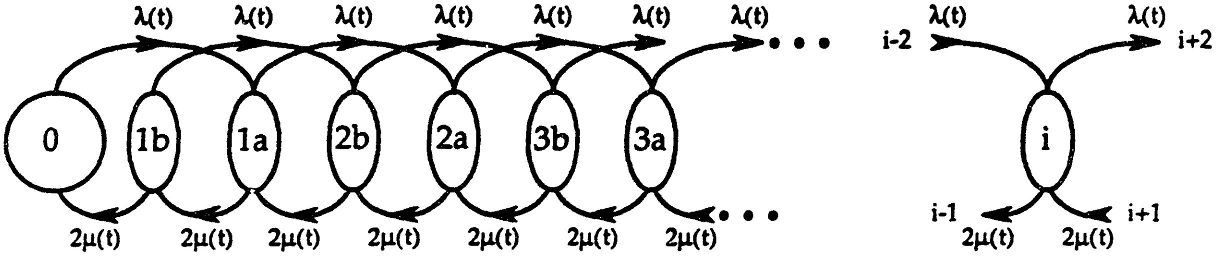
$$\sum_{i=0}^{\infty} P_i = 1$$

The evolution of this system is described by the differential equations that relate the rate of change of probability of each state with the rate of inflow and outflow. The differential equations can be written directly from the balance equations. We can discretize the differential system by replacing the instantaneous time derivatives of flow by incremental flows for small time steps. This results in a system of difference equations. If the time increments are small enough, the higher order transitions (i.e. those resulting from a jump to two or more states) in the difference equation system are assumed to have negligible probability, and are dropped. The resulting difference equations are shown below.

$$
\begin{array}{llll}
P_0^{t+1} & = P_0^t(1-\lambda(t)) & + P_1^t\mu(t) & \\
P_1^{t+1} & = P_1^t(-\lambda(t)-\mu(t)) & + P_0^t\lambda(t) & + P_2^t\mu(t) \\
P_2^{t+1} & = P_2^t(-\lambda(t)-\mu(t)) & + P_1^t\lambda(t) & + P_3^t\mu(t) \\
P_3^{t+1} & = P_3^t(-\lambda(t)-\mu(t)) & + P_2^t\lambda(t) & + P_4^t\mu(t) \\
\vdots & \vdots & \vdots & \vdots \\
P_i^{t+1} & = P_i^t(-\lambda(t)-\mu(t)) & + P_{i-1}^t\lambda(t) & + P_{i+1}^t\mu(t)
\end{array}
$$

The probabilities of the states at any point in time can be solved for using standard difference equation solution techniques. The states are initialized with some probability. Then the solver increments by some small δt and calculates the new state probabilities using the above equations. Delay statistics can then be calculated from the resulting probability distribution of the states of the queue.

In order to represent accurately the transitions of an Erlang process in a Markov system the Erlang must be viewed as a sum of exponentials. Translated into the language of Markov systems this means that in order to make one 'Erlang' transition the system must make a number of exponential or 'Poisson' transitions equal to the order of the Erlang. Thus a customer in our system will need to complete a sum of exponential services to complete a single Erlang service. This manifests itself in the Markov system as stages of service states. An arrival causes the system to jump up a full set of service states, while service requires the customer to pass through each service state before the queue has one less customer. The Markov system representing a second order Erlang is shown below. The balance equations, differential equations, and difference equations can all be written directly from the Markov system, as for the Poisson system above.



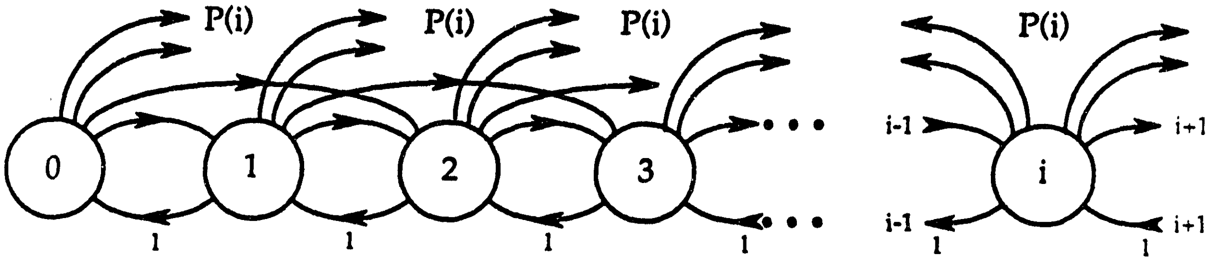
Markov State Space Representing the Queue Length of the M/E2/1 Queue

FIGURE 4.4

Unfortunately this 'method of stages' increases the number of states required to represent the system, and hence the number of calculations required to solve it. Since we replace each of the original queue states with a number of service states equal to the order of the Erlang service process, the number of states increases with the order of the Erlang. Very high order Erlang processes, possibly used to approximate a deterministic process, increase computation costs enormously.

A solution to this problem in the case of approximating deterministic service processes is to construct a Markov model where the time increment is equivalent to the deterministic service time. We know that only one service will be performed in each service increment, and so there are no higher order combinations of arrivals and services to contend with. The service increment will usually imply a significant probability of more than one arrival, but since arrivals will only be coupled with a single service, the Markov property is preserved. In each time increment, the probability of each state is recalculated with respect to the potential for arrivals. Then, the state probabilities are transferred deterministically down one state in the queue to represent the deterministic service that was executed during the interval. As long as this system is observed only at the conclusion of a service and has a very small chance of being empty, it will closely approximate an infinite order Erlang or truly deterministic service queue.

The Markov representation of such a system is shown below. By assuming that the time increment exactly equals the service cycle and increments at the end of each cycle, the system is forced to be Markovian. The representation below is of the discrete time version, so the transitions are labeled with probabilities. P(i) stands for the Poisson probability of i arrivals in each service interval.



Markov State Space Representing the Queue Length of the M/D/1 Queue

FIGURE 4.5

The use of Poisson (M(t)/M(t)/1) and deterministic (M(t)/D(t)/1) variations of the difference equation model as tools for predicting air traffic delays were first explored by Koopman. [KOOP 72] Extensions to the Erlang (M(t)/E_r(t)/k) case and refinements were made by Odoni, Hengsbach and Roth.

[HENG 75] [HENG 74] [ROTH 79] [ODON 83]

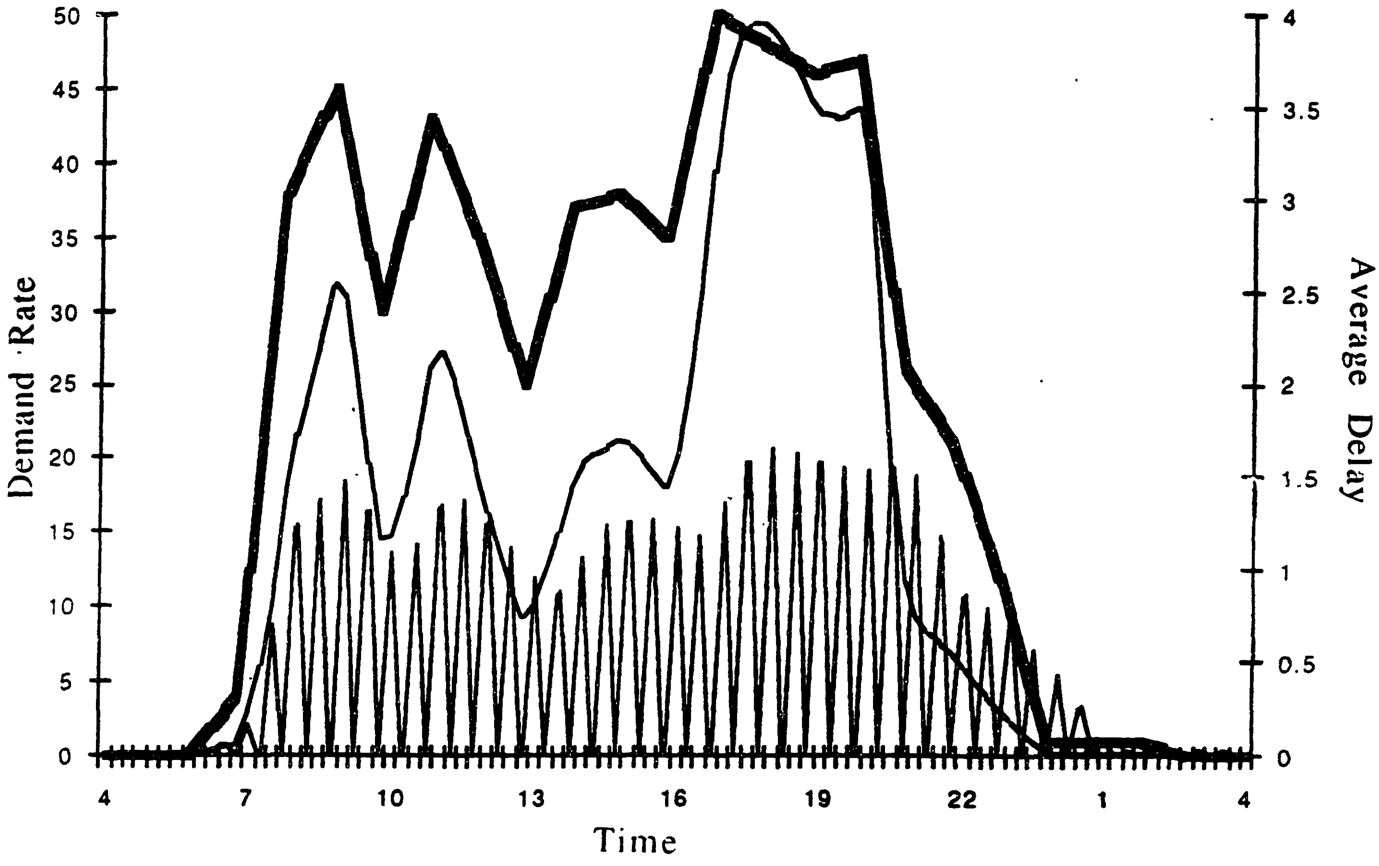
Of prime consideration in the implementation and use of these difference equation models are their computational characteristics. We have seen that an Erlang model requires proportionately more states by its order than the Poisson. This makes the number of computations in the Erlang model increase as the order increases. Another area of concern is the time increment used in solving the difference equations. The time increment in the Erlang and Poisson models must be chosen such that there is little probability of two events occurring at once. As the demand and service rates increase, the time increment must decrease at approximately the same rate. This of course increases the number of time intervals required to solve the system over a specified time period, and therefore the number of calculations. The model with deterministic service does not exhibit this property since the time increment is preset to the service interval. However, the number of significant upward transitions will grow as the arrival rate increases, also increasing the number of computations. Thus the computational cost of all of the difference equation models increases in proportion to the service and demand rates.

In any standard implementation of these models, the states which have insignificant probability would not require any calculations. As the probable queue length grows the number of significant states requiring calculation increases. The rate of growth of the queue is roughly proportional to the utilization ratio, that is, the demand rate divided by the service rate. If the demand rate is much higher than the service rate, the queue is bound to grow, and vice versa. Thus computation also grows in proportion the utilization ratio.

An additional computational consideration is that the rate of transition between substages of an Erlang system must be set multiplicatively higher by the order of Erlang. This higher rate between the substages insures that the expected number of services will be the same as would have been performed by an equivalent Poisson system. This higher transition rate implies a decrease in the time increment required for the solution to converge. The smaller time increment means increased computations.

Thus the computational costs of the Erlang system are doubly sensitive to the order of the Erlang. A higher order increases the number of states and decreases the time increment. This combination causes the amount of computation of an Erlang system to vary in proportion to the square of the order of the Erlang. If one were to make the demand process Erlangian, as well as the service process, as in the $E_i/E_j/k$ queue, the amount of

A final computational consideration is that the maximum time increment that results in a solution which converges to the actual continuous time solution cannot be determined exactly. Most algorithms use a rough heuristic to guess an initial time increment, and then test a smaller time increment by redoing the calculations and checking that the solution does not change substantially. If it does, the smaller increment is made the current guess and an even smaller increment is tested. If too large an increment is used, the solution does not converge and the system can exhibit strange behavior. The figure below shows a converging solution for the transient delay of an M/M/1 queue. The demand rate is shown in heavy black, and the service rate is constant at 55 operations per minute (not shown). It also shows a non-convergent solution (rapidly oscillating) which used a time increment that was too large.



The proper time increment must be redetermined by this method each time the characteristics of the system change significantly, which can be often in the case of air terminals. Thus the difference equation systems are also sensitive to the rate of change of the inputs. A high rate of change will increase the number of wrong guesses as to a good time increment.

In summary, the amount of computation for these models increases in proportion to the rate of demand and service rates, the rate of change in demand and service rates, and the utilization ratio. In the case of the Erlang, computation also increases in proportion to the square of the order of the Erlang. While the difference equation models require far more computation than the previous two models, it is still less than the amount required by simulation. In general simulation requires computation that increases exponentially with the size of the problem. The next two sections discuss approximations to the full difference equation models that require less computation.

4.2.4 Interpolated Model ($M(t)/E_{r(t)}/1$)

The intuition behind an interpolation model is simple and is based on the relationship between M/M/1, M/D/1, and $M/E_r/1$ queues. We might conjecture that by the nature of the different service processes of these queues, the transient result for the $M/E_r/1$ queue is bounded above and below by the transient solution to the M/M/1 and M/D/1 queues. In addition, we know that the transient solution to the Erlang system is equivalent to the solution for the M/M/1 model for Erlang systems of order one, and therefore would suspect that the solution moves asymptotically toward the solution for the M/D/1 queue as the order of the Erlang goes to infinity. This leads to a rough method of approximating the M/E$_r$/1 queue by interpolation between the results of the M/M/1 and M/D/1 queues.

The M/M/1 queue has a service process that is in some sense as random as possible. If an observer checks the queue at any instant, the probability distribution of the remaining service time is exactly the same. Thus the past events in the queue, such as when the present customer entered service, give no information as to its future behavior. The exponential distribution has a coefficient of variation equal to one. This presents a possible limitation in that distributions with higher coefficients of variation are not well approximated by the exponential. It is possible to postulate distributions with greater variance, but none of them will independent of the time the customer has already spent in service.

The M/D/1 queue, on the other hand, has a deterministic service process. The service for all customers always takes the same amount of time. This process has no variance, and its coefficient of variation is zero.

The M/E$_r$/1 queue has a service process with a service time variance that varies from that of the M/M/1 queue for order one, to deterministic, as in the M/D/1 queue, when the order of the Erlang approaches infinity. It is this characteristic that allows us to match service time variances by changing the order of the Erlang. The Erlang family allows us to specify coefficients of variation between zero and one.

The steady state solution to M/E$_r$/1 queueing systems is equivalent to the solution to the M/M/1 at order one (since they are the same system exactly) and approaches asymptotically the solution to the M/D/1 as the order goes to infinity. This is clear by observing the behavior of the steady state waiting solution to the M/G/1 queue.

$$W_{M/G/1} = \frac{\lambda\sigma_S{}^2+\rho^2}{2(1-\rho)}$$
$$\text{where } \rho = \frac{\lambda}{E[S]}$$

By substituting the variance of the M/M/1, M/D/1, and M/E$_r$/1 into this formula, we obtain:

Poisson

$$\sigma = \frac{1}{\mu^2} \quad \therefore \quad W_{M/M/1} = \left(\frac{\lambda}{\mu(\mu-\lambda)}\right)$$

Erlang

$$\sigma = \frac{1}{r\mu^2} \quad \therefore \quad W_{M/E_r/1} = \frac{1+r}{2r}\left(\frac{\lambda}{\mu(\mu-\lambda)}\right)$$

Deterministic

$$\sigma = 0 \quad \therefore \quad W_{M/D/1} = \frac{1}{2}\left(\frac{\lambda}{\mu(\mu-\lambda)}\right)$$

It is clear that the steady state solution to the M/E$_r$/1 system should always be ((1+r)/2r) of the steady state solution to the M/M/1 system. The figure below shows different Erlang systems approaching the steady state after a step in demand.

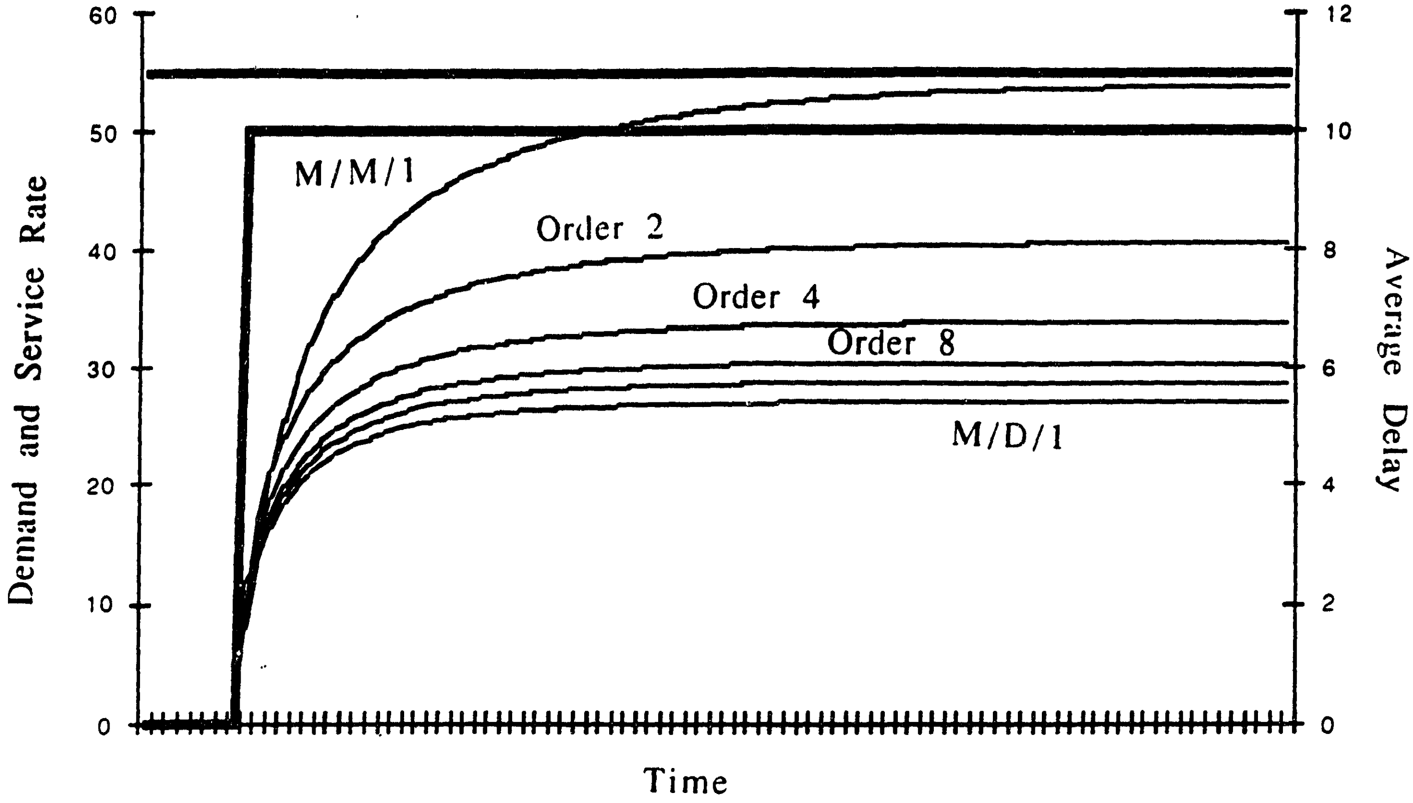


FIGURE 4.7

One might also suspect that the transient solution (as opposed to the steady state) to the M/E_r/1 queue exhibits similar behavior. This is roughly true, but not exactly. In the short run the transient solution of the M/E_r/1 queue is not simply a fraction of the transient solution of the simpler queues, although it is in the long run (i.e. steady state). The difference lies in the way the different systems approach steady state.

Each system approaches steady state in an exponential manner. The rate at which the path reaches the steady state is determined by the time constant of the system. Unfortunately, queueing systems with more variable service processes have higher time constants and require longer to reach the steady state than those with lower variance. This causes them to react faster to transient changes in the inputs and results in different transient response. The inputs to our systems are always changing, implying that the very short term transient response is more important in determining the transient response than the eventual steady state level.

In trying to approximate the transient response of the M/E$_r$/1 queue, we assume that it is always between the the transient response to the M/M/1 and M/D/1 queues. This limits our search to a convex interpolation. This assumption is supported by the behavior of the steady state solutions outlined above. We also assume the transient solution moves monotonically toward the deterministic solution as the order of the Erlang rises, and that it approaches the deterministic solution asymptotically. These assumptions are also supported by the behavior of the steady state solution. The figure below shows the transient response for a number of orders of Erlang queues, further supporting this assumption.

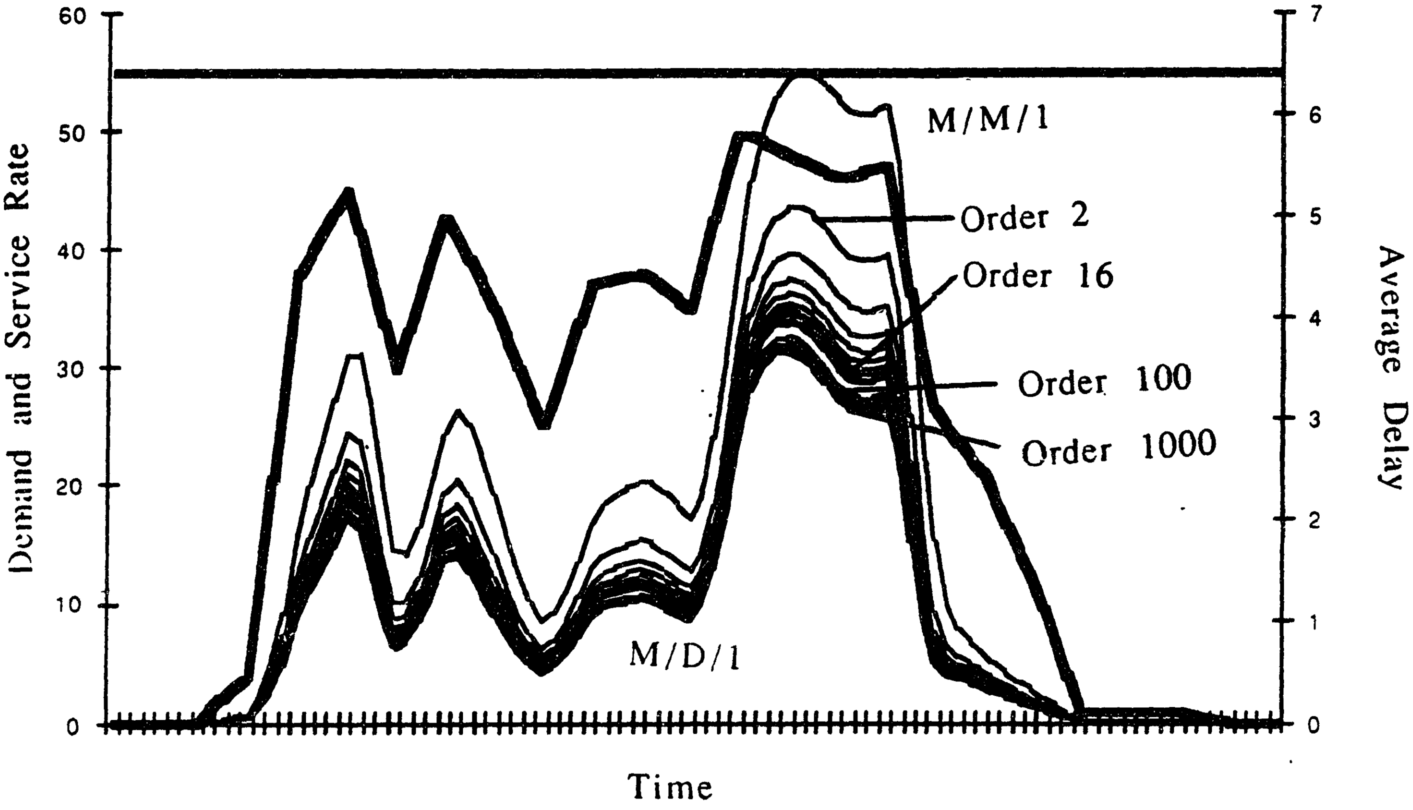


FIGURE 4.8

It seems that one might be able to construct an approximation to the transient solution to the $M/E_r/1$ queue by combining the results of the transient solutions to the M/M/1 and M/D/1. This is exactly what was attempted. First a broad search was made of a variety of functional forms, and then the forms that worked well were refined. The functional combinations were tested by producing transient results to simple step input functions using M/D/1, M/M/1, and a variety of orders of $M/E_r/1$ queues. Then the solutions to the M/D/1 and M/M/1 queues were used to construct the terms of the functional form in question. The combination of terms was then regressed against the $M/E_r/1$ solutions. From this regression the coefficients of the terms were estimated, which completed the functional form, and the residuals could be analyzed to see how well the functional form worked. A number of cycles of solution generation, transformation, regression, and modification of the functional form were performed as the approximation moved closer and closer to the true values.

The best interpolation function found is shown below:

$$W(t)_{M/E_r/1} = \left(\frac{1}{r}\right)W(t)_{M/M/1} + \left(\frac{r-1}{r}\right)W(t)_{M/D/1}$$

It is interesting that the Poisson coefficient plus one half of the deterministic coefficient sums to (r+1)/2r, the steady state scaling factor for the Erlang. Thus this interpolation not only works well in approximating the transient solution, but it approximates the steady state solution exactly. By placing a larger weight on the Poisson solution at low orders, it imposes a slower rate of change consistent with the higher time constant. As the order of the

Erlang gets larger, it places more weight on the deterministic solution, imposing the faster rate of change inherent in the falling time constant.

The approximation works surprisingly well. In numeric tests with standard input functions the transient solution was correct to within 2 or 3%. Shown below is the transient response to the demand profile shown above of a second order Erlang queue. Also shown is the interpolated solution. In heavy black, corresponding to the right hand axis, is the vertical distance between the two.

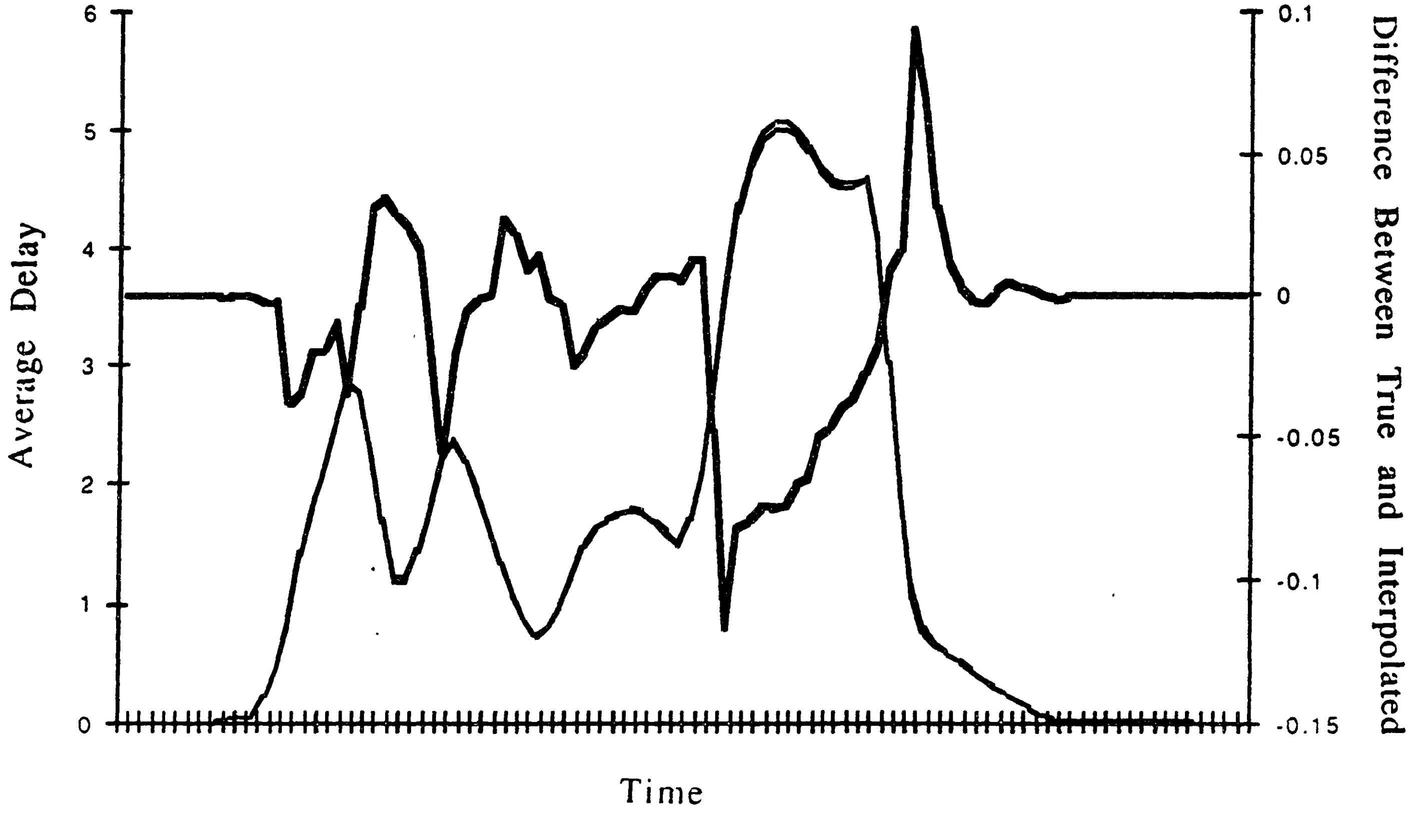


FIGURE 4.9

One can see that the interpolated solution drops below the true solution when the input is rising, and rises above when the input is falling. This is a result primarily of the time constant difference. The interpolated model takes longer to ramp up and longer to fall. The differences are not substantial, though. Overall, the solutions match extremely well, as is demonstrated by the next two figures showing the results for a fifth and tenth order Erlang queueing system.

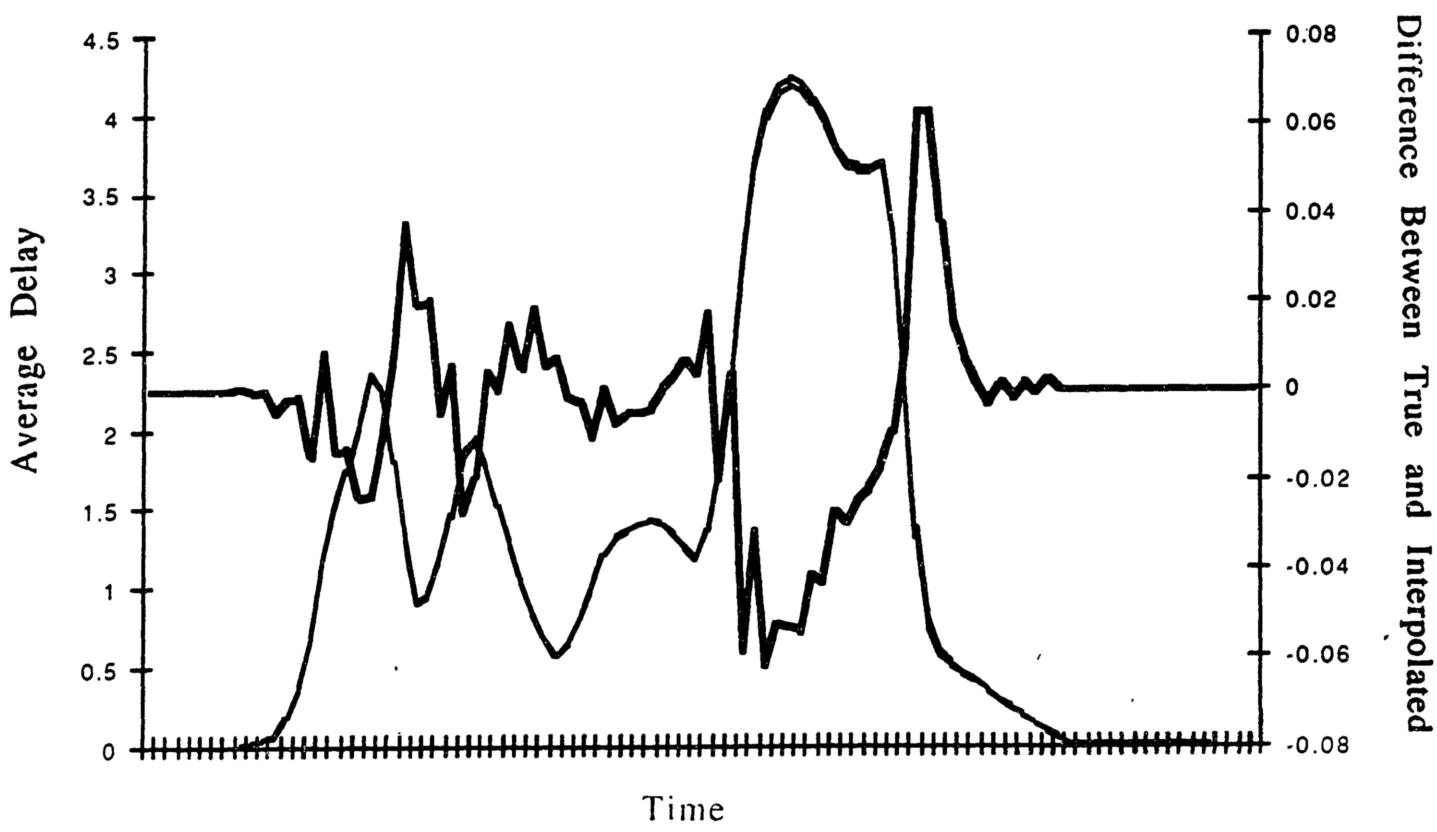


FIGURE 4.10

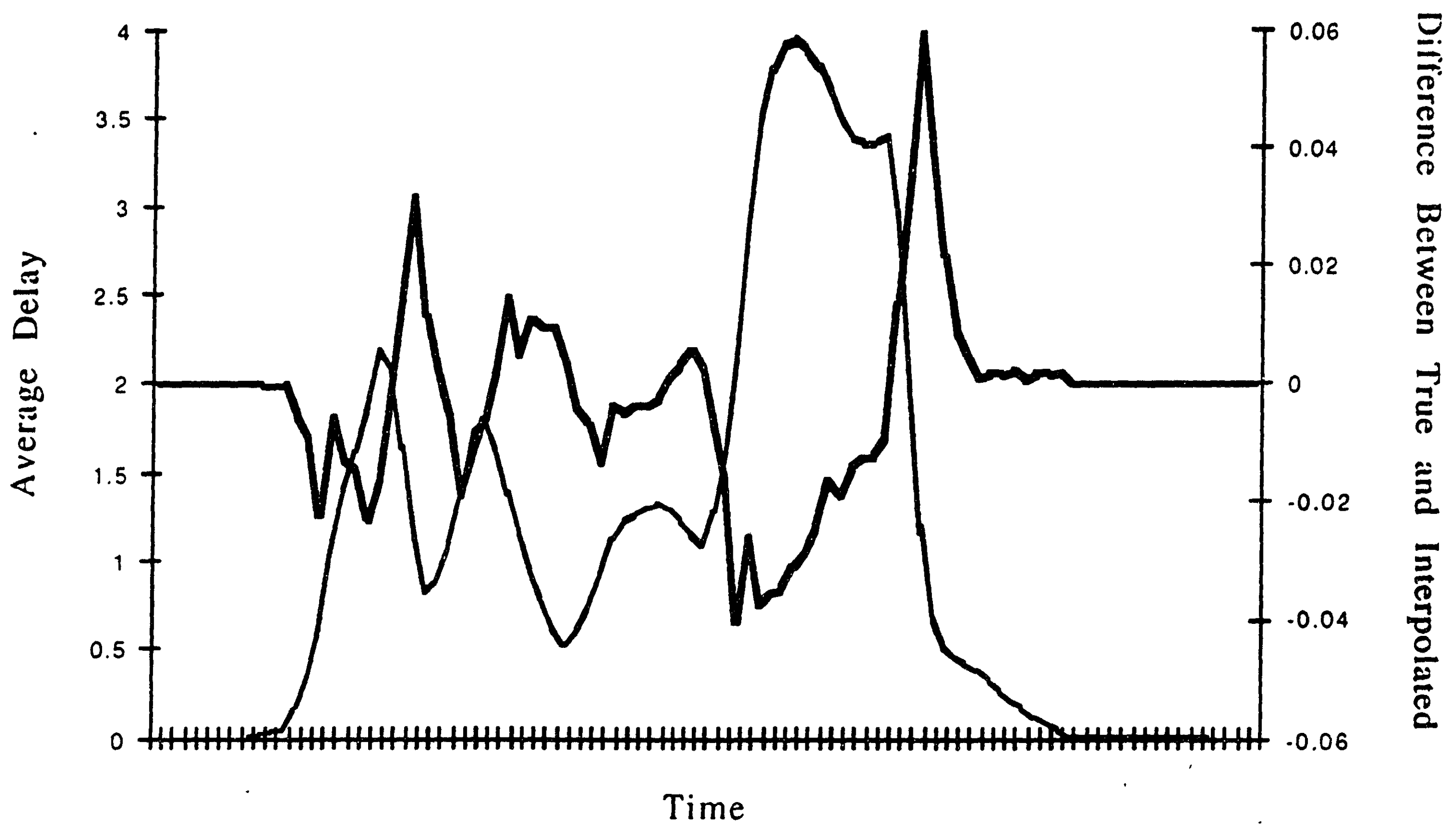


FIGURE 4.11

The primary benefit of this Erlang approximation is that its order of computation is independent of the order of the Erlang. It requires the calculation of the solution to the deterministic and Poisson queues, and then is able to generate approximate Erlang queue solutions of any order with relatively little effort. The next section outlines a model due to Kivestu that does better in that it only requires the solution to a deterministic queueing system. It also matches the time constant of the Erlang queue better than the interpolation model did. It is not, however, as intuitive as the interpolation model.

4.2.5 Kivestu Approximation Model $(M_{(t)}/E_{r(t)}/1)$

The Kivestu model is another approximation of the transient solution to the $M/E_r/k$ queue. It was developed by Peeter Kivestu as part of his masters thesis in 1976. [KIVE 76] The idea comes from the aforementioned Bessel function decomposition for constructing transient solutions. Kivestu investigated the behavior of the time constant in the transient response to a step function using this Bessel function decomposition to form the transient solutions. The time constant determines the rate at which the solution moves exponentially toward its steady state. The actual steady state level is determined without the time constant. However, in the short term the time constant dictates how quickly the solution responds to a transient step. Since the typical time varying input functions to these queues can be viewed as many small steps close together, the short term responses to all of these steps becomes the transient response as the time increment of the steps goes to zero.

The base of the model is the standard set of difference equations used to iteratively solve for the M/D/k transient response. In this model the time step is assumed to be equal to the deterministic service time. This is because at the end of each service cycle the queue state depends only on the state at the end of the previous cycle. Thus the system has the Markov memory property and can be represented as a time varying Markov state system. In the Kivestu model, though, the calculations in each service increment are performed assuming that the increment is the service cycle, but the solver actually takes a time step that is different than the service cycle. If the system is approximating a high order Erlang the time step is very close to the actual deterministic time step. As the order of the Erlang being approximated gets lower, though, the time step is increased until at order one (Poisson), the time step actually taken is double the deterministic one.

Since the time step goes up as the order comes down, fewer planes are deterministically served in a given period, increasing the length of the queue and approximating the increased length that would occur with lower order, more variable Erlangs. When the order is one, only half of the planes are deterministically served, doubling the length of the queue and the waiting time, thus perfectly approximating a Poisson, at least in steady state. This change in the time increment also has the effect of making the queue respond faster at higher orders and slower at low orders, approximating the change in the time constant.

This model comes very close to approximating the $M(t)/E_r(t)/1$ queue while requiring only as much computation as the deterministic service $M(t)/D(t)/1$ queue. It is therefore the least costly approximation of the $M(t)/E_r(t)/1$ queue. The accuracy of approximation is at least as good as the interpolation model described previously. This model in the one used for most of the analysis in Section 5.

4.3 Other Possible Models

For models where the processes are not exponentially based and thus cannot be represented as Markov systems, simulation is an obvious solution. Simulation, though, is generally very computationally expensive. It is possible to arrive at the transient solution for models that do not have exponentially based processes using analytical methods, however. The procedures requires decomposition of the inputs into the space of Bessel functions, manipulation of the transforms, and recomposition into the queue length space. This procedure is so computationally expensive that it is rarely used.

An alternative decomposition that is less expensive has recently been developed by Nakazoto and Bertsimas. [BERT 89] This decomposition requires that the interarrival distributions of the processes be composed of sums of exponentials, but they do not need to be the same exponentials, as the Erlang requires. Any interarrival distribution, discrete or continuous, can be closely approximated by some combination of exponential distributions. Thus this method is very flexible. It requires far more computational resources, though, than the above Erlangian approximation models. Such a system would be useful only when the characteristics of the underlying process preclude it from being represented as a queue with Erlang service times and Erlang arrivals.

5. Logan Analysis

5.1 Model of Logan Airport

An approximate analysis of Logan airport delays was conducted using a number of the models investigated in this thesis. The models were implemented on a Macintosh II in a menu driven system that generated transient solutions for the models when given as input the demand and service rates during the day. The solution could be viewed within the program, or saved in flat files. The flat file data was then analyzed using Excel or Mathematica. The system was written in C.

The demand function used as input in the analysis was the average weekday profile of scheduled arrivals to Logan in 1987, with an added number of unscheduled, general aviation flights. In order to test the sensitivity of the model to different levels of total demand, the profile was simply scaled proportionately up or down so that the integral under the profile was equal to the total number of flights per day. In all of these analyses the service rate at Logan was presumed to be constant throughout the day. In order to test the sensitivity of the model to service rate changes, the service level was simply increased or decreased. The demand and service rate functions were represented as piecewise linear approximations, usually with one hour time increments.

5.2 Sensitivity to Service and Demand Rates

The first tests which were run investigated the delays over a whole day with respect to changes in the utilization ratio, that is, the ratio of demand rate to service rate. The Kivestu approximation of a third order Erlang service queue with time varying Poisson demand was used as the airport model. The solutions are plotted below. Note that operations includes only landings. Arrivals were not considered customers of the landing server.

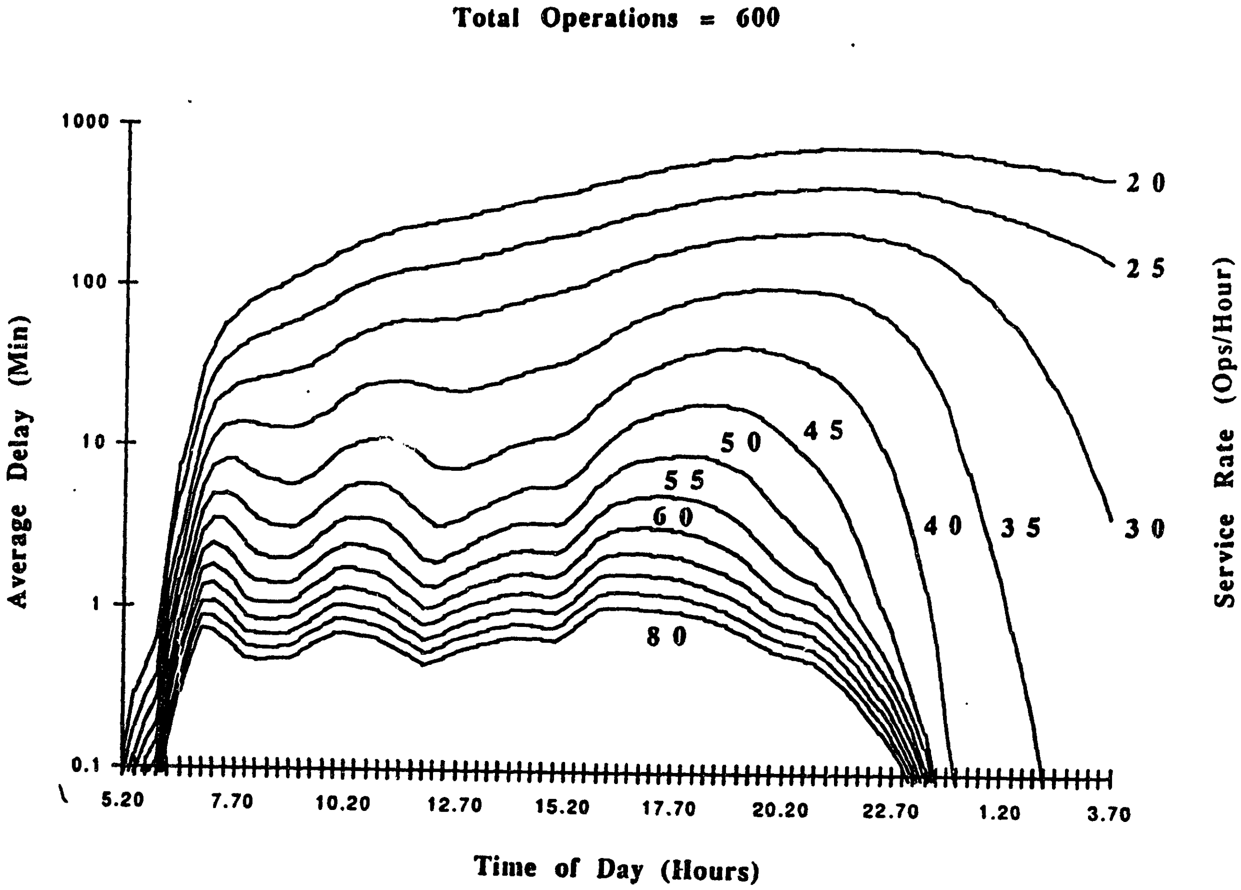


FIGURE 5.1

The graph shows time of day plotted against the log of the average delay. This is a macroscopic view of what happens to the system over a large range of utilization ratios. Two characteristics are obvious from this data. First, at an average service rate for arrivals of 35 and below, Logan could not even land all of the planes that would probably arrive during the day. Planes would have to be turned back or delayed at their origin. The second characteristic evident from the graph above is that total delays increase exponentially as the service rate falls.

A second graph, below, shows average delay plotted against the time of day for different service rates u. The capacity (service rate) of the runway system at Logan for arrivals can vary between 32 and 60 operations per hour. The characteristic increase of delays exponentially with respect to changes in the service rate is evident in the relation between the peaks and troughs.

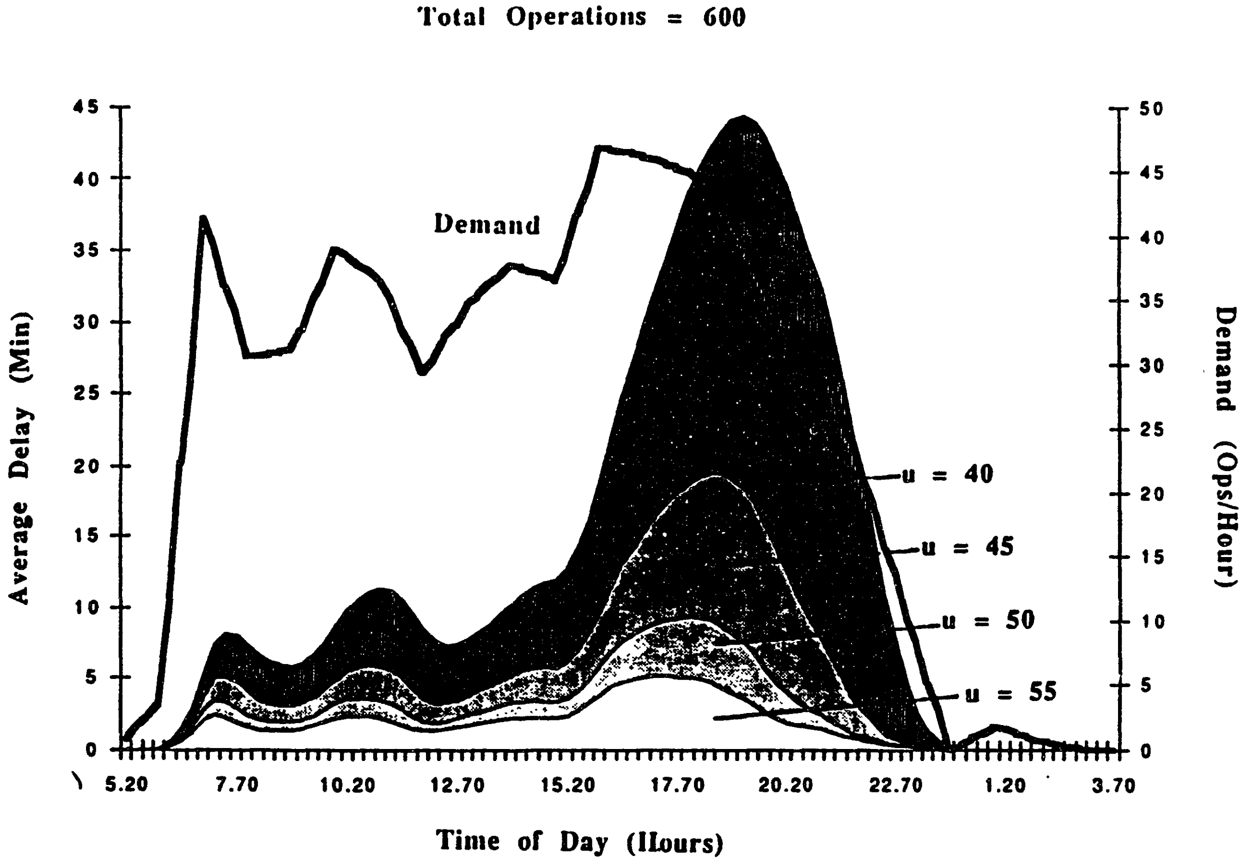


FIGURE 5.2

Further investigation of the behavior of delays over a day demonstrated an interesting characteristic of saturation. As the system becomes more and more utilized, one might expect the difference between the delays in the system with probabilistic service time and the delays in the system with deterministic service time to increase. Indeed this is the case up to a point.

When the queue is very heavily saturated, though, the variable service time system seems to behave more and more like the deterministic system. The difference between the two becomes smaller. This effect is manifested in the hump in the graphs below.

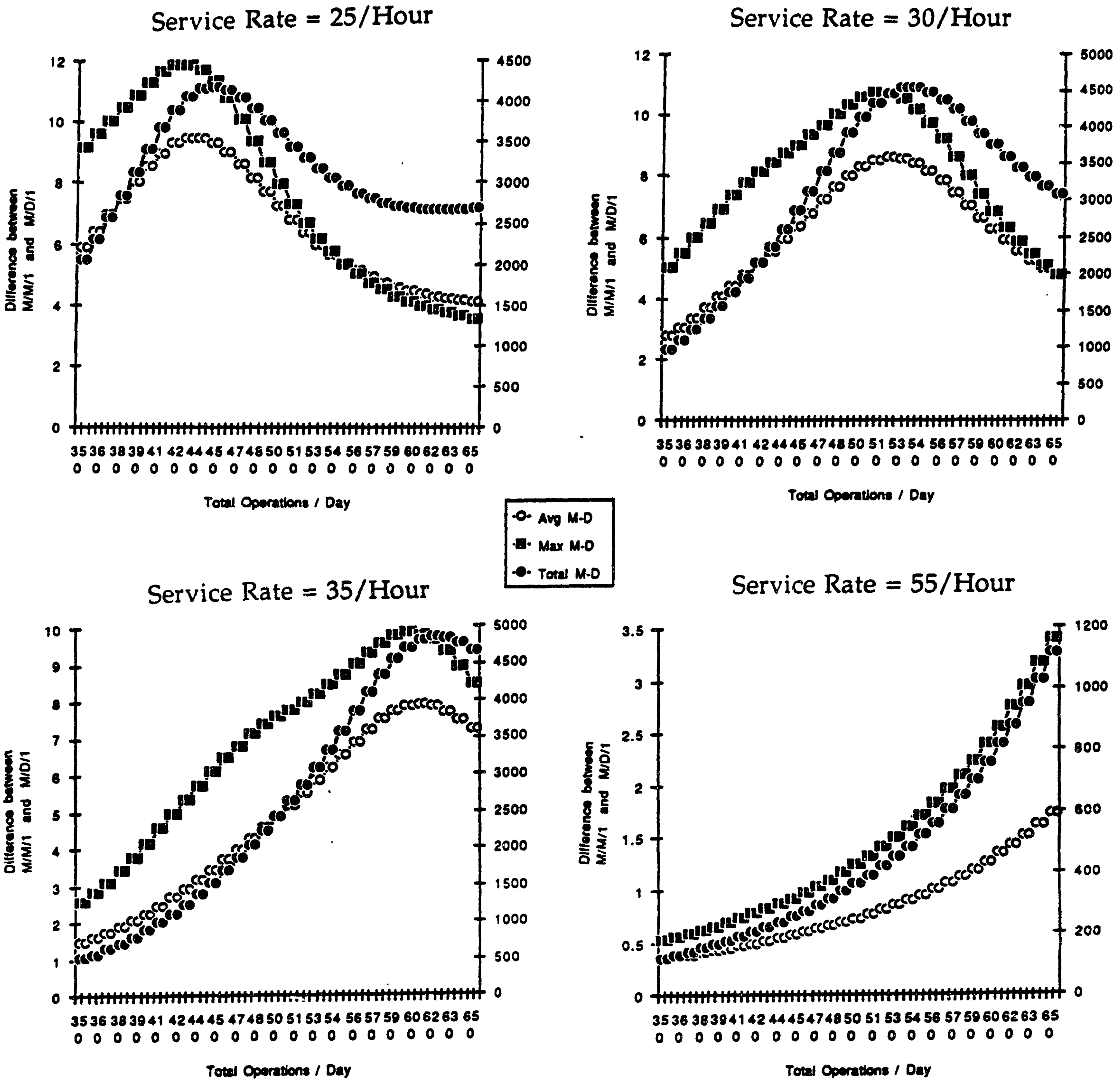


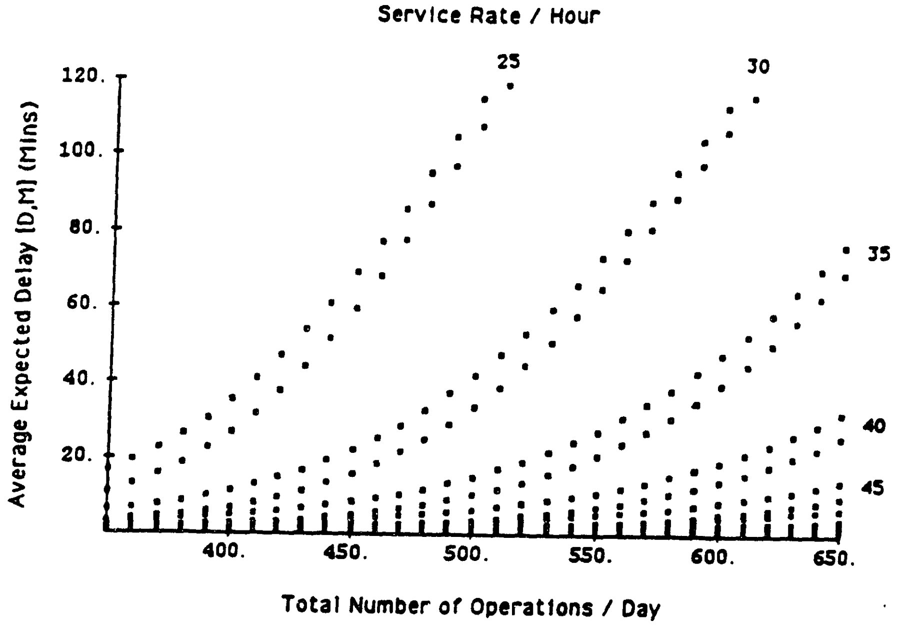
FIGURE 5.3

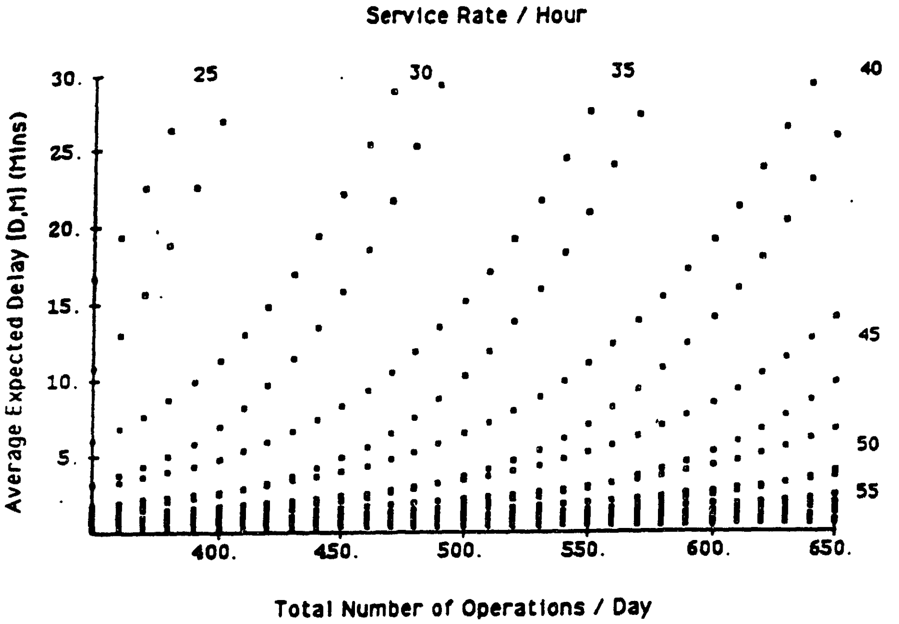
From the first graph we can see that as total operations per day increases, the difference between the solutions to the M/M/1 and M/D/1 queue diverge. The delay predicted by the M/M/1 queue grows faster than the delay predicted by the M/D/1. At some point, however, the solutions stop diverging and begin to converge. This occurs at about 440 operations per day in the first graph. The difference between the solutions falls, and then levels off and becomes steady as the number of operations rises above this point. From the other three graphs, one can see that this hump moves into higher ranges of demand when the service rate rises. That is, as the utilization ratio falls, the hump does not occur until higher levels of demand.

5.3 Sensitivity to Service Rate Variance

In this study the delays induced by a Poisson and a high (30) order Erlang queue were generated for various combinations of average service rate and total operations using the Kivestu approximation program. The solutions for the average delay over the whole day are plotted as two curves, one for the various demand rates using a Poisson server, and the other using the 30-order Erlang server. The Erlang curve is the lower one in all cases. The two curves form a region in which systems with server time variance between Poisson and order 30 Erlang would fall.

The magnified versions of the graphs below indicate that there is a definite tradeoff between variance and delay in low saturation scenarios. In other words, when the queue is not heavily saturated, a reduction in the variance of the landing times can be more beneficial than an increase in the rate of landings. This result implies that any modifications to the terminal server that would reduce landing time variance would be as beneficial in low saturation scenarios as modifications which simply increased the rate of service. This tradeoff is manifested in the plots where the Erlang curve for a higher utilization ratio intersects the Poisson curve for a lower utilization ratio. That is, the average delay caused by an Erlang server at a lower service rate drops below that of a Poisson server at a higher service rate.





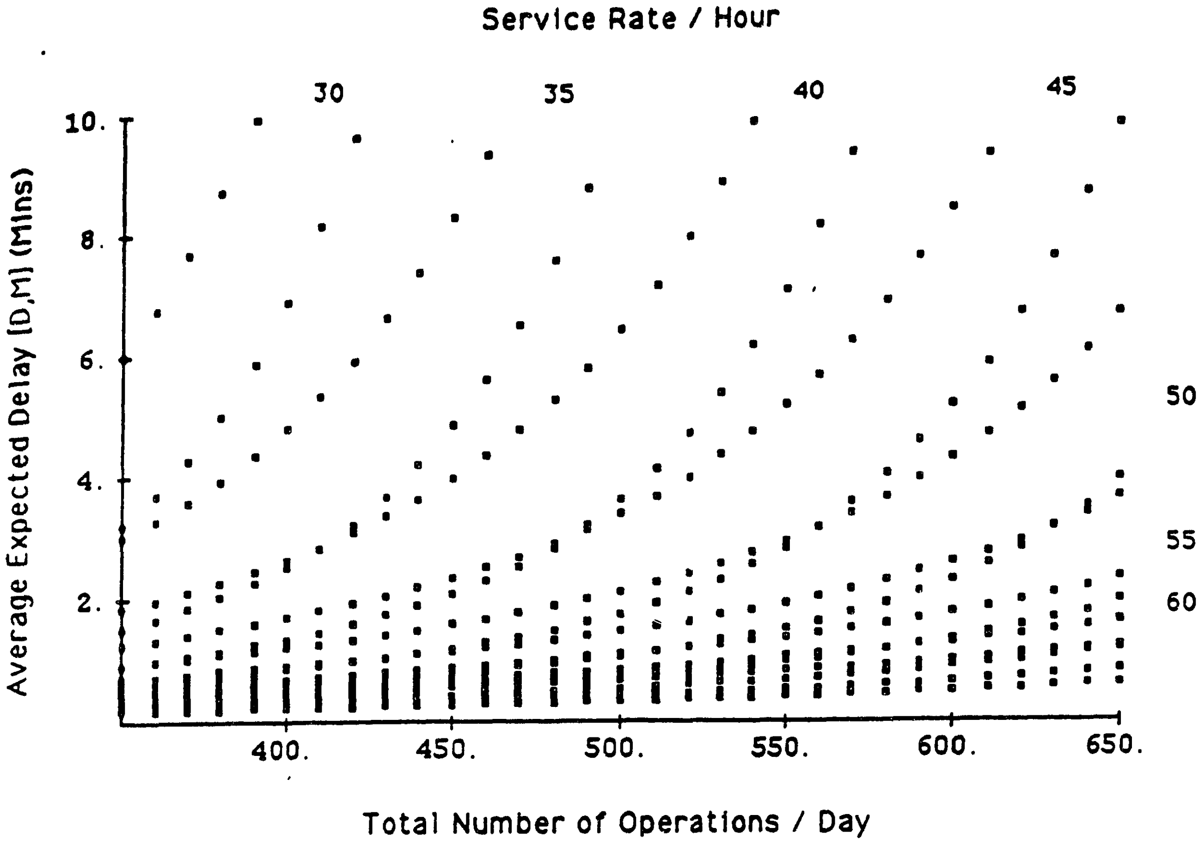


FIGURE 5.4

In the bottom graph, which is simply the first graph magnified with respect to average delay, we observe that the region encompassed by the Poisson (top) and Erlang (bottom) curves for each service rate intersect the regions corresponding to other service rates. This implies that in these areas service time variance is relatively more important in affecting delays than service rate. Even in the areas where the regions do not intersect, though, the substantial vertical distance (difference in average delay) between the Poisson (high variance) and Erlang (low variance) delays demonstrates the importance of service time variance in reducing delays.

5.4 Accuracy of Poisson Arrivals Assumption

To test the accuracy of the Poisson arrivals assumption, the actual Logan schedule for arriving flights was used to generate randomized arrival patterns that incorporated an uncertain deviation from the scheduled arrival time for each flight. A large number of these randomized patterns were compared to what one would have expected from a Poisson generating process. This amounted to testing whether a monte carlo simulation of arrivals, meant to be a close approximation to the actual process governing arrivals, was distinguishable from Poisson generated arrivals.

Each scheduled flight was given a random deviation from its schedule drawn from a lateness distribution. The arrivals were then sorted into what would be their arrival order with the deviations in their schedules. A large sample of these patterns were generated for lateness distributions with differing variances. The patterns were assumed to have the same properties that the actual stochastic process for generating arrivals might have. In order to test the validity of the assumption of Poisson governed arrivals, the rate of arrival, smoothed by moving average over time, that resulted from the schedule was used as the rate of the Poisson process to be tested. The characteristics of the randomized arrivals were then compared to what one would have expected from this Poisson.

A third order Erlang was used as the lateness distribution. Its variance was specified by scaling each draw. The mean of lateness was set at zero. Each scaled draw was added to a scheduled arrival time to generate a simulated late or early arrival time. By adding an Erlang draw to the original scheduled time, the left half of the Erlang distribution was forced to represent earliness. As shown below, this end of the distribution has a hump and is bounded, similar to the behavior one would expect an early arriving plane to exhibit. The infinite tail end of the distribution (right hand side) represents late arriving planes. Just as one would assume earliness to be bounded, one would expect lateness to be open ended. Thus the shape of the Erlang fits well what one would predict to be the characteristics of early and late arrival among scheduled aircraft.

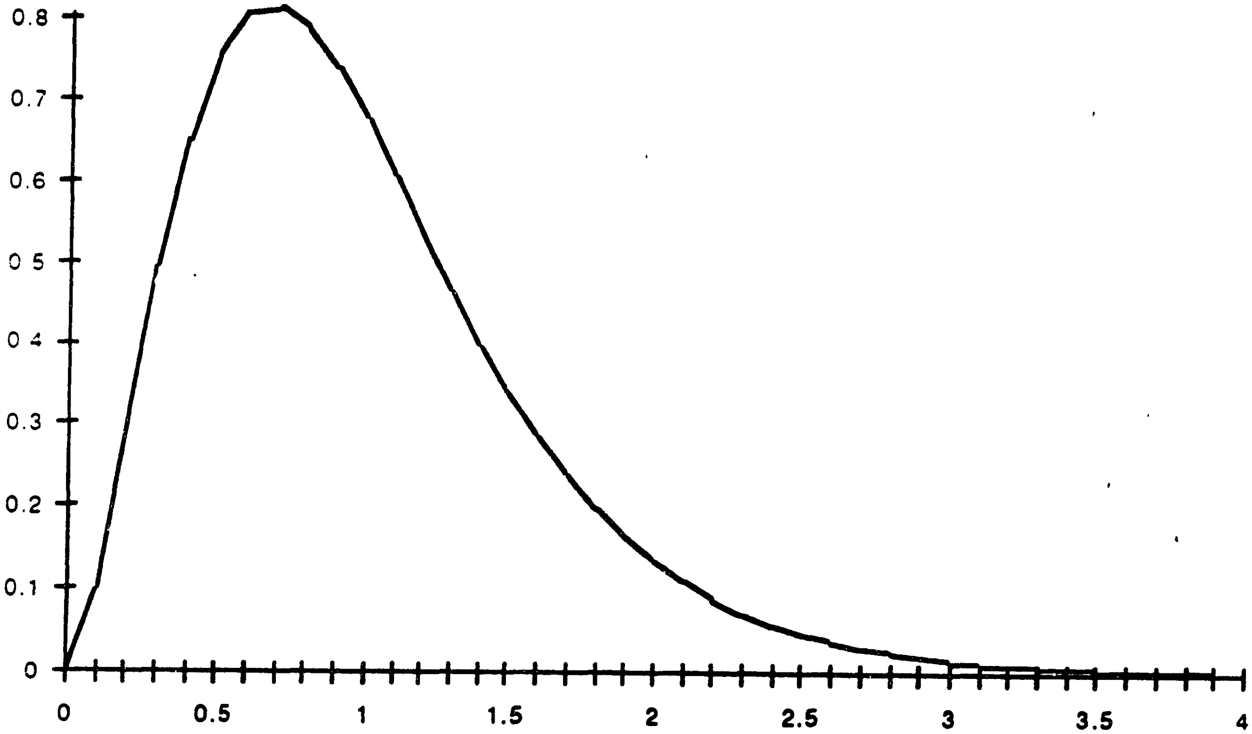
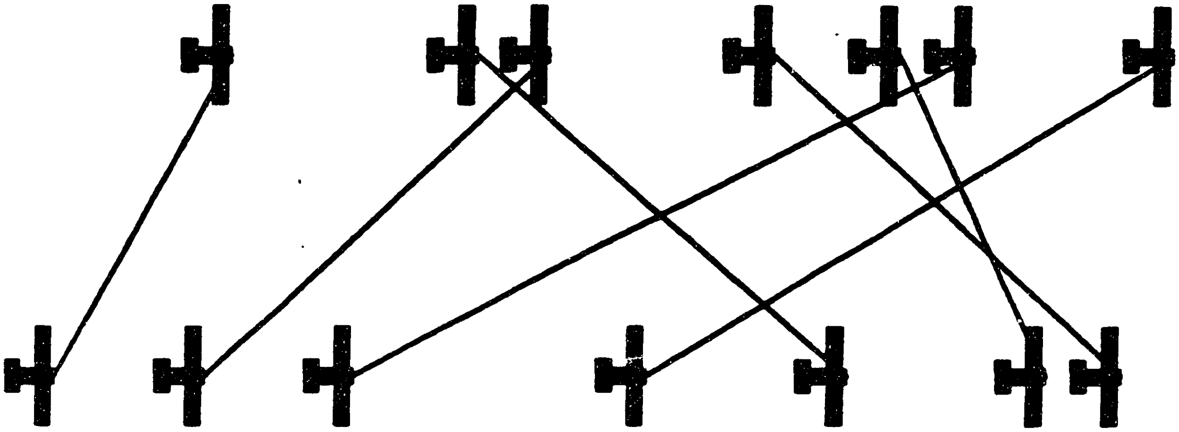


FIGURE 5.5

After the arrival times were transformed by the lateness function, they were re-sorted by their modified arrival time. This constituted a simulation run of scheduled arrivals.



Example of Monte Carlo rearrangement of arrival sequences.

FIGURE 5.6

The comparison to the Poisson was done in two ways. First, the moments of the distribution of simulation arrival times were calculated for the simulation runs of the schedule. This was repeated using lateness distributions of differing variance. The differing variances were generated by scaling the lateness draw by some number of minutes.

When compared to each other, the mean and variance of simulation runs using different variances of lateness distributions were remarkably similar. The mean of the randomized arrivals were 1.56 for both the scheduled arrivals and lateness standard deviations from 1 to 9. Standard deviations higher than 9 caused the mean to rise. This result is reasonable, since the simulation is randomly perturbing 400 densely packed arrivals. One would not tend to see an increased mean interarrival time until a significant number of arrivals spilled over the original borders at either end of the time scale. Since the same number of arrivals are occurring within roughly the same boundaries, the mean will be the same.

The variance of the scheduled arrivals was 3.284. This is not a genuine measure, though, due to the characteristic clumping of predicted arrival times around specific parts of the hour (i.e. half past, quarter of). The estimates of interarrival variance over a broad range of lateness variances centered around 2.5.

The table below shows the results for estimated variance.

Standard Deviation of Lateness Distribution	1	2	3	4	5	6	9	20
Estimated Simulation Variance	2.46	2.41	2.55	2.42	2.56	2.57	2.48	5.28

This data implies that the interarrival distribution for a heavily utilized facility is insensitive to increases in the variance of the lateness of arrivals. It also implies that if the Poisson is a good representation of this process, then it is robust to changes in lateness variance. Of course at very large variances of the lateness distribution the samples did tend toward increasing interarrival variances. At a standard deviation of 20 the variance rises to 5.28. In addition, the distributions of the estimates were more variable at these levels.

Second, the moments of the interarrival distribution of the simulation runs were compared to the moments of the interarrival distribution predicted by the Poisson (an exponential). With an arrival rate of 1.56 per minute, we would expect a variance equal to 1.56, yet the monte carlo simulations reveal a variance more on the range of 2.5. Part of this increase could be explained by remaining macroscopic clumping effects throughout the day.

However, the shape of the distribution was different than would have been expected from the Poisson, lacking the initial peak and infinite tail characteristic of an exponential. In fact, the distribution looks like a second or slightly higher order Erlang since it starts at zero, peaks early, and dies away slower than the exponential. This implies that the Poisson may not be a good representation of the arrival process. A typical sample distribution of interarrival times from our simulation is shown below with what one would expect to be the corresponding exponential distribution.

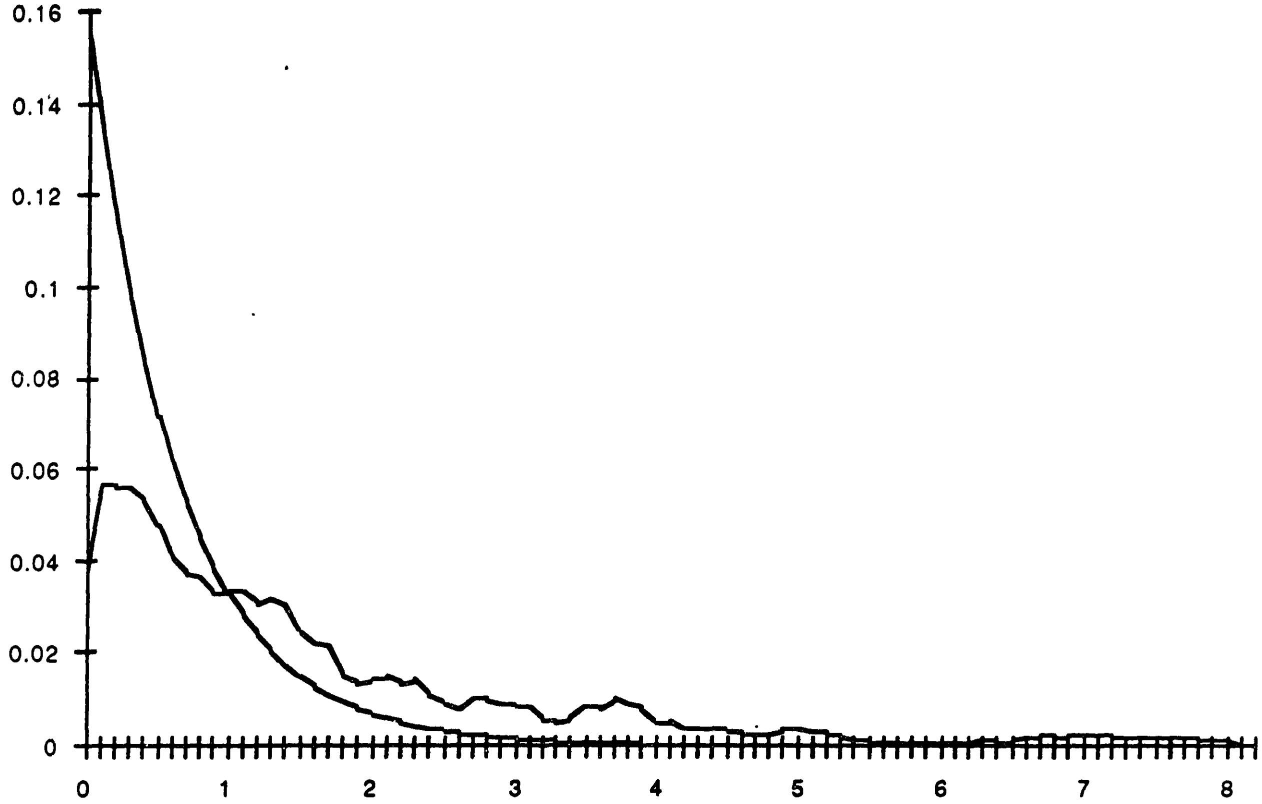


FIGURE 5.7

A characteristic of scheduled arrivals that does not support the Poisson assumption is that scheduled arrivals, while they vary from their schedule by being early or late, all must eventually arrive since the airlines have scheduled them. This implies if the Poisson generated relatively fewer arrivals than one would have expected in the beginning of the day, then it should generate more than would be expected at the end of the day. In other words the total number of arrivals each day should be roughly the same in order to manifest this characteristic accurately.

One can be tricked by just looking at the interarrival time distribution and not the correlations in the stream of interarrival times. The simulation system may produce overall a sample that has an interarrival time distribution that roughly matches that of the Poisson, yet that is actually restricted so that the same amount of arrivals will be generated each day. This is definitely not how the Poisson works, though. The past history of the Poisson process does not affect its generation of arrivals in the future. In addition, the formula for the Poisson distribution shows that the variance in the number of arrivals the Poisson might generate could be considered significant. The variance falls, proportionately, for busier airports though, reducing this problem. For instance, with 625 arrivals over the day, we would expect a standard deviation of 25. Since a Poisson with a high rate approximates a normal, we know that 95% of the arrivals will be within two standard deviations above and below the expected value. (575 < # Arrivals < 675) This is reasonable if a number of unscheduled, general aviation flights are included in the total with a majority of scheduled flights.

A possible solution to this discrepancy is to substitute a low order Erlang for the Poisson demand generator. While it still does not guarantee that the same number of arrivals will be generated each day, it does improve the problem in two areas. The shape of the interarrival distribution will more closely match the sample. Also, the variance of the total number of arrivals generated will be lower, increasing the chances that about the same number of arrivals would be generated each day. One would like to incorporate the correlation effects on the interarrival gaps exhibited by the simulation and arrivals in reality, but doing so would violate the Markov property and then we would not be able to solve the systems using the techniques presented in the previous section. A low order Erlang is a good compromise between the Poisson and such an unmanageable process.

Overall, though, the Poisson assumption seems reasonable. The only stipulations are that the arrival profile should have some amount of unscheduled, general aviation flights to make up for the total variance of the Poisson. For systems with heavy activity the Poisson assumption is most applicable, and it seems robust to changes in the interarrival time variance.

6. Conclusion

This thesis investigated queueing models of air traffic delays. We concluded that the most important area of the air traffic system with respect to delays is the landing service which airports provide to planes, including usage of the runways, terminal airspace, and controller resources. We also concluded that a queueing model with time varying Poisson arrivals and Erlangian service time (with Poisson and deterministic service time as special cases) was the most useful for investigating the effects of various aspects of the system on delays.

The general Erlang service model was chosen for four reasons. First, it is mathematically tractable, and therefore it provides an attractive alternative to simulation. More complex models, such as network queueing models, are not normally tractable and make analysis susceptible to the vagaries and enormous costs of simulation. Second, the assumption of time varying Poisson arrivals is a good one, especially if demand is heavy and composed of some amount of unscheduled aircraft. Third, the Erlangian server allows us to vary the service time mean and variance. This allows us to check the effect of changes in the air traffic system that affect both mean landing time and landing time variance. Fourth, since the model has time varying inputs it has a transient solution. The transient solution is far more valuable in investigating the behavior of delays, since terminal servers are rarely in steady state.

Various implementations of this model were investigated. The standard Fluid Flow, Steady State, and Difference Equation models were analyzed for their computation costs and implemented. Then an interpolated model that produced transient solutions to any order Erlang queue by interpolating between the Poisson and deterministic solutions was developed, implemented and tested. The model was found to perform well, approximating the steady state exactly and lagging the transient response only slightly during steep changes. The interpolation formula is shown below, where W(t) is the average delay at time T.

$$W(t)_{M/E_r/1} = \left(\frac{1}{r}\right)W(t)_{M/M/1} + \left(\frac{r-1}{r}\right)W(t)_{M/D/1}$$

A second approximation model, the Kivestu model, was investigated and implemented and found to be even less expensive computationally and more accurate than the above model. It requires only the computation of the transient response to the M/D/k queue.

In the last section of this thesis the above models, especially the Kivestu model, were used in an analysis of the delay situation at Logan airport. A number of points were noted. First, transient and steady state delays increase polynomially as the service rate is lowered. Second, the delays generated by systems with low server variance can often be lower than those generated by systems with higher service rates and higher variance. This implies that variance of service times is an important area for improvement in the air traffic system. Third, when the systems are heavily saturated, the difference between the delay induced by a Poisson system moves closer to that produced by a deterministic system.

Finally, some simulation tests of the accuracy of the Poisson arrivals process were executed for the scheduled arrival profile at Logan. The Poisson was found to be fairly accurate, with some reservations. The simulation distribution of interarrival times was found to have greater variance than would be predicted by the Poisson, but this could be attributed to clumping over the day in the schedule. The variance of interarrival times was found to be remarkably insensitive to the variance of the lateness distribution used in the simulation, implying that whatever the process generating arrivals is, it is robust to changes in the lateness variance. The distribution of interarrival times resembled a low order Erlang more than than exponential, which counts against the Poisson assumption. The variance of the Poisson over a whole day was investigated, and found to be reasonable if one assumes some small percentage of unscheduled aircraft are included in the demand profile. It was concluded that time varying Poisson arrivals is a good assumption given heavy utilization and some amount of unscheduled aircraft.

Bibliography

[ANDR 89]
Andrews, John W. & Welch, Jerry D. "The Challenge of Terminal Air Traffic Control Automation" 34th Annual Air Traffic Control Association Conference Proceedings, Fall 1989.

[ANDR 88]
Research Plan for Terminal ATC Automation (TATCA)

[ASHF 79]
Ashford, N. & Wright. Airport Engineering. 1979.

[BERT 89]
Bertsimas, Dimitris J., & Nakazato, Daisuke. Transient and Busy Period Analysis of the GI/G/1 Queue: Part 1, The Method of Stages. Part II, Solution as a Hilbert Problem. 1989.

[BLUM 76]
Blumer, T., Simpson, R., & Wiley, J. A Computer Simulation of Tampa International Airport's Landside Terminal and Shuttles. FTL Report R-76-5.

[BROW 76]
Brown, T.H. A Comparison of Runway Capacity and Delay Using Computer Simulation and Analytic Models. M.S. Thesis C.E. 9/76.

[CONO 75]
Conolly, Brian. Lecture Notes on Queueing Systems. John Wikey & Sons, 1975.

[HENG 75]
Hengsbach, Gerd & Odoni, Amedeo R. Time Dependent Estimates of Delays and Delay Costs at Major Airports. FTL Report R75-4, 1/1975.

[HENG 74]
Hengsbach, G. Computer Estimates of Delays and Delay Costs at Conjested Airports. FTL MS Thesis 1/74.

[HORO 83]
Horonjeff, R. Airport Planning and Design. 1979.

[KIVE 76]
Kivestu, Peeter. Alternative Methods of Investigating the Time Dependent M/G/k Queue. M.S. Thesis Aero 1976.

[KOOP 72]
Koopman, Bernard O. "Air Terminal Queues under Time Dependent Conditions.", Operations Research 20, 1089-1114 (1972).

[KLIE 76]
Klienrock, L. Queueing Systems V1. 1975.

[LARS 81]
Odoni & Larson. Urban Operations Research. 1981.

[MOOR 89]
Moore, Margaret L., Description of ATC Operations and facilities at Boston TRACON and Logan Tower. Lincoln Lab ATC Project Memorandum No. 42PM-TATCA-0004, 1989.

[MOOR 89b]
Moore, M. L. & Crone, C. W. Modes of Operation for Runway Configuration 4R&L/9 At Logan International Airport. Lincoln Labs Memo (Draft).

[NORD 78]
Nordin, J.P. Principles of a Flexible Simulation Model of Airport Airside Operations. M.S. Thesis C.E. 8/78.

[OAG 89]
Official Airline Guide. June 1989.

[ODON 69]
Odoni, A.R. An Analytical Investigation of Air Traffic in the Vicinity of Terminal Areas, ORC Technical Report #46, 12/69.

[ODON 71]
Odoni, A.R. Modelling for Air Traffic Control Systems. FTL Memo M71-4.

[ODON 75]
Odoni, A.R., & Kivestu, P. A Handbook for the Estimation of Airside Delays at Major Airports (Quick Approximation Method). FTL Report 75-10, 6/76.

[ODON 76]
Odoni, Simpson, Estimates of Capacity and Delay For Proposed Runway Systems: Schiphol Airport, Amsterdam. FTL Report R76-12 12/76.

[ODON 83]
Odoni, Amedeo R. & Roth, Emily. "An Empirical Investigation of the Transient Behavior of Stationary Queueing Systems", Operations Research, May, 1983.

[OLIV 64]
Oliver, Robert M. "Delays in Terminal Air Traffic Control", Journal of Aircraft, V1 #3 1964.

[ROTH 79]
Roth, Emily. An Advanced Time-Dependent Queueing Model For Airport Delay Analysis. FTL Report FTL-R-79-9, 10/1979.

[SCAL 76]
Scalea, J.C. A Comparison of Several Methods for the Calculation of Airside Airport Delay. M.S. Thesis C.E. 6/76.

[SIMP 88]
Simpson, Robert W.. The Merging Process for Metering ATC Streams. FTL Memorandum M88-6, 11/88.

[SIMP 89]
Simpson, R. W. "The Operation of Holding Stacks for Terminal Area ATC", FTL TATCA Working Paper #3.