AN ANALYSIS OF RUNWAY CAPACITY

by

Charles F. Booth

Thesis submitted to the Graduate Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE
in
Industrial Engineering and Operations Research

APPROVED:

J. W. Schmidt, Chairman

G. K. Bennett

J. W. Eyster

July, 1972

Blacksburg, Virginia

ACKNOWLEDGMENTS

The author wishes to express his sincere appreciation to all those who have contributed to the completion of this thesis.

Dr. J. W. Schmidt, the author's major advisor, has been particularly helpful during the preparation of this thesis. He has spent many hours counseling and guiding the author and his help has been greatly appreciated.

Dr. G. K. Bennett and Dr. J. W. Eyster have also contributed their knowledge and encouragement to the author during the preparation of the thesis.

The author is deeply indebted to the National Aeronautics and Space Administration, without whose financial support under Grant Number NGR-47-004-090, this past year of graduate study would not have been possible. was particularly helpful to the author during his stay at Langley Research Center in July and August, 1971.

TABLE OF CONTENTS

Chapter Page

LIST OF FIGURES

Figure Page

Figure Page

LIST OF TABLES

DEFINITIONS OF FREQUENTLY USED TERMS

ATC:	Air traffic control.
Capacity:	The total number of operations of a runway per time (usually per hour).
Departure Control:	A function of approach control providing service for departing IFR aircraft and on occasion, VFR aircraft.
Decision Point:	That point located on the approach path at decision height. The point at which a decision must be made to either continue the approach or execute a missed approach.
FAR'S:	Federal aviation regulations.
Final Approach-IFR:	The flight path of an aircraft which is in bound to the airport on an approved final instrument approach course, beginning at the final approach fix and extending to the runway.
Final Approach Fix:	The fix over which final approach (IFR) to an airport is executed.
Final Controller:	That controller providing final approach guidance utilizing radar equipment.
Flight Plan:	Specified information relating to the intended flight of an aircraft that is filed orally or in writing with an ATC facility.
IFR:	Instrument flight rules.
ILS:	Instrument landing system.

Knot:	One nautical mile per hour or 1.151 statute miles per hour.
Operation:	Either a takeoff or landing.
Outer Marker:	A radio marker beacon indicating where an aircraft at the appropriate altitude on the localizer course will intercept the ILS glide path.
Threshold:	The end of the runway.
VFR:	Visual flight rules.

CHAPTER I: INTRODUCTION

Certainly no industry has experienced such dramatic growth during the last thirty years as has the aviation industry. It has passed swiftly from the era of the famous Douglas DC-3 to the load toting Boeing 747. In fact, the MACH 2 Anglo-French Concorde SST is a good indication of what the plane of the future will be. Just how much progress has been made is easily seen by comparing performance characteristics for the planes of the past, present, and future. The Douglas DC-3 carried 28 passengers 1,400 miles with a cruise velocity of about 165 miles per hour. The Boeing 747 carries 360 passengers 3,800 miles averaging about 585 miles per hour. The Concorde SST carries 144 passengers 4,000 miles at speeds up to 1,450 miles per hour.

Not only are aircraft performance characteristics soaring but the number of aircraft in use is increasing equally as fast. The commercial fleet increases by about thirty aircraft per month while the registry of private aircraft increases by about four hundred fifty aircraft per month. Aviation forecasts indicate that the total number of operations will double in approximately ten-year intervals. However, the number of aircraft requesting control service is expected to grow even faster with three times as many requests in 1990 than at present. If the challenge of projected increased demand on an already taxed ATC system is to be met, our available alternatives must be evaluated immediately. It would seem that there are two choices:

(1) Build new and more adequate air hubs in conjunction with the installation of new generation ATC equipment. (2) Analyze our present ATC system to determine what areas appear promising with respect to increasing the capacity of our airways.

Upon examining (1) above it is found that rising costs of land acquisition and building make expansion of existing facilities and construction of new facilities a far from optimal choice. Along with construction of new airports comes the problems of airspace interference in the areas of major hubs and community resistance due to anticipation of noise, property devaluation, etc.

It is believed that (2) above offers the most promise for increased capacity of present and future airways. In particular, there is great room for improvement in the airport and its associated airspace; for this reason an analysis of runway capacity constraints was conducted. It is felt that this study will result in a better understanding of the interrelationships between system parameters for the purpose of increasing throughput rate in the ATC system.

The ATC System

The ATC system consists of a vast network of radar covering the entire USA. A pilot enters into this system by filing a flight plan prior to departure. The flight plan may be submitted to the nearest flight service station or the airport control tower. This document contains all information pertinent to the flight: destination, arrival time, altitude, velocity, radio call sign, aircraft instrumentation, etc. By filing a flight plan a pilot is notifying ATC of his intentions and allowing the controllers to plan accordingly (ATC may modify a pilot's request). When the pilot is ready to depart he calls control tower personnel for runway selection, weather information, and final takeoff clearance. After takeoff, a second group of controllers (departure controllers) in the near terminal area assume radar observation of the aircraft and inform the pilot of departure procedure that will deliver him safely to the edge of the terminal's positive control area, at which time he will be "handed off" to a regional control center. Each regional control center will have been advised by computer of the aircraft's flight plan and will thus be expecting the aircraft. The aircraft will be "handed off" from each region to the next, all the while being assured of continual radar surveilance and notification of approaching traffic. Thus, on a cross country flight an aircraft would pass through several regions with the final "hand off" being made to the approach controller in the near terminal area at the destination. Assuming saturated conditions the approach controller then directs the inbound aircraft to a holding "stack" to await its turn to land.

When the approach controller notifies the inbound aircraft to exit the "stack" the pilot will begin his approach course in order to intercept the glide slope. Once in the terminal area an aircraft must observe minimum intrail separation rules (the three mile rule). If the pilot is able to stabilize his aircraft in landing configuration by the time minimum decision altitude is reached a normal landing will ensue: the aircraft will touchdown, decelerate, and exit the active runway as soon as possible.

Method of Approach

To reiterate, it was felt that in reviewing the Air Traffic Control system, the terminal area is the component whose increased efficiency would have the most immediate impact on total ATC system capacity. For this reason the focus of this study was the airport runway and its approach and departure path airspace.

Fundamental to addressing the problem of runway capacity is the identification of all independent variables, their interrelationships, and their relations to capacity. Two solution approaches were considered: an analytic model and a simulation model. Of the two approaches, simulation was chosen after careful consideration of the goals involved in the research. One goal was to model the system under study in such a way as to enable further research using the model developed. Although a mathematical model might have been developed for a simplified system, it is unlikely that the full effect of the random variables in the system could have been captured through such a model. In addition, a mathematical model is not as easily extended and adapted to further study as is a simulation model. In other words, a simulation model is not as application dependent as is a mathematical model. With little effort the basic simulator envisioned could be extended to analyze sequencing techniques, delay times, and total airport simulation including taxiways, passenger and baggage loading and unloading.

To illustrate how simulation can be used in runway capacity analysis, experiments were designed in which multiple linear regression equations were fitted to simulated results for typical aircraft mixes. These regression equations were used to show the relationships between capacity and independent variables of the system for the different mixes considered. A complete simulation validation was performed and each experiment was statistically analyzed.

CHAPTER II: REVIEW OF THE LITERATURE

In approaching the problem of runway capacity it was deemed of utmost importance to first fully understand the total Air Traffic Control environment. With this in mind a review of selected ATC literature of a general nature is presented, followed by a review of studies addressing topics of particular importance to this study.

The General ATC System

Perhaps the best source of general information concerning all phases of aircraft operation in the United States is the *Airman's Information Manual*, Part I (8). For the reader's benefit an outline of topics covered is included in Appendix A.

More detailed information on terminal ATC procedures and constraints may be found in *Terminal Air Traffic Control* (10). This manual is one to which reference is made in Part 65 of the Federal Aviation regulations and with which controllers are required to be familiar. A publication describing standardized methods for use in designing instrument flight procedures is *United States Standard for Terminal Instrument Procedures* (11).

It was desired to include all types of aircraft in this study, yet it was felt that certain aircraft have so many common characteristics that they could, for all practical purposes, be considered of the same type. For this reason all aircraft were considered classifiable into five categories as given in the *Airport Capacity Handbook* (3):

Class 1 Aircraft: All single-engine aircraft with the exception of the Mustang (3), Bonanza (2), and Debonair (2), and small STOL aircraft some of which may have two engines (Twin Otter).

The most common types of class 1 aircraft are

Cessna series 150 through 210

Mooney 20 series

Piper series, Tri-pacer, Colt, Comanche, and Cherokee

Other miscellaneous types are Beech Musketeer, D. H. Beaver, Bellanca, Helio Courier, Luscombe, Navion, and the Stinson.

Class 2 Aircraft: All light twin engine piston and turbo prop aircraft having a normal loaded weight less than 8000 pounds, and some high-performance single-engine light aircraft (such as The Beech Bonanza). They are normal, small, light, twin-engine aircraft with the exception of those marked with an asterisk (*).

Aero Commander (500, 600, 700 series, Grand and Turbo)

Beech Bonanza*

Debonair*

Beech Baron

Travel Air
Queen Air
Twin Bonanza
Cessna 310, 320, 411, 336/337
Piper Apache
Aztec
Twin Commanche

Class 3 Aircraft:

1. Piston and turbo prop aircraft having a normal loaded weight greater than 8,000 pounds and less than 36,000 pounds.
2. Jet aircraft having a normal loaded weight greater than 25,000 pounds.

Aero Commander Jet Commander
Beech 18
Beech King Air
Dassult Fan Jet Falcon
Douglas DC-3
Grumman Gulfstream I, II
Lear Jet 23, 24
North American Saberliner

Class 4 Aircraft:

1. Piston and turbo prop aircraft having a normal weight

in excess of 36,000 pounds.

2. Jet aircraft not included in class 5 but having a normal loaded weight in excess of 25,000 pounds.

BAC 111

Boeing 727, 737

Convair 240/340/440/580/600

Douglas DC-4

DC-6

DC-7

DC-9

Lockheed Constellation

Martin 404

Class 5 Aircraft: All jet aircraft normally requiring runway lengths exceeding 6,000 feet (corrected to sea level) for takeoff and/or landing.

BAC (Vickers) VC 10

Boeing 707

720

747

Convair 880/990

Douglas DC-8

DC-10

SUD Caravelle

Lockheed 1011

Runway Capacity

In 1969 the Bureau of Standards conducted a brief study (12) concerning a "maximum throughput rate" concept for the capacity of a runway and its final approach path airspace. This concept is new in the sense that it does not assume an underlying arrival distribution (such as Poisson), as have other capacity studies. The approach taken permits one to evaluate existing and proposed single runway (arrival only) configurations in terms of arrivals per unit time. The capacity concept described is shown to be representable by a simple mathematical equation. An expected value model is derived, with only the aircraft population mix, facility tieup times, and error distribution being required.

Knox, in conjunction with NASA is presently conducting a study entitled "An Investigation of the Air Traffic Control Procedures and Aircraft Performance Characteristics Around the Runway Environment" (15). The purpose of this investigation is to identify the pertinent aircraft performance parameters and ATC constraints around the runway environment that could cause major runway congestion. The sensitivity of each aircraft performance parameter and the ATC constraint that affect runway capacity rate will be determined by fixing all variables with the exception of one. This constraint will be varied and the corresponding change in runway capacity observed.

E. N. Hooten, et al (14), have developed extensive mathematical models for practical runway capacity. These models furnish the basis for the Airport Capacity Handbook (3) and are applicable in the following environments: (1) single runways and runway/taxiway crossings, (2) intersecting runways in VFR, (3) dual arrival feed in VFR to multiple runways, (4) IFR operations for all runway configurations. It should be emphasized that the models are intended to arrive at practical runway capacity and have been based on a preset minimum acceptable delay level. Also discussed by Hooten, et al, is the establishment of arrival demand as a Poisson Process.

Baran, Benezra, and Blumenthal (6) of the Boeing Company have developed several analytical runway capacity models; however, they are not dependent upon a "minimum acceptable delay level".

An attempt to gain the proper perspective relative to the functional characteristics and interrelationships of all factors involved in simulating an airport environment was made. Invaluable in this attempt was "Factors Affecting Airport Capacity and their Applicability to Simulation" (5).

Data Input to the Simulator

Much literature is available on characteristic data required as input to the simulator. Runway Characteristics and Performance of Selected Propeller-Driven Aircraft in Routine Operation (1) and Runway Characteristics and Performance of Jet Transports in Routine Operation, Volume I (2) are appropriate references. Possibly the best and most easily accessible reference for aircraft performance data is "Aviation Week and Space Technology" magazine. Periodically, complete performance tables are published for virtually every major type of aircraft.

An idea of the type of data required concerning the aircraft population, or mix, may be obtained from the Airport Capacity Handbook (3) Chapter 19. However, the best source of information is the airport to be studied, except in the case of a proposed airport in which planners must furnish the required information.

For the example analysis included in Chapter V the range considered for aircraft separation was 1-4 nautical miles. This was considered essential in view of preliminary findings by Charles Knox (15) indicating that intrail separation must be reduced to below 2 N. miles before the exit speed, deceleration rate, and touchdown point parameters are likely to influence the runway capacity.

Simulation Techniques

Simulation and Analysis of Industrial Systems (22) was the source of all process generators and of the random number generator used. The reader should be cognizant of the fact that the random number generator employed is not valid on some machines.

Regression Analysis

Once the simulation model is developed and validated, its output will be analyzed. In particular, a relationship will be sought which describes a certain dependent variable, capacity, as a function of the independent variables under study. When a suitable relationship is found, it will be used to predict the behavior of the dependent variable as the independent variables change. Criteria for the functional relationship sought will require not only that it fit the data, but that it explain a significant amount of variation in the dependent variable as well. To satisfy these criteria, regression and correlation techniques (19) were used.

CHAPTER III: SYSTEM DESCRIPTION

This study was focused on a single independent runway, with associated approach and departure path airspace, serving both arriving and departing aircraft. With regard to this runway an operation was defined as either a takeoff or a landing and capacity as the number of operations per hour. An arriving aircraft was dealt with from the time it entered the holding stack (located over the outer marker for modeling purposes) until it exited the active runway. A departing aircraft was considered from entry onto the active runway until the point in time that ATC regulations permitted a subsequent arrival or departure. The following ATC constraints, aircraft performance parameters, and assumptions were used in the system analysis:

ATC Constraints:

1. Minimum aircraft intrail separation on the landing approach.
2. Minimum aircraft entrail separation on the departure path.
3. Only one aircraft on the active runway at a time.
4. Length of final approach path.

Aircraft Performance Parameters:

1. Velocity at entry to active runway for a departing aircraft.

2. Runway acceleration rate for a departing aircraft.
3. Lift off velocity for a departing aircraft.
4. Climb acceleration rate for a departing aircraft.
5. Final approach and touchdown velocity for an arriving aircraft.
6. Touchdown point on the runway for an arriving aircraft.
7. Runway exit velocity for an arriving aircraft.
8. Runway deceleration rate for an arriving aircraft.

Assumptions:

1. Runway acceleration rate is constant for a departing aircraft.
2. Climb acceleration is constant for a departing aircraft.
3. Runway acceleration rate and climb acceleration rate are considered equal over the regime for which a departing aircraft is considered.
4. Instrument flight rules apply to the entire system.
5. Operations are in a radar environment.
6. Approach velocity equals touchdown velocity for an arriving aircraft.
7. Runway deceleration rate is constant for an arriving aircraft.
8. Departing aircraft follow divergent paths.

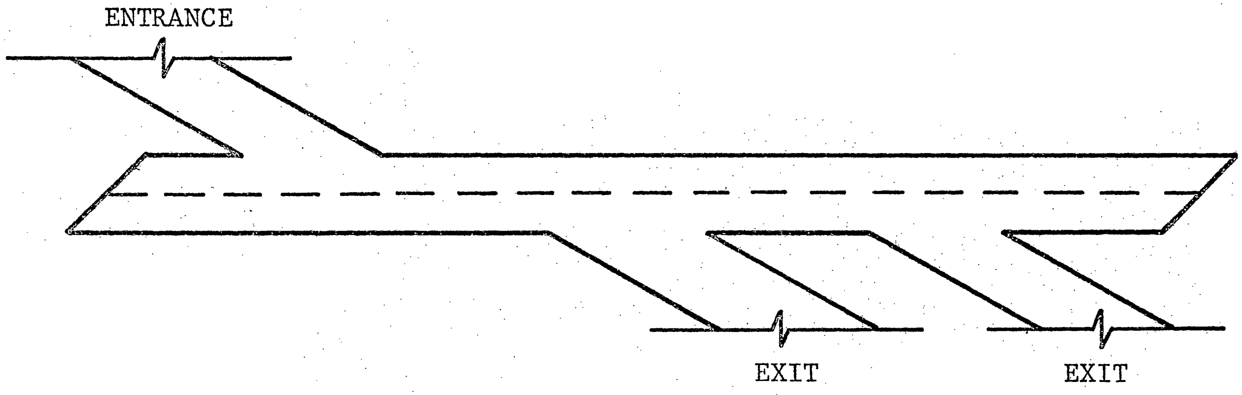


FIGURE 1: SINGLE RUNWAY CONFIGURATION

Characterizing the ATC minimum intrail separation for approaching aircraft is the three mile rule: within 40 nautical miles of the radar antenna site all aircraft operating under a radar environment at the same altitude must maintain a longitudinal separation of three nautical miles.

Departing aircraft were considered to follow divergent paths so that a departure was allowed when either of two conditions was satisfied (11):

A. The previous aircraft was airborne and the following minimum distance criteria was met:
   (1) When only class 1 aircraft were involved--3,000 feet.
   (2) When a class 1 aircraft was preceded by a class 2 aircraft--3,000 feet.
   (3) When either the succeeding or both were class 2 aircraft--4,500 feet.
   (4) When either was a class 3, 4, or 5 aircraft--6,000 feet.

B. A preceding, arriving aircraft had taxied off the active runway.

Deceleration rates investigated fell between 1.8 and 9 FT./sec./sec. depending on factors such as touchdown point and location of runway exits. This variable was relatively constant over different aircraft classes due mainly to passenger comfort considerations. Touchdown points were relatively consistent for class 3, 4, and 5 aircraft due to the fact that these aircraft generally employ visual or electronic approach slope aids and are flown by professional pilots. Touchdown points for class 1 and class 2 aircraft varied greatly reflecting differing pilot proficiencies and techniques, runway exit locations, and meteorological conditions.

The Stochastic Nature of the System

As previously stated, the reason for approaching the problem at hand through simulation was the stochastic nature of the system. To focus attention on the random variables present, the progress of both an arriving and a departing aircraft will be traced from system entry to exit.

Approaching Aircraft

The system under study begins at the outer marker, over which, for programming convenience only, is situated the holding stack. To enter an aircraft into the system we must first know the interarrival time and the assigned velocity of the aircraft. These two quantities can never be predicted with certainty and are thus considered to be random variables. If mean arrival rates for aircraft classes were examined, it would be found that the arrival rates vary according to time of day. Therefore, the random variable interarrival time (the reciprocal of arrival rate) is time dependent. As the aircraft waits in the holding stack (assuming saturated conditions) its turn eventually comes to enter onto the final approach path. Prior to the release of the aircraft an approach controller estimates, by a specified rule, a release time that will ensure a safe approach and landing for the aircraft. The point is that the controller will never use exactly the same rule (over some range) twice; thus within a certain range, the actual separation between aircraft is a random variable.

Once on final approach the pilot will attempt to hold a predetermined approach velocity. However, due to factors such as pilot error, faulty instrumentation, equipment failure, and weather conditions this is, in general, not possible. This indicates another stochastic phenomena to be accounted for--that of slightly varying velocity on the final approach path. It is important that the pilot have the aircraft stabilized and in landing configuration by the time decision height is reached, for if he does not he must execute a missed approach.

It now seems intuitively obvious to state that within certain ranges touchdown point, deceleration rate, and runway exit velocity are all random variables.

Departing Aircraft

As with arriving aircraft the stochastic nature of system arrival time for departures must be accounted for. The simulation model developed was intended to be as general as possible. Therefore, the runway entrance velocity was treated as a random variable, characterizing an aircraft beginning its takeoff roll before actually entering the active runway. If takeoffs are desired to begin from a full stop a mean of zero and standard deviation of zero are used for generating entrance velocity. Again it is intuitively justifiable to treat runway acceleration rate and lift off velocity as random variables.

CHAPTER IV: MODEL DESCRIPTION

The main goal in constructing the simulator was complete versatility in describing a system of the type discussed in the previous chapter. That is, it was required that the model be independent of the experiment at hand. This goal was achieved by using the FORTRAN language and modularized construction techniques. The program may easily be modified or extended by changing only specific subroutines. Landings only, take offs only, or landings and take offs combined may be simulated for any length of time by reading in appropriate input data.

The simulator was constructed by the author and run on the IBM 370/155 computer. The region used was 118k and execution time may be estimated at 4 seconds per hour of simulated time. The general operational concepts of the simulator will be presented in this chapter while specific details of each subprogram may be found in Appendix E.

Model Concepts

Real and Phantom Operations

Relative to the system under study two types of operations were identified--arrivals and departures. The concept of a phantom aircraft was introduced to enable a departing aircraft to be sequenced, scheduled, and operated on by the same program facilities as used by landing aircraft. The system "sees" a phantom aircraft as an ordinary aircraft moving along the approach path towards the runway. In actuality the phantom aircraft simply represents a time slot reserved for a prescheduled take off. Thus a landing is a real operation and a departure is a phantom operation.

Storage Facilities

Three storage arrays were used for retaining all information on aircraft present in the system: WAIT, STACK, and APROCH. Matrix manipulation routines were included to allow "push up stack" operation within the arrays, that is, the matrices are allocated a fixed amount of core storage which is used over and over--by incorporating this technique considerable savings in computer storage space was realized.

The array "WAIT" (Table I) consists of all possible next event types (takeoffs and landings) and associated descriptive information. It is from this array that the next event is selected, by choosing that event which occurs next temporally. As stated in Chapter II this study considers five different aircraft categories. Therefore, WAIT will consist of ten rows: a next take off and next landing for each of the five classes. As next events are selected and deleted from WAIT they are immediately replaced with an identical event, thus always assuring a complete population of next events from which to draw aircraft.

The array "STACK" (Table II) represents a holding stack consisting of all aircraft (phantom and real) that have entered into the system but are awaiting sequencing onto the final approach path. When a new system arrival is selected from WAIT it is given a number. The air-

TABLE I: TYPICAL WAIT ARRAY

A/C Class	Type Operation	Assigned Velocity (kts)	System Arrival Time (seconds)
1	landing	60	54249
2	landing	75	54345
3	landing	95	54767
4	landing	105	55292
5	landing	130	55535
1	takeoff	--	55575
2	takeoff	--	54434
3	takeoff	--	54107
4	takeoff	--	54634
5	takeoff	--	54244

TABLE II: TYPICAL STACK ARRAY

Aircraft Number	Aircraft Class	Type Operation	Assigned Velocity (kts)	System Arrival Time (seconds)
9	2	Landing	69	54628
10	4	Take off	--	54634
11	3	Landing	93	54767

TABLE III: TYPICAL APROCH ARRAY

Aircraft Number	Aircraft Class	Type Operation	Assigned Velocity (kts)	Last Velocity	Last Node	Time at Last Node (seconds)
13	2	Takeoff	96	96	21	54485
14	1	Landing	62	60	8	54484
15	4	Takeoff	62	62	5	54485

craft and its parameters are then transferred from WAIT to STACK to await sequencing. The array "APROCH" (Table III) consists of all aircraft and their parameters either on final approach or on the runway.

Generating System Arrivals

Either from historical data or forecasts of future operations some indication of expected hourly facility demand by an aircraft class (either take offs or landings) is available to the analyst. This information furnishes a time dependent mean arrival rate for any given system time, which in turn will provide the parameter necessary to generate exponential interarrival times, characterizing the Poisson Arrival Process (a normal interarrival time generator is also included).

Let us define an approach or departure class profile as a graph of arrivals or departures per hour vs. time (Figure 2). As a system input, an approach and departure profile is required for each aircraft class. This profile provides the means by which the aircraft population to be simulated is characterized. In analyzing the profile it would be expected that operations would peak during the morning and evening rushes and to diminish somewhat in between.

Define:

t = Time between successive arrivals or departures (exponential)

$\alpha(t)$ = Mean arrival or departure rate at time t

$f[t|\alpha(t)]$ = Conditional density function of t given $\alpha(t)$

An approach or departure profile can be approximated as follows

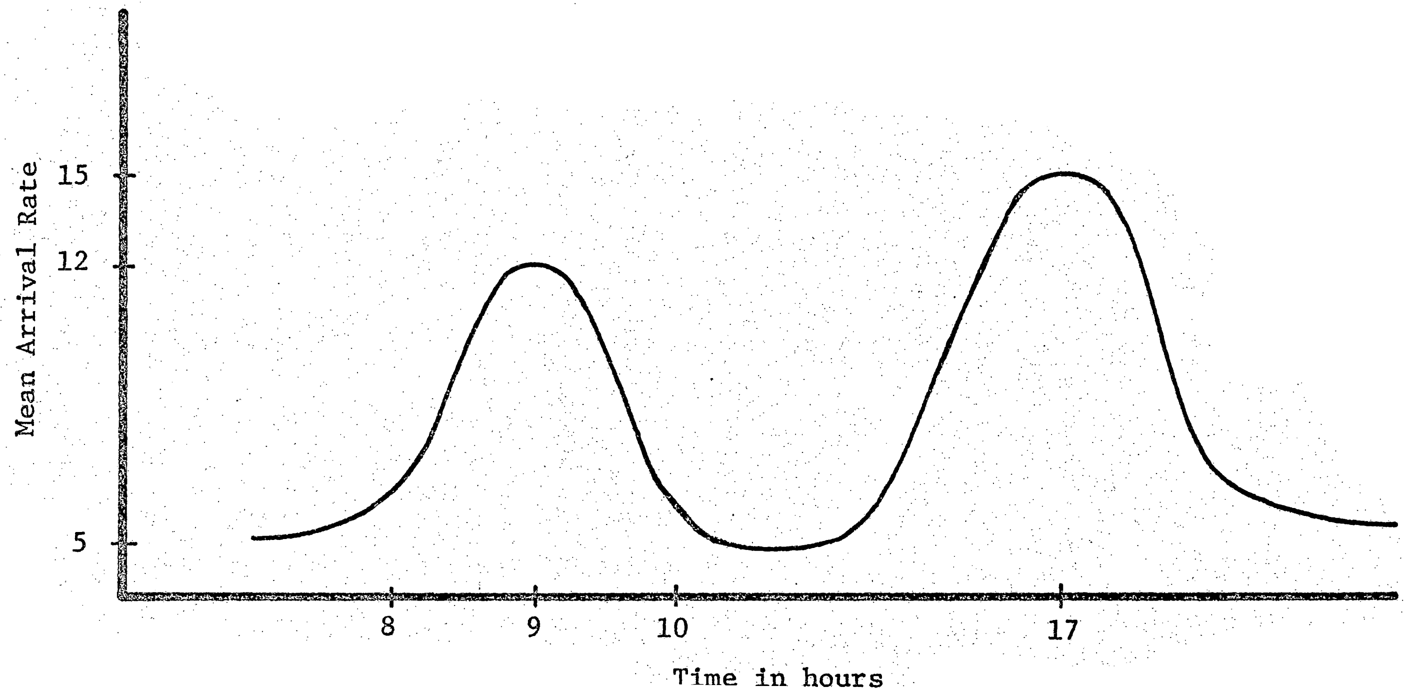


FIGURE 2: TYPICAL APPROACH CLASS PROFILE ILLUSTRATING THE TIME DEPENDENCE OF MEAN ARRIVAL RATE

$$\alpha(t) = B_1 \frac{1}{\gamma_1\sqrt{2\pi}} e^{-\frac{(T-\delta_1)^2}{2\gamma_1^2}} + B_2 \frac{1}{\gamma_2\sqrt{2\pi}} e^{-\frac{(T-\delta_2)^2}{2\gamma_2^2}} + \rho \tag{4.1}$$

where

T = Master clock time

Subscript 1 denotes operations from 0-12 hours

Subscript 2 denotes operations from 12-24 hours

δ_1 = the morning time of peak operations

δ_2 = the afternoon time of peak operations

ρ = the minimum daily class operational level

γ_1 = the standard deviation of morning operations times

γ_2 = the standard deviation of afternoon operations times

B_1 = a scale parameter for morning operations

B_2 = a scale parameter for afternoon operations

Using the information given in Figure ~~5~~ 2? a morning, time dependent, mean arrival rate may be solved for.

It is known that 99% of the area under a normal curve lies within ± 3 standard deviations away from the mean. Therefore

$$6\gamma_1 = 2 \text{ hours}$$

$$\gamma_1 = 1/3 \text{ hour}$$

Knowing that $\rho = 5$, β can be solved for by entering $T = 9$ into the equation for $\alpha(t)$ (using only that part of $\alpha(t)$ representing morning operations).

$$12 = \beta \frac{1}{1/3\sqrt{2\pi}} e^{-\frac{(9-9)^2}{2\gamma_1^2}} + 5$$

$$\beta = 7/3\sqrt{2\pi} \qquad 4.3$$

Having solved for all parameters a time dependent mean arrival rate may be obtained by substituting the system master clock time for T and finding $\alpha(T)$.

For an exponential interarrival time, t is generated at time T by

$$t = \frac{-1}{\alpha(T)} \ln(r) \qquad 4.4$$

where r is a uniformly distributed random number between zero and one.

To begin a simulation the initial master clock time is used to generate the first system arrival time for each of ten possible first events. The minimum time event is then selected and entered directly into the APROCH array.

The System as Seen by the Simulator

Within the simulator, the runway and its associated airspace are represented by a series of discrete nodes (Figures 3 and 4). When an aircraft enters the system it starts at node one and progresses node by node until it lands or takes off. One aircraft is always designated the "present event aircraft" or AC. To select AC the time of arrival at the next node is calculated for all aircraft in the APROCH array. The aircraft with the minimum time then becomes AC and is advanced to the

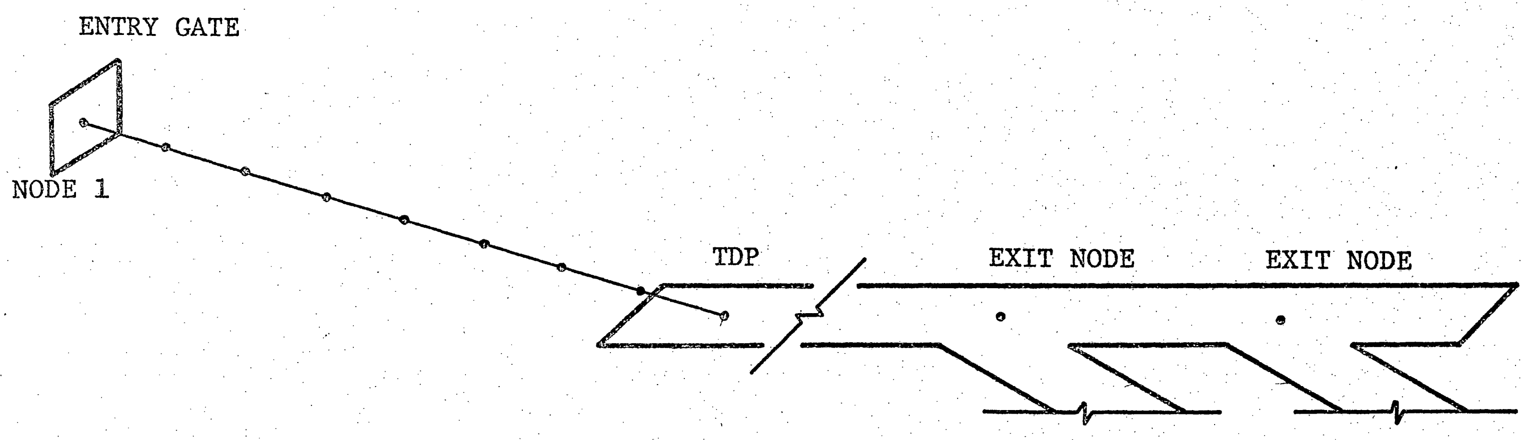


FIGURE 3: APPROACH PATH AND RUNWAY AS SEEN BY ARRIVALS

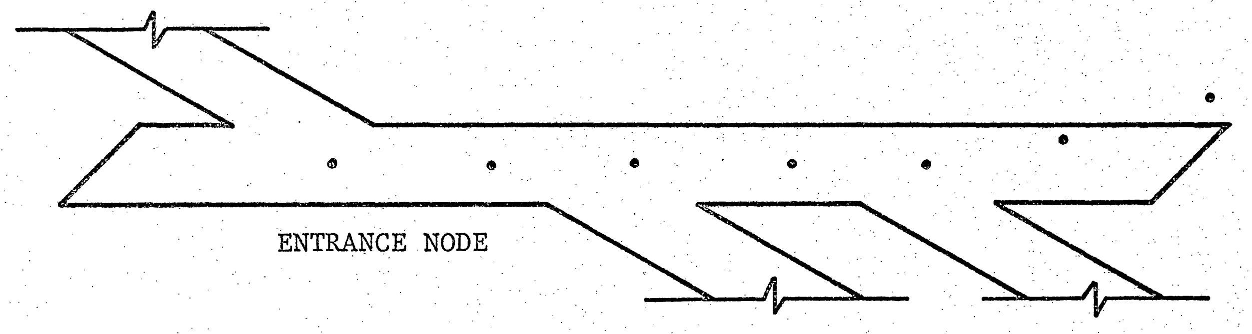


FIGURE 4: RUNWAY AND DEPARTURE PATH AS SEEN BY DEPARTURES

next node. Once AC has been advanced, system statistics are collected, the system updated, and the cycle repeated until the end of the simulation.

All calculations of time between nodes are performed in subroutine TIMNOD using the following relationships:

$$V = V_o + AT \tag{4.5}$$

$$S = S_o + V_o T + 1/2\ AT^2 \tag{4.6}$$

$$A(S-S_o) = \frac{V^2 - V_o^2}{2} \tag{4.7}$$

where

V = Final velocity

V_o = Initial velocity

A = Acceleration (Deceleration)

T = Time

S = Final position

S_o = Initial position

System Time Keeping

System time is kept by means of a master clock aircraft denoted MCAC. The MCAC is always the closest real aircraft to the end of the runway or, in the event that all aircraft in the system are phantom (take offs), the phantom aircraft nearest to exiting the system is considered to be the MCAC aircraft. To begin system time keeping the initial time of the simulation is read in and a master clock aircraft designated--then as the MCAC moves from node to node the system maste clock time is updated by the internode travel time. When a MCAC exits the system a new MCAC is immediately designated, thus insuring that no system time lag occurs.

Introducing New Aircraft to the System

To introduce a new aircraft into the system it is always necessary to know the type aircraft (and its parameters) immediately preceding the releasing aircraft (RAC). For this reason the aircraft closest to the entry gate (see Figure 3) is always designated the release point aircraft (RPAC). Thus instead of searching the APROCH array for the aircraft on which the release decision is to be made, information can be summoned about the RPAC. This notation not only saves program execution time in this manner but also by allowing the system to be reviewed for new arrivals only when the RPAC is the present event aircraft. This technique is valid because of the discrete node concept of the model. That is, as far as the program is concerned the aircraft move one by one, from node to node, and only when the RPAC moves can the release of a new aircraft be considered.

Each time the RPAC advances a node there is an internode time during which a new system arrival may have occurred or an aircraft may have been released from the holding stack to the final approach path. Subroutine REVUE is the facility by which the possibility of these occurances is considered. In REVUE all system arrival times contained in the WAIT array are compared to the time (TIME) the RPAC arrived at its present node. If the system arrival time of any aircraft is found to be less than TIME then the aircraft and its parameters are transferred to the STACK array and the next system arrival time for this aircraft class is generated and entered into WAIT. In this manner WAIT is revued until no aircraft can be transferred to STACK. At this point in time it remains to be determined which aircraft in STACK is to be released next and what its release time should be. This information is determined in subroutine SEQ.

Sequencing Aircraft onto Final Approach

For the purpose of this study all that was desired was a random sequence of aircraft onto the runway. For this reason the priority rule for service was first come-first served. However, the reader should note that any priority rule may be used by changing subroutine SEQ. Then, to determine which of the aircraft present in STACK is to be eligible for next release the aircraft with minimum system arrival time is searched for. This aircraft is designated RAC (releasing aircraft). With Federal Aviation Regulation separation criteria in mind it remains to be determined at what point in time the RAC can be released (RELTIM) so as to insure this minimum separation. When RELTIM is determined it can be compared to TIME; if RELTIM is less than or equal to TIME then the RAC and its parameters are transferred to the APROCH array, as the RAC has started its final approach.

Determining Time Separation (SEP) Between RAC and RPAC

For the purpose of scheduling runway use, phantom aircraft were treated as though they were real. Therefore, if a phantom aircraft is to be released following either a real or another phantom aircraft all that is necessary is to insure that there is a time opening available during which the aircraft can safely execute a takeoff. This time opening must include not only actual, on runway time, but total runway occupancy time: that time during which no other aircraft may use the runway. To insure that this time slot is available, the sequencing algorithm adds the estimated total runway occupancy time to the time that the RPAC passed through the entry gate (always designated the entry gate master clock time--EGMCT). This time becomes the release time. If RELTIM is less than or equal to TIME the phantom aircraft is assigned a velocity equal to that of the RPAC (this insures that the time slot for the takeoff "arrives" when it should) and a first node arrival time equal to RELTIM.

For the case of a real RAC and real RPAC or a real RAC and Phantom RPAC equations of motion are used to determine the time separation between aircraft. The only difference in the two cases is the distance separation upon which the time separation is based. As Federal Aviation Regulations now stand, aircraft on final approach are required to maintain three nautical miles separation while a departure cannot take place if an arrival is on final approach and closer than two nautical miles. Let

L = Length of final approach in nautical miles.

S = Required separation in nautical miles.

V_{RPAC} = Assigned velocity of the RPAC.

V_{RAC} = Assigned velocity of the RAC.

Case 1. Suppose $V_{RPAC} \geq V_{RAC}$. Then the two aircraft will be closest together at the moment that the RAC enters onto the final approach path. In this case

$$\text{SEP(in seconds)} = \frac{S}{V_{RPAC}} \cdot 3600 \qquad 4.8$$

Case 2. Suppose $V_{RPAC} < V_{RAC}$. Then the two aircraft will be closest together at the last moment they share the common final approach path (at the threshold). It will take the RPAC $\frac{L}{V_{RPAC}}$ hours to get to the runway threshold from the entry gate. Knowing that at this time the RPAC should be S nautical miles back (taking $\frac{L - S}{V_{RAC}}$ to arrive at this position), we have

$$\text{SEP} = \frac{L}{V_{RPAC}} - \frac{L - S}{V_{RAC}}$$

$$\text{SEP(in seconds)} = \left[\frac{S}{V_{RPAC}} + L\left(\frac{1}{V_{RPAC}} - \frac{1}{V_{RAC}}\right)\right] 3600 \qquad 4.9$$

release time is now

$$\text{RELTIM} = \text{SEP} + \text{EGMCT}$$

As discussed in Chapter III it is realized that the controller does not schedule aircraft for absolute minimum spacing but rather allows differing margins for error. To allow for this the quantity S used in the above equations is a normally distributed random variable. The choice of a normal distribution for this random variable as well as others arises from the intuitive notion that the random variables in question are the sum of other random variables, in which case the Central Limit Theorem supports the normal distribution for the sum.

Decisions Made at Each Node

Depending on whether the present event aircraft is real or phantom there are certain decisions concerning system operation to be made at each node. Define

DN as the decision node or that node located at the decision point.

THN as the threshold node or the node located over the threshold.

TDP as the node located at the touch down point.

Real Aircraft, Nodes 2 to DN-1. When a real aircraft arrives at one of these nodes the first check made is to see if minimum separation distance is being observed. Since a phantom aircraft is just as its name implies, approach separation criteria is only enforced between real aircraft. Therefore, on arrival of a real aircraft at a node (say at time t_1) subroutine error will scan the APROCH array to find the immediately preceding real aircraft and the time it arrived at its present position node (say at time t_2). If the separation between the two aircraft is greater than or equal to the minimum separation the program continues execution. However, if separation is less than the allowable minimum there are two cases to be considered (due to the discrete system of nodes).

Case 1. If $t_1 \leq t_2$ then the aircraft must wave off (execute a missed approach).

Case 2. If $t_1 \geq t_2$ then possibly the immediately preceding aircraft will be able to increase the separation to greater than or equal to minimum separation in time $t_2 - t_1$. The program will find the present velocity of the preceding aircraft from the APROCH array and from this calculate its estimated position in time $t_2 - t_1$. If this calculated position implies a separation that equals or exceeds minimum the program will continue in a normal fashion. Otherwise, the present event aircraft will be forced to wave off and an immediate system update will be called for. Since this study is concerned with random sequences of aircraft only, any aircraft violating minimum separation criteria is simply destroyed rather than actually reentering the holding stack.

If minimum separation criteria is met, the next check is that of ~~present velocity~~ against assigned velocity. Realizing the approach velocity is not constant an algorithm is included that introduces variability. The concept used is that of drawing a new velocity at each node from a normal distribution with the target velocity as the mean and a variance that reflects the deviation of present velocity from target velocity. Define EHR(I) as the error half range for a

Class I aircraft. That is, EHR is 3 standard deviations away from the mean of a normal curve. Further, define EHRMAX(I) as the maximum EHR allowable for a class I aircraft and EHRMIN(I) as the minimum allowable EHR for a class I aircraft. Thus, if the present velocity of the aircraft is within ± EHRMIN the new velocity, or correction velocity, will be drawn from a normal distribution with a mean equal to target velocity and standard deviation of $\frac{\text{EHRMIN}}{3}$. If the present velocity is greater than EHRMIN the standard deviation will be $\frac{\text{EHRMAX}}{3}$ with the same mean.

Real Aircraft at DN. When a real aircraft reaches the decision node it must either commit itself to a landing or execute a missed approach. Assuming the aircraft to be stabilized in landing configuration, the runway is checked for aircraft. If an aircraft is present on the runway, greater than one node from its estimated exit or estimated take off node, the approaching aircraft is forced to execute a missed approach. This is an attempt to remain consistent with FAR'S which permit an approaching aircraft to proceed if it is apparent that the aircraft on the runway will indeed make its exit or take off. Since a real aircraft is assumed stabilized for landing once the decision node is reached, no more correction velocities are generated once DN is reached.

Real Aircraft at THN. When a real aircraft reaches the threshold node a normally distributed touchdown point is generated. The mean and standard deviations used in the process generator are read in values determined from actual observations or published data for a particular aircraft class.

Real Aircraft at TDP. At the touchdown point a normally distributed deceleration rate is generated based upon read in values of the mean and standard deviation for a particular aircraft class.

Real Aircraft at Exit Nodes After TDP. At each exit node encountered after touchdown the present velocity is compared to the maximum allowable exit velocity for the particular aircraft class. If the velocity is less than or equal to the maximum exit velocity, an exit occurs.

It should be noted here that many default conditions are included in the program to prevent unrealistic conditions from occurring. For example, provisions are included in subroutine TIMNOD to keep an aircraft that barely misses an exit from coming to a complete stop on the runway. However, the user should make every effort to read in realistic data, or unrealistic situations may ensue.

Phantom Aircraft at the Entrance Node. When a phantom aircraft reaches the entrance node, in effect what has happened is that the predetermined departure time for the particular aircraft has arrived. The approach path is automatically checked for inbound real aircraft and if none are found program execution continues. Otherwise, the takeoff is rescheduled by putting the phantom aircraft back into the STACK array. This is in keeping with the program requirements of a random sequence of arrivals. If the aircraft is not rescheduled, the runway is checked for aircraft. If any aircraft are found then the departure is delayed until the aircraft presently occupying the runway clears the system. If no aircraft are found on the runway the departure's entrance velocity, acceleration rate, and take off velocity are generated from a normal generator with parameters for the particular aircraft class that have been read in.

Phantom Aircraft at Nodes After the Entrance Node. After entering the runway the aircraft's velocity is checked at each node against the previously generated liftoff velocity. When lift off occurs the system is checked to see what type aircraft is demanding service next. If the next aircraft is attempting to land the departure is considered "clear" of the system. However, if the next aircraft is waiting to take off, FAR'S must be consulted for appropriate separation criteria to determine when the departing aircraft may be considered "clear" of the system, enabling another operation to occur. The separation criteria for departing aircraft are read in values.

Validation of the Model

A complete discussion on the techniques used to validate the simulation model is included in Appendix C. However, to lend credibility to the study at the outset, graphs were prepared showing theoretical capacity (see Appendix C for how theoretical capacity was determined) versus separation for landings only aircraft. Three different homogeneous mixes were considered: 160 knot, 125 knot, and 100 knot approach velocity aircraft. Then all random variation except that occurring in the generation of interarrival times was removed from the model. Having made these changes to the model, the same three mixes used to find theoretical capacity were simulated. The simulated capacities were plotted over the theoretical capacities for easy comparisons and may be seen for the three mixes in Figures 5, 6, and 7. These unintuitive results may be more easily understood if the relationship between separation and waveoffs is considered for a particular approach velocity category. For example consider 160 knot aircraft. Remembering that capacity is defined as operations per hour, there would (in a deterministic system) be 40 aircraft entering the system per hour at 4 N. miles separation and 160 aircraft entering the system per hour at 1 N. mile separation. Were it not for the fact that separation standards are violated and waveoffs occur, capacity would continue to climb in a linear fashion from 40 operations per hour to 160 operations per hour. This violation of separation standards (the number of violations at decision height will continue to grow with decreased separation) and the ensuing waveoffs are what cause the "discontinuities" in the capacity curve. Between 4 and 2.7 N. miles separation there are 0% waveoffs, between 2.6 and 1.4 N. miles there are 50% waveoffs and between 1.5 and 1 N. miles there are 66 2/3% waveoffs. As can be seen the model follows the predicted results very closely. Variations between predicted values (from the mathematical model) and simulated values may be explained by the method of generating arrivals to the system. Due to the difficulty in supplying constants for all interarrival times

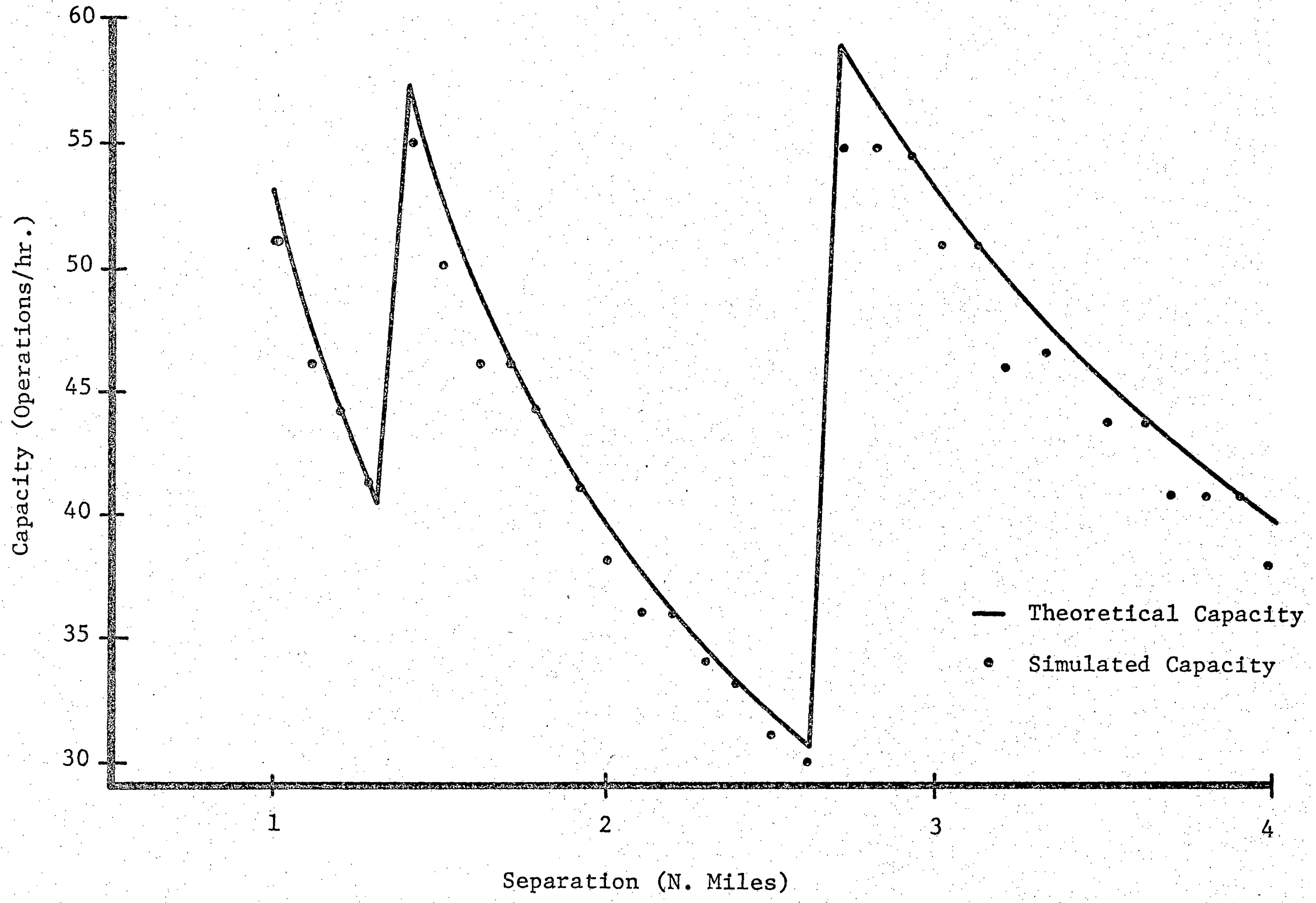


FIGURE 5: COMPARISON OF THEORETICAL CAPACITY AND SIMULATED CAPACITY (WITH ALL RANDOM VARIATION REMOVED EXCEPT IN INTERARRIVAL TIME) FOR LANDINGS ONLY, V = 160 KNOTS TDP = 800 FEET

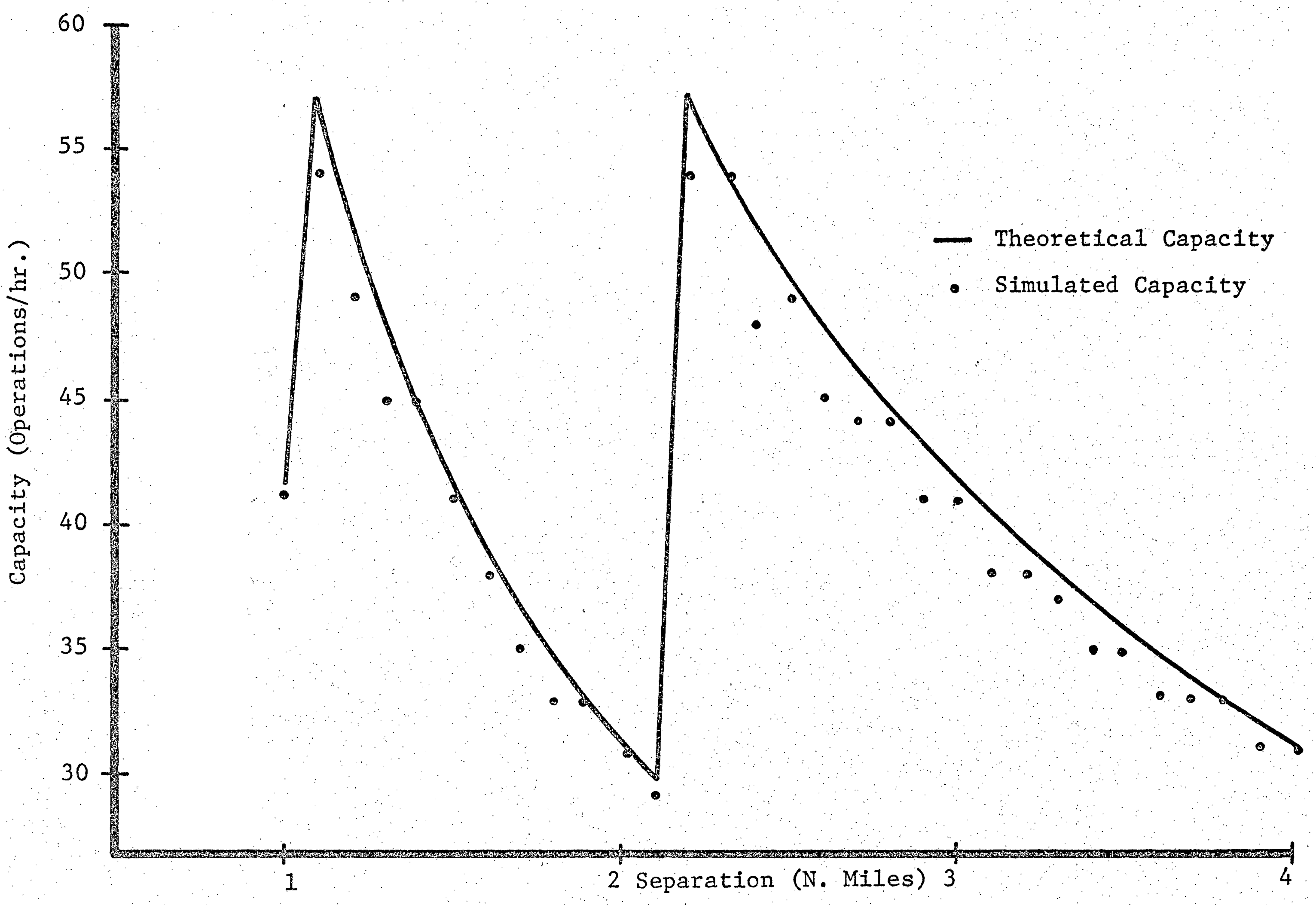


FIGURE 6: COMPARISON OF THEORETICAL CAPACITY AND SIMULATED CAPACITY (WITH ALL RANDOM VARIATION REMOVED EXCEPT IN INTERARRIVAL TIME) FOR LANDINGS ONLY, V= 125 KNOTS TDP = 1600 FEET

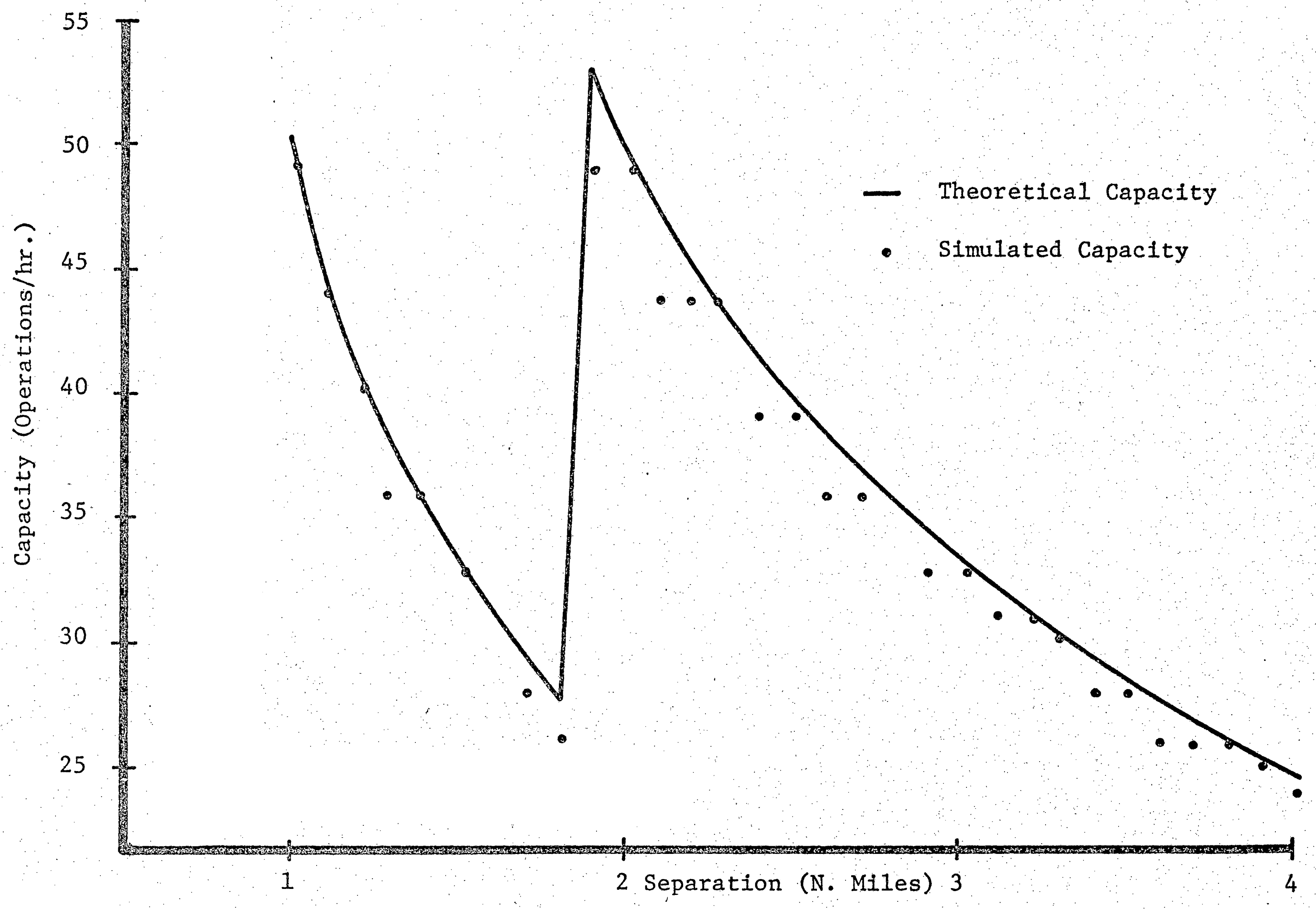


FIGURE 7: COMPARISON OF THEORETICAL CAPACITY AND SIMULATED CAPACITY (WITH ALL RANDOM VARIATION REMOVED EXCEPT IN INTERARRIVAL TIME) FOR LANDINGS ONLY, V = 100 KNOTS TDP = 4000 FEET

necessary in the simulation, it was decided to try to "load up" the system to the point that there was no lag between the exact time an aircraft was demanded and the time it was supplied. This technique worked well in most cases; however, due to the critical time factor in the releasing of aircraft to the final approach all time lags that did occur were reflected by a decrease in simulated capacity.

Using the Simulator

To use the simulator it is necessary that the user specify completely the characteristics of the runway, final approach path, departure path, and the aircraft using the system. With one exception, all these specifications are read in at the beginning of the simulation. The exception is the switch located in subroutine Alpha. If exponential interarrival times are desired set norm = 0. If normal interarrival times are desired set norm = 1 (the usefulness of the normal interarrival time is discussed in Chapter V).

A listing of the variables (with dimensions) to be read in follows, with a brief description of each. Remember that five aircraft classes are considered which implies each aircraft parameter has five values, one for each class.

1)	ACCMU	Mean runway acceleration rate in ft./sec./sec.
2)	ACCSD	Standard deviation of acceleration rate in ft./sec./sec.

3)	ALLDMN	The minimum allowable daily level for landings (reference Figure 2).
4)	ALTOMN	The minimum allowable daily level for take offs.
5)	DCELMU	The mean runway deceleration rate in ft./sec./sec.
6)	DCELSD	The standard deviation of deceleration rate in ft./sec./sec.
7)	DIST	1/number of nodes per N. mile or the distance in N. miles between nodes.
8)	DN	The decision node, located at decision height.
9)	DSTEN1	The distance in feet of the first exit from the threshold.
10)	DSTEN2	The distance in feet of the second exit from the threshold.
11)	DSTEN3	The distance in feet of the third exit from the threshold.
12)	DSTEN4	The distance in feet of the fourth exit from the threshold.
13)	DSTENT	The distance in feet from the threshold to the runway entrance for departing aircraft.
14)	EN1	Exit node #1.
15)	EN2	Exit node #2.
16)	EN3	Exit node #3.
17)	EN4	Exit node #4.
18)	ENT	Entrance node for departing aircraft.
19)	EHRMAX	Maximum error half range.

20)	EHRMIN	Minimum error half range.
21)	EXVLMX	Maximum runway exit velocity.
22)	F	The length of the final approach path in N. miles.
23)	MCT	The beginning time of the simulation, in seconds (12 o'clock noon = 12 hours X 3600 secs/hr).
24)	NP	The number of nodes per N. mile or the number of parts a N. mile is divided into.
25)	NROWAP	The number of rows desired in the array APROCH.
26)	NROWST	The number of rows desired in the array STACK.
27)	NROWWT	The number of rows desired in the array WAIT.
28)	PERLD	Peak evening rate for landings (landings/hr.).
29)	PERTO	Peak evening rate for take offs (take offs/hr.).
30)	PMRLD	Peak morning rate for landings (landings/hr.).
31)	PMRTO	Peak morning rate for take offs (take offs/hr.).
32)	PTELD	Peak time of evening landings (secs.).
33)	PTETO	Peak time of evening take offs (secs.).
34)	PTMLD	Peak time of morning landings (secs.).
35)	PTMTO	Peak time of morning take offs (secs.).
36)	RWYLDT	Average time on runway for a landing.
37)	RWYTOT	Average time on runway for a take off.
38)	S	Distance (in N. miles) the controller algorithm uses for separation criteria in sequencing aircraft. This is naturally somewhat larger than the minimum separation criteria to allow a buffer.

39)	SMIN	Minimum separation allowed between aircraft (in N. miles).
40)	TDDMU	Mean touchdown distance measured from the threshold in feet.
41)	TDDSD	Standard deviation of touchdown distance in feet.
42)	TIMLIM	The time limit of the simulation in seconds.
43)	VAPMU	Mean approach velocity (in knots).
44)	VAPSD	Standard deviation of approach velocity (in knots).
45)	VARTIM	The time in seconds over which variance is to be calculated.
46)	VENTMU	Mean runway entrance velocity for a departing aircraft (in knots).
47)	VENTSD	Standard deviation of runway entrance velocity (in knots).
48)	VTOMU	Mean take off velocity (in knots).
49)	VTOSD	Standard deviation of takeoff velocity (in knots).
50)	XDD	The distance of the decision node (DN) from the threshold (in N. miles). XDD in fractional form must be evenly divisible by 1/NP.

Output From the Simulator

The program will print out: 1) a list of all input data, 2) the time and type of each operation, waveoff, or aborted take off along with the aircraft class and the number of aircraft present in the system at the time (excluding that aircraft) and 3) a simulation summary.

The simulation summary gives: 1) the beginning time of the simulation, 2) the end time of the simulation, 3) total time simulated, 4) total missed approaches, 5) total aborted take offs, 6) total landings, 7) total take offs, 8) a breakdown of take offs and landings by class, and 9) the variance and standard deviation of capacity, calculated over the read in time value, VARTIM. The output from an example problem may be seen in Appendix F.

CHAPTER V. ANALYSIS AND RESULTS

Method of Analysis

The purpose in analyzing the simulator output was to determine whether or not there was some underlying relationship between capacity and the independent variables investigated. Furthermore, it was desired to know if any relationship found could be useful in predicting capacity from knowledge of the independent variables, separation, and approach velocity for each of the five aircraft types.

The technique employed to identify and evaluate this relationship was least squares regression analysis. In using the method of least squares a functional relation between the dependent and independent variables is hypothesized. Then the parameters of the hypothesized equations are estimated. A hypothesized relation may be of any form, for example:

$$y_i = \beta_o + \sum_{j=1}^{n} \beta_j x_{ji} + \varepsilon_i, \qquad 5.1$$

where n is the number of independent variables under study, $\beta_o, \ldots, \beta_n$ are parameters of the function, and ε represents the random variation in the dependent variable. Equation 5.1 may be estimated by an equation of the form

$$\hat{y}_i = b_o + \sum_{j=1}^{n} b_j x_{ji} \qquad 5.2$$

where $b_o \ldots b_n$ are estimates of $\beta_o \ldots \beta_n$. Once the hypothesized functional relationship has been approximated by y_i it is desired to know whether or not y_i adequately describes the true (unknown) underlying relationship. This is accomplished by the F test for lack of fit. Define

MS(LF) = Mean square for lack of fit

MS(E) = Mean square for error

The appropriate F test for lack of fit is then

$$F = \frac{MS(LF)}{MS(E)} \qquad 5.3$$

Let μ_{y_i} be the true (unknown) mean value for the i^{th} combination of independent variables, and let $\hat{\mu}_{y_i}$ be the mean as determined by the true best fitting regression equation of the form in 5.1. Then the F test for lack of fit tests the hypothesis

$$H_o: \mu_{y_i} = \hat{\mu}_{y_i} \; (\hat{\mu}_{y_i} = \beta_o + \sum_{j=1}^{n} \beta_j x_{ji}) \text{ for all } i$$

against the alternative

$$H_1: \mu_{y_i} \neq \hat{\mu}_{y_i}, \text{ for some } i$$

Rejection of the null hypothesis indicates that the hypothesized regression equation does not describe the variation in the dependent variable adequately, in the sense that it is likely that additional variation can be explained by some other equation.

It should be emphasized that in order to run a test for lack of fit it is necessary to observe more than one value of the dependent variable at at least one combination of values of the independent variables. This is necessary to obtain an unbiased estimate of the error variance, MS(E).

Once a function is found that adequately describes the relationship between the dependent and independent variables under study, it is desired to know whether or not the function is useful as a predictor of the dependent variable. An indication of this usefulness may be evaluated through the index of correlation. The index of correlation (R^2) tells what percentage of the variation of the dependent variable, capacity, has been explained by the fitted equation. At this point the reader should be cautioned about using the index of correlation blindly. It is possible to have a high index of correlation that is statistically insignificant. Hence, it is necessary to test for significance of the index of correlation. Define

$$SS(b) = \sum_{i=1}^{n} (\text{Sum of squares for } b_i)$$

The appropriate F test for significance of the index of correlation is

$$F = \frac{\frac{SS(b)}{n}}{MS(E)} \qquad 5.4$$

The hypothesis tested is

$$H_o: R^2 = 0$$

against the alternative

$$H_1: \; R^2 \neq 0$$

If R^2 is found to be high yet statistically insignificant, it is indicated that the value found for R^2 was meaningless.

When any functional relationship is hypothesized and accepted or rejected as being the true underlying relation, there are two types of error possible. The error of the first type (TYPE I) is the error made if H_o is rejected when it should be accepted. The error of the second type (TYPE II) is the error made if H_o is accepted when it should be rejected. The probabilities of making a TYPE I and TYPE II error are α and β respectively. That is:

$$\alpha = P(\text{rejecting } H_o \,|\, \hat{\mu}_{y_i} = \mu_{y_i}, \text{ for all } i)$$

$$\beta = P(\text{accepting } H_o \,|\, \mu_{y_i} - \hat{\mu}_{y_i} = \delta_i \neq 0 \text{ for some } i)$$

As a result of the above, define a non-centrality parameter, λ, as:

$$\lambda = \frac{n}{2\sigma^2} \sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2$$

where

n = the number of observations or replications per combination (level) of independent variables

m = number of combinations of independent variables

σ^2 = the variance of the dependent variable,

$\hat{\mu}_{y_i}$ = the estimated mean of the dependent variable for the i^{th} combination of the independent variables.

μ_{y_i} = the true overall mean of the dependent variable for the i^{th} combination of the independent variables.

The central F density function (see Appendix B) with, non-centrality parameter zero, is given by $f[y|0, m-k-1, m(n-1)]$, with m-k-1 and m(n-1) degrees of freedom. It is a simple matter to solve for α. Letting C_u be the upper limit for the F test for lack of fit, then

$$\int_0^{C_u} f[y|0, m-k-1, m(n-1)]dy = 1 - \alpha \qquad 5.5$$

This question is most easily solved by using standard F tables. Note that in assuming the non-centrality parameter to be zero it is implied that the mean value of the dependent variable as given by the estimating equation does not differ from the true mean for any combination of independent variables. That is, since

$$\lambda = \frac{n}{2\sigma^2} \sum_{i=1}^{m} (\hat{\mu}_{y_i} = \mu_{y_i})^2$$

if

$$\lambda = 0$$

then

$$\sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2 = 0$$

and

$$\hat{\mu}_{y_i} = \mu_{y_i}, \quad i = 1, 2, \ldots, m$$

If the mean value of the dependent variable as given by the estimating equation, $\hat{\mu}_{y_i}$, does differ from the true mean for some combination of independent variables, then λ is no longer equal to zero. In this case the F density function no longer has a non-centrality parameter equal to zero and is termed the non-central F density (see Appendix B) with m-k-1 and m(n-1) degrees of freedom. If the non-central F were integrated over the same interval as equation 5.5, the result would be

$$\int_0^{C_u} f[y|\lambda, m-k-1, m(n-1)]dy = \beta \qquad 5.6$$

A graphical comparison of equations 5.5 and 5.6 is shown in Figure 8.

Equation 5.6 offers a convenient method of determining m and n for the simulation experiment, corresponding to given values of α and β. If the true mean value of the dependent variable is not equal to the value given by the estimating equation, $\hat{\mu}_{y_i}$, for every combination of levels of independent variables H_0 may be accepted when it should be rejected with probability β. Thus, the power of the regression analysis is the probability of rejecting the null hypothesis that the hypo-

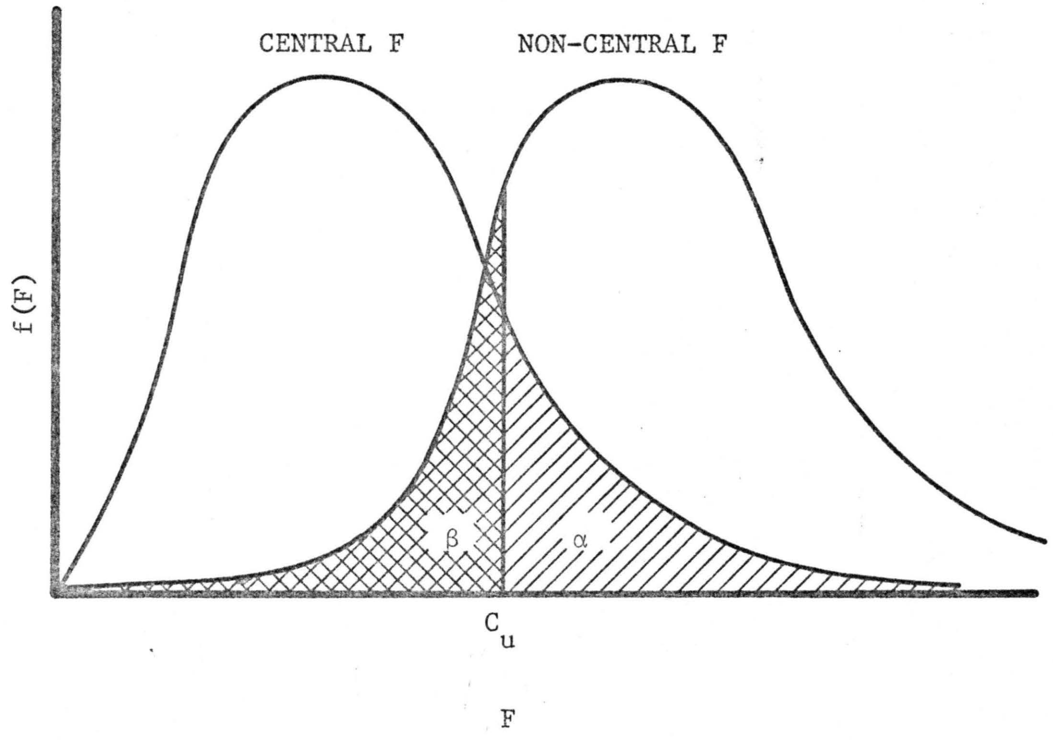


FIGURE 8: GRAPHICAL REPRESENTATION OF α AND β

thesized equation explains the variation in y_i when it is false, or 1-β.

The problem in the method suggested above for finding m and n is the difficulty in integrating the non-central F density function. To avoid this cumbersome task a standard approximation is used.

Patnaik's Approximation

Patnaik's approximation is:

$$\beta = \int_0^{\frac{C_u}{p}} f[y|0, r, m(n-1)]dy \qquad 5.7$$

where

$$p = \frac{(m-k-1) + 2\lambda}{m-k-1}$$

$$r = \frac{[(m-k-1) + 2\lambda]^2}{(m-k-1) + 4\lambda} \qquad 5.8$$

$$\lambda = \frac{n}{2\sigma^2} \sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2 \qquad 5.9$$

$$C_u = F_{1-\alpha}[0, m-k-1, m(n-1)] \qquad 5.10$$

k = the number of independent variables.

Designing the Experiment

With Patnaik's approximation, a simulation experiment was designed in which the level of significance (α) was .10 and the desired value of β was .10. As an example six independent variables were chosen, with their effects on capacity to be determined. The six variables chosen were:

1) Separation.
2) Mean approach velocity for class 1 aircraft.
3) Mean approach velocity for class 2 aircraft.
4) Mean approach velocity for class 3 aircraft.
5) Mean approach velocity for class 4 aircraft.
6) Mean approach velocity for class 5 aircraft.

The first requirement in designing the experiment was to find the number of combinations, or levels, of independent variables, and the number of replications per level, at which to observe capacity. This was accomplished using the following rules and Patnaik's approximation:

1) Arbitrarily set values for m and n.
2) For each combination of m and n arbitrarily set several values of λ.
3) Calculate p, r, and C_u for each of the above combinations of m, n, and λ.
4) Look up β for each combination of m, n, and λ.
5) Repeat these steps for several combinations of m and n.

Since β has no meaning without an associated detectable difference between the true and estimated means, it is now necessary to estimate the variance of y_i and return to each combination of m, n, and λ above and see what detectable difference, $\sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2$, is implied (an operating characteristic curve analysis). The combination of m and n that appears to give the best results is then chosen as the design. The following example is offered using 29 levels with 4 replications per level. A pilot study was conducted and the best estimate for σ^2 was found to be 28.

λ	C_u	C_u/p	r	β	$\frac{2\sigma^2\lambda}{n} = \sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2$
0.0645	1.483	1.474	22.001	0.895	00.903
1.0000	1.483	1.359	22.154	0.840	14.000
2.0000	1.483	1.255	22.533	0.771	28.000
6.0000	1.483	0.960	25.130	0.468	84.000
9.0000	1.483	0.816	27.586	0.227	126.000
12.0000	1.483	0.680	31.135	0.116	182.000

This particular combination of m and n was chosen as the design to be used because of its superior power in detecting deviations of the regression equation from the true equation. Take for example m = 29, n = 4, $\lambda = 13$. It is seen that $\sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2 = 182$ which implies that the regression equation will be accepted as fitting the data with probability .116 under the following circumstances:

1) if any one population mean, as determined by the regression equation, differs from the true mean by as much as 13.491.

2) if any two population means, as determined by the regres-

sion equation, differ from the true mean by as much as 9.539.

3) If any three population means, as determined by the regression equation, differ from the true mean by as much as 7.789.
4) If any four population means, as determined by the regression equation, differ from the true mean by as much as 6.745.
5) If any five population means, as determined by the regression equation, differ from the true mean by as much as 6.033.
6) If any six population means, as determined by the regression equation, differ from the true mean by as much as 5.508.

Of course the other end of the spectrum should be analyzed to ensure that the design is not too powerful. Taking $m = 29$, $n = 4$, $\lambda = .0645$ the regression equation obtained will be accepted as fitting data with probability .895 under the following circumstances:

1) If any one population mean, as determined by the regression equation, differs from the true population mean by as much as .950.
2) If any two population means, as determined by the regression equation, differ from the true population mean by

as much as .672.

3) If any three population means, as determined by the regression equation, differ from the true population mean by as much as .549.

4) If any four population means, as determined by the regression equation, differ from the true population mean by as much as .475.

5) If any five population means, as determined by the regression equation, differ from the true population mean by as much as .425.

6) If any six population means, as determined by the regression equation, differ from the true population mean by as much as .388.

It must be remembered that the above calculations are all based on the best estimate for σ^2 available. As soon as updated estimates are available for σ^2, revised calculations should be made to insure the adequacy of the design. If power is lacking, more data may be collected.

Conducting the Experiment

Input to the simulator

Input data common to every combination of independent variables are listed in Appendix G. The variable names are those defined in Chapter IV. This data is used for the example only and does not necessarily reflect real life situations at every air terminal, although every attempt was made to keep the values of the independent variable within reasonable limits. The ranges used for mean velocity for each aircraft class are given below (all velocities are in nautical miles per hour).

Class I	Class II	Class III	Class IV	Class V
45-65	65-85	85-115	115-135	140-165

The range considered for minimum separation was from one to four nautical miles. For purposes of scheduling arrivals it was necessary to read in a scheduled separation distance. As discussed in Chapter IV, the scheduled separation distance will never be the minimum allowable separation in order to allow a margin for error. In all cases the scheduled separation distance was taken to be the minimum plus three-tenths nautical mile.

In order to draw conclusions about capacity over the whole range of values considered, combinations of the independent variables were generated randomly over the specified ranges. The values of the independent variables unique to each set of four replications are listed in Appendix G.

The simulator was run for each of three aircraft speed mixes: fast, medium, and slow. The input associated with each mix is given in Appendix G. The numbers used for peak arrival and departure rates may seen extremely high; however, they were necessary to insure that completely saturated conditions existed for the time period simulated. To further insure a time period saturated with operations, the arrival-departure process was simulated for 45 minutes prior to actually counting operations. Thus, as seen in Appendix G, the master clock was initialized at 48,500 seconds and the end of the simulated time period was at 64,800. This corresponds to 4:15 p.m. - 6:00 p.m.

Results of the Experiment

Using the experimental design previously described, each mix was used to simulate the operation of an airport for 116 independent trials (29 combinations of independent variables with 4 replications for each combination). The results of these simulation trials were analyzed using analysis of variance techniques to arrive at regression equations describing capacity as related to the independent variables for each mix. The hypotheses tested regarding the relationship between capacity and the independent variables were that: 1) the relationship is linear, 2) the relationship is quadratic, 3) the relationship is cubic. For all three mixes, the most reasonable relationship for each mix was found to be a quadratic equation of the form

$$\hat{y} = b_o + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6 + b_7x_1^2 + b_8x_2^2 + b_9x_3^2 + b_{10}x_4^2 + b_{11}x_5^2 + b_{12}x_6^2 \qquad 5.11$$

where

$\hat{y}$ = an estimate of capacity in operations per hour.

x_1 = separation in N. miles.

x_2 = Mean approach velocity for class 1 aircraft.

x_3 = Mean approach velocity for class 2 aircraft.

x_4 = Mean approach velocity for class 3 aircraft.

x_5 = Mean approach velocity for class 4 aircraft.

x_6 = Mean approach velocity for class 5 aircraft.

b_i = The regression coefficients, $i = 0, 1, \ldots, 12$

The three mixes will hereafter be referred to as either M1E, M2E, or M3E where

M1E = Mix 1 with exponential interarrival times.

M2E = Mix 2 with exponential interarrival times.

M3E = Mix 3 with exponential interarrival times.

The analysis of variance table for each of the three mixes may be found in Tables IV, V, and VI. Regression coefficients for each of the three mixes are also given in Tables IV, V, and VI.

To reiterate, there are two questions to be answered concerning the regression equations:

(1) is the hypothesized equation the correct equation and

(2) is the equation useful as a predictor of capacity from a knowledge of the independent variables.

To answer these questions, all F ratios for lack of fit and regression were calculated and the tests for lack of fit and the significance of the regression equations were conducted. The results are summarized in Tables VII, VIII, and IX. These results indicate that the correct equations have been found and that they have, in fact, explained a

TABLE IV: ANALYSIS OF VARIANCE TABLE FOR M1E

Source of Variation	Value of Regression Coefficient	Degrees of Freedom	Sum of Squares	Mean Squares
b_o	-320.36	1	175,968.30	175,968.30
b_1	0.36	1	1,737.24	1,737.24
b_2	- 3.38	1	152.87	152.87
b_3	3.73	1	35.08	35.08
b_4	0.91	1	293.76	293.76
b_5	- 3.25	1	146.66	146.66
b_6	5.81	1	0.54	0.54
b_7	- 0.92	1	3.89	3.89
b_8	- 0.02	1	45.13	45.13
b_9	- 0.02	1	12.62	12.62
b_{10}	0.00	1	10.88	10.88
b_{11}	- 0.01	1	13.52	13.52
b_{12}	- 0.02	1	55.06	55.06
Lack of fit		16	498.45	31.15
Experimental Error		87	2,048.00	23.54

Index of Correlation = .496

TABLE V: ANALYSIS OF VARIANCE TABLE FOR M2E

Source of Variation	Value of Regression Coefficient	Degrees of Freedom	Sum of Squares	Mean Square
b_o	-547.63	1	152,648.80	152,648.80
b_1	3.92	1	925.45	925.45
b_2	- 0.18	1	8.70	8.70
b_3	- 2.61	1	311.89	311.89
b_4	0.57	1	16.77	16.77
b_5	1.20	1	13.76	13.76
b_6	7.42	1	0.36	0.36
b_7	- 1.48	1	88.20	88.20
b_8	0.00	1	19.88	19.88
b_9	- 0.02	1	28.94	28.94
b_{10}	0.00	1	17.46	17.46
b_{11}	0.00	1	4.31	4.31
b_{12}	- 0.02	1	89.08	89.08
Lack of fit		16	333.88	20.87
Experimental Error		87	1,974.50	22.70

Index of Correlation = .398

TABLE VI: ANALYSIS OF VARIANCE TABLE FOR M3E

Source of Variation	Value of Regression Coefficient	Degrees of Freedom	Sum of Squares	Mean Square
b_o	−495.85	1	104,580.10	104,580.10
b_1	3.10	1	773.73	773.73
b_2	− 2.11	1	488.55	588.55
b_3	1.39	1	32.76	32.76
b_4	1.91	1	1.32	1.32
b_5	5.33	1	0.01	0.01
b_6	1.23	1	8.84	8.84
b_7	− 1.21	1	45.15	45.15
b_8	− 0.02	1	27.53	27.53
b_9	− 0.01	1	10.47	10.47
b_{10}	− 0.01	1	51.01	51.01
b_{11}	− 0.02	1	42.20	42.20
b_{12}	0.00	1	2.73	2.73
Lack of fit		16	111.88	6.99
Experimental Error		87	1,146.75	13.18

Index of Correlation = .541

TABLE VII: F RATIOS FOR M1E

Source of Variation	Degrees of Freedom	Sum of Squares	Mean Square	F_{exp}	Tabular $F_{.90}$
Regression	12	2507.24	208.94	9.20	1.63
Lack of Fit	16	498.45	31.15	1.32	1.57
Experimental Error	87	2048.00	23.54		

TABLE VIII: F RATIOS FOR M2E

Source of Variation	Degrees of Freedom	Sum of Squares	Mean Square	F_{exp}	Tabular $F_{.90}$
Regression	12	1524.73	127.06	5.59	1.63
Lack of Fit	16	333.88	20.87	0.92	1.57
Experimental Error	87	1974.50	22.70		

TABLE IX: F RATIOS FOR M3E

Source of Variation	Degrees of Freedom	Sum of Squares	Mean Square	F_{exp}	Tabular $F_{.90}$
Regression	12	1484.23	123.69	9.38	1.63
Lack of Fit	16	111.88	6.99	0.53	1.57
Experimental Error	87	1146.75	13.18		

significant amount of variation in y_i. Just how much variation is explained by the equations is indicated by the index of correlation. This index is given in Tables IV, V, and VI.

Having obtained the regression equations that were sought it was desired to check the actual power of the equations as a predictor of capacity against their intended power. This was required because it was realized that that original estimate for variance may have been incorrect. As in the original design, using Patnaik's approximation, values of λ were assumed and the corresponding β errors found. This yielded the operating characteristic curve shown in Figure 9. However, as previously stated, β is a meaningless term without stating what difference between actual and predicted capacity is detectable by the experimental design. To answer this question the pooled estimate of variance was obtained from the analysis of variance table and the sum of squared deviations about the true mean capacity calculated. From this sum of squared deviations it is possible to calculate how great the estimated mean capacity may deviate from the true mean capacity at combinations of the 1st, 2nd, 3rd, 4th, 5th, or 6th (called points from here on) independent variable and yet still be considered a "good" fitting equation with probability β. Consider M1E:

$$\sigma^2 = 24.0000$$

$$\lambda = 0.0645$$

From Patnaik's approximation:

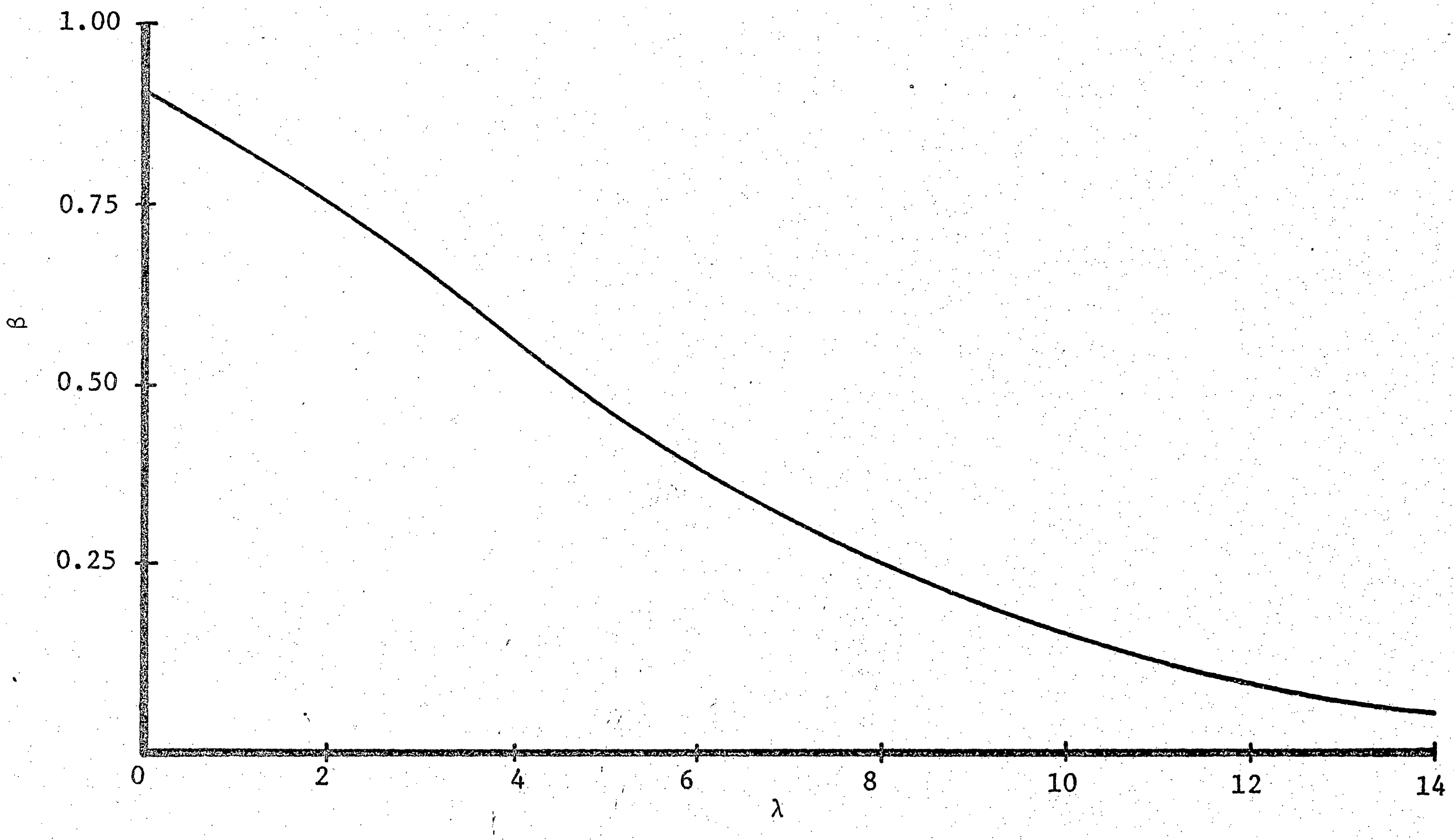


FIGURE 9: OPERATING CHARACTERISTIC CURVE FOR QUADRATIC FIT, ONE SIDED (UPPER TAIL) F TEST, $\alpha = .10$

$$\frac{2\sigma^2\lambda}{n} = \sum_{i=1}^{29} (\hat{\mu}_{y_i} - \mu_{y_i})^2 = .774$$

This implies that the regression equation found may differ from the true equation:

at any 1 point by $\sqrt{.774} = .880$

at any 2 points by $\sqrt{\frac{.774}{2}} = .662$

at any 3 points by $\sqrt{\frac{.774}{3}} = .508$

at any 4 points by $\sqrt{\frac{.774}{4}} = .440$

at any 5 points by $\sqrt{\frac{.774}{5}} = .393$

at any 6 points by $\sqrt{\frac{.774}{6}} = .359$

and still be accepted as a reliable fit with probability .895. A similar analysis was conducted for 10 values of λ for each of the three mixes investigated and the results are presented in Tables X, XI, and XII. It was felt that the design was entirely adequate, in fact more powerful than the original design due to the overestimation of variance in the pilot study used to derive the initial design.

Having ascertained that the design was, in fact, adequate it remains to examine the regression equations for any insight they may

TABLE X: β CORRESPONDING TO DEVIATIONS OF THE REGRESSION MEAN FROM THE TRUE MEAN

AT 1, 2, 3, 4, 5 AND 6 POINTS FOR MlE

$$\text{Sum of Squared Deviations} = \sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2$$

λ	Beta	1 Point	2 Points	3 Points	4 Points	5 Points	6 Points	Sum of Squared Deviations
0.0645	0.895	0.880	0.622	0.508	0.440	0.393	0.359	0.774
1.0	0.840	3.464	2.449	2.000	1.732	1.549	1.414	12.000
2.0	0.771	4.899	3.464	2.828	2.449	2.191	2.000	24.000
3.0	0.698	6.000	4.243	3.464	3.000	2.683	2.449	36.000
4.0	0.619	6.928	4.899	4.000	3.464	3.098	2.828	48.000
6	0.468	8.485	6.000	4.899	4.243	3.795	3.464	72.000
8.0	0.335	9.798	6.928	5.657	4.899	4.382	4.000	96.000
9.0	0.227	10.392	7.348	6.000	5.196	4.648	4.243	108.000
11.0	0.183	11.489	8.124	6.633	5.745	5.138	4.690	132.000
13.0	0.116	12.490	8.832	7.211	6.245	5.586	5.099	156.000

TABLE XI: β CORRESPONDING TO DEVIATIONS OF THE REGRESSION MEAN FROM THE TRUE MEAN AT 1, 2, 3, 4, 5 AND 6 POINTS FOR M2E

$$\text{Sum of Squared Deviations} = \sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2$$

λ	Beta	1 Point	2 Points	3 Points	4 Points	5 Points	6 Points	Sum of Squared Deviations
0.0645	0.895	0.861	0.609	0.497	0.431	0.385	0.352	0.742
1.0	0.840	3.391	2.398	1.958	1.696	1.517	1.384	11.500
2.0	0.771	4.796	3.391	2.769	2.398	2.145	1.958	23.000
3.0	0.698	5.874	4.153	3.391	2.937	2.627	2.398	34.500
4.0	0.619	6.782	4.796	3.916	3.391	3.033	2.769	46.000
6.0	0.468	8.307	5.874	4.796	4.153	3.715	3.391	69.000
8.0	0.335	9.592	6.782	5.538	4.796	4.290	3.916	92.000
9.0	0.227	10.173	7.194	5.874	5.087	4.550	4.153	103.500
11.0	0.183	11.247	7.953	6.494	5.624	5.030	4.592	126.500
13.0	0.116	12.227	8.646	7.059	6.114	5.468	4.992	149.500

TABLE XII: β CORRESPONDING TO DEVIATIONS OF THE REGRESSION MEAN FROM THE TRUE MEAN AT 1, 2, 3, 4, 5 AND 6 POINTS FOR M3E

$$\text{Sum of Squared Deviations} = \sum_{i=1}^{m} (\hat{\mu}_{y_i} - \mu_{y_i})^2$$

λ	Beta	1 Point	2 Points	3 Points	4 Points	5 Points	6 Points	Sum of Squared Deviations
0.0645	0.895	0.647	0.458	0.374	0.324	0.289	0.264	0.419
1.0	0.840	2.550	1.803	1.472	1.275	1.140	1.041	6.500
2.0	0.771	3.606	2.550	2.082	1.803	1.612	1.472	13.000
3.0	0.698	4.416	3.122	2.550	2.208	1.975	1.803	19.500
4.0	0.619	5.099	3.606	2.944	2.550	2.280	2.082	26.000
6.0	0.468	6.245	4.416	3.606	3.122	2.793	2.550	39.000
8.0	0.335	7.211	5.099	4.163	3.606	3.225	2.944	52.000
9.0	0.227	7.649	5.408	4.416	3.824	3.421	3.122	58.500
11.0	0.183	8.456	5.979	4.882	4.228	3.782	3.452	71.500
13.0	0.116	9.192	6.500	5.307	4.596	4.111	3.753	84.500

furnish concerning runway capacity. The regression equations obtained are of the form given in equation 5.11, where the values of the regression coefficients are given in Tables IV, V, and VI. By choosing values of the independent variables over the range of the design and substituting into equation 5.11, capacity can be determined for any specific set of conditions for any of the three mixes. As an example it was desired to investigate the affects of separation criteria on capacity for the three mixes of arriving and departing aircraft. For each case the values of the five approach velocity categories used were:

x_2 = 60 knots.

x_3 = 80 knots

x_4 = 100 knots.

x_5 = 125 knots.

x_6 = 160 knots.

Separation was varied from 1 to 4 N. miles in increments of .1 N. mile. Graphs of capacity versus separation under the above conditions are shown in Figures 10, 11, and 12.

In analyzing Figures 10, 11, and 12 two facts are readily apparant:

(1) by decreasing the number of slower aircraft in the total population, capacity can be significantly increased.

(2) there are considerable affects on runway capacity caused by the interaction of arriving and departing aircraft. This is

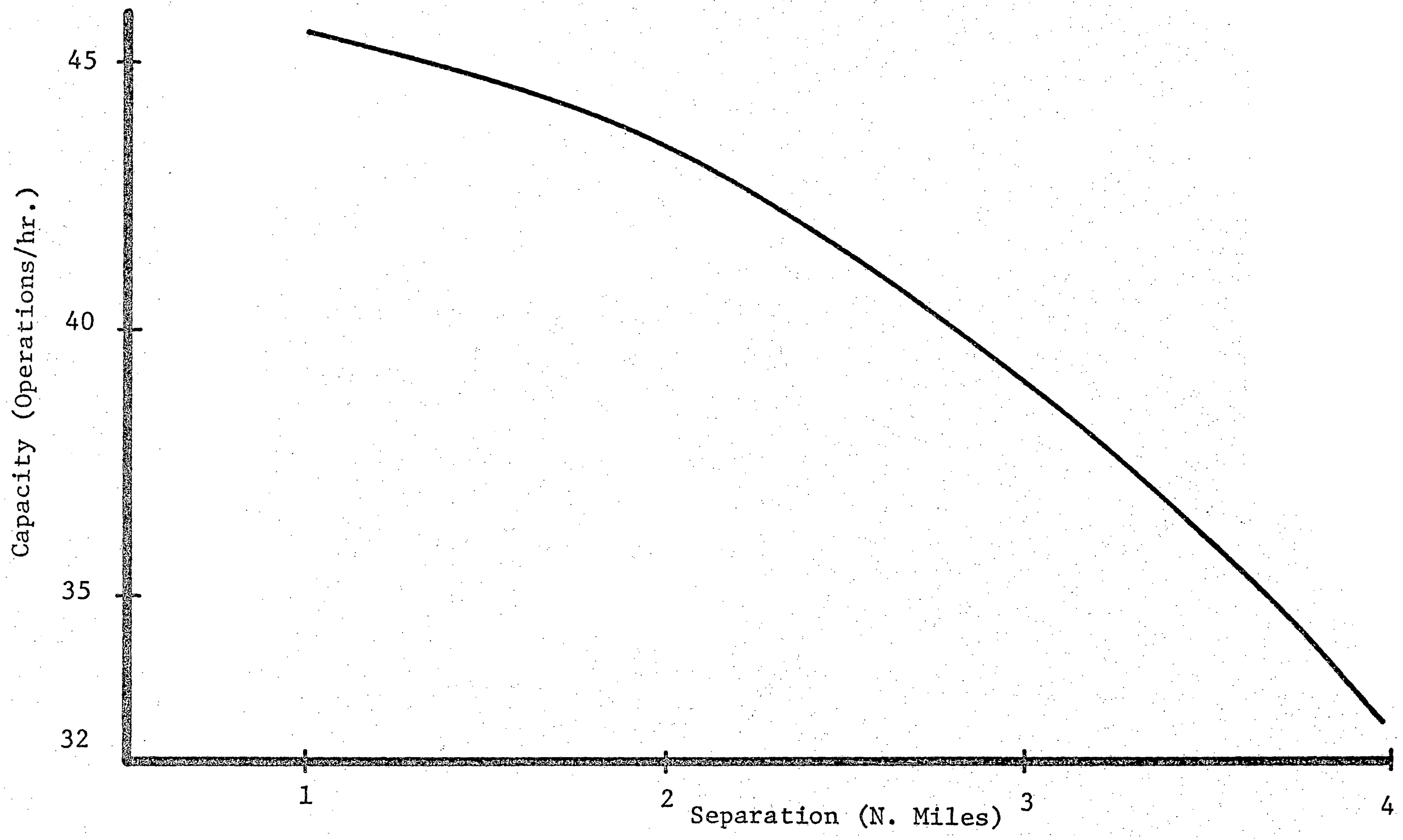


FIGURE 10: CAPACITY VS SEPARATION FOR LANDINGS AND TAKE OFFS COMBINED, MIX 1 (M1E)

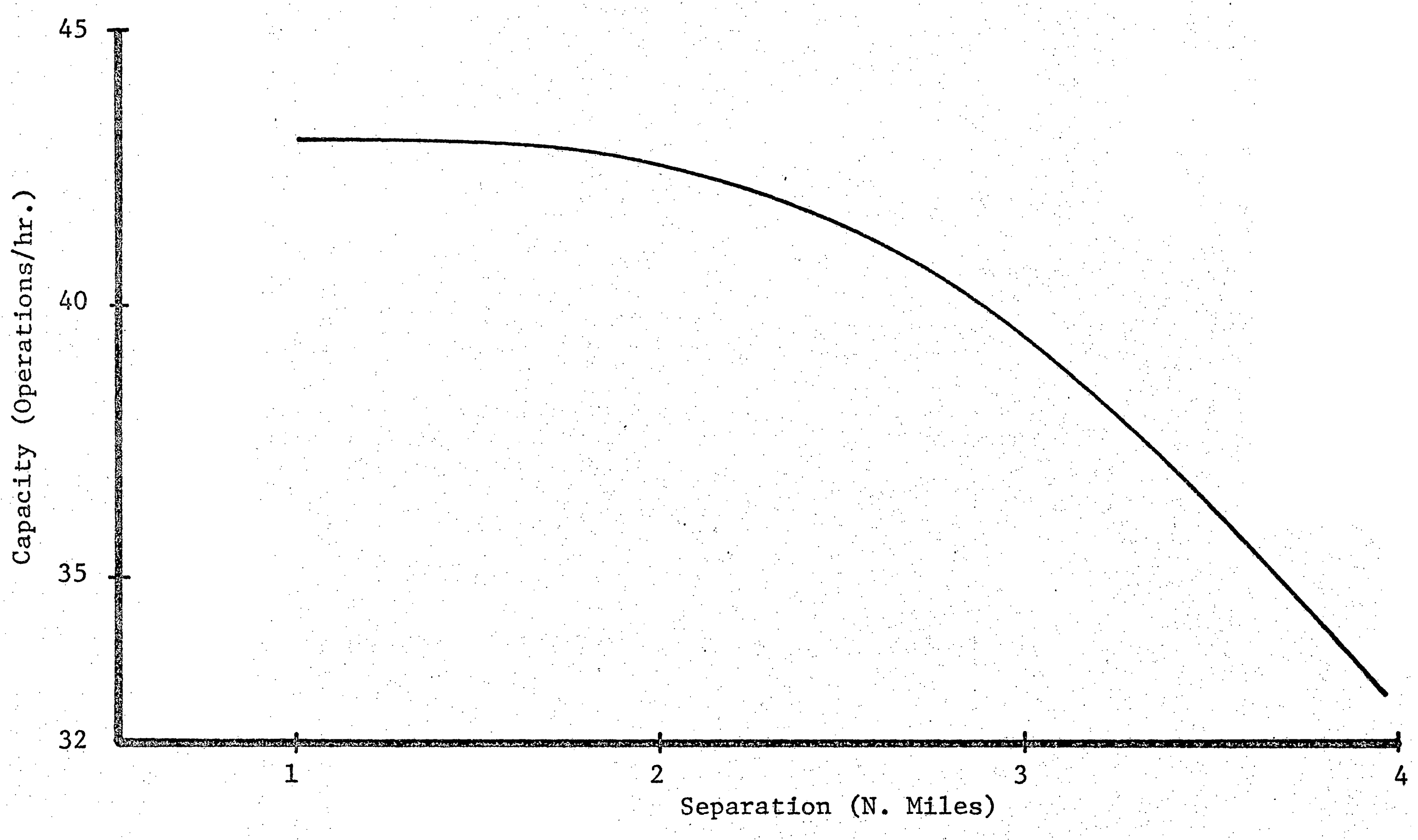


FIGURE 11: CAPACITY VS SEPARATION FOR LANDINGS AND TAKEOFFS COMBINED, MIX 2 (M2E)

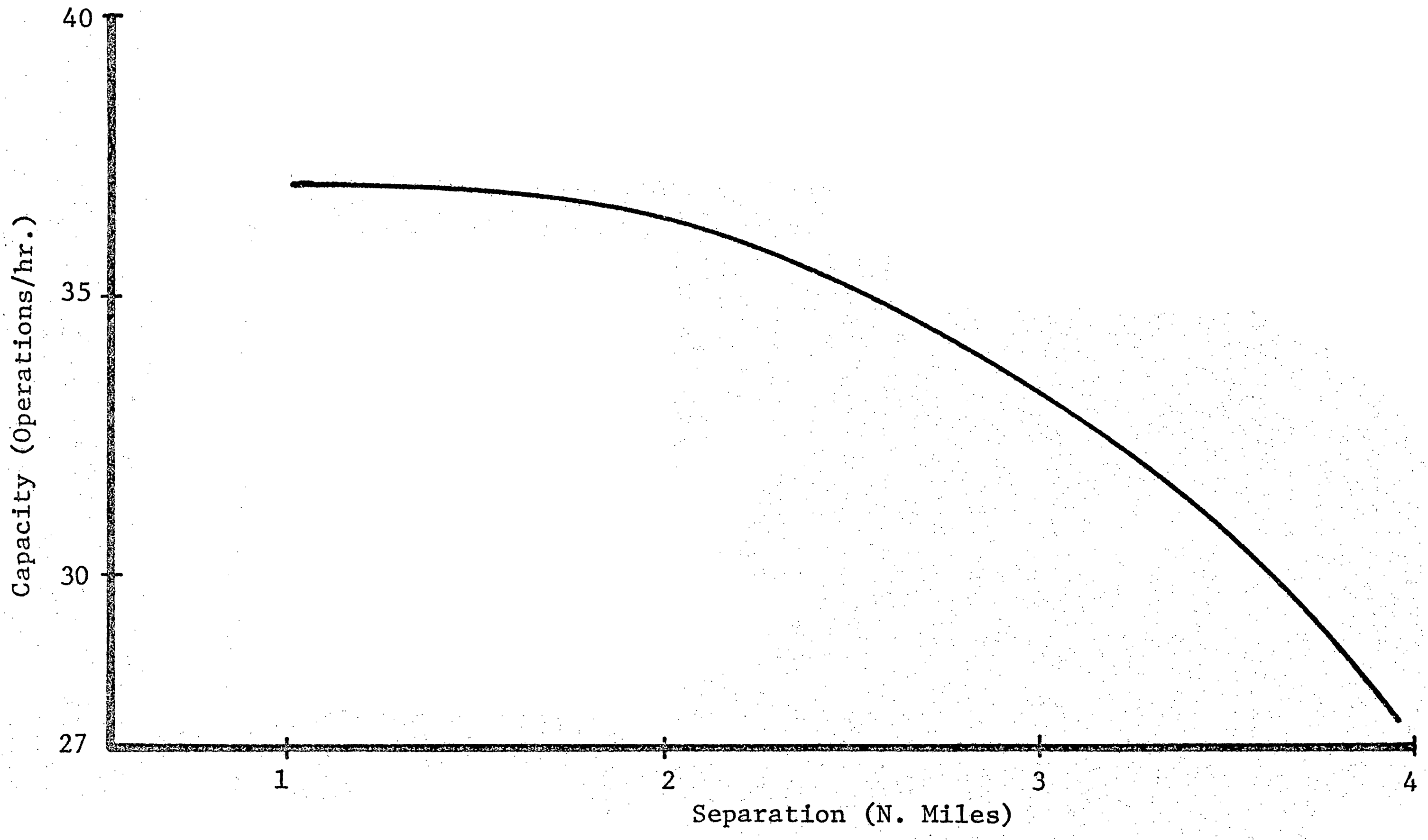


FIGURE 12: CAPACITY VS SEPARATION FOR LANDINGS AND TAKEOFFS COMBINED, MIX 3 (M3E)

easily seen by comparing Figures 13, 14, and 15 for arrivals only to Figures 10, 11, and 12 for arrivals and departures combined.

It should be stressed that any conclusions reached based on the simulation model are valid only for the population and range of input data considered. Extrapolation outside the range of values considered may lead to invalid conclusions.

Affect of random variation on capacity

A detailed discussion of the general validation of the simulation model is given in Appendix C; however, the results obtained from the first efforts at validation were considered significant and are presented here.

A topic of frequent discussion in runway capacity studies is the affect of reduced separation criteria on capacity. It is generally accepted as fact that as separation decreases, capacity will increase to a certain point, and then decrease. It was felt that if the model were correct, this result would be readily apparent if simulations were run for landings only, under varying separation criteria. Three homogenous approach velocity categories were chosen (160 knots, 125 knots, and 100 knots) and landings only were simulated for each category at separations ranging from 1 to 4 N. miles in increments of .1 N. mile. Next a mathematical model was developed for the case where there is no random variation. With this model it was possible to predict the theoretical percentage of waveoffs occurring at the decision node, and the resulting capacity. The theoretical results were then compared to the simulated results and may be seen graphically in Figures 13, 14, and 15. Development of the model for predicting percentage waveoffs and capacity is given in Appendix C along with tabular results.

In simulating the three approach categories with random variation it is expected that simulated capacity would, in general, be less than theoretical capacity as determined by the math model. This is due to the fact that the math model did not provide for waveoffs to occur at any place other than the decision node. In other words the theoretical results assume that the aircraft will all adhere to minimum separation criteria while the simulation model does not. The simulation model introduces error to the approach velocities and if minimum separation is not met, a waveoff may occur not only at the decision node, but prior to the decision node as well.

Not only does the comparison above serve to validate the model but it also illustrates the affects of random variation on capacity. As can be seen, capacity varies most about the separation criteria dividing 66 2/3% waveoffs from 50% waveoffs, and 50% waveoffs from 0% waveoffs. This is due to random variation. For example, even though 160 knot aircraft are assigned an intrail distance of 2.6 N. miles (the theoretical division between 50% and 0% waveoffs), random variation may cause the aircraft to have a greater or lesser intrail distance, thus increasing or decreasing capacity to a noticeable degree.

Of greater importance than simulator validation and illustration of the affects of random variation on capacity, these results show the

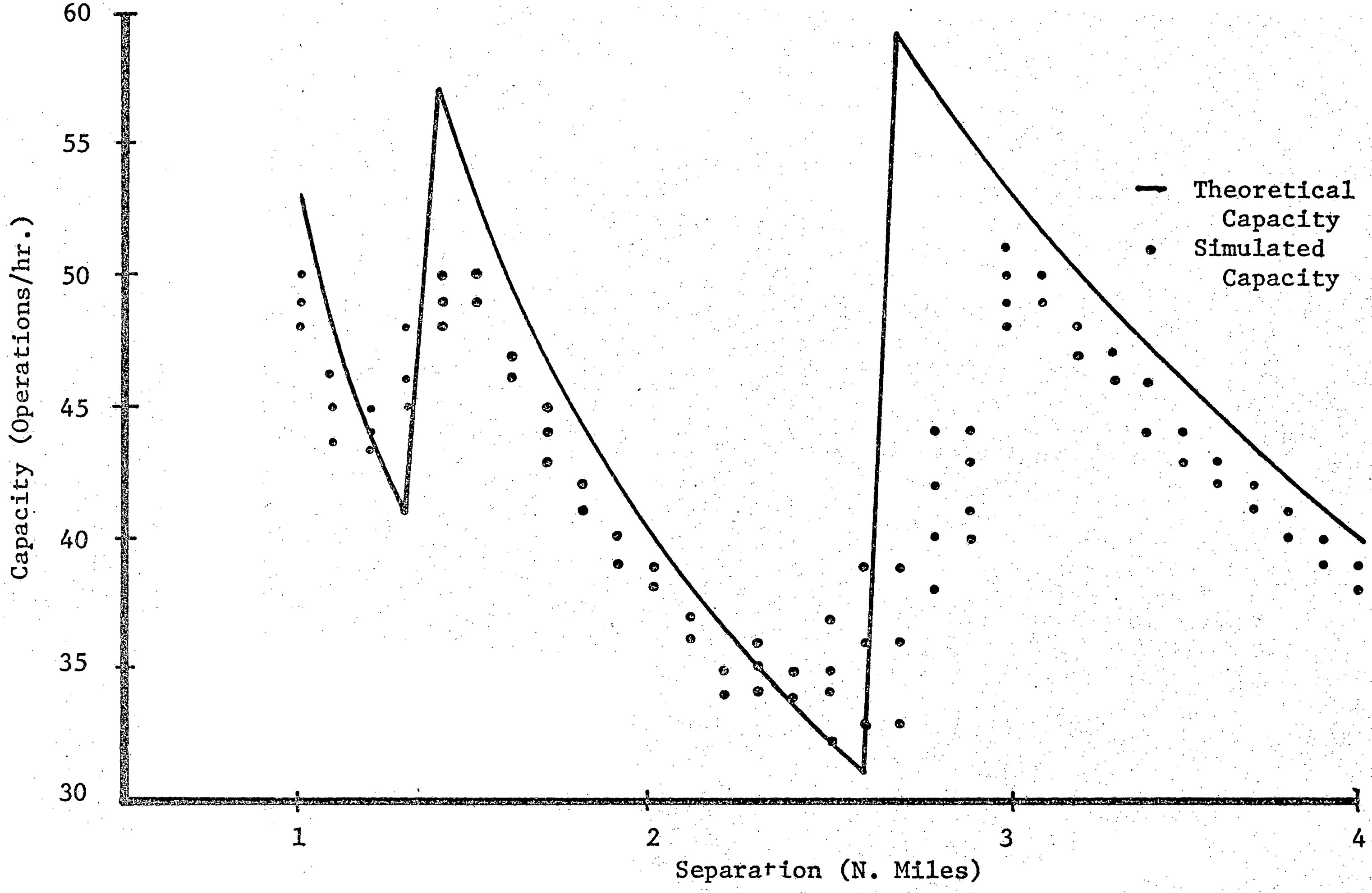


FIGURE 13: COMPARISON OF THEORETICAL CAPACITY AND SIMULATED CAPACITY (WITH RANDOM VARIATION) FOR LANDINGS ONLY, V = 160 KNOTS TDP = 800 FEET

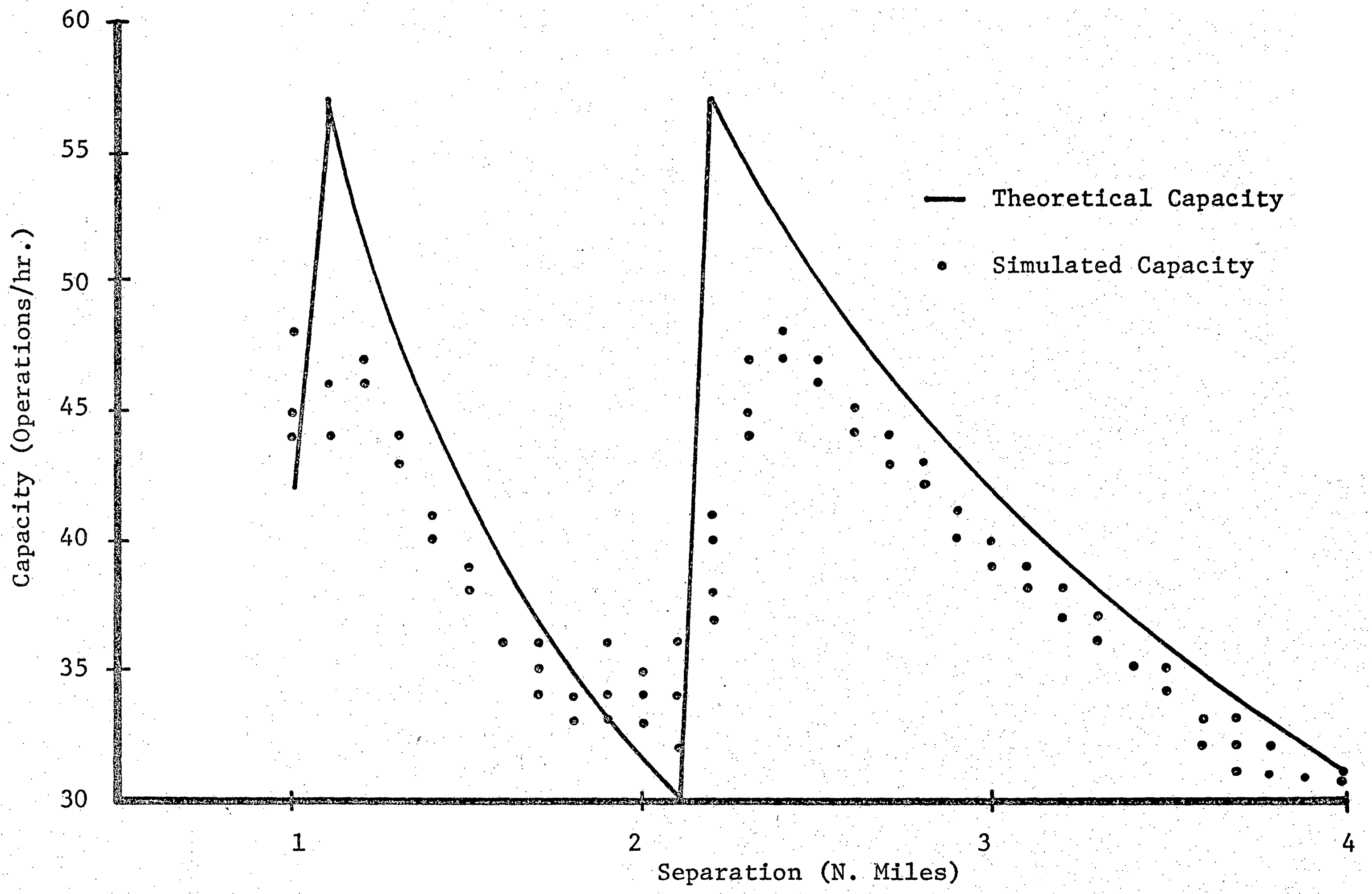


FIGURE 14: COMPARISON OF THEORETICAL CAPACITY AND SIMULATED CAPACITY (WITH RANDOM VARIATION) FOR LANDINGS ONLY, V = 125 KNOTS TDP = 1600 FEET

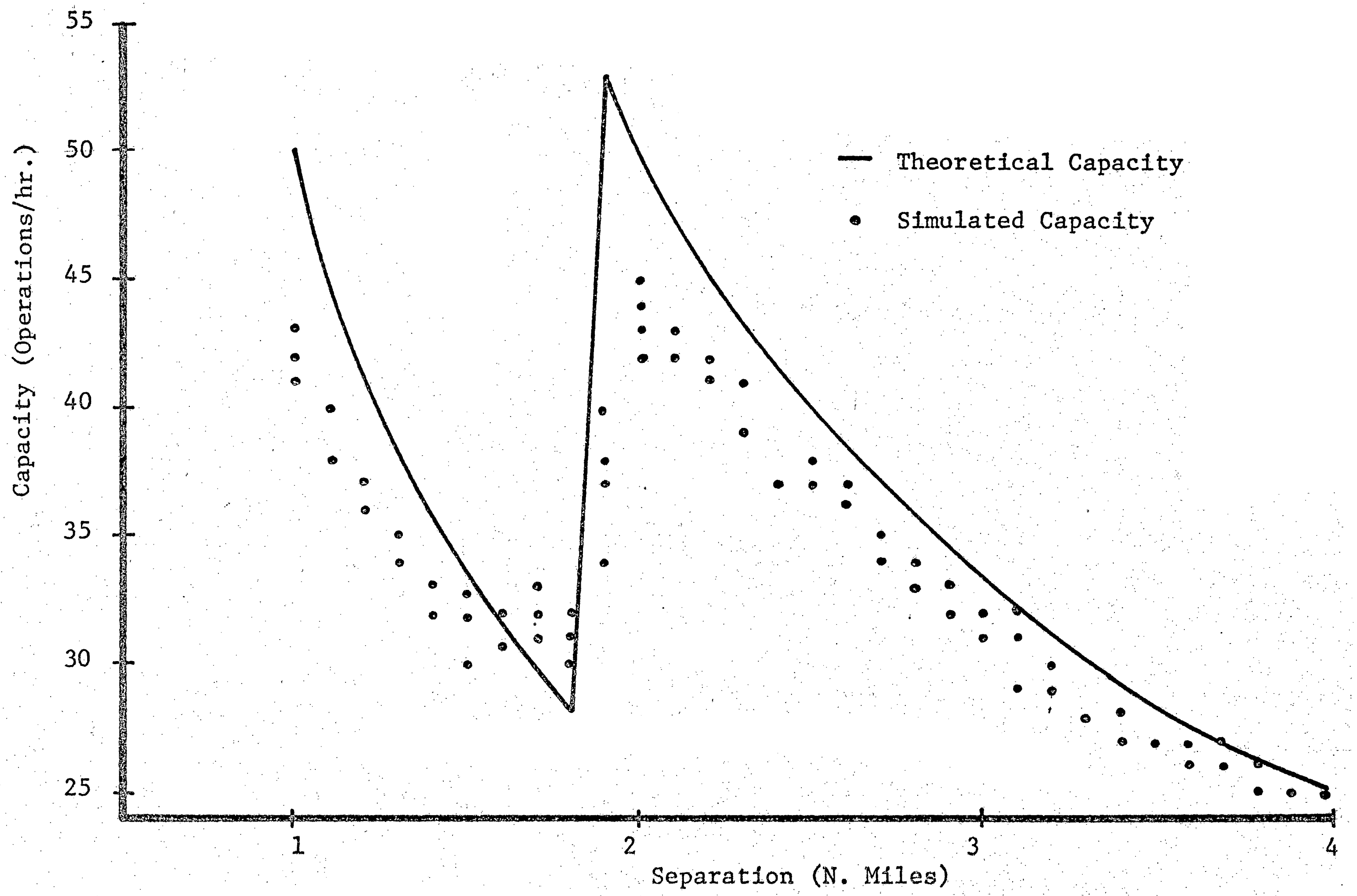


FIGURE 15: COMPARISON OF THEORETICAL CAPACITY AND SIMULATED CAPACITY (WITH RANDOM VARIATION)

FOR LANDINGS ONLY, V = 100 KNOTS TDP = 4000 FEET need for careful realization of the exact affects of separation on runway capacity. This is particularly true in investigating speed class sequencing techniques for present, as well as future, runway operations. For example, from Figure 13 it can be seen that by reducing separation by as little as .4 N. mile from the present 3 N. miles, runway capacity could actually be decreased by 41%. The simulation model developed provides one tool for investigating the affects of separation criteria on runway capacity in a stochastic environment.

CHAPTER VI. SUMMARY AND RECOMMENDATIONS

To furnish a tool with which to study runway capacity, a simulation model was developed for a single runway. The system modeled includes the runway and its associated approach and departure path airspace, including a holding stack located over the outer marker (for programming convenience only). In analyzing the system, the following ATC constraints, aircraft performance parameters, and assumptions were used:

ATC Constraints:

1. Minimum aircraft intrail separation on the landing approach path.
2. Minimum aircraft intrail separation on the departure path.
3. Only one aircraft on the active runway at a time.
4. Length of final approach path.

Aircraft Performance Parameters

1. Velocity at entry to active runway for a departing aircraft.

2. Runway acceleration rate for a departing aircraft.
3. Lift off velocity for a departing aircraft.
4. Climb acceleration rate for a departing aircraft.
5. Final approach and touchdown velocity for an arriving aircraft.
6. Touchdown point on the runway for an arriving aircraft.
7. Runway exit velocity for an arriving aircraft.
8. Runway deceleration rate for an arriving aircraft.

Assumptions

1. Runway acceleration rate is constant for a departing aircraft.
2. Climb acceleration rate is constant for a departing aircraft.
3. Runway acceleration rate and climb acceleration rate are considered equal over the regime for which a departing aircraft is considered.
4. Instrument flight rules apply to the entire system.
5. Operations are in a radar environment.
6. Approach velocity equals touchdown velocity for an arriving aircraft.
7. Runway deceleration rate is constant for an arriving aircraft.
8. Departing aircraft follow divergent paths.

Once the simulator was developed it was desired that it be validated. To check the performance of the simulator in a landings only mode (for a single aircraft class), a mathematical model was developed to predict the percentage of waveoffs occurring at the decision node for any given separation criteria. In comparing the results of the simulator and the math model, not only was the simulator validated, but significant observations were made as well. It was observed that if capacity were plotted against separation for various approach categories, there occur points of "discontinuity" separating regions of 0% waveoffs from 50% waveoffs and 50% waveoffs from 66 2/3% waveoffs. In particular it was observed that for 160 knot aircraft, a decrease in separation of as little as .4 N. mile from the present 3 N. miles could actually decrease capacity by 41% in a tightly packed approach environment.

As an example of how the simulator may be used to study runway capacity a regression analysis was performed on the simulator output in order to arrive at equations with which to predict runway capacity.

Before any analysis could take place it was necessary to know how much data should be collected through simulation to ensure statistically sound results. The combinations of levels of the independent variables were selected at random and the number of replications of the simulation at each combination of levels was defined in accordance with prescribed α and β values. A complete statistical analysis was performed after data collection to ensure that the desired power of the regression equations was present.

Three mixes of arriving and departing aircraft were simulated--fast, medium, and slow. For each case six independent variables were considered:

1) Separation, ranging from 1 to 4 N. miles.

2) Mean approach velocity for class 1 aircraft, ranging from 45 to 65 knots.
3) Mean approach velocity for class 2 aircraft, ranging from 65-85 knots.
4) Mean approach velocity for class 3 aircraft, ranging from 85-115 knots.
5) Mean approach velocity for class 4 aircraft, ranging from 115-135 knots.
6) Mean approach velocity for class 5 aircraft, ranging from 140-165 knots.

A quadratic regression equation was hypothesized for each of the three mixes, and was verified using the appropriate F test. In addition, it was ascertained that a significant amount of variation was, in fact, explained by each equation. Using the regression equation for each of the three aircraft mixes all variables except separation were held constant. Separation was varied from 1 to 4 N. miles in increments of .1 N. mile and the corresponding capacity noted. Two observations were made:

1) By decreasing the number of slower aircraft in the total population, capacity can be significantly increased.
2) There are considerable affects on runway capacity caused by the interaction of arriving and departing aircraft.

Recommendations for Future Research

Any information obtained from a simulation is only as good as the source from which it came; that is, the more true to life the simulator, the better the results. In attempting to make the simulator developed more realistic several additions would help enormously:

1) Incorporate different sequencing algorithms into the simulator. It would be interesting to investigate priority rules other than first come-first served. These could be based on total delay time, fuel costs, number of passengers, etc.
2) Extend the model to include various runway configurations including parallel and intersecting runways.
3) Add factors such as weather, and wake turbulence.
4) If total airport simulation were desired, taxiways could be added along with passenger and baggage loading and unloading.

As a further recommendation, it would be interesting to perform a sensitivity analysis on the simulator with respect to the various distritutions assumed throughout the model.

BIBLIOGRAPHY

1. Airborne Instruments Laboratory, "Runway Characteristics and Performance of Selected Propeller-Driven Aircraft in Routine Operation," Airborne Instruments Laboratory, 1960.

2. __________, "Runway Characteristics and Performance of Jet Transports in Routine Operation, Volume I," Airborne Instruments Laboratory, 1960.

3. __________, "Airport Capacity Handbook," Airborne Instruments Laboratory, 1969.

4. __________, "Airport Capacity--A Handbook for Analyzing Airport Designs to Determine Practical Movement Rates and Aircraft Operating Costs," Airborne Instruments Laboratory, 1963.

5. __________, "Factors Affecting Airport Capacity and their Applicability to Simulation," Volumes I, II, III, and IV., Airborne Instruments Laboratory, 1959.

6. Baran, G., Benezra, J. N., and Blumenthal, R. W., "Runway Capacity Augmentation Study," The Boeing Company.

7. Cornell Aeronautical Laboratory, Inc., "Investigation of Response and Maneuver Times for use in Air Traffic Control Simulators," Cornell Aeronautical Laboratory, Inc., 1958.

8. Department of Transportation Federal Aviation Administration, "Airman's Information Manual," Washington, D. C.: U. S. Gov't. Printing Office, 1965.

9. __________, "Runway Length Requirements for Airport Design," Washington, D. C.: U. S. Gov't. Printing Office, 1965.

10. __________, "Terminal Air Traffic Control," Washington, D. C.: U. S. Gov't. Printing Office, 1971.

11. __________, "United States Standard for Terminal Instrument Procedures," Washington, D. C.: U. S. Gov't. Printing Office, 1970.

12. Goldman, A. J., "Analysis of a Capacity Concept for Runway and Final Approach Path Airspace," Journal of the Institute of Navigation, Vol. 17, No. 3, Fall, 1970.

13. Gray Bill, F. A., An Introduction to Linear Statistical Models, Vol. I, McGraw Hill Book Company, Inc., New York, 1961.

14. Hooton, E. N., et al., "Operational Evaluation of Airport Runway Design and Capacity (A Study of Methods and Techniques)," Airborne Instruments Laboratory, 1963.

15. Knox, Charles, "An Investigation of the Air Traffic Control Procedures and Aircraft Performance Characteristics Around the Runway Environment," Incomplete Ph.D. Dissertation, Kansas State University, 1971.

16. Lundquist, Gustav E, "The FAA Program for Increased Airport and Airway Capacity," AIAA paper No. 70-1315.

17. Morse, Phillip M., "Stochastic Properties of Waiting Lines," Operations Research, 1955, Vol. 3 #3.

18. Oliver, Robert M, "Distribution of Gaps and Blocks in a Traffic Stream" Operations Research, 1962, Vol. #10, pp. 197-217.

19. Ostle, Bernard, Statistics in Research, Iowa State University Press, 1963.

20. Pestalozzi, Gerald, "Priority Rules for Runway Use," Institute of Transportation and Traffic Engineering, University of California, 1969.

21. Porter, Lynn W., "On Optimal Scheduling and Holding Strategies for the Air Traffic Control Problem," Massachusetts Institute of Technology, 1969.

22. Schmidt, J. W. and Taylor, R. E., Simulation and Analysis of Industrial Systems, Richard D. Irwin, Inc., Homewood, Illinois, 1970.

23. Sullivan, Thomas M., "Design and Development of a New Air Terminal," U. S. Department of Commerce, National Bureau of Standards.

24. Warskow, M., et al, "Factors affecting Airport Capacity and their Applicability to Simulation," Airborne Instruments Laboratory, 1959.

APPENDIX A: OUTLINE OF TOPICS COVERED IN THE AIRMAN'S INFORMATION MANUAL, PART I

APPENDIX A: OUTLINE OF TOPICS COVERED IN THE AIRMAN'S INFORMATION MANUAL, PART I

I. General Aeronautical Terms

II. Navigation Aids
    A. General
    B. VHF omnidirectional range (VOR)
    C. Distance measuring equipment
    D. Marker beacons
    E. Instrument landing systems
    F. Radar
    G. Surveillance radar
    H. Precision radar
    I. Aeronautical light beacons
    J. Instrument approach light systems
    K. Runway marking

III. The Airspace
    A. General
    B. VFR requirements
    C. IFR requirements
    D. Terminal control area

IV. Air Traffic Control
    A. Services available to pilots
        1. Control towers
        2. Flight service stations

B. Airport operations

C. ATC clearances/separations

D. Departures--IFR

   1. Instrument departures

   2. Departure control

E. Enroute--IFR

   1. Airway/route systems

   2. Holding

F. Arrival--IFR

   1. Approach control

   2. Instrument approach procedures

   3. Speed adjustment of arriving aircraft

   4. Landing priority

   5. Missed approach

APPENDIX B: THE CENTRAL AND NON-CENTRAL F DISTRIBUTIONS

APPENDIX B: THE CENTRAL AND NON-CENTRAL F DISTRIBUTIONS

The density function of the central F distribution (with mean zero) with ν_1 and ν_2 degrees of freedom is given by

$$f(F_o) = \frac{\Gamma\left(\frac{\nu_1 + \nu_2}{2}\right)}{\Gamma\left(\frac{\nu_1}{2}\right)\Gamma\left(\frac{\nu_2}{2}\right)} \left(\frac{\nu_1}{\nu_2}\right)^{\frac{\nu_1}{2}} \frac{F_o^{\frac{\nu_1 - 2}{2}}}{\left(1 + \frac{\nu_1 F_o}{2}\right)^{\frac{\nu_1 + \nu_2}{2}}}$$

The density function of the non-central F distribution (with non-zero mean and parameter λ) with ν_1 and ν_2 degrees of freedom is given by

$$f(F_\lambda) = e^{-\lambda} \sum_{j=0}^{\infty} \frac{\lambda^j}{j!} \left(\frac{\nu_1}{\nu_2}\right)^{\frac{\nu_1 + 2j}{2}} \frac{\Gamma\left(\frac{\nu_1 + \nu_2 + 2j}{2}\right) F_\lambda^{\frac{\nu_1 + 2j - 2}{2}}}{\Gamma\left(\frac{\nu_2}{2}\right)\left(\frac{\nu_1 + 2j}{2}\right)\left(1 + \frac{\nu_1 F_\lambda}{\nu_2}\right)^{\frac{{}_1 + {}_2 + 2j}{2}}}$$

APPENDIX C: VALIDATION OF THE MODEL

APPENDIX C: VALIDATION OF THE MODEL

In validating any simulation model, two questions must be considered:

1) Is the simulator performing as desired?
2) Does the desired operation of the simulator match the operation of its real life counterpart?

To answer the first question several simulations were performed in which all random variables except interarrival time were removed and replaced by constants. The program was then made to print out the node by node progress of each aircraft. In this manner such statistics as time from node to node, release time from the holding stack, and runway occupancy time could be checked against their expected values. In all cases good agreement was found. An interesting example of model validation, already discussed in Chapter V, is the case in which theoretical capacity is compared to simulated capacity of landings only, for single aircraft classes. Before explaining how theoretical capacity was arrived at, the rule (in the simulator) used to determine waveoffs at the decision node will be reviewed.

In the landings only case, when an aircraft arrives at the decision node, the simulator looks at the approach path ahead to see if any other aircraft are present. If an aircraft is found, the program calculates the estimated node at which the aircraft should exit. The node immediately preceeding this exit node is then considered the "safe" point (this safe point is never further than 1500 feet from the estimated exit node due to the spacing of the nodes). If the last position of the aircraft found ahead of the decision node equals or exceeds the "safe" point then the aircraft at the decision node may proceed, otherwise it must execute a missed approach.

The steps used to determine theoretical capacity and theoretical percentage of waveoffs occurring at the decision node are listed below. All calculations were made with equations 4.5, 4.6, and 4.7 and the runway configuration previously given.

1. Divide the length of the final approach path by separation between aircraft. Round this number down and add 1 to find the maximum number of aircraft on final approach at any given time.
2. Allow the maximum number of aircraft calculated above to occupy the final approach path, the last being placed at the entry gate.
3. Take the aircraft closest to the runway threshold and calculate the time that will elapse until it reaches its "safe" point.
4. For each aircraft following the lead aircraft, calculate the time that will elapse until it reaches the decision node. Any aircraft whose arrival time at the decision node is less than the arrival time of the lead aircraft at its "safe" point, must waveoff.

5. Define a "block" of aircraft to be the number of aircraft beginning with the lead aircraft, up to and including the last aircraft to waveoff. The percentage of aircraft in the "block" that wavesoff is the same as the percentage of aircraft that will waveoff during any time period. This is because following the last plane in a "block" the entire cycle of landings and waveoffs is repeated identically.
6. Calculate the time separation between aircraft. Divide the total time period considered by the time separation between aircraft to calculate the total releases. Theoretical capacity equals total releases multiplied by 1—the percentage of waveoffs.

An example will clear up any misconceptions concerning the above rules. Consider 160 knot aircraft, separated by 1.3 N. miles and touching down 800 feet past the runway threshold.

1. $\frac{\text{Length of final approach}}{\text{Separation}} = \frac{5}{1.3} = 3.8$

   This implies that there can be at most 3 + 1 = 4 aircraft on final approach at any one time.
2. See Figure C-1 for the spacing considered between aircraft.
3. Time until aircraft #1 reaches its "safe" point

   = time until it reaches THN from original position +

   time until it reaches TPD from THN +

   time until safe point is reached from TDP

   = 24.750 + 2.962 + 33.795

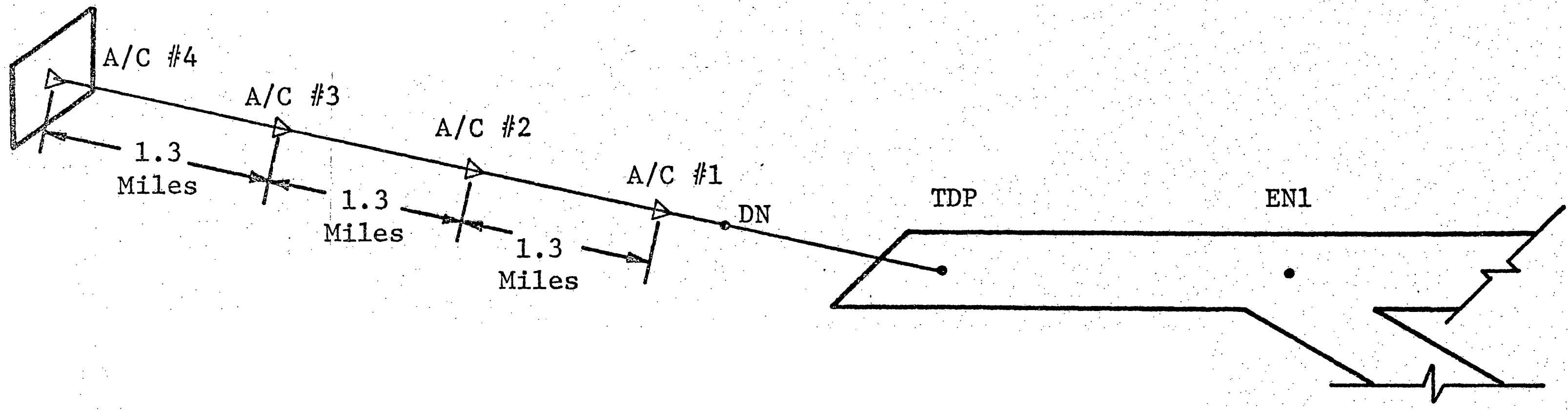


FIGURE C-1: MAXIMUM THEORETICAL NUMBER OF AIRCRAFT ON FINAL AT ONCE, V = 160 KNOTS TDP = 800 FEET

= 61.507 seconds

4. Time for aircraft #2 to reach

   DN from its original position = 31.5 seconds

   Time for aircraft #3 to reach

   DN from its original position = 60.75 seconds.

   Time for aircraft #4 to reach

   DN from entry gate = 90 seconds

   It is seen that aircraft #2 and #3 must wave off.

5. The number of aircraft in the "block" is 3. The percentage of aircraft in the block and in any time period that will waveoff is 2/3 · 100 = 66 2/3%.

6. Time separation between aircraft = 29.25 seconds.

   Total number of releases in 1 hour = $\frac{3600}{29.25}$

   Theoretical capacity for 1 hour = $\frac{3600}{29.25} \cdot (1 - .667) = 41.$

The analysis above was programmed and carried out for three example approach categories:

1) 160 knots, touch down point = 800 feet
2) 125 knots, touch down point = 1600 feet
3) 100 knots, touch down point = 4000 feet.

When the theoretical capacities were obtained the simulation was run under the same conditions, both with and without random variation, and the results compared to the theoretical capacities. Tabular comparisons are given in Tables C-I, C-II, and C-III. Graphical compari-

TABLE C-I: CAPACITY (IN OPERATIONS PER HOUR) FOR 160 KNOT APPROACH CATEGORY, TDP = 800 FEET

Separation (N. miles)	% Waveoffs at DN	Capacity (Theoretical)	Capacity over 4 replications (simulated, with random variation)	Capacity (simulated, without random variation*)
1.0	66 2/3	53	49, 48, 48, 50	51
1.1	66 2/3	48	45, 46, 45, 44	46
1.2	66 2/3	44	44, 45, 46, 44	44
1.3	66 2/3	41	45, 45, 46, 48	41
1.4	50	57	50, 48, 49, 50	55
1.5	50	53	50, 50, 50, 49	50
1.6	50	50	47, 47, 46, 46	46
1.7	50	47	44, 44, 43, 45	46
1.8	50	44	42, 41, 42, 42	44
1.9	50	42	39, 40, 40, 40	41
2.0	50	40	38, 38, 38, 39	38
2.1	50	38	36, 37, 37, 36	36
2.2	50	36	35, 35, 34, 35	36
2.3	50	35	36, 35, 34, 34	34
2.4	50	33	35, 34, 34, 35	33
2.5	50	32	32, 37, 35, 34	31

*Except in interarrival times

TABLE C-I Continued.

Separation (N. miles)	% Waveoffs at DN	Capacity (Theoretical)	Capacity over 4 replications (simulated, with random variation)	Capacity (simulated, without random variation*)
2.6	50	31	33, 39, 36, 36	30
2.7	0	59	42, 38, 40, 44	55
2.8	0	57	43, 44, 41, 40	55
2.9	0	55	50, 48, 49, 52	55
3.0	0	53	50, 50, 51, 51	51
3.1	0	52	49, 49, 49, 50	51
3.2	0	50	48, 47, 48, 48	46
3.3	0	48	46, 46, 47, 47	47
3.4	0	47	44, 46, 46, 46	47
3.5	0	46	43, 43, 44, 44	44
3.6	0	44	42, 42, 43, 43	44
3.7	0	43	41, 41, 42, 41	41
3.8	0	42	41, 41, 40, 41	41
3.9	0	41	40, 40, 40. 39	41
4.0	0	40	39, 38, 38, 49	38

*Except in interarrival times

TABLE C-II: CAPACITY (IN OPERATIONS PER HOUR) FOR 125 KNOT APPROACH CATEGORY, TDP = 1600 FEET

Separation (N. miles)	% Waveoffs at DN	Capacity (Theoretical)	Capacity over 4 replications (simulated, with random variation)	Capacity (simulated, without random variation*)
1.0	66 2/3	42	44, 44, 45, 45	41
1.1	50	57	46, 46, 48, 44	54
1.2	50	52	47, 47, 46, 47	48
1.3	50	48	43, 43, 44, 44	45
1.4	50	45	41, 40, 41, 40	45
1.5	50	42	39, 39, 39, 38	41
1.6	50	39	36, 36, 36, 36	38
1.7	50	37	35, 35, 36, 34	35
1.8	50	35	34, 33, 33, 34	33
1.9	50	33	34, 36, 33, 34	33
2.0	50	31	33, 34, 35, 33	31
2.1	50	30	36, 34, 32, 36	29
2.2	0	57	40, 37, 41, 38	54
2.3	0	54	44, 47, 44, 35	54
2.4	0	52	48, 48, 47, 48	48
2.5	0	50	47, 46, 47, 47	49

*Except in interarrival times

TABLE C-II Continued.

Separation (N. miles)	% Waveoffs at DN	Capacity (Theoretical)	Capacity over 4 replications (simulated, with random variation)	Capacity (simulated, without random variation*)
2.6	0	48	45, 44, 44, 44	45
2.7	0	46	43, 43, 44, 44	44
2.8	0	45	42, 42, 42, 43	44
2.9	0	43	41, 41, 41, 40	41
3.0	0	42	40, 40, 40, 39	41
3.1	0	40	38, 38, 38, 39	38
3.2	0	39	37, 38, 37, 37	38
3.3	0	38	37, 36, 36, 36	37
3.4	0	37	35, 35, 35, 35	35
3.5	0	36	34, 34, 34, 35	35
3.6	0	35	32, 33, 33, 33	33
3.7	0	34	33, 34, 32, 34	33
3.8	0	33	32, 31, 31, 31	33
3.9	0	32	30, 30, 31, 29	31
4.0	0	31	28, 29, 28, 28	31

*Except in interarrival times

TABLE C-III: CAPACITY (IN OPERATIONS PER HOUR) FOR 100 KNOT APPROACH CATEGORY, TDP = 4000 FEET

Separation (N. miles)	% Waveoffs at DN	Capacity (Theoretical)	Capacity over 4 replications (simulated, with random variation)	Capacity (simulated, without random variation*)
1.0	50	50	41, 43, 42, 43	49
1.1	50	45	40, 40, 38, 40	44
1.2	50	42	36, 36, 37, 37	40
1.3	50	38	34, 34, 34, 35	36
1.4	50	36	33, 33, 32, 33	36
1.5	50	33	33, 32, 30, 33	33
1.6	50	31	32, 32, 32, 31	31
1.7	50	29	30, 33, 31, 32	28
1.8	50	28	31, 30, 30, 32	26
1.9	0	53	38, 37, 40, 34	49
2.0	0	50	45, 44, 43, 42	49
2.1	0	48	43, 42, 43, 43	44
2.2	0	45	42, 41, 41, 41	44
2.3	0	43	39, 41, 39, 39	43
2.4	0	42	38, 38, 38, 38	39
2.5	0	40	38, 37, 37, 37	39

*Except in interarrival times

TABLE C-III Continued.

Separation (N. miles)	% Waveoffs at DN	Capacity (Theoretical)	Capacity over 4 replications (simulated, with random variation)	Capacity (simulated, without random variation*)
2.6	0	38	36, 36, 36, 37	36
2.7	0	37	35, 35, 35, 34	36
2.8	0	36	33, 34, 33, 34	36
2.9	0	34	32, 33, 33, 33	33
3.0	0	33	32, 32, 31, 32	33
3.1	0	32	29, 31, 31, 32	31
3.2	0	31	30, 30, 29, 30	31
3.3	0	30	29, 29, 29, 29	30
3.4	0	29	29, 28, 28, 29	28
3.5	0	29	27, 27, 27, 27	28
3.6	0	28	27, 26, 26, 26	26
3.7	0	27	26, 27, 26, 26	26
3.8	0	26	25, 25, 26, 25	26
3.9	0	26	24, 25, 25, 26	25
4.0	0	25	23, 24, 24, 23	24

*Except in interarrival times

sons may be seen in Figures 5, 6, 7, 13, 14, and 15. A detailed discussion of the results of the above comparisons may be found in Chapter IV.

In answering the second question posed at the beginning of this section, many different mixes were simulated, with the progress of each aircraft being checked at each node (by printing out the storage arrays). The performance of the simulator was found to be as desired and as close to real life as the constraints given in Chapter III permit.

Out of curiosity, two comparisons were made of simulated capacity values to those found in the Airport Capacity Handbook, those results may be found in Appendix D.

APPENDIX D: COMPARISON OF _AIRPORT CAPACITY HANDBOOK_ VALUES TO SIMULATED VALUES OF CAPACITY

APPENDIX D: COMPARISON OF AIRPORT CAPACITY HANDBOOK VALUES TO SIMULATED VALUES OF CAPACITY

For those familiar with the Airport Capacity Handbook (3), a comparison of Handbook capacity values and simulator values was made for two example mixes. All input and data used was the same as that given in Appendix G. Each mix was simulated for 1 3/4 hour with operations being counted only during the last hour. This assured the system of being "loaded" with aircraft.

Mix 1

Information for analysis:

1) PHOCAP A airport.
2) SINGLE RUNWAY, instrument airport weather.
3) MIXED OPERATIONS, no touch and go traffic.
4) RUNWAY LENGTH--9500 feet.
5) ANGLED TURNOFFS at 5000, 6500, 8000, and 9500 feet.
6) NORMAL AIRSPACE.
7) RUNWAY ALTITUDE--250 feet above mean sea level

Class	Population %	Runway Length Correction factor	Corrected length	Exit Rating	R_R
E	0	1	9500 feet	5	59 sec.
D	12.82	1	9500 feet	5	55 sec.
C	38.46	1	9500 feet	5	61 sec.
B	25.64	1	9500 feet	4	59 sec.
A	23.08	1	9500 feet	3	54 sec.

final $R_R = .1282(55) + .3846(61) + .2564(59) + .2308(54) = 58.10$ seconds. For the simulation there were 27 landings and 12 take offs with the above percentages of each.

$$\lambda_L = 27 \quad \lambda_T = 12$$

$$\text{Ratio} = \frac{27}{12} = 2.25$$

Calculating accurate IAW PHOCAP with RADAR CONTROL and an instrument landing system:

HDC = 12 from figure 3-9 (<u>Handbook</u>)

HAC = 35 from figure 3-9 (<u>Handbook</u>)

arrival demand = 12(2.25) = 27

PHOCAP = 12 + 27 = 39

The simulated capacity was 39.

Mix 2

Information for analysis:

1) PHOCAP A airport.
2) SINGLE RUNWAY, instrument airport weather.
3) MIXED OPERATIONS, no touch and go traffic.
4) RUNWAY LENGTH--9500 feet.
5) ANGLED TURNOFFS at 5000, 6500, 8000, and 9500 feet.
6) NORMAL AIRSPACE.
7) RUNWAY ALTITUDE--250 feet above mean sea level.

Population Class	%	Runway length Correction factor	Corrected length	Exit Rating	R_R
E	5.26	1	9500 feet	5	59 sec.
D	28.95	1	9500 feet	5	55 sec.
C	34.21	1	9500 feet	5	61 sec.
B	26.32	1	9500 feet	4	59 sec.
A	5.26	1	9500 feet	3	54 sec.

final $R_R = .0526(59) + .2895(55) + .3421(61) + .2632(59) + .0526(54) =$ 58.36. For the simulation there were 24 landings and 14 take offs

$$\lambda_L = 24 \qquad \lambda_T = 14$$

$$\text{Ratio} = \frac{24}{14} = 1.7143$$

Calculating accurate IAW PHOCAP with radar control and an instrument landing system:

HOC = 16 from figure 3-9 (Handbook)

HAC = 38 from figure 3-14 (Handbook)

arrival demand = 16(1.7143) ≐ 27.4288

PHOCAP = 16 + 27 = 43

Simulated capacity was 38.

APPENDIX E: DESCRIPTION OF THE SUBROUTINES

APPENDIX E: DESCRIPTION OF THE SUBROUTINES

As an aid to using the computer simulation developed in this study a brief description of each subroutine has been included. These descriptions, along with Chapter IV and comments interspersed throughout the program itself should allow implementation of the program with minimum effort.

Main Program

The main program reads in all data and prints it out in easily read tabular form, directly before the output. All storage arrays are initialized and the first event aircraft selected prior to statement #15. There are no computations performed whatsoever in the main program; its sole purpose is to make all decisions concerning system operation, and call the appropriate subroutine to perform necessary calculations.

Subroutine REPLCE

Subroutine REPLCE is called when an aircraft has been drawn from the WAIT array and entered into the APROCH array. REPLCE will generate a new aircraft of the type that was just removed from WAIT, thus assuring a constant supply of all possible types of next event aircraft.

Subroutine ALPHA

The purpose of subroutine ALPHA is to calculate first a time dependent mean arrival rate and from this an interarrival time for both arriving and departing aircraft.

Function RAND

Function RAND generates uniformly distributed random numbers on the interval from zero to one. The reader is referred to Schmidt and Taylor (22) for details concerning the random number generator.

Subroutine TIMNOD

TIMNOD calculates the time that every aircraft in the APROCH array will reach the next preceding node. The aircraft with the minimum arrival time is designated the present event aircraft and control is returned to the main program.

Subroutine VLOCTY(I)

Subroutine VLOCTY calculates a normally destributed velocity assignment for a class I aircraft.

Subroutine RWYCHK

Subroutine RWYCHK is called when a departing aircraft has been assured that no aircraft are within two miles of the threshold. RWYCHK then checks to make sure no aircraft are on the active runway. If the runway is clear a switch will be set enabling the departing aircraft to proceed. If the switch is not set, the departure will be delayed until the runway is clear.

Subroutine TOINFO

Subroutine TOINFO generates normally distributed take off parameters. Generated are: 1) runway entrance velocity, 2) take off velocity, and 3) runway acceleration rate.

Subroutine VVELTO

VVELTO checks to see if take off velocity has been achieved for a departing aircraft, if so it shecks to see if separation criteria has been met. When both lift off and separation criteria have been satisfied, NCODE is set equal to 4, indicating that the departing aircraft has cleared the system.

Subroutine DECNOD

DECNOD is called when an aircraft reaches the decision node. Its purpose is to check the runway for aircraft. If an aircraft is found, DECNOD calculates a "safe" point as described in Appendix C. If the safe point has been equaled or exceeded by the aircraft on the runway, the aircraft at the dicision node is allowed to continue its approach. Otherwise a missed approach is executed.

Subroutine TDPT

Subroutine TDPT generates a normally distributed runway touchdown point for a landing aircraft.

Subroutine TDINFO

TDINFO generates normally distributed deceleration rates for landing aircraft.

Subroutine UPDATE

All storage arrays are updated to their present time conditions in UPDATE. In addition, statistics concerning number of take offs, landings, aborted take offs, and waveoffs are collected. Within UPDATE information is collected with which to calculate the variance of operations over any time period desired.

Subroutine STKDEL

As first row entries in STACK enter the APROCH array, STKDEL deletes the old row and moves all other rows up one, thus conserving storage space.

Subroutine APRDEL

As first row entries in APROCH exit the system, APRDEL deletes the first row entries and moves all succeeding rows up one, thus reusing the same storage space over and over.

Subroutine VELEN1

VELEN1 compares the velocity of a landing aircraft at an exit node to the maximum safe exit velocity to determine if an exit can be made.

Subroutine AMCAC

AMCAC selects a master clock aircraft from APROCH. This may be when a take off, landing, or aborted landing occurs.

Subroutine REVUE

REVUE checks arrival times in the WAIT array to see if any system arrivals have occurred since the last system revue. If any arrivals have occurred, REVUE will call STKREG and REPLCE and check again for arrivals. In this manner the possibility of two arrivals in a row, of the same class, is not excluded.

Subroutine APRREG

Subroutine APRREG registers an arriving aircraft and its parameters in the APROCH array.

Subroutine STKREG

Subroutine STKREG registers an aircraft and its parameters in the STACK array.

Subroutine APRROW

When APRROW is called, it searches the rows of APROCH to determine the row number of the present event aircraft.

Subroutine STKROW

When STKROW is called, it searches the rows of STACK to determine the row number of the particular aircraft in question.

Subroutine SEQ

Subroutine SEQ sequences aircraft onto the runway according to whatever rules the user chooses to use. This program uses a first come-first served priority rule. When SEQ determines the next aircraft eligible for release to the approach path, it determines the spacing required between the releasing aircraft and the one immediately preceeding. Based on this spacing, SEQ calculates an optimal release time, by which it can be determined whether or not the aircraft may be released.

Subroutine THCHK

When a departing aircraft wishes to enter onto the active runway, it must first know if any aircraft are within 2 miles of the threshold. If any aircraft are within 2 miles the departing aircraft must delay its departure. To check for this condition is the function of THCHK.

Subroutine CNTRL1

Realizing that a controller will not be entirely consistent in scheduling aircraft, CNTRL1 attempts to introduce some variability into the scheduling process. This is done by allowing the "aimed at" separation to be a mean value only, with some variation allowed.

Subroutine ERROR

Subroutine ERROR, when called, checks to see if minimum separation criteria are being observed, and if not initiates a missed approach.

Subroutine VELCOR

VELCOR generates a correction velocity based on the deviation of present aircraft velocity from target velocity.

Subroutine OUTPUT

When called, subroutine OUTPUT writes out the time and type of operation having just occurred, the aircraft class and the total aircraft present in the system at that time (see Appendix F for sample output).

Subroutine SUMARY

At the end of the simulation, SUMARY will print out a summary table of all events that occurred. Also contained in the summary table will be the variance of operations, calculated over whatever time period was specified (see Appendix F for a sample summary table).

APPENDIX F: SAMPLE OUTPUT

APPENDIX F: SAMPLE OUTPUT

As previously explained, before printing out the type operation, time, aircraft class, and aircraft in the system all input data is listed in tabular form. This output, including the simulation summary, is shown in Table F-I as it is printed by the computer.

TABLE F-I: SAMPLE OUTPUT

A/C Class

	1	2	3	4	5
ACCMU	5.0	5.0	5.0	5.0	5.0
ACCSD	1.0	1.0	1.0	1.0	1.0
ALLDMN	5.0	8.0	9.0	8.0	5.0
ALTOMN	1.0	2.0	4.0	8.0	5.0
DCELMU	1.8	2.4	3.1	5.0	6.0
DCELSD	0.09	0.05	0.02	0.01	0.01
EHRMAX	3.00	3.00	2.00	2.00	2.00
EHRMIN	2.00	2.00	1.00	1.00	1.00
EXVLMX	30.0	25.0	20.0	20.0	20.0
PERLD	5.0	9.0	23.0	28.0	13.0
PERTO	3.0	3.0	12.0	12.0	3.0
PMRLD	5.0	10.0	12.0	20.0	15.0
PMRTO	5.0	10.0	12.0	20.0	15.0
PTELD	61200.00	61200.00	61200.00	61200.00	61200.00
PTETO	61200.00	61200.00	61200.00	61200.00	61200.00
PTMLD	32400.00	32400.00	32400.00	28800.00	32400.00
PTMTO	32400.00	32400.00	32400.00	28800.00	32400.00
RWYLDT	30.0	35.0	45.0	45.0	45.0

TABLE F-I Continued.

A/C Class

RWYTOT	32.0	32.0	37.0	46.0	50.0
SIGEL	7200.00	7200.00	7200.00	7200.00	7200.00
SIGETO	7200.00	7200.00	7200.00	7200.00	7200.00
SIGML	7200.00	7200.00	7200.00	7200.00	7200.00
SIGMTO	7200.00	7200.00	7200.00	7200.00	7200.00
TDDMU	2500.00	1800.00	800.00	800.00	800.00
TDDSD	100.0	100.0	50.0	50.0	50.0
VAPSD	5.0	5.0	5.0	5.0	5.0
VENTMU	8.0	8.0	8.0	8.0	8.0
VENTSD	1.0	1.0	1.0	1.0	1.0
VTOMO	55.0	70.0	90.0	100.0	125.0
VTOSO	2.0	2.0	2.0	2.0	2.0

The master clock is initialized at $\overset{4}{5}$8500.0 seconds.

The simulation time limit is 64800.00 seconds.

DN = 17 THN = 21 TDP = 22 DIST = 0.250 EN1 = 25 EN2 = 26 EN3 = 27

EN4 = 28 ENT = 22 ENT1 = 23 ENT2 = 24 ENT3 = 25 ENT4 = 26

F = 5.0 NP = 4. XSDD = 1.0 DSTENT = 500.00

DSTEN1 = 5000.00 DSTEN2 = 6500.00 DSTEN3 = 8000.00 DSTEN4 = 9500.00

TABLE F-I Continued.

Time	Type Operation	A/C Class	A/C in System (STACK and APROCH)
17.048	Take off	3	1.
17.143	Landing	1	4.
17.187	Landing	2	9.
17.196	Waveoff	3	9.
17.197	Take off	5	8.
17.250	Landing	3	7.
17.324	Landing	1	11.
17.364	Landing	2	14.
17.373	Take off	4	13.
17.382	Take off	4	12.
17.400	Take off	5	13.
17.413	Take off	4	12.
17.422	Take off	4	11.
17.433	Take off	4	12.
17.446	Take off	4	15.
17.482	Landing	2	16.
17.488	Take off	2	15.

TABLE F-I Continued.

```
                      Simulation Summary

Beginning time of simulation was 17.00 hours.
End time of simulation was  17.50 hours.
Total time simulated was  0.50 hours.
Total number of missed approaches was 1.
Total number of aborted takeoffs was 0.
Total number of landings was 6.
Total number of takeoffs was 10.
Total number of operations was 16.
Total number of type 1 landings was 2.
Total number of type 1 takeoffs was 0.
Total number of type 2 landings was 3.
Total number of type 2 takeoffs was 1.
Total number of type 3 landings was 1.
Total number of type 3 takeoffs was 1.
Total number of type 4 landings was 0.
Total number of type 4 takeoffs was 6.
Total number of type 5 landings was 0.
Total number of type 5 takeoffs was 2.
Variance of operations per 450. seconds was    12.000.
Standard deviation of operations per 450. seconds was     3.464.
```

APPENDIX G: INPUT TO THE SIMULATOR

APPENDIX G: INPUT TO THE SIMULATOR

If it is desired to verify the results obtained in this study, all input data used are listed in this appendix. Table G-I contains simulator input data common to all runs. Table G-II contains input data for 29 combinations of the independent variables under study. Table G-III contains simulator input data unique to each mix investigated.

TABLE G-I: SIMULATOR INPUT DATA COMMON TO ALL RUNS

	A/C Class				
	1	2	3	4	5
ACCMU	5.0	5.0	5.0	5.0	5.0
ACCSD	1.0	1.0	1.0	1.0	1.0
ALLDMN	3.0	5.0	5.0	8.0	5.0
ALTOMN	3.0	5.0	5.0	8.0	5.0
DCELMU	1.8	2.4	3.1	5.0	6.0
DCELSD	0.09	0.05	0.02	0.01	0.01
EHRMAX	3.00	3.00	2.00	2.00	2.00
EHRMIN	2.00	2.00	1.00	1.00	1.00
EXVLMX	30.0	25.0	20.0	20.0	20.0
PMRLD	5.0	10.0	12.0	20.0	15.0
PMRTO	5.0	10.0	12.0	20.0	15.0
PTELD	61200.00	61200.00	61200.00	61200.00	61200.00
PTETO	61200.00	61200.00	61200.00	61200.00	61200.00
PTMLD	32400.00	32400.00	32400.00	28800.00	32400.00
PTMTO	32400.00	32400.00	32400.00	28800.00	32400.00
RWYLDT	30.0	35.0	45.0	45.0	45.0

TABLE G-II: SIMULATOR INPUT DATA FOR 29 COMBINATIONS OF THE INDEPENDENT VARIABLES

Combination	Minimum Separation	Scheduled Separation	Mean Velocity for Class I	II	III	IV	V
1	3.58	3.88	61.14	81.86	108.82	118.51	162.66
2	1.30	1.60	64.90	69.84	99.99	131.35	150.25
3	1.68	1.98	63.22	76.35	91.16	117.46	162.23
4	1.17	1.47	52.27	67.63	100.57	133.58	162.57
5	3.75	4.05	49.46	81.27	111.28	133.69	158.02
6	1.96	2.26	45.35	81.46	109.81	126.09	162.14
7	3.99	4.29	64.12	81.94	99.40	120.12	145.42
8	2.39	2.69	45.60	69.12	114.05	134.07	140.18
9	2.40	2.70	59.13	66.63	88.98	116.21	144.20
10	3.58	3.88	50.71	75.28	100.51	124.46	144.66
11	2.16	2.46	54.82	68.82	106.90	128.19	149.67
12	1.06	1.36	61.76	75.99	107.66	126.71	157.87
13	2.02	2.32	58.62	83.34	96.16	134.58	153.12
14	2.11	2.41	51.13	80.86	85.02	132.34	144.90

TABLE G-II Continued.

Combination	Minimum Separation	Scheduled Separation	Mean Velocity for Class I	II	III	IV	V
15	1.22	1.52	54.31	73.96	99.97	134.28	147.29
16	2.50	2.80	61.48	70.40	91.15	130.98	163.71
17	1.73	2.03	54.09	69.84	95.85	134.83	157.36
18	2.74	3.04	49.12	66.02	98.61	120.25	152.26
19	1.15	1.45	46.29	68.45	98.61	118.37	163.19
20	3.97	4.27	64.02	70.16	114.63	127.09	158.46
21	3.26	3.56	50.93	82.32	100.90	122.68	153.37
22	3.70	4.00	59.12	74.08	96.19	117.99	153.53
23	3.53	3.83	55.76	67.25	110.06	134.97	151.79
24	3.29	3.59	61.26	70.88	98.38	115.63	144.36
25	3.38	3.68	45.34	69.41	90.11	115.73	157.18
26	2.65	2.95	56.45	71.02	104.65	119.38	150.51
27	1.79	2.09	55.52	68.83	97.46	130.32	161.49
28	2.89	3.19	61.74	78.19	97.75	127.29	161.50
29	2.84	3.14	45.91	76.85	89.32	125.66	162.56

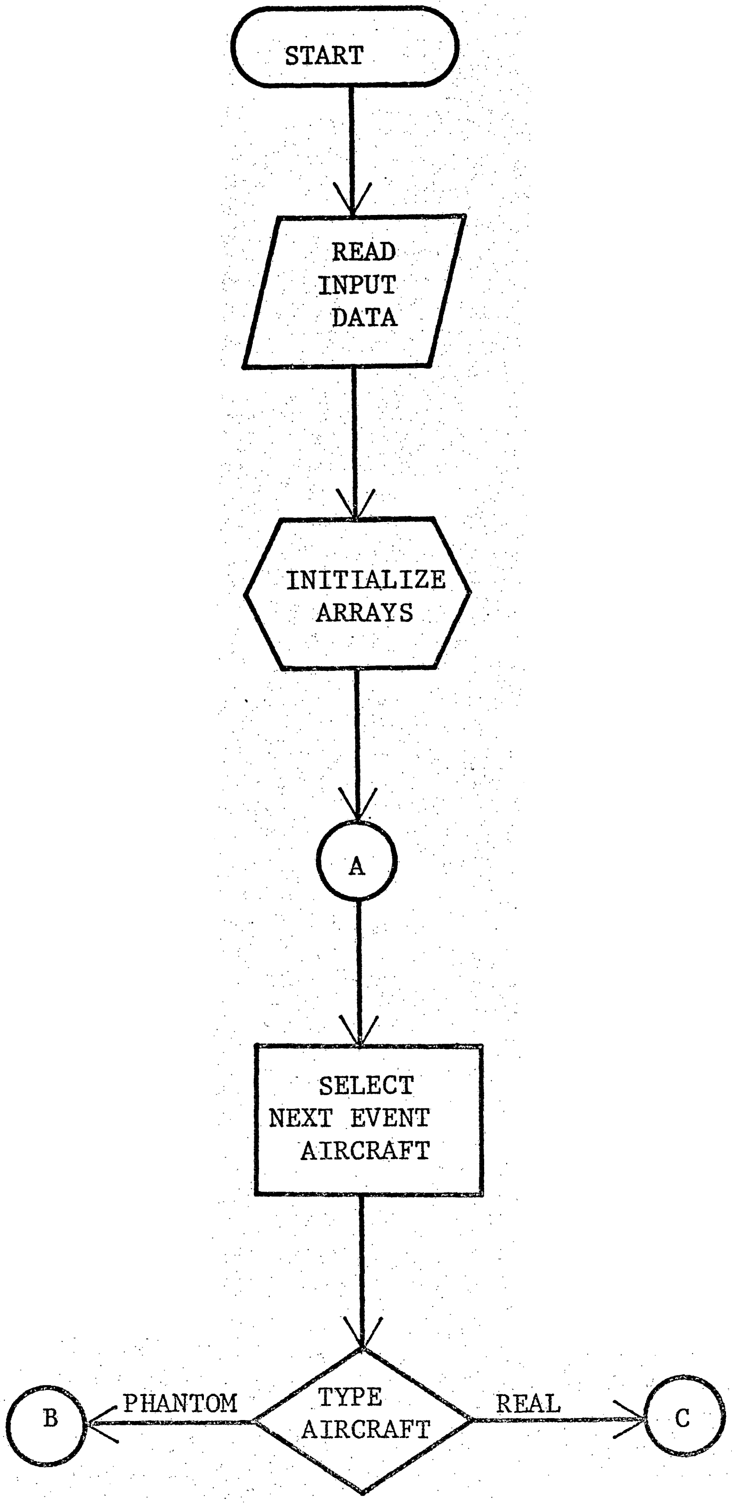
TABLE G-III: INPUT DATA UNIQUE TO EACH MIX

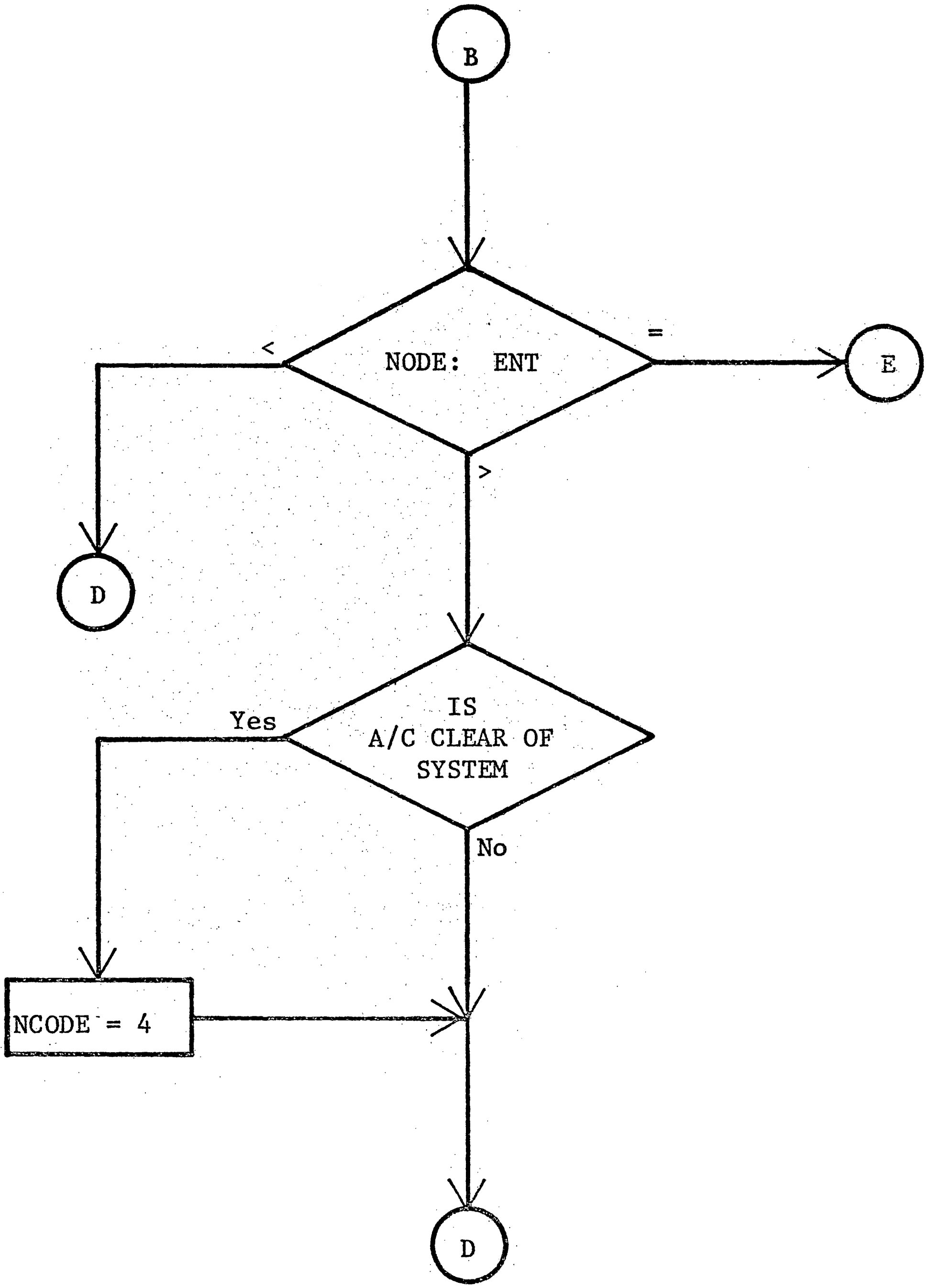
Variable Names Are Those Defined in Chapter IV

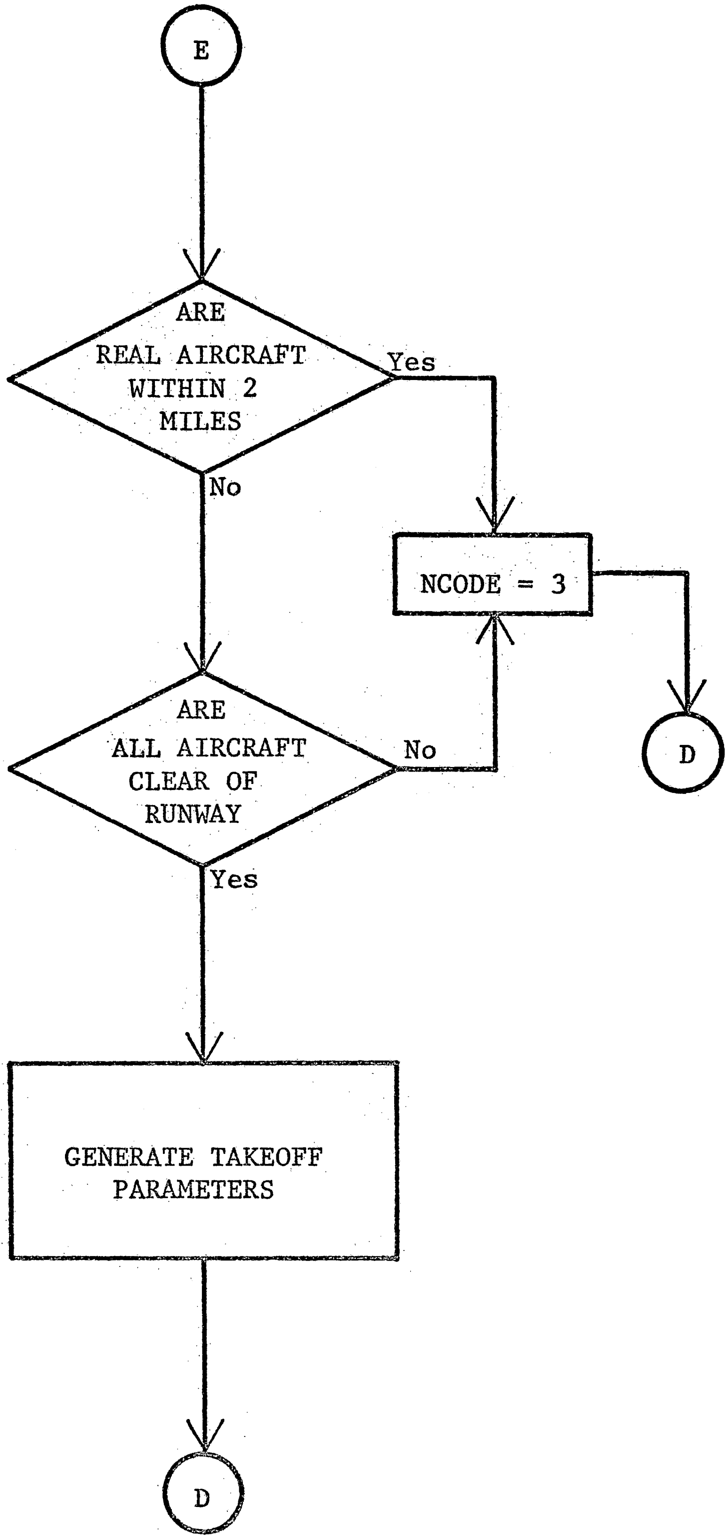
	Aircraft Class				
	1	2	3	4	5
Mix 1--Fast					
ALLDMN	5	8	9	8	5
ALTOMN	1	2	4	8	5
PERLD	5	9	23	28	13
PERTO	3	3	12	12	3
Mix 2--Medium					
ALLDMN	3	7	10	11	6
ALTOMN	3	3	4	5	2
PERLD	8	23	25	10	8
PERTO	3	10	18	8	3
Mix 3--Slow					
ALLDMN	6	14	10	5	3
ALTOMN	4	5	3	2	1
PERLD	25	32	5	5	0
PERTO	15	18	4	2	0

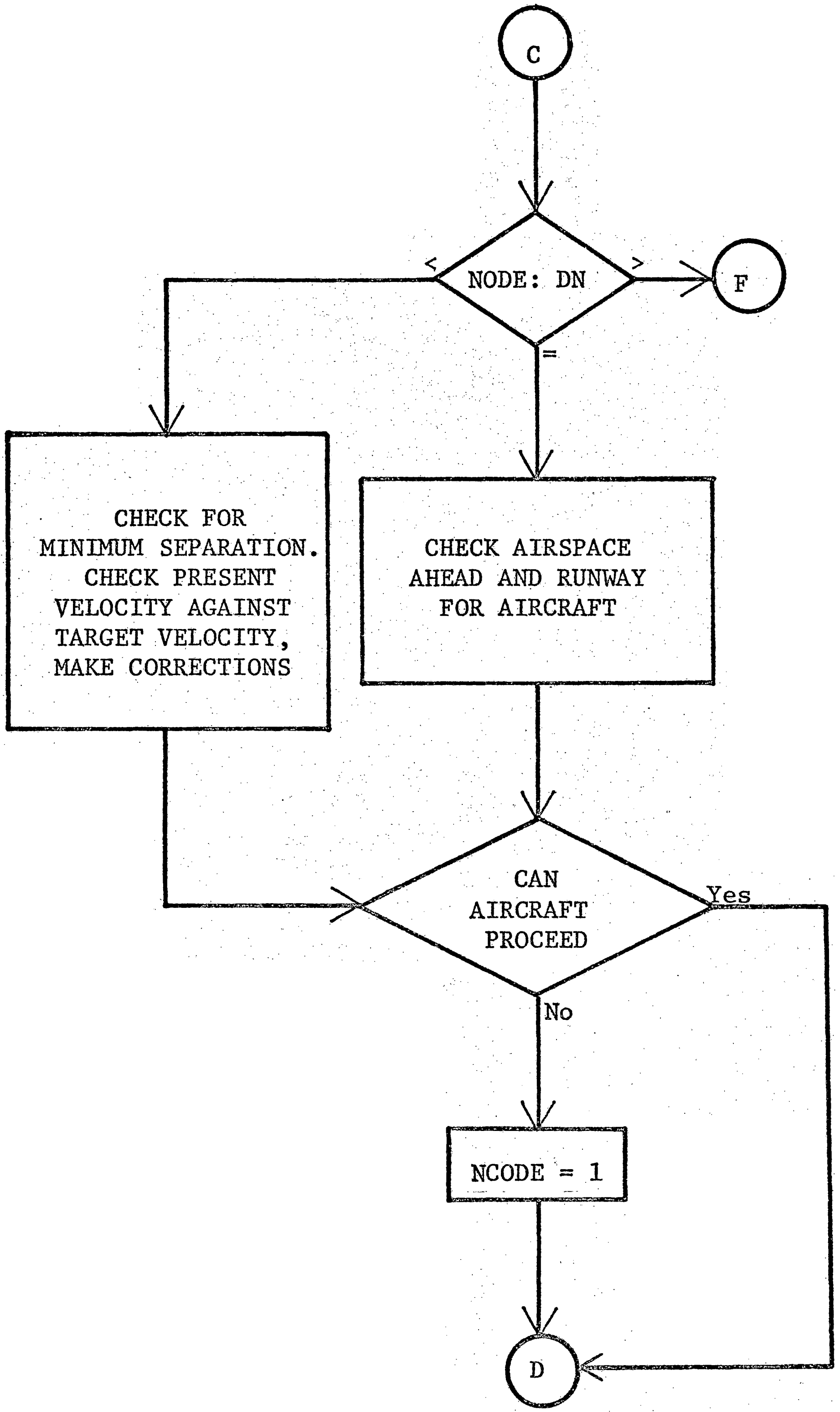
APPENDIX H: MACRO COMPUTER FLOW CHART

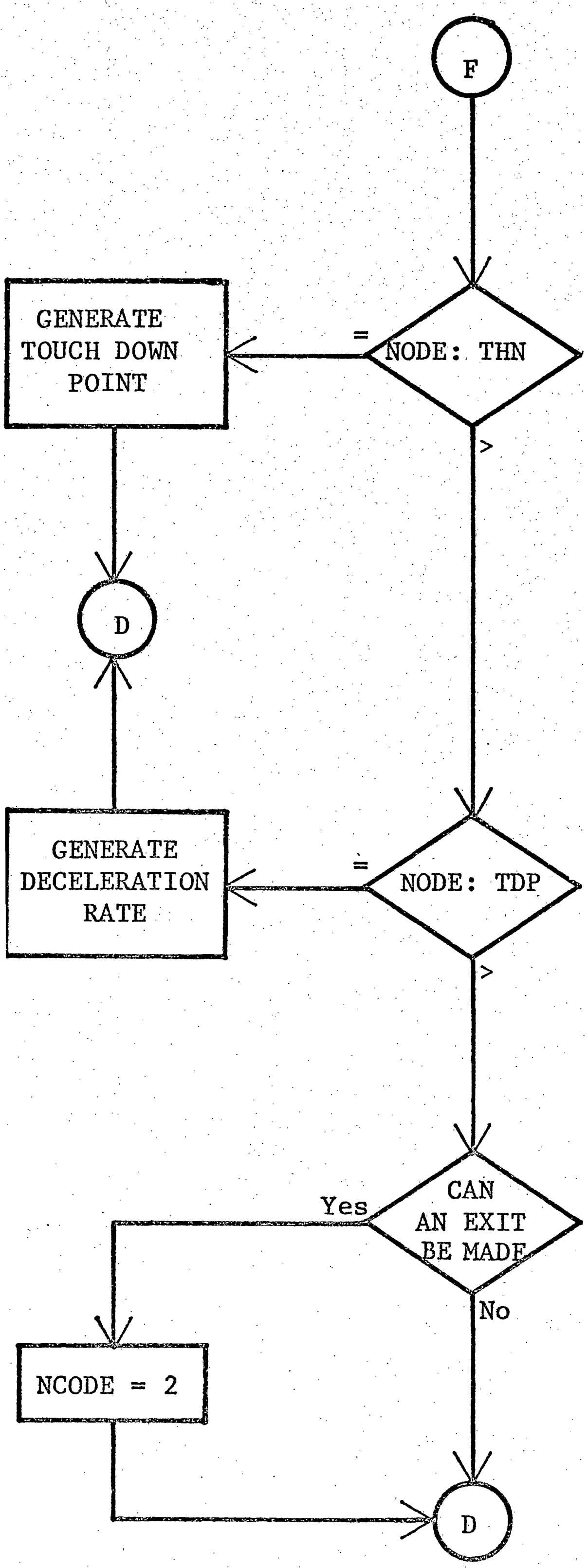
APPENDIX H: MACRO COMPUTER FLOW CHART

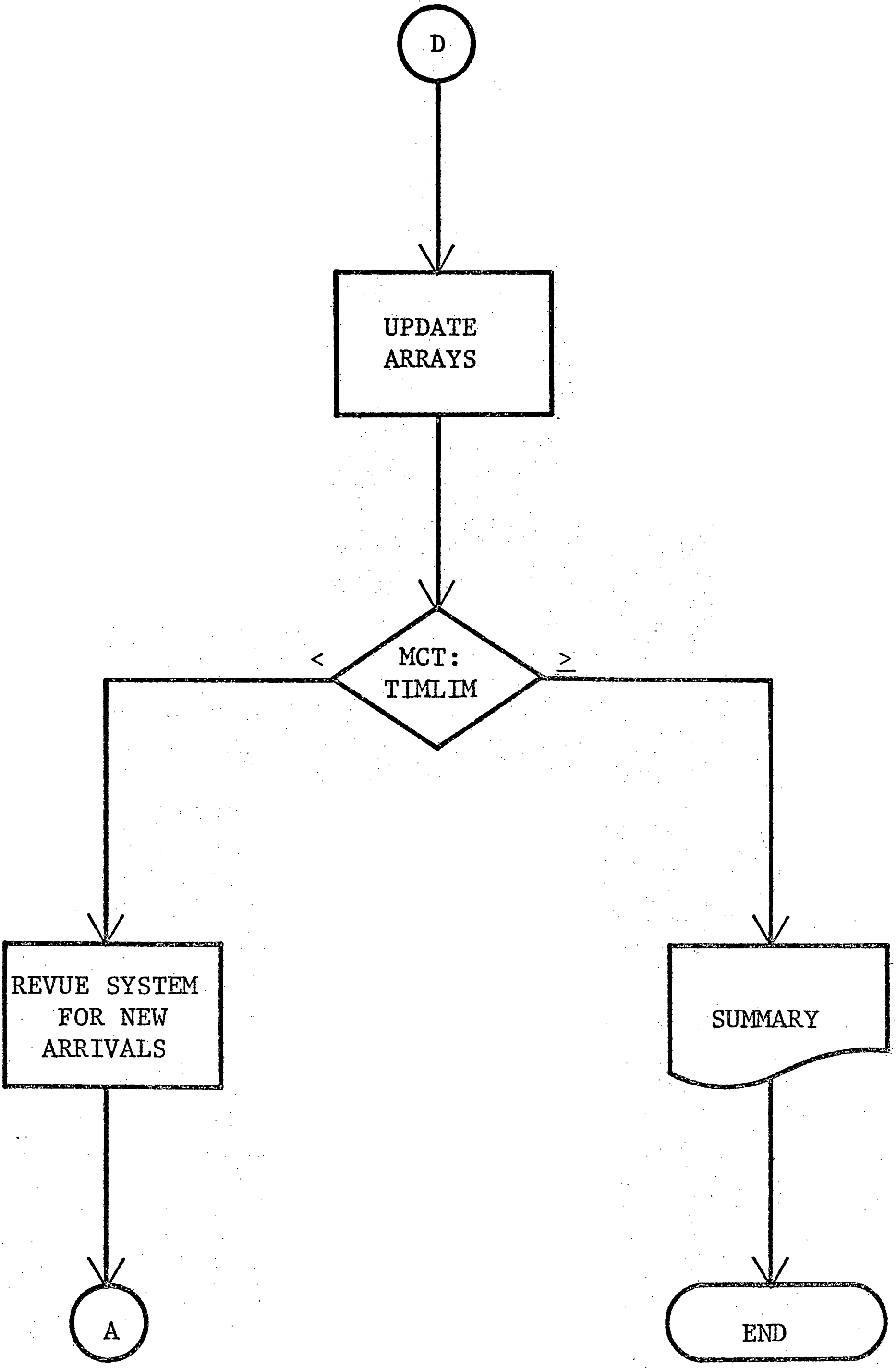



B
NODE: ENT
<
=
>
E
D
IS
A/C CLEAR OF
SYSTEM
Yes
No
NCODE = 4
D


E
ARE REAL AIRCRAFT WITHIN 2 MILES
Yes
No
NCODE = 3
D
ARE ALL AIRCRAFT CLEAR OF RUNWAY
No
Yes
GENERATE TAKEOFF PARAMETERS
D


C
NODE: DN
<
>
=
F
CHECK FOR MINIMUM SEPARATION. CHECK PRESENT VELOCITY AGAINST TARGET VELOCITY, MAKE CORRECTIONS
CHECK AIRSPACE AHEAD AND RUNWAY FOR AIRCRAFT
CAN AIRCRAFT PROCEED
Yes
No
NCODE = 1
D


F
NODE: THN
=
>
GENERATE TOUCH DOWN POINT
D
NODE: TDP
=
>
GENERATE DECELERATION RATE
CAN AN EXIT BE MADE
Yes
No
NCODE = 2
D


D
UPDATE ARRAYS
MCT: TIMLIM
<
≥
REVUE SYSTEM FOR NEW ARRIVALS
SUMMARY
A
END

APPENDIX I: COMPUTER PROGRAM

```
C     AC ALWAYS REPRESENTS THE PRESENT EVENT A/C.
C     ACCMU(I) IS THE MEAN ACC RATE (FT/SEC/SEC) OF A CLASS I A/C.
C     ACCSD(I) IS THE S.D. OF ACC RATE (FT/SEC/SEC) OF A CLASS I A/C.
C     ALLDMN(I) IS THE MIN DAILY ARRIVAL LEVEL (ARRIVALS/HR) FOR CLASS I
C          LANDINGS.
C     ALTOMN(I) IS THE MIN DAILY ARRIVAL LEVEL (ARRIVALS/HR) FOR CLASS I
C          TAKEOFFS.
C     DCELMU(I) IS THE MEAN DCEL RATE (FT/SEC/SEC) OF A CLASS I A/C.
C     DCELSD(I) IS THE S.D. OF DCEL RATE (FT/SEC/SEC) OF A CLASS I A/C.
C     DIST IS 1/NP OR THE DISTANCE IN N. MILES BETWEEN NODES FROM THE
C          ENTRY GATE TO TH  AND FROM ENT TO ENT4.
C     DN IS THE DECISION NODE, I.E. THE POINT AT WHICH DECISION
C          HEIGHT IS REACHED.  PAST DN THE A/C IS COMMITTED TO LAND.
C     DSTEN1 IS THE DISTANCE IN FEET TO EN1 FROM THE THRESHOLD.
C     DSTEN2 IS THE DISTANCE IN FEET TO EN2 FROM THE THRESHOLD.
C     DSTEN3 IS THE DISTANCE IN FEET TO EN3 FROM THE THRESHOLD.
C     DSTEN4 IS THE DISTANCE IN FEET TO EN4 FROM THE THRESHOLD.
C     DSTENT IS THE DISTANCE IN FEET TO THE RUNWAY ENTRANCE FROM THE
C          THRESHOLD.
C     EGMCT IS THE ENTRY GATE MCT (SEE THESIS).
C     ERRLST IS NOT ACTUALLY USED BUT IS INCLUDED IN CASE THE USER
C          WISHES TO INCORPORATE THE LAST VELOCITY ERROR INTO A NEW
C          VELOCITY ASSIGNMENT.
C     EN1 IS EXIT NODE 1.
C     EN2 IS EXIT NODE 2.
C     EN3 IS EXIT NODE 3.
C     EN4 IS EXIT NODE 4.
C     ENT IS THE ENTRANCE NODE FOR TAKE OFFS.
C     ENT1,2,3,4,5,6 ARE SIMPLY COMPUTATION NODES PLACED ON THE RUNWAY
C          FOR DEPARTING A/C.
C     EHRMAX AND EHRMIN ARE DEFINED IN SUBROUTINE VELCOR.
C     EXVLMX(I) IS THE MAX SAFE EXIT VEL (IN KTS) OF A CLASS I A/C.
```

```
C     F IS LENGTH OF FINAL APROCH PATH IN N. MILES.
C     IDEX IS AN INDICATOR USED IN SUBROUTINE REVUE.  WHEN IDEX=1, NO
C          A/C ARE IN THE STACK ARRAY.
C     INDEX IS AN INDICATOR.  WHEN INDEX=1 THERE IS AN A/C WAITING FOR
C          IMMEDIATE DEPARTURE WHEN THE RUNWAY IS CLEAR.  WHEN INDEX
C          CHANGES TO 11 THE PROGRAM IS INFORMED THAT THERE IS AN A/C
C          WAITING FOR IMMEDIATE DEPARTURE AND THE RUNWAY HAS JUST
C          CLEARED.
C     JCL IS THE A/C CLASS.
C     LN IS USED TO DENOTE THE LAST NODE VISITED.
C     MCAC IS THE MASTER CLOCK A/C (THE A/C WE'RE KEEPING TIME BY).
C     MCT IS THE MASTER CLOCK TIME (SECS) .  IT IS INITIALIZED AT THE
C          TIME OF DAY THE SIMULATION BEGINS .
C     N IS A COUNTER WHICH NUMBERS THE A/C AS THEY ENTER THE SYSTEM.
C     NCODE TELLS WHAT EVENT HAS HAPPENED
C          NCODE=1. . . WAVEOFF
C          NCODE=2. . . EXIT
C          NCODE=3. . . ABORTED T/O
C          NCODE=4. . . TAKEOFF
C          NCODE=5. . . REGULAR NODE WAS ENCOUNTERED.
C     NP IS THE NUMBER OF PARTS A N. MILE IS DIVIDED INTO ON FINAL APR.
C     NROWAP IS THE NUMBER OF ROWS IN THE ARRAY APROCH.
C     NROWST IS THE NUMBER OF ROWS IN THE ARRAY STACK.
C     NROWWT IS THE NUMBER OF ROWS IN THE ARRAY WAIT.
C     PERLD(I) IS THE PEAK EVENING RATE (ARRIVALS/HR) FOR CLASS I
C          LANDINGS.
C     PERTO(I) IS THE PEAK EVENING RATE (ARRIVALS/HR) FOR CLASS I
C          TAKEOFFS.
C     PMRLD(I) IS THE PEAK MORNING RATE (ARRIVALS/HR) FOR CLASS I
C          LANDINGS.
C     PMRTO(I) IS THE PEAK MORNING RATE (ARRIVALS/HR) FOR CLASS I
C          TAKECFFS.
```

```
C     PTELD(I) IS THE PEAK TIME (SECS) OF EVENING LANDINGS FOR CLASS I
C          A/C.
C     PTETO(I) IS THE PEAK TIME (SECS) OF EVENING TAKEOFFS FOR CLASS I
C          A/C.
C     PTMLD(I) IS THE PEAK TIME (SECS) OF MORNING LANDINGS FOR CLASS I
C          A/C.
C     PTMTO(I) IS THE PEAK TIME (IN SECONDS, I.E. 2 O'CLOCK EQUALS
C          14 HOURS TIMES 3600 SECS PER HOUR) OF MORNING TAKEOFFS FOR
C          CLASS I A/C.
C     RELTIM IS RELEASE TIME.
C     RPAC IS THE RELEASE POINT A/C (SEE THESIS).
C     RWYLDT(I) IS THE AVG TIME FROM OVER THRESHOLD TO EXIT FOR A CLASS I A/C.
C     RWYTOT(I) IS THE AVERAGE TIME ON RWY (SECS) FOR A CLASS I T/O.
C     S IS THE SEPARATION USED BY THE CONTROLLER TO SEQUENCE A/C.
C     SIGELD(I) IS 3 S.D. IN SECS FOR EVENING LANDINGS OF CLASS I A/C.
C     SIGETO(I) IS 3 S.D. IN SECS FOR EVENING TAKEOFFS OF CLASS I A/C.
C     SIGMLD(I) IS 3 S.D. IN SECS FOR MORNING LANDINGS OF CLASS I A/C.
C     SIGMTO(I) IS 3 S.D. IN SECS FOR MORNING TAKEOFFS OF CLASS I A/C.
C     SMIN IS THE MINIMUM SEPARATION ALLOWED BY FAR'S.
C     SMIN AND S ARE IN N. MILES.
C     T IS THE TYPE A/C (REAL OR PHANTOM).
C     TDDMU(I) IS THE MEAN TOUCH DOWN DISTANCE (IN FT) FROM THE
C          THRESHOLD FOR A CLASS I LANDING.
C     TDDSD(I) IS THE S.D. OF TOUCH DOWN DISTANCE (FT) FROM THE
C          THRESHOLD FOR A CLASS I LANDING.
C     TDP IS THE TOUCH DOWN POINT.
C     TIMLIM IS THE TIME LIMIT OF THE SIMULATION IN SECONDS, I.E. THE
C          TIME OF DAY THE SIMULATION WILL END.
C     TIMLN IS TIME AT LAST NODE.
C     THN IS THE THRESHOLD NODE.
C     TOSEP(I,J) IS THE TAKE OFF SEPARATION (FT) REQUIRED BY FARS IF A
C          CLASS I TAKE OFF IS FOLLOWED BY A CLASS J TAKE OFF,
```

```
C         ASSUMING DIVERGENT FLIGHT PATHS.
C     THE THREE STORAGE ARRAYS ARE SET UP AS FOLLOWS. . .
C         ARRAY APROCH. . . CONTAINS A/C MAKING APPROACH TO RUNWAY
C         COLUMN 1. . . A/C NUMBER
C         COLUMN 2. . . A/C CLASS
C         COLUMN 3. . . TAKEOFF OR LANDING (T = 2 OR T = 1)
C         COLUMN 4. . . ASSIGNED VELOCITY
C         COLUMN 5. . . LAST VELOCITY
C         COLUMN 6. . . LAST NODE ENCOUNTERED
C         COLUMN 7. . . TIME AT LAST NODE (IN SECS)
C         ARRAY STACK. . . CONTAINS A/C IN HOLDING STACK
C         COLUMN 1. . . A/C NUMBER
C         COLUMN 2. . . A/C CLASS
C         COLUMN 3. . . TAKEOFF OR LANDING (T = 2 OR T = 1)
C         COLUMN 4. . . ASSIGNED VELOCITY
C         COLUMN 5. . . ARRIVAL TIME TO SYSTEM
C         ARRAY WAIT. . . CONTAINS ALL POSSIBLE NEXT EVENT A/C
C         COLUMN 1. . . A/C CLASS
C         COLUMN 2. . . TAKEOFF OR LANDING (T = 2 OR T = 1)
C         COLUMN 3. . . ASSIGNED VELOCITY (IF A/C IS LANDING)
C         COLUMN 4. . . ARRIVAL TIME TO SYSTEM
C     THIS RUNWAY HAS FOUR EXITS WITH VARIABLE SPACING BETWEEN EXITS.
C     VAPMU(I) IS THE MEAN APROCH VEL (KTS) OF A CLASS I A/C.
C     VAPSD(I) IS THE S.D. OF APROCH VELOCITY (KTS) OF A CLASS I A/C.
C     VARMCT MARKS THE BEGINNING OF THE TIME PERIOD OVER WHICH VARIANCE
C         IS BEING CALCULATED.
C     VARTIM IS THE TIME IN SECONDS OVER WHICH VARIANCE IS TO BE
C        CALCULATED.
C     VELLST IS LAST VELOCITY.
C     VENTMU(I) IS THE MEAN RWY ENTRANCE VEL (IN KTS) FOR A CLASS I A/C.
C     VENTSD(I) IS THE S.D. OF ENTRANCE VEL (IN KTS) OF A CLASS I A/C.
C     VTOMU(I) IS THE MEAN TAKE OFF VEL (IN KTS) OF A CLASS I A/C.
```

```
C     VTOSD(I) IS THE S.D. OF TAKE OFF VEL (IN KTS) OF A CLASS I A/C.
C     XDD IS THE DECISION DISTANCE WHICH MUST BE DIVISIBLE BY 1/NP.
C          TO CALCULATE XDD FIRST DETERMINE THE DECISION HEIGHT, THEN
C          CALCULATE HOW FAR FROM THRESHOLD XDD IS (IN N. MILES).
C     XNIS(I) IS THE TOTAL OPERATIONS OCCURRING IN THE SYSTEM DURING
C          TIME PERIOD I.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      EXTERNAL RAND
      REAL MCT,LN,NP
      INTEGER DN,THN,TDP,EN1,DUMTPE,    EN2,ENT,ENT1,ENT2,ENT3,ENT4,RPAC,
     1MCAC
      NWO=0
      NEXT=0
      NTO=0
      NABTO=0
      INDEX=0
      J1=0
      N1=0
      IDEX=0
```

```
      INDERR=0
      INDRT=0
      SW=0.
      N=0
      OD=0.
      NCODE=5
      TH=0.
      READ(5,204)SMIN,DSTEN3,DSTEN4
  204 FORMAT(F4.2,2F7.2)
      READ(5,205)S,F,NP,XDD,DSTEN1,DSTEN2,DSTENT
  205 FORMAT(F3.1,F4.1,F3.0,F3.1,3F7.2)
      READ(5,210)NROWAP,NROWST,NROWWT
  210 FORMAT(3I2)
      READ(5,215)(VENTMU(I),I=1,5),(VENTSD(I),I=1,5),(VTOMU(I),I=1,5),(V
     1TOSD(I),I=1,5)
  215 FORMAT(10F6.2)
      READ(5,220)(ACCMU(I),I=1,5),(ACCSD(I),I=1,5),(DCELMU(I),I=1,5),(DC
     3ELSD(I),I=1,5)
  220 FORMAT(10F6.3)
      READ(5,225)(TDDMU(I),I=1,5),(TDDSD(I),I=1,5)
  225 FORMAT(10F7.2)
      READ(5,230)(EXVLMX(I),I=1,5),(VAPMU(I),I=1,5),(VAPSD(I),I=1,5)
  230 FORMAT(5F6.2)
      READ(5,235)(PTMTO(I),I=1,5),(PTETO(I),I=1,5),(PTMLD(I),I=1,5),(PTE
     1LD(I),I=1,5)
  235 FORMAT(10F8.2)
      READ(5,240)(SIGMTO(I),I=1,5),(SIGETO(I),I=1,5),(SIGML(I),I=1,5),(S
     5IGEL(I),I=1,5)
  240 FORMAT(10F7.2)
      READ(5,245)(ALTOMN(I),I=1,5),(ALLDMN(I),I=1,5),(PMRTO(I),I=1,5),(P
     1ERTO(I),I=1,5)
  245 FORMAT(20F3.0)
```

```
      READ(5,250)(PMRLD(I),I=1,5),(PERLD(I),I=1,5)
  250 FORMAT(10F3.0)
      READ(5,255)(RWYTOT(I),I=1,5),(RWYLDT(I),I=1,5)
  255 FORMAT(10F4.1)
      READ(5,257)MCT,TIMLIM
  257 FORMAT(2F7.1)
      READ(5,258)(EHRMAX(I),I=1,5),(EHRMIN(I),I=1,5)
  258 FORMAT(10F4.2)
      READ(5,259)VARTIM
  259 FORMAT(F6.0)
      BEGMCT=MCT
      DN=(F*NP)-(XDD*NP)+1.
      THN=F*NP+1.
      TDP=THN+1
      DIST=1./NP
      EN1=TDP+3
      EN2=EN1+1
      EN3=EN2+1
      EN4=EN3+1
      ENT=THN+1
      ENT1=ENT+1
      ENT2=ENT1+1
      ENT3=ENT2+1
      ENT4=ENT3+1
      ENT5=ENT4+1
      ENT6=ENT5+1
      WRITE(6,600)
      WRITE(6,605)
      WRITE(6,610)
      WRITE(6,635)(ACCMU(I),I=1,5)
      WRITE(6,610)
      WRITE(6,640)(ACCSD(I),I=1,5)
```

```
WRITE(6,610)
WRITE(6,730)(ALLDMN(I),I=1,5)
WRITE(6,610)
WRITE(6,725)(ALTOMN(I),I=1,5)
WRITE(6,610)
WRITE(6,645)(DCELMU(I),I=1,5)
WRITE(6,610)
WRITE(6,650)(DCELSD(I),I=1,5)
WRITE(6,610)
WRITE(6,651)(EHRMAX(I),I=1,5)
WRITE(6,610)
WRITE(6,652)(EHRMIN(I),I=1,5)
WRITE(6,610)
WRITE(6,665)(EXVLMX(I),I=1,5)
WRITE(6,610)
WRITE(6,750)(PERLD(I),I=1,5)
WRITE(6,610)
WRITE(6,740)(PERTO(I),I=1,5)
WRITE(6,610)
WRITE(6,745)(PMRLD(I),I=1,5)
WRITE(6,610)
WRITE(6,735)(PMRTO(I),I=1,5)
WRITE(6,610)
WRITE(6,696)(PTELD(I),I=1,5)
WRITE(6,610)
WRITE(6,686)(PTETO(I),I=1,5)
WRITE(6,610)
WRITE(6,691)(PTMLD(I),I=1,5)
WRITE(6,610)
WRITE(6,681)(PTMTO(I),I=1,5)
WRITE(6,610)
WRITE(6,760)(RWYLDT(I),I=1,5)
```

```
WRITE(6,610)
WRITE(6,600)
WRITE(6,605)
WRITE(6,610)
WRITE(6,755)(RWYTOT(I),I=1,5)
WRITE(6,610)
WRITE(6,720)(SIGEL(I),I=1,5)
WRITE(6,610)
WRITE(6,711)(SIGETO(I),I=1,5)
WRITE(6,610)
WRITE(6,715)(SIGML(I),I=1,5)
WRITE(6,610)
WRITE(6,705)(SIGMTO(I),I=1,5)
WRITE(6,610)
WRITE(6,655)(TDDMU(I),I=1,5)
WRITE(6,610)
WRITE(6,660)(TDDSD(I),I=1,5)
WRITE(6,610)
WRITE(6,671)(VAPMU(I),I=1,5)
WRITE(6,610)
WRITE(6,676)(VAPSD(I),I=1,5)
WRITE(6,610)
WRITE(6,615)(VENTMU(I),I=1,5)
WRITE(6,610)
WRITE(6,620)(VENTSD(I),I=1,5)
WRITE(6,610)
WRITE(6,625)(VTOMU(I),I=1,5)
WRITE(6,610)
WRITE(6,630)(VTOSD(I),I=1,5)
WRITE(6,610)
WRITE(6,765)MCT,TIMLIM
WRITE(6,770)DN,THN,TDP,DIST,EN1,EN2
```

```
      WRITE(6,775)ENT,ENT1,ENT2,ENT3,ENT4
      WRITE(6,780)S,F,NP,XDD
      WRITE(6,785)DSTEN1,DSTEN2,DSTENT
      TOSEP(1,1)=3000.
      TOSEP(2,1)=3000.
      TOSEP(1,2)=4500.
      TOSEP(2,2)=4500.
      DO265 I=1,5
      DO 260 J=3,5
      TOSEP(I,J)=6000.
      QLAND(I)=0.
      QTO(I)=0.
  260 CONTINUE
  265 CONTINUE
      DO 275 I=3,5
      DO 270 J=1,2
      TOSEP(I,J)=6000.
  270 CONTINUE
  275 CONTINUE
      IX=75432
      DO 285 I=1,NROWAP
      DO 280 J=1,7
      APROCH(I,J)=0.
  280 CONTINUE
  285 CONTINUE
      DO 295 I=1,NROWST
      DO 290 J=1,5
      STACK(I,J)=0.
  290 CONTINUE
  295 CONTINUE
      DO 305 I=1,NROWWT
      DO 300 J=1,4
```

```
      WAIT(I,J)=0.
  300 CONTINUE
  305 CONTINUE
      WRITE(6,1)
    1 FORMAT(1H1,T10,'TIME',T23,'TYPE OPERATION',T46,'A/C CLASS',T64,'A/
     1C IN SYSTEM (STACK AND APROCH)',///)
C           THE FOLLOWING FILLS THE FIRST FIVE ROWS OF WAIT WITH
C           POSSIBLE NEXT EVENT LANDINGS
      DO 2 I=1,5
      CALL VLOCTY(I)
      CALL ALPHA(SIGML(I),SIGEL(I),PMRLD(I),PERLD(I),ALLDMN(I),PTMLD(I),
     1PTELD(I))
      WAIT(I,1)=I
      WAIT(I,2)=1.
      WAIT(I,3)=VELASS
      WAIT(I,4)=ARTMIN+MCT
    2 CONTINUE
C           THE FOLLOWING FILLS ROWS 6-10 WITH POSSIBLE NEXT EVENT
C           TAKE OFF TYPES
      DO 3 I=1,5
      CALL ALPHA(SIGMTO(I),SIGETO(I),PMRTO(I),PERTO(I),ALTOMN(I),PTMTO(I
     1),PTETO(I))
      WAIT(I+5,1)=I
      WAIT(I+5,2)=2.
      WAIT(I+5,4)=ARTMIN+MCT
    3 CONTINUE
C           NOW THE WAIT MATRIX CONSISTING OF POSSIBLE NEXT EVENT TYPES
C           IS FULL.  WE NOW INTRODUCE THE FIRST A/C TO THE SYSTEM BY
C           SEARCHING WAIT FOR THE MINIMUM ARRIVAL TIME.  MM IS THE ROW
C           IN WAIT THAT THE ARRIVING A/C IS STORED IN.
      MM=1
      DO 4 I=2,10
```

```
      IF(WAIT(I,4).LT.WAIT(MM,4))MM=I
    4 CONTINUE
      N=N+1
      RPAC=N
      AC=N
      MCAC=N
      T=WAIT(MM,2)
      JCL=WAIT(MM,1)
      IF(T.EQ.1.)GO TO 5
      APROCH(1,1)=N
      APROCH(1,2)=JCL
      APROCH (1,3)=T
      APROCH(1,6)=ENT
      APROCH(1,7)=WAIT(MM,4)
      VARMCT=WAIT(MM,4)
      NODE=ENT
      SEPTO=0.
      CALL TOINFO
      APROCH(1,4)=VELENT
      APROCH(1,5)=VELENT
      GO TO 10
C          HERE THE FIRST A/C ENTERING THE SYSTEM IS REGISTERED IN APROCH.
    5 APROCH(1,1)=N
      APROCH(1,2)=JCL
      APROCH(1,3)=T
      APROCH(1,4)=WAIT(MM,3)
      APROCH(1,5)=WAIT(MM,3)
      APROCH(1,6)=1.
      APROCH(1,7)=WAIT(MM,4)
   10 MCT=WAIT(MM,4)
      EGMCT=MCT
      CALL REPLCE
```

```
C
C          THE PROGRAM UP TO THIS POINT IS CONCERNED ONLY WITH THE
C          INITIALIZATION OF STORAGE ARRAYS AND LISTING ALL INPUT DATA.
C
   15 CALL TIMNCD
      T=APROCH(NROW,3)
      IF(T.EQ.1.)GO TO 60
C
C          THE PROGRAM FROM HERE UP TO STATEMENT 60 DEALS ONLY WITH
C          PHANTOM AIRCRAFT.
C
      IF(NODE-ENT)125,50,20
   20 IF(NODE-ENT1)125,40,21
   21 IF(NODE-ENT2)125,40,25
   25 IF(NODE-ENT3)125,40,30
   30 IF(NODE-ENT4)125,40,31
   31 IF(NODE-ENT5)125,40,32
   32 NCODE=4
      IF(INDEX-1.)45,35,45
   35 INDEX=11.
      GO TO 45
   40 CALL VVELTO
   45 GO TO 125
   50 CALL THCHK
      IF(TH.NE.1.)GO TO 55
      NCODE=3
      CALL UPDATE
      GO TO 15
   55 CALL RWYCHK
      GO TO 125
C
C          FROM HERE ON REAL AIRCRAFT ARE DELT WITH.
```

```
C
   60 IF(NODE-DN)63,65,90
   63 CALL ERROR
      GO TO 125
   65 CALL DECNOD
      GO TO 125
   90 IF(NODE-THN)125,95,100
   95 CALL TDPT
      GO TO 125
  100 IF(NODE-TDP)125,105,110
  105 CALL TDINFO
      GO TO 125
  110 IF(NODE-EN1)125,115,120
  115 CALL VELEN1
      GO TO 125
  120 IF(NODE-EN2)125,121,122
  121 CALL VELEN1
      GO TO 125
  122 IF(NODE-EN3)125,123,124
  123 CALL VELEN1
      GO TO 125
  124 IF(NODE-EN4)125,127,127
  127 NCODE=2
      IF(INDEX.EQ.1.)INDEX=11.
  125 CALL UPDATE
      IF(IFIX(AC).NE.RPAC)GO TO 15
      CALL REVUE
      IF(IDEX-1.)133,130,133
  130 IDEX=0.
      GO TO 15
  133 CALL APRROW
  135 IF(RELTIM-APROCH(NROW,7))140,140,15
```

```
140 RPAC=RAC
    EGMCT=RELTIM
    LN=1.
    TIMLN=RELTIM
    ERRLST=0.
    CALL APRREG
    DUMAC=AC
    AC=RAC
    CALL STKDEL
    AC=MCAC
    CALL APRROW
    IF(APROCH(NROW,3).EQ.1.)GO TO 144
    AC=RAC
    CALL APRROW
    IF(APROCH(NROW,3).EQ.2.)GO TO 144
    MCAC=RAC
    MCT=APROCH(NROW,7)
144 AC=DUMAC
    GO TO 15
145 STOP
600 FORMAT(1H1,///////////,T61,'A/C CLASS',/)
605 FORMAT(T42,'|',T47,'1',T53,'|',T58,'2',T64,'|',T69,'3',T75,'|',T80
   1,'4',T86,'|',T91,'5',T97,'|')
610 FORMAT(T32,33('--'))
615 FORMAT(T33,'VENTMU   |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
620 FORMAT(T33,'VENTSD   |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
625 FORMAT(T33,'VTOMU    |',T46,F5.1,T53,'|',T57,F5.1,T64,'|',T68,F5.1
   1,T75,'|',T79,F5.1,T86,'|',T90,F5.1,T97,'|')
630 FORMAT(T33,'VTOSD    |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
```

```
635 FORMAT(T33,'ACCMU    |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
640 FORMAT(T33,'ACCSD    |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
645 FORMAT(T33,'DCELMU   |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
650 FORMAT(T33,'DCELSD   |',T46,F4.2,T53,'|',T57,F4.2,T64,'|',T68,F4.2
   1,T75,'|',T79,F4.2,T86,'|',T90,F4.2,T97,'|')
651 FORMAT(T33,'EHRMAX   |',T46,F4.2,T53,'|',T57,F4.2,T64,'|',T68,F4.2
   1,T75,'|',T79,F4.2,T86,'|',T90,F4.2,T97,'|')
652 FORMAT(T33,'EHRMIN   |',T46,F4.2,T53,'|',T57,F4.2,T64,'|',T68,F4.2
   1,T75,'|',T79,F4.2,T86,'|',T90,F4.2,T97,'|')
655 FORMAT(T33,'TDDMU    |',T45,F7.2,T53,'|',T56,F7.2,T64,'|',T67,F7.2
   1,T75,'|',T78,F7.2,T86,'|',T89,F7.2,T97,'|')
660 FORMAT(T33,'TDDSD    |',T46,F5.1,T53,'|',T57,F5.1,T64,'|',T68,F5.1
   1,T75,'|',T79,F5.1,T86,'|',T90,F5.1,T97,'|')
665 FORMAT(T33,'EXVLMX   |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
671 FORMAT(T33,'VAPMU    |',T46,F4.1,T53,'|',T57,F5.1,T64,'|',T68,F5.1
   1,T75,'|',T79,F5.1,T86,'|',T90,F5.1,T97,'|')
676 FORMAT(T33,'VAPSD    |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
681 FORMAT(T33,'PTMTO    |',T44,F8.2,T53,'|',T55,F8.2,T64,'|',T66,F8.2
   1,T75,'|',T77,F8.2,T86,'|',T88,F8.2,T97,'|')
686 FORMAT(T33,'PTETO    |',T44,F8.2,T53,'|',T55,F8.2,T64,'|',T66,F8.2
   1,T75,'|',T77,F8.2,T86,'|',T88,F8.2,T97,'|')
691 FORMAT(T33,'PTMLD    |',T44,F8.2,T53,'|',T55,F8.2,T64,'|',T66,F8.2
   1,T75,'|',T77,F8.2,T86,'|',T88,F8.2,T97,'|')
696 FORMAT(T33,'PTELD    |',T44,F8.2,T53,'|',T55,F8.2,T64,'|',T66,F8.2
   1,T75,'|',T77,F8.2,T86,'|',T88,F8.2,T97,'|')
705 FORMAT(T33,'SIGMTO   |',T45,F7.2,T53,'|',T56,F7.2,T64,'|',T67,F7.2
   1,T75,'|',T78,F7.2,T86,'|',T89,F7.2,T97,'|')
```

```
711 FORMAT(T33,'SIGETO    |',T45,F7.2,T53,'|',T56,F7.2,T64,'|',T67,F7.2
   1,T75,'|',T78,F7.2,T86,'|',T89,F7.2,T97,'|')
715 FORMAT(T33,'SIGML     |',T45,F7.2,T53,'|',T56,F7.2,T64,'|',T67,F7.2
   1,T75,'|',T78,F7.2,T86,'|',T89,F7.2,T97,'|')
720 FORMAT(T33,'SIGEL     |',T45,F7.2,T53,'|',T56,F7.2,T64,'|',T67,F7.2
   1,T75,'|',T78,F7.2,T86,'|',T89,F7.2,T97,'|')
725 FORMAT(T33,'ALTOMN    |',T42,'|',T47,F2.0,T53,'|',T58,F2.0,T64,'|',
   1T69,F2.0,T75,'|',T80,F2.0,T86,'|',T91,F2.0,T97,'|')
730 FORMAT(T33,'ALLDMN    |',T42,'|',T47,F2.0,T53,'|',T58,F2.0,T64,'|',
   1T69,F2.0,T75,'|',T80,F2.0,T86,'|',T91,F2.0,T97,'|')
735 FORMAT(T33,'PMRTO     |',T42,'|',T47,F3.0,T53,'|',T58,F3.0,T64,'|',
   1T69,F3.0,T75,'|',T80,F3.0,T86,'|',T91,F3.0,T97,'|')
740 FORMAT(T33,'PERTO     |',T42,'|',T47,F3.0,T53,'|',T58,F3.0,T64,'|',
   1T69,F3.0,T75,'|',T80,F3.0,T86,'|',T91,F3.0,T97,'|')
745 FORMAT(T33,'PMRLD     |',T42,'|',T47,F3.0,T53,'|',T58,F3.0,T64,'|',
   1T69,F3.0,T75,'|',T80,F3.0,T86,'|',T91,F3.0,T97,'|')
750 FORMAT(T33,'PERLD     |',T42,'|',T47,F3.0,T53,'|',T58,F3.0,T64,'|',
   1T69,F3.0,T75,'|',T80,F3.0,T86,'|',T91,F3.0,T97,'|')
755 FORMAT(T33,'RWYTOT    |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
760 FORMAT(T33,'RWYLDT    |',T46,F4.1,T53,'|',T57,F4.1,T64,'|',T68,F4.1
   1,T75,'|',T79,F4.1,T86,'|',T90,F4.1,T97,'|')
765 FORMAT(//,T32,'THE MASTER CLOCK IS INITIALIZED AT',F8.1,' SECONDS.
   1',//,T32,'THE SIMULATION TIME LIMIT IS',F8.1,' SECONDS.',/)
770 FORMAT(T32,'DN = ',I2,'  THN = ',I2,'  TDP = ',I2,'  DIST = ',F5.3
   1,'  EN1 = ',I2,'  EN2 = ',I2,/)
775 FORMAT(T32,'ENT = ',I2,'  ENT1 = ',I2,'  ENT2 = ',I2,'  ENT3 = ',I
   12,'  ENT4 = ',I2,/)
780 FORMAT(T32,'S = ',F3.1,'  F = ',F4.1,'  NP = ',F2.0,'  SDD = ',F3.
   11,/)
785 FORMAT(T32,'DSTEN1 = ',F7.2,'  DSTEN2 = ',F7.2,'  DSTENT = ',F6.2)
    END
```

```
      SUBROUTINE REPLCE
C
C     WHEN AN A/C ENTERS IN THE SYSTEM ITS PARAMETERS ARE TRANSFERRED
C     FROM WAIT (THE MATRIX OF POSSIBLE NEXT EVENTS) TO APROCH.
C     SUBROUTINE REPLCE WILL GENERATE AN A/C OF THE TYPE THAT JUST LEFT
C     WAIT AND ENTER IT INTO WAIT SO THAT THE NEXT EVENT MATRIX WILL
C     AGAIN BE FULL.
C     J IS THE A/C CLASS.
C     T IS THE TYPE OPERATION (T=2 FOR TAKEOFFS, T=1 FOR LANDINGS).
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      J=WAIT(MM,1)
      T=WAIT(MM,2)
      IF(T.EQ.1.)CALL ALPHA(SIGML(J ),SIGEL(J ),PMRLD(J ),PERLD(J ),ALLD
     1MN(J ),PTMLD(J ),PTELD(J ))
      IF(T.EQ.2.)CALL ALPHA(SIGMTO(J ),SIGETO(J ),PMRTO(J ),PERTO(J ),AL
     1TOMN(J ),PTMTO(J ),PTETO(J ))
      IF(T.EQ.1.)CALL VLOCTY(J)
      WAIT(MM,1)=WAIT(MM,1)
```

```
WAIT(MM,2)=WAIT(MM,2)
WAIT(MM,4)=WAIT(MM,4)+ARTMIN
IF(T.EQ.1.)WAIT(MM,3)=VELASS
IF(T.EQ.2.)WAIT(MM,3)=0.
RETURN
END
```

```
      SUBROUTINE ALPHA(S1,S2,PMR,PER,RHO,PMT,PET)
C     THE PURPOSE OF THIS ROUTINE IS TO CALCULATE FIRST A TIME
C     DEPENDENT ARRIVAL RATE AND FROM THIS AN INTERARRIVAL TIME.
C     DEFINITION OF VARIABLES. . .
C          PER  PEAK EVENING RATE
C          PET  EVENING PEAK ARRIVAL TIME
C          RHO  MINIMUM RATE FOR THE DAY
C          PMR  PEAK MORNING RATE
C          PMT  MORNING PEAK ARRIVAL TIME
C          S1  3 STANDARD DEVIATIONS IN SECONDS FOR MORNING OPERATIONS
C          S2  3 STANDARD DEVIATIONS IN SECONDS FOR EVENING OPERATIONS
C     NORM=1 INDICATES THAT NORMAL INTERARRIVAL TIMES WILL BE GENERATED.
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      EXTERNAL RAND
      NORM=1
      NORM=0
      IF(MCT.GT.43200..AND.PER.EQ.0.)GO TO 10
      IF(MCT.LE.43200..AND.PMR.EQ.0.)GO TO 10
      OD=0.
```

```
      SD1=S1/3.
      SD2=S2/3.
      B1=(PMR-RHO)*SD1*SQRT(2.*3.14156)
      B2=(PER-RHO)*SD2*SQRT(2.*3.14156)
      XPON2=-(ABS(MCT-PET))**2./(2.*SD2**2.)
      XPON1=-(ABS(MCT-PMT))**2./(2.*SD1**2.)
      IF(XPON2.LT.-179.)EXPON2=1.
      IF(XPON2.LT.-179.)GO TO 9
      IF(XPON1.LT.-179.)EXPON1=1.
      IF(XPON1.LT.-179.)GO TO 9
      EXPON1=EXP(XPON1)
      EXPON2=EXP(XPON2)
    9 IF(MCT.GT.43200.)ALPHAT=(B2/(SD2*SQRT(2.*3.14159)))*EXPON2+RHO
      IF(MCT.LE.43200.)ALPHAT=(B1/(SD1*SQRT(2.*3.14159)))*EXPON1+RHO
      ALPHAT=(B1/(SD1*SQRT(2.*3.14156)))*EXP(-(MCT-PMT)**2./(2.*SD1**2.)
     1)+(B2/(SD2*SQRT(2.*3.14156)))*EXP(-(MCT-PMT)**2./(2.*SD2**2.))+RHO
      IF(NORM.EQ.1)GO TO 5
      ALPHAT=ALPHAT/3600.
      ARTMIN=-(1./ALPHAT)*ALOG(RAND(00))
      RETURN
    5 RR=RAND(00)
      AL=1./ALPHAT
      W=4./AL**2.
      Z=SQRT(-2.*ALOG(.5*(1.0-ABS(1.-2.*RR))))
      T3=Z-(2.515517+0.802853*Z+.010328*Z**2.)/(1.+1.432788*Z+0.189269*Z
     1**2.+.001308*Z**3.)
      T2=(RR-0.5)/(ABS(RR-0.5))
      ARTMIN=AL+T2*W*T3
      RETURN
   10 ARTMIN=10.**32.
      RETURN
      END
```

```
      FUNCTION RAND(OO)
C
C     THIS FUNCTION GENERATES UNIFORMLY DISTRIBUTED RANDOM NUMBERS
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      OO=0.
      IY=IX*65539
      IF(IY)5,6,6
    5 IY=IY+2147483647+1
    6 YFL=IY
      YFL=YFL*.4656613E-9
      IX=IY
      RAND=YFL
      RETURN
      END
```

```
      SUBROUTINE TIMNOD
C
C     SUBROUTINE TIMNOD CALCULATES THE TIME THAT EACH A/C IN APROCH
C     WILL ARRIVE AT A NODE.  THE MINIMUM ARRIVAL TIME IS THEN SEARCHED
C     FOR OVER ALL A/C.  THE A/C WITH THE MINIMUM ARRIVAL TIME IS THEN
C     DESIGNATED THE NEXT EVENT A/C (AC) ARRIVING AT THE NEXT EVENT
C     NODE (NODE).
C     DD IS IN FEET.
C     DIST IS IN NAUTICAL MILES.
C     TIM IS INTERNODE TIME.
C     TIME IS TIME OF ARRIVAL AT NEXT NODE (IN TERMS OF MCT).
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      AC=APROCH(1,1)
      CALL APRROW
      INTEGER THN,EN1
      INTEGER EN2
      V=APROCH(1,5)
C     IF THE A/C HAS PASSED THE THRESHOLD, GO TO 5
      IF(APROCH(1,6).GE.THN) GO TO 5
```

```
      AC=APROCH(1,1)
      NODE=APROCH(1,6)+1.
      TIM=(DIST/APROCH(1,5))*3600.
      TIME=TIM+APROCH(1,7)
      GO TO 25
C     IF THE A/C IS ON THE RUNWAY GO TO 10
    5 IF(APROCH(1,6).GT.THN)GO TO 10
      AC=APROCH(1,1)
      NODE=APROCH(1,6)+1.
      IF(APROCH(1,3).EQ.2..AND.INDEX.EQ.11.)TIME=EXITIM
      IF(APROCH(1,3).EQ.2..AND.INDEX.EQ.11.)TIM=TIME-APROCH(1,7)
      IF(APROCH(1,3).EQ.2..AND.INDEX.EQ.11.)GO TO 25
      IF(APROCH(1,3).EQ.1.)TIM=(TDD/APROCH(1,5))*(3600./6076.115)
      IF(APROCH(1,3).EQ.2.)TIM=(DSTENT/APROCH(1,5))*(3600./6076.115)
      TIME=TIM+APROCH(1,7)
      GO TO 25
   10 IF(APROCH(1,3).EQ.1.)GO TO 15
      AC=APROCH(1,1)
      NODE=APROCH(1,6)+1.
      V=SQRT(2.*ACEL*(3600.**2./6076.115     )*DIST+APROCH(1,5)**2.)
      TIM=((V-APROCH(1,5))/ACEL)*(6076.115/3600.)
      TIME=TIM+APROCH(1,7)
      GO TO 25
   15 IF(APROCH(1,6).EQ.EN1)GO TO 20
      IF(APROCH(1,6).EQ.EN2)GO TO 22
      IF(APROCH(1,6).EQ.EN3)GO TO 23
      DUMVEL=-2.*DCEL*(3600.**2./6076.115**2.)*DD+APROCH(1,5)**2.
      IF(DUMVEL.LT.(EXVLMX(APROCH(NROW,2))/1.15)**2.)DUMVEL=(EXVLMX(APRO
     1CH(NROW,2))/1.15)**2.
      V=SQRT(DUMVEL)
      AC=APROCH(1,1)
      NODE=APROCH(1,6)+1.
```

```
      TIM=((V-APROCH(1,5))/(-DCEL))*(6076.115/3600.)
      TIME=TIM+APROCH(1,7)
      GO TO 25
   20 AC=APROCH(1,1)
      NODE=APROCH(1,6)+1.
      DUMVEL=-2.*DCEL*(3600.**2./6076.115**2.)*(DSTEN2-DSTEN1)+APROCH(1,
     15)**2.
      IF(DUMVEL.LT.(EXVLMX(APROCH(NROW,2))/1.15)**2.)DUMVEL=(EXVLMX(APRO
     1CH(NROW,2))/1.15)**2.
      V=SQRT(DUMVEL)
      TIM=((V-APROCH(1,5))/(-DCEL))*(6076.115/3600.)
      TIME=TIM+APROCH(1,7)
      GO TO 25
   22 AC=APROCH(1,1)
      NODE=APROCH(1,6)+1.
      DUMVEL=-2.*DCEL*(3600.**2./6076.115**2.)*(DSTEN3-DSTEN2)+APROCH(1,
     15)**2.
      IF(DUMVEL.LT.(EXVLMX(APROCH(NROW,2))/1.15)**2.)DUMVEL=(EXVLMX(APRO
     1CH(NROW,2))/1.15)**2.
      V=SQRT(DUMVEL)
      TIM=((V-APROCH(1,5))/(-DCEL))*(6076.115/3600.)
      TIME=TIM+APROCH(1,7)
      GO TO 25
   23 AC=APROCH(1,1)
      NODE=APROCH(1,6)+1.
      DUMVEL=-2.*DCEL*(3600.**2./6076.115**2.)*(DSTEN4-DSTEN3)+APROCH(1,
     15)**2.
      IF(DUMVEL.LT.(EXVLMX(APROCH(NROW,2))/1.15)**2.)DUMVEL=(EXVLMX(APRO
     1CH(NRCW,2))/1.15)**2.
      V=SQRT(DUMVEL)
      TIM=((V-APROCH(1,5))/(-DCEL))*(6076.115/3600.)
      TIME=TIM+APROCH(1,7)
```

```
25 DO 55 I=2,NROWAP
   VV=APROCH(I,5)
   IF(APROCH(I,1).EQ.0.)GO TO 60
   DUMAC=AC
   AC=APROCH(I,1)
   CALL APRROW
   AC=DUMAC
   IF(APROCH(I,6).GE.THN) GO TO 30
   AAC=APROCH(I,1)
   NNODE=APROCH(I,6)+1.
   TTIM=(DIST/APROCH(I,5))*3600.
   TTIME=TTIM+APROCH(I,7)
   GO TO 50
30 IF(APROCH(I,6).GT.THN)GO TO 35
   AAC=APROCH(I,1)
   NNODE=APROCH(I,6)+1.
   IF(APROCH(I,3).EQ.2..AND.INDEX.EQ.11.)TTIME=EXITIM
   IF(APROCH(I,3).EQ.2..AND.INDEX.EQ.11.)TTIM=TTIME-APROCH(I,7)
   IF(APROCH(I,3).EQ.2..AND.INDEX.EQ.11.)GO TO 50
   IF(APROCH(I,3).EQ.1.)TTIM=(TDD/APROCH(I,5))*(3600./6076.115)
   IF(APROCH(I,3).EQ.2.)TTIM=(DSTENT/APROCH(I,5))*(3600./6076.115)
   TTIME=TTIM+APROCH(I,7)
   GO TO 50
35 IF(APROCH(I,3).EQ.1.)GO TO 40
   AAC=APROCH(I,1)
   NNODE=APROCH(I,6)+1.
   VV=SQRT(2.*ACEL*(3600.**2./6076.115     )*DIST+APROCH(I,5)**2.)
   TTIM=((VV-APROCH(I,5))/ACEL)*(6076.115/3600.)
   TTIME=TTIM+APROCH(I,7)
   GO TO 50
40 IF(APROCH(I,6).EQ.EN1)GO TO 45
   IF(APROCH(I,6).EQ.EN2)GO TO 47
```

```
      IF(APROCH(I,6).EQ.EN3)GO TO 48
      DUMVEL=-2.*DCEL*(3600.**2./6076.115**2.)*DD+APROCH(I,5)**2.
      IF(DUMVEL.LT.(EXVLMX(APROCH(NROW,2))/1.15)**2.)DUMVEL=(EXVLMX(APRO
     1CH(NROW,2))/1.15)**2.
      VV=SQRT(DUMVEL)
      AAC=APROCH(I,1)
      NNODE=APROCH(I,6)+1.
      TTIM=((VV-APROCH(I,5))/(-DCEL))*(6076.115/3600.)
      TTIME=TTIM+APROCH(I,7)
      GO TO 50
   45 AAC=APROCH(I,1)
      NNODE=APROCH(I,6)+1.
      DUMVEL=-2.*DCEL*(3600.**2./6076.115**2.)*(DSTEN2-DSTEN1)+APROCH(I,
     15)**2.
      IF(DUMVEL.LT.(EXVLMX(APROCH(NROW,2))/1.15)**2.)DUMVEL=(EXVLMX(APRO
     1CH(NROW,2))/1.15)**2.
      VV=SQRT(DUMVEL)
      TTIM=((VV-APROCH(I,5))/(-DCEL))*(6076.115/3600.)
      TTIME=TTIM+APROCH(I,7)
      GO TO 50
   47 AAC=APROCH(I,1)
      NNODE=APROCH(I,6)+1.
      DUMVEL=-2.*DCEL*(3600.**2./6076.115**2.)*(DSTEN3-DSTEN2)+APROCH(I,
     15)**2.
      IF(DUMVEL.LT.(EXVLMX(APROCH(NROW,2))/1.15)**2.)DUMVEL=(EXVLMX(APRO
     1CH(NROW,2))/1.15)**2.
      VV=SQRT(DUMVEL)
      TTIM=((VV-APROCH(I,5))/(-DCEL))*(6076.115/3600.)
      TTIME=TTIM+APROCH(I,7)
      GO TO 50
   48 AAC=APROCH(I,1)
      NNODE=APROCH(I,6)+1.
```

```
      DUMVEL=-2.*DCEL*(3600.**2./6076.115**2.)*(DSTEN4-DSTEN3)+APROCH(I,
     15)**2.
      IF(DUMVEL.LT.(EXVLMX(APROCH(NROW,2))/1.15)**2.)DUMVEL=(EXVLMX(APRO
     1CH(NROW,2))/1.15)**2.
      VV=SQRT(DUMVEL)
      TTIM=((VV-APROCH(I,5))/(-DCEL))*(6076.115/3600.)
      TTIME=TTIM+APROCH(I,7)
   50 IF(TTIME.LT.TIME)TIM=TTIM
      IF(TTIME.LT.TIME)V=VV
      IF(TTIME.LT.TIME)AC=AAC
      IF(TTIME.LT.TIME)NODE=NNODE
      IF(TTIME.LT.TIME)TIME=TTIME
   55 CONTINUE
   60 CALL APRROW
      IF(APROCH(NROW,6).EQ.THN.AND.INDEX.EQ.11.)INDEX=0.
      RETURN
      END
```

```
      SUBROUTINE VLOCTY(I)
C
C     SUBROUTINE VLOCTY(I) CALCULATES A NORMALLY DISTRIBUTED VELOCITY
C     ASSIGNMENT FOR A CLASS I A/C.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      EXTERNAL RAND
      OD=0.
      RR=RAND(OC)
      AL=VAPMU(I)
      W=VAPSD(I)
      Z=SQRT(-2.*ALOG(.5*(1.0-ABS(1.-2.*RR))))
      T3=Z-(2.515517+0.802853*Z+.010328*Z**2.)/(1.+1.432788*Z+0.189269*Z
     1**2.+.001308*Z**3.)
      T2=(RR-0.5)/(ABS(RR-0.5))
      VELASS=AL+T2*W*T3
      RETURN
      END
```

```
      SUBROUTINE RWYCHK
C     RWYCHK CHECKS THE RUNWAY FOR A/C ,IF ANY ARE FOUND IT SETS INDEX=1
C     SW IS A SWITCH TO INDICATE WHEN ENTRANCE VELOCITY IS TO BE ENTERED
C     INTO THE APROCH ARRAY.  WHEN THE SWITCH (SW) = 1, THE ENTRANCE
C     VELOCITY WILL BE ENTERED INTO APROCH.
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      INTEGER ENT,ENT1,ENT2,ENT3,THN
      DO 5 I=1,NRCWAP
      IF(APROCH(I,6).EQ.ENT.OR.APROCH(I,6).EQ.ENT1.OR.APROCH(I,6).EQ.ENT
     12.OR.APROCH(I,6).EQ.ENT3)GO TO 10
    5 CONTINUE
      CALL TCINFO
      CALL APRROW
      APROCH(NROW,5)=VELENT
      SW=1.
    9 GO TO 20
   10 INDEX=1.
      NODE=THN
   20 RETURN
      END
```

```
      SUBROUTINE TOINFO
C
C     SUBROUTINE TOINFO GENERATES NORMALLY DISTRIBUTED TAKEOFF
C     PARAMETERS.  GENERATED ARE. . . ENTRANCE VELOCITY, TAKEOFF
C     VELOCITY, AND ACCELERATION.  TAKEOFF SEPARATION (SEPTO) IS ALSO
C     CALCULATED.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1CMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTC,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      INTEGER DUMTPE
      CO=0.
      RR=RAND(CO)
      M=1
      CALL APRROW
      AL=VENTMU(APROCH(NROW,2))
      W=VENTSD(APROCH(NROW,2))
      GO TO 15
    5 AL=VTOMU(APROCH(NROW,2))
      W=VTOSD(APROCH(NROW,2))
      M=2
```

```
      GO TO 15
   10 AL=ACCMU(APROCH(NROW,2))
      W=ACCSD(APROCH(NROW,2))
      M=3
   15 Z=SQRT(-2.*ALOG(.5*(1.0-ABS(1.-2.*RR))))
      T3=Z-(2.515517+0.802853*Z+.010328*Z**2.)/(1.+1.432788*Z+0.189269*Z
     1**2.+.001308*Z**3.)
      T2=(RR-0.5)/(ABS(RR-0.5))
      GEN=AL+T2*W*T3
      IF(M.EQ.1)VELENT=GEN
      IF(M.EQ.1)GO TO 5
      IF(M.EQ.2)VELTO=GEN
      IF(M.EQ.2)GO TO 10
      ACEL=GEN
      DUM=APROCH(1,6)
      DUMTPE=APROCH(1,2)
      DUMOP=APROCH(1,3)
      DO 20 I=2,10
      IF(APROCH(I,6).GT.DUM.AND.APROCH(I,1).NE.AC)DUM=APROCH(I,6)
      IF(APROCH(I,6).GT.DUM.AND.APROCH(I,1).NE.AC)DUMTPE=APROCH(I,2)
      IF(APROCH(I,6).GT.DUM.AND.APROCH(I,1).NE.AC)DUMOP=APROCH(I,3)
   20 CONTINUE
      IF(DUM.EQ.NODE-1)SEPTO=0.
      IF(DUM.EQ.NODE-1)GO TO 25
      IF(SEPTO.EQ.0.)GO TO 25
      IF(DUMOP.EQ.1)SEPTO=0.
      IF(DUMOP.EQ.1)GO TO 25
      SEPTO=TOSEP(APROCH(NROW,2),DUMTPE)
   25 RETURN
      END
```

```
      SUBROUTINE VVELTO
C
C     SUBROUTINE VVELTO CHECKS TO SEE IF TAKEOFF VELOCITY HAS BEEN
C     ACHIEVED, IF SO VVELTO CHECKS TO SEE IF TAKEOFF SEPARATION HAS
C     BEEN ACHIEVED.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1CMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      INTEGER ENT
      IF(V.LT.VELTO)GO TO 10
      CALL APRROW
      IF(APROCH(NROW+1,3).EQ.1.)GO TO 5
      IF(((APROCH(NROW,6)+1.)-ENT)*DIST*5280..LT.SEPTO)GO TO 10
    5 IF(INDEX.EQ.1.)INDEX=11.
      NCODE=4
   10 RETURN
      END
```

```
      SUBROUTINE DECNOD
C
C     SUBROUTINE DECNOD DECIDES WHETHER OR NOT A MISSED APPROACH IS TO
C     BE EXECUTED DUE TO AN A/C ON THE RUNWAY.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      INTEGER THN,EN1,DN
      INTEGER EN2
      J=0
      K=0
      CALL APRROW
      IF(NROW.EQ.1)GO TO 25
      DO 10 I=1,NROWAP
      IF(APROCH(I,1).EQ.AC)GO TO 15
      IF(APROCH(I,6).GT.DN.AND.APROCH(I,3).EQ.1.)J=I
      IF(APROCH(I,6).GT.THN.AND.APROCH(I,3).EQ.2.)K=I
   10 CONTINUE
      GO TO 25
   15 IF(K.EQ.I)GO TO 30
      I=J
```

```
      XMUTDP=TDDMU(APROCH(I,2))
      XMUDCL=DCELMU(APROCH(I,2))
      XMUVAP=VAPMU(APROCH(I,2))
      XMUEXV=EXVLMX(APROCH(I,2))
C     TTE IS THE ESTIMATED TIME UNTIL EXIT VELOCITY IS REACHED MEASURED
C     FROM THE TOUCHDOWN POINT (IN SECONDS).
      TTE=((XMUEXV-XMUVAP)/(-XMUDCL))*(6076.115/3600.)
C     DTE IS THE ESTIMATED DISTANCE UNTIL EXIT VELOCITY IS REACHED,
C     FROM THE THRESHOLD (IN FEET).
      DTE=XMUTDP+(XMUVAP*TTE)*(6076.115/3600.)-(1./2.)*(XMUDCL)*TTE*TTE
C     XITNOD IS THE ESTIMATED NODE AT WHICH AN EXIT WILL OCCUR.
      IF(DTE.LE.DSTEN1)XITNOD=EN1
      IF(DTE.GT.DSTEN1.AND.DTE.LE.DSTEN2)XITNOD=EN2
      IF(DTE.GT.DSTEN2.AND.DTE.LE.DSTEN3)XITNOD=EN3
      IF(DTE.GT.DSTEN3)XITNOD=EN4
      IF(APROCH(I,6).LT.(XITNOD-1.))NCODE=1
      IF(NCODE.EQ.1)WOTIME=TIME
   25 RETURN
   30 NCODE=1
      WOTIME=TIME
      RETURN
      END
```

```
      SUBROUTINE TDPT
C
C     SUBROUTINE TDPT GENERATES NORMALLY DISTRIBUTED RUNWAY TOUCHDOWN
C     POINTS (TDD) FOR LANDING A/C.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      OO=0.
      RR=RAND(CO)
      CALL APRROW
      AL=TDDMU(APROCH(NROW,2))
      W=TDDSD(APROCH(NROW,2))
      Z=SQRT(-2.*ALOG(.5*(1.0-ABS(1.-2.*RR))))
      T3=Z-(2.515517+0.802853*Z+.010328*Z**2.)/(1.+1.432788*Z+0.189269*Z
     1**2.+.001308*Z**3.)
      T2=(RR-0.5)/(ABS(RR-0.5))
      TDD=AL+T2*W*T3
      RETURN
      END
```

```
      SUBROUTINE TDINFO
C
C     SUBROUTINE TDINFO GENERATES NORMALLY DISTRIBUTED DECELERATION
C     RATES FOR LANDING A/C.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      OO=0
      RR=RAND(OO)
      CALL APRROW
      AL=DCELMU(APROCH(NROW,2))
      W=DCELSD(APROCH(NROW,2))
      Z=SQRT(-2.*ALOG(.5*(1.0-ABS(1.-2.*RR))))
      T3=Z-(2.515517+0.802853*Z+.010328*Z**2.)/(1.+1.432788*Z+0.189269*Z
     1**2.+.001308*Z**3.)
      T2=(RR-0.5)/(ABS(RR-0.5))
      DCEL=AL+T2*W*T3
      DD=(DSTEN1-TDD)/3.
      RETURN
      END
```

```
      SUBROUTINE UPDATE
C
C     SUBROUTINE UPDATE TAKES CARE OF UPDATING ALL STORAGE ARRAYS TO
C     THEIR PRESENT TIME CONDITIONS.  ALL STATISTICS CONCERNING NUMBER
C     OF TAKEOFFS, ETC. ARE COLLECTED IN UPDATE.
C     INDRT IS A RELEASE TIME INDICATOR FOR THE PURPOSE OF SCHEDULING A
C     RELEASE TIME WHEN THE RPAC HAS WAVED OFF.
C     J1 COUNTS THE NUMBER OF OPERATIONS WITHIN A PERIOD OF VARTIM SECS.
C     N1 COUNTS THE NUMBER OF PERIODS OVER WHICH VARIANCE IS CALCULATED.
C     QLAND(I) COUNTS THE NUMBER OF TYPE I LANDINGS.
C     QTO(I) COUNTS THE NUMBER OF TYPE I TAKEOFFS.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1DMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      INTEGER ENT,RPAC
      XAC=AC
      IF(AC.EQ.RPAC.AND.NCODE.EQ.1)INDRT=1
      GO TO (5,20,30,25,35),NCODE
    5 NWO=NWO+1
      IF(IFIX(AC).NE.MCAC)CALL APRROW
```

```
      IF(IFIX(AC).NE.MCAC)CALL APRDEL
      IF(IFIX(AC).NE.MCAC)NCLAS=APROCH(NROW,2)
      IF(IFIX(AC).NE.MCAC)GO TO 16
   10 MCT=MCT+TIM
      XMCT=MCT
      IF(NCODE.EQ.2)EXITIM=TIME
      IF(NCODE.EQ.4)EXITIM=TIME
      CALL APRROW
      NCLAS=APROCH(NROW,2)
      T=APROCH(NROW,3)
      CALL AMCAC
      GO TO 16
   15 CALL APRROW
   16 CALL OUTPUT
      IF(NCODE.EQ.1.OR.NCODE.EQ.3)GO TO 17
      IF(T.EQ.1..AND.NCLAS.EQ.1)QLAND(1)=QLAND(1)+1.
      IF(T.EQ.1..AND.NCLAS.EQ.2)QLAND(2)=QLAND(2)+1.
      IF(T.EQ.1..AND.NCLAS.EQ.3)QLAND(3)=QLAND(3)+1.
      IF(T.EQ.1..AND.NCLAS.EQ.4)QLAND(4)=QLAND(4)+1.
      IF(T.EQ.1..AND.NCLAS.EQ.5)QLAND(5)=QLAND(5)+1.
      IF(T.EQ.2..AND.NCLAS.EQ.1)QTO(1)=QTO(1)+1.
      IF(T.EQ.2..AND.NCLAS.EQ.2)QTO(2)=QTO(2)+1.
      IF(T.EQ.2..AND.NCLAS.EQ.3)QTO(3)=QTO(3)+1.
      IF(T.EQ.2..AND.NCLAS.EQ.4)QTO(4)=QTO(4)+1.
      IF(T.EQ.2..AND.NCLAS.EQ.5)QTO(5)=QTO(5)+1.
      J1=J1+1
      Q=XMCT-VARMCT
      IF(Q.LT.VARTIM)GO TO 17
      VARMCT=MCT
      N1=N1+1
      XNIS(N1)=J1
      J1=0
```

```
   17 IF(AC.NE.RPAC)GO TO 50
      JJ=1
      DO 18 I=2,NROWAP
      IF(APROCH(I,1).EQ.0.)GO TO 19
      IF(APROCH(I,6).LT.APROCH(JJ,6))JJ=I
   18 CONTINUE
   19 RPAC=APROCH(JJ,1)
      IF(NCODE.EQ.1)AC=RPAC
      GO TO 50
   20 NEXT=NEXT+1
      GO TO 10
   25 NTO=NTO+1
      IF(IFIX(AC).EQ.MCAC)GO TO 10
      EXITIM=TIME
      CALL APRROW
      NCLAS=APROCH(NROW,2)
      T=APROCH(NROW,3)
      CALL APRDEL
      GO TO 16
   30 NABTO=NABTO+1
      CALL APRROW
      NCLAS=APROCH(NROW,2)
      CALL OUTPUT
      DO 32 I=1,NROWST
      IF(STACK(I,1).EQ.0.)GO TO 33
   32 CONTINUE
C     HERE WE ENTER THE A/C INTO THE STACK ARRAY.
   33 STACK(I,1)=APROCH(NROW,1)
      STACK(I,2)=APROCH(NROW,2)
      STACK(I,3)=APROCH(NROW,3)
      STACK(I,4)=0.
      STACK(I,5)=APROCH(NROW,7)
```

```
      CALL APRDEL
C     IF THE RPAC ABORTED A NEW RPAC WILL BE SELECTED.
      IF(IFIX(AC).NE.RPAC)GO TO 34
      J=1
      DO 36 I=2,NROWAP
      IF(APROCH(I,1).EQ.0.)GO TO 37
      IF(APROCH(I,6).LT.APROCH(J,6))J=I
   36 CONTINUE
   37 RPAC=APROCH(J,1)
   34 IF(IFIX(AC).NE.MCAC)GO TO 55
      CALL AMCAC
      GO TO 55
   35 IF(IFIX(AC).NE.MCAC)GO TO 45
      DUMAC=AC
      AC=MCAC
      CALL APRROW
      AC=DUMAC
      IF(APROCH(NROW,3).EQ.1.)GO TO 40
      IF(NODE.LE.ENT)GO TO 45
   40 MCT=MCT+TIM
   45 CALL APRROW
      APROCH(NROW,7)=APROCH(NROW,7)+TIM
      APROCH(NROW,6)=NODE
      APROCH(NROW,5)=V
      IF(SW.EQ.1.)APROCH(NROW,5)=VELENT
      IF(INDERR.EQ.1)APROCH(NROW,5)=VELLST
      INDERR=0
      SW=0.
   50 IF(MCT.GE.TIMLIM)CALL SUMARY
   55 NCODE=5
      RETURN
      END
```

```
      SUBROUTINE STKDEL
C
C     SUBROUTINE STKDEL DELETES THE FIRST ROW ENTRIES OF STACK AND
C     MOVES THE OTHER ROWS UP IN ORDER TO CONSERVE STORAGE SPACE.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      CALL STKROW
      STACK(NROW,1)=0.
      I=NROW+1
      DO 5 J=I,NROWST
      IF(STACK(J,1).EQ.0.)GO TO 10
      STACK(J-1,1)=STACK(J,1)
      STACK(J-1,2)=STACK(J,2)
      STACK(J-1,3)=STACK(J,3)
      STACK(J-1,4)=STACK(J,4)
      STACK(J-1,5)=STACK(J,5)
      STACK(J,1)=0.
    5 CONTINUE
   10 RETURN
      END
```

```
      SUBROUTINE APRDEL
C
C     SUBROUTINE APRDEL DELETES THE FIRST ROW ENTRIES OF APROCH AND
C     MOVES THE OTHER ROWS UP IN ORDER TO CONSERVE STORAGE SPACE.
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLOMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      APROCH(NROW,1)=0.
      I=NROW+1
      DO 5 J=I,NROWAP
      IF(APROCH(J,1).EQ.0.)GO TO 10
      APROCH(J-1,1)=APROCH(J,1)
      APROCH(J-1,2)=APROCH(J,2)
      APROCH(J-1,3)=APROCH(J,3)
      APROCH(J-1,4)=APROCH(J,4)
      APROCH(J-1,5)=APROCH(J,5)
      APROCH(J-1,6)=APROCH(J,6)
      APROCH(J-1,7)=APROCH(J,7)
    5 APROCH(J,1)=0.
   10 RETURN
      END
```

```
      SUBROUTINE VELENI
C
C     SUBROUTINE VELENI COMPARES THE VELOCITY OF A LANDING A/C AT THE
C     EXIT RAMP TO THE MAXIMUM SAFE EXIT VELOCITY TO DETERMINE
C     IF AN EXIT CAN BE SAFELY MADE.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,CD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      CALL APRROW
      IF(V-EXVLMX(APROCH(NROW,2)))5,5,10
    5 NCODE=2
      IF(INDEX.EQ.1.)INDEX=11.
   10 RETURN
      END
```

```
      SUBROUTINE AMCAC
C
C     SUBROUTINE AMCAC SELECTS FROM APROCH A MASTER CLOCK AIRCRAFT
C     I.E. AN AIRCRAFT TO KEEP TIME BY.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      INTEGER ENT,RPAC
      REAL MCT,LN
      CALL APRDEL
      IF(APROCH(1,1).EQ.0.)GO TO 25
      DO 5 I=1,NROWAP
      IF(APROCH(I,3).EQ.1..AND.APROCH(I,1).NE.0.)MM=I
      IF(APROCH(I,3).EQ.1..AND.APROCH(I,1).NE.0.)GO TO 15
    5 CONTINUE
      JJ=1
      DO 10 I=2,NROWAP
      IF(APROCH(I,1).EQ.0.)GO TO 11
      IF(APROCH(I,6).GT.APROCH(JJ,6))JJ=I
   10 CONTINUE
   11 MCAC=APROCH(JJ,1)
```

```
14 GO TO 65
15 J=MM+1
   DO 20 I=J,NROWAP
   IF(APROCH(I,1).EQ.0.)GO TO 21
   IF(APROCH(I,6).GT.APROCH(MM,6).AND.APROCH(I,3).EQ.1.)MM=I
20 CONTINUE
21 MCAC=APROCH(MM,1)
24 GO TO 65
25 IF(STACK(1,1).EQ.0.)GO TO 35
   MM=1
   DO 30 I=2,NROWST
   IF(STACK(I,1).EQ.0.)GO TO 31
   IF(STACK(I,5).LT.STACK(MM,5))MM=I
30 CONTINUE
31 AC=STACK(MM,1)
   IF(STACK(MM,3).EQ.1.)LN=1
   IF(STACK(MM,3).EQ.2.)LN=ENT
   TIMLN=MCT
   ERRLST=0.
   CALL APRREG
   CALL STKDEL
   MCAC=AC
   RPAC=AC
   EGMCT=MCT
34 GO TO 65
35 MM=1
   DO 40 I=2,NROWWT
   IF(WAIT(I,4).LT.WAIT(MM,4))MM=I
40 CONTINUE
   N=N+1
   DO 45 I=1,NROWAP
   IF(APROCH(I,1).EQ.0.)GO TO 50
```

```
45 CONTINUE
50 JJ=I
   AC=N
   APROCH(JJ,1)=N
   APROCH(JJ,2)=WAIT(MM,1)
   APROCH(JJ,3)=WAIT(MM,2)
   APROCH(JJ,7)=WAIT(MM,4)
   IF(MCT.GT.WAIT(MM,4))APROCH(JJ,7)=MCT
   IF(WAIT(MM,2).EQ.2.)GO TO 55
   APROCH(JJ,4)=WAIT(MM,3)
   APROCH(JJ,5)=WAIT(MM,3)
   APROCH(JJ,6)=1.
   GO TO 60
55 CALL TCINFO
   APROCH(JJ,4)=VELENT
   APROCH(JJ,5)=VELENT
   APROCH(JJ,6)=ENT
60 IF(WAIT(MM,4).GT.MCT)MCT=WAIT(MM,4)
   IF(MCT.EQ.WAIT(MM,4))TIME=MCT
   EGMCT=APROCH(JJ,7)
   MCAC=N
   RPAC=N
   JCL=WAIT(MM,1)
   T=WAIT(MM,2)
   CALL REPLCE
65 RETURN
   END
```

```
      SUBROUTINE REVUE
C
C     SUBROUTINE REVUE CHECKS ARRIVAL TIMES IN THE WAIT ARRAY TO SEE IF
C     ANY A/C HAVE ARRIVED INTO THE SYSTEM SINCE THE LAST REVUE.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      MM=1
      DO 5 I=2,NROWWT
      IF(WAIT(I,4).LT.WAIT(MM,4))MM=I
    5 CONTINUE
      IF(WAIT(MM,4).LT.TIME)GO TO 10
      IF(STACK(1,1).EQ.0.)IDEX=1.
      IF(STACK(1,1).EQ.0.)GO TO 30
      CALL SEQ
      GO TO 30
   10 N=N+1
      CALL STKREG
      CALL REPLCE
   15 MM=1
```

```
      DO 20 I=2,NROWWT
      IF(WAIT(I,4).LT.WAIT(MM,4))MM=I
   20 CONTINUE
      IF(WAIT(MM,4).GT.TIME)GO TO 25
      N=N+1
      CALL STKREG
      CALL REPLCE
      GO TO 15
   25 CALL SEQ
   30 RETURN
      END
```

```
      SUBROUTINE APRREG
C
C     SUBROUTINE APREG REGISTERS AN ARRIVING A/C AND ITS PARAMETERS IN
C     APROCH.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTC,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      REAL LN
C     HERE WE'RE LOOKING FOR THE FIRST VACANT ROW IN APROCH.
      DO 5 I=1,NROWAP
      IF(APROCH(I,1).EQ.0.)GO TO 10
    5 CONTINUE
   10 DUMAC=AC
      AC=RAC
C     WHAT ROW IN STACK IS THE RAC STORED IN?
      CALL STKROW
      AC=DUMAC
      APROCH(I,1)=STACK(NROW,1)
      APROCH(I,2)=STACK(NROW,2)
      APROCH(I,3)=STACK(NROW,3)
```

```
APROCH(I,4)=STACK(NROW,4)
APROCH(I,5)=STACK(NROW,4)
APROCH(I,6)=LN
APROCH(I,7)=TIMLN
RETURN
END
```

```
      SUBROUTINE STKREG
C
C     SUBROUTINE STKREG REGISTERS AN ARRIVING A/C AND ITS PARAMETERS IN
C     STACK.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLOMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      DO 5 I=1,NROWST
      IF(STACK(I,1).EQ.0.)GO TO 10
    5 CONTINUE
   10 STACK(I,1)=N
      STACK(I,2)=WAIT(MM,1)
      STACK(I,3)=WAIT(MM,2)
      STACK(I,4)=WAIT(MM,3)
      STACK(I,5)=WAIT(MM,4)
      RETURN
      END
```

```
      SUBROUTINE APRROW
C
C     SUBROUTINE APRROW SEARCHES APROCH TO DETERMINE WHAT ROW AC (THE
C     A/C UNDER PRESENT CONSIDERATION AND ITS PARAMETERS ARE STORED
C     IN.  THE ROW NUMBER IS RETURNED AS NROW.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TOP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XOD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTC,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      DO 5 I=1,NROWAP
      IF(APRCCH(I,1).EQ.AC)GO TO 10
    5 CONTINUE
   10 NROW=I
      RETURN
      END
```

```
      SUBROUTINE STKROW
C
C     SUBROUTINE STKROW SEARCHES STACK TO DETERMINE WHAT ROW A
C     PARTICULAR A/C AND ITS PARAMETERS ARE STORED IN.  THE ROW
C     NUMBER IS RETURNED AS NROW.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      DO 5 I=1,NROWST
      IF(STACK(I,1).EQ.AC)GO TO 10
    5 CONTINUE
      GO TO 15
   10 NROW=I
   15 RETURN
      END
```

```
      SUBROUTINE SEQ
C
C     SUBROUTINE SEQ SEQUENCES THE A/C ONTO THE RUNWAY IN A MANNER
C     DETERMINED BY WHATEVER ALGORITHM THE USER CHOOSES TO USE. THIS
C     PROGRAM IS CONCERNED ONLY WITH LANDINGS AND TAKEOFFS AS THE
C     RUNWAY SEES THEM SO ARRIVALS AND DEPARTURES ARE GENERATED
C     RANDOMLY AND TAKEN ON A FIRST COME FIRST SERVED BASIS.
C     ALSO CONTAINED IN SEQ ARE THE CONTROLLER ALGORITHMS FOR
C     DETERMINING A/C SPACING.
C     INDRT IS DEFINED IN UPDATE.
C     M IS THE ROW IN STACK THAT THE RELEASING A/C IS STORED IN.
C     NRELPT IS THE NODE THAT THE RPAC MUST BE AT FOR A RELEASE TO OCCUR.
C     RAC IS THE RELEASING A/C.
C     SEP IS THE TIME SEPARATION BETWEEN A/C IN SECONDS.
C     SS IS THE SEPARATION USED BY THE CONTROLLER FOR SEQUENCING A/C.
C     XDISTIS THE DISTANCE SEPARATION BETWEEN A/C IN N. MILES.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT,NP
      INTEGER RPAC
```

```
      M=1
C     HERE WE'RE LOOKING FOR THE A/C WITH THE LOWEST ARRIVAL TIME.
      DO 5 I=2,NROWST
      IF(STACK(I,5).LT.STACK(M,5).AND.STACK(I,1).NE.0.)M=I
    5 CONTINUE
      RAC=STACK(M,1)
      IF(STACK(M,3).NE.2.)GO TO 10
      DUMAC=AC
      AC=RPAC
      CALL APRROW
      AC=DUMAC
C     HERE A PHANTOM A/C IS BEING RELEASED-IT IS ASSIGNED THE VELOCITY
C     OF THE A/C IMMEDIATELY PRECEDING IT (THE RPAC).
      STACK(M,4)=APROCH(NROW,4)
      GO TO 11
   10 DUMAC=AC
      AC=RPAC
      CALL APRROW
      AC=DUMAC
C     UP TO THIS POINT WE KNOW ABOUT THE RELEASING A/C AND THE RPAC.
   11 IF(APROCH(NROW,3).EQ.1.)GO TO 25
      IF(STACK(M,3).EQ.1.)GO TO 15
C     RPAC IS A T/O, RAC IS A T/O
      SEP=RWYTOT(APROCH(NROW,2))
      GO TO 40
   15 IF(APROCH(NROW,4).LT.STACK(M,4))GO TO 20
C     RPAC IS A T/O, RAC IS A LANDING
      SEP=(2.25/APROCH(NROW,5))*3600.
      XDIST=2.25
      GO TO 41
   20 SEP=(2.25/STACK(M,4)+F*(1/APROCH(NROW,5)-1./STACK(M,4)))*3600.
      GO TO 40
```

```
   25 IF(STACK(M,3).EQ.1.)GO TO 30
C     RPAC IS LANDING, RAC IS A T/O
      SEP=RWYLDT(APROCH(NROW,2))
      GO TO 40
   30 IF(APROCH(NROW,4).LT.STACK(M,4))GO TO 35
C     RPAC IS LANDING, RAC IS LANDING
      CALL CNTRL1
      SEP=(SS/APROCH(NROW,4))*3600.
      GO TO 40
   35 CALL CNTRL1
      SEP=(SS  /STACK(M,4)+F*(1/APROCH(NROW,5)-1./STACK(M,4)))*3600.
   40 XDIST=SEP*APROCH(NROW,5)/3600.
   41 NRELPT=IFIX(XDIST*NP)+2
      IF(NODE.LE.NRELPT+1)RELTIM=EGMCT+SEP
      IF(NODE.GT.NRELPT+1)RELTIM=STACK(M,5)
      IF(INDRT.EQ.1.AND.NODE.GT.NRELPT+1)RELTIM=TIME
      IF(INDRT.EQ.1.AND.NODE.GT.NRELPT+1)INDRT=0
      RETURN
      END
```

```
      SUBROUTINE THCHK
C
C     SUBROUTINE THCHK CHECKS TO SEE IF ANY REAL A/C ARE WITHIN 2 MILES
C     OF THE THRESHOLD.  IF NO A/C ARE WITHIN 2 MILES TH=0, IF SO TH=1.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTC,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WDTIME,INDERR,EHRMAX(5),EHRMIN(5 ),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      INTEGER THN
      REAL NP
      TH=0.
      DO 5 I=1,NROWAP
      IF(APROCH(I,6).GT.(THN-2.*NP).AND.APROCH(I,6).LT.THN.AND.APROCH(I,
     11).NE.AC.AND.APROCH(I,1).NE.0..AND.APROCH(I,3).NE.2.)GO TO 10
    5 CONTINUE
    9 GO TO 15
   10 TH=1.
   15 RETURN
      END
```

```
      SUBROUTINE CNTRL1
C
C     THIS RCUTINE ATTEMPTS TO INTRCDUCE SOME VARIABILITY INTO THE
C     SEPARATICN CRITERIA USED BY A CONTROLLER TO SEQUENCE LANDING A/C.
C     W IS THE STANDARD DEVIATION USED IN THE NORMAL GENERATOR.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDC,DSTEN1,DSTEN2,DSTENT,NROWAP,NRCWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCCDE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      OO=0.
      RR=RAND(OO)
      AL=S
      W=.2/6.0
      Z=SQRT(-2.*ALOG(.5*(1.0-ABS(1.-2.*RR))))
      T3=Z-(2.515517+0.802853*Z+.010328*Z**2.)/(1.+1.432788*Z+0.189269*Z
     1**2.+.001308*Z**3.)
      T2=(RR-0.5)/(ABS(RR-0.5))
      SS=AL+T2*W*T3
      IF(SS.LT.SMIN-.1)SS=SMIN-.1
      IF(SS.GT.SMIN+.1)SS=SMIN+.1
      RETURN
      END
```

```
      SUBROUTINE ERROR
C
C     SUBROUTINE ERROR FINDS OUT WHETHER DISTANCE SEPARATION,SMIN, IS
C     BEING MAINTAINED AND INTRODUCES ERROR VELOCITIES FOR APPROACHING
C     REAL AIRCRAFT.
C     INDERR IS AN INDICATOR TELLING UPDATE THAT A VELOCITY ERROR HAS
C     BEEN INTRODUCED.  IF INDERR=1 AN ERROR HAS BEEN INTRODUCED.
C     LROW IS THE ROW NUMBER OF THE CLOSEST REAL A/C PRECEDING AC.
C     SEPDIS IS THE INCREASE IN LONGITUDINAL SEPARATION (IN N. MI.)
C     ACHIEVABLE IN DIFF SECONDS.
C     WOTIME IS THE MASTER CLOCK TIME IF AND WHEN A WAVEOFF OCCURS.
C     Z IS THE NUMBER OF NODES BETWEEN TWO A/C.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,CD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL NP
      CALL APRRCW
      IF(APROCH(NROW,3).EQ.2.)GO TO 25
      INDERR=1
      IF(NROW.EQ.1)GO TO 20
      K=NROW-1
```

```
C      ARE THERE REAL A/C IN FRONT OF AC?
       DO 5 I=1,K
       IF(APROCH(I,3).EQ.1.)J=I
       IF(APROCH(I,3).EQ.1.)GO TO 6
     5 CONTINUE
       GO TO 20
     6 DO 8 I=J,K
       IF(APROCH(I,3).EQ.1)LROW=I
     8 CONTINUE
       IF(APROCH(LROW,6).GE.THN)GO TO 20
       Z=APROCH(LROW,6)-FLOAT(NODE)
       ZZDIS=(1./NP)*Z
       IF(ZZDIS.GE.SMIN.AND.TIME.GE.APROCH(LROW,7))GO TO 20
       IF(ZZDIS.EQ.SMIN.AND.TIME.LT.APROCH(LROW,7))GO TO 9
       IF(ZZDIS.LT.SMIN.AND.TIME.GT.APROCH(LROW,7))GO TO 10
     9 WOTIME=TIME
       NCODE=1
       GO TO 25
    10 DIFF=TIME-APROCH(LROW,7)
       SEPDIS=(APROCH(LROW,5)*DIFF)/3600.
       IF(Z/NP+SEPDIS.LT.SMIN)GO TO 9
    20 CALL VELCOR
    25 RETURN
       END
```

```
      SUBROUTINE VELCOR
C
C     SUBROUTINE VELCOR COMPUTES A CORRECTION VELOCITY (CALLED VELLST)
C     BASED ON THE DEVIATION FROM ASSIGNED VELOCITY OF AN A/C ON FINAL
C     APPROACH.
C     EHR IS THE ERROR HALF RANGE OR THE PLUS MINUS DEVIATION FROM
C     ASSIGNED APPROACH VELOCITY.
C     EHRMAX(I) IS THE MAX EHR EXPECTED FOR A CLASS I A/C.
C     EHRMIN(I) IS THE MIN EHR EXPECTED FOR A CLASS I A/C.
C     SDMAX IS THE MAX POSSIBLE S.D. FOR A PARTICULAR CLASS A/C.
C     SDMIN IS THE MIN POSSIBLE S.D. FOR A PARTICULAR CLASS A/C.
C
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTC,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      EXTERNAL RAND
      OO=0
      RR=RAND(OO)
      AL=APROCH(NROW,4)
      SDMAX=EHRMAX(APROCH(NROW,2))/3.
      SDMIN=EHRMIN(APROCH(NROW,2))/3.
      IF(APROCH(NROW,5).GE.APROCH(NROW,4)-EHRMIN(APROCH(NROW,2)).AND.
```

```
1   APROCH(NROW,5).LE.APROCH(NROW,4)+EHRMIN(APROCH(NROW,2)))W=SDMIN
 IF(APROCH(NROW,5).GT.APROCH(NROW,4)+EHRMIN(APROCH(NROW,2)).AND.
1   APROCH(NROW,5).LE.APROCH(NROW,4)+EHRMAX(APROCH(NROW,2)).OR.
1   APROCH(NROW,5).GE.APROCH(NROW,4)-EHRMAX(APROCH(NROW,2)).AND.
1   APROCH(NROW,5).LT.APROCH(NROW,4)-EHRMIN(APROCH(NROW,2)))W=SDMAX
 Z=SQRT(-2.*ALOG(.5*(1.0-ABS(1.-2.*RR))))
 T3=Z-(2.515517+0.802853*Z+.010328*Z**2.)/(1.+1.432788*Z+0.189269*Z
1**2.+.001308*Z**3.)
 T2=(RR-0.5)/(ABS(RR-0.5))
 VELLST=AL+T2*W*T3
 IF(VELLST.GT.APROCH(NROW,4)+EHRMAX(APROCH(NROW,2)))VELLST=APROCH(N
1ROW,4)+EHRMAX(APROCH(NROW,2))
 IF(VELLST.LT.APROCH(NROW,4)-EHRMIN(APROCH(NROW,2)))VELLST=APROCH(N
1ROW,4)-EHRMIN(APROCH(NROW,2))
 RETURN
 END
```

```
      SUBROUTINE OUTPUT
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NRCWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCCDE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,QLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      ZTIME=(MCT/3600.)
      IF(NCODE.EQ.2.OR.NCODE.EQ.4)ZTIME=(EXITIM/3600.)
      IF(NCODE.EQ.1)ZTIME=(WOTIME/3600.)
      SYSTEM=0.
      DO 5 I=1,NROWAP
      IF(APROCH(I,1).NE.0..AND.APROCH(I,1).NE.AC)SYSTEM=SYSTEM+1.
      IF(APROCH(I,1).EQ.0.)GO TO 10
    5 CONTINUE
   10 DO 15 I=1,NRCWST
      IF(STACK(I,1).NE.0.)SYSTEM=SYSTEM+1.
      IF(STACK(I,1).EQ.0.)GO TO 20
   15 CONTINUE
   20 GO TO (25,35,45,55),NCODE
   25 WRITE(6,30)ZTIME,NCLAS,SYSTEM
   30 FORMAT(T9,F7.3,T25,'WAVEOFF',T50,I2  ,T78,F3.0,//)
      GO TO 65
   35 WRITE(6,40)ZTIME,NCLAS,SYSTEM
```

```
40 FORMAT(T9,F7.3,T25,'LANDING',T50,I2  ,T78,F3.0,//)
   GO TO 65
45 WRITE(6,50)ZTIME,NCLAS,SYSTEM
50 FORMAT(T9,F7.3,T24,'ABORTED T/O',T50,I2  ,T78,F3.0,//)
   GO TO 65
55 WRITE(6,60)ZTIME,NCLAS,SYSTEM
60 FORMAT(T9,F7.3,T25,'TAKE OFF',T50,I2  ,T78,F3.0,//)
65 RETURN
   END
```

```
      SUBROUTINE SUMARY
      COMMON APROCH(12,7),STACK(35,5),WAIT(10,4),VENTMU(5),PTMLD(5),V
     1ENTSD(5),VTOMU(5),VTOSD(5),ACCMU(5),ACCSD(5),TOSEP(5,5),TDDMU(5),T
     1DDSD(5),DCELMU(5),DCELSD(5),EXVLMX(5),VAPMU(5),VAPSD(5),PTMTO(5),
     1PTETO(5),PTELD(5),SIGMTO(5),SIGETO(5),SIGML(5),SIGEL(5),ALT
     1OMN(5),ALLDMN(5),PMRTO(5),PERTO(5),PMRLD(5),RWYTOT(5),RWYLDT(5),
     1MCT,LN,DN,THN,TDP,EN1,DUMTPE,AC,EN2,ENT,ENT1,ENT2,ENT3,ENT4,TH,S,F
     1,NP,XDD,DSTEN1,DSTEN2,DSTENT,NROWAP,NROWST,NROWWT,
     1IX,IY,NODE,ARTMIN,TIME,TIM,VELASS,INDEX,IDEX,NROW,V,VELENT,VELTO,A
     1CEL,DCEL,SEPTO,NCODE,MAP,TDD,DD,NEXT,NTO,NABTO,NWO,MCAC,RPAC,EGMCT
     1,MM,TIMLN,RELTIM,PERLD(5),DIST,TIMLIM,N,RAC,NCLAS,SMIN,EXITIM,SW
     1,WOTIME,INDERR,EHRMAX(5),EHRMIN(5),VELLST,BEGMCT,INDRT,SS
     1,XNIS(100),N1,VARMCT,VARTIM,J1,ZZZ,GLAND(5),QTO(5)
     1,DSTEN4,DSTEN3,EN3,EN4
      REAL MCT
      A=BEGMCT/3600.
      B=MCT/3600.
      C=B-A
      NTOT=NEXT+NTO
      SUMX=0.
      SUMXSQ=0.
      DO 1 I=1,N1
      SUMXSQ=SUMXSQ+XNIS(I)**2.
      SUMX=SUMX+XNIS(I)
    1 CONTINUE
      XN1=N1
      VAR=(SUMXSQ-(SUMX**2./XN1))/(XN1-1.)
      SD=SQRT(VAR)
    2 WRITE(6,5)
    5 FORMAT(T14,'SIMULATION SUMMARY',//)
      WRITE(6,10)A,B,C,NWO,NABTO,NEXT,NTO,NTOT
   10 FORMAT(T9,'BEGINNING TIME OF SIMULATION WAS ',F7.2,' HOURS.',/,
```

```
   1T9,'END TIME OF SIMULATION WAS ',F7.2,' HOURS.',/,
   1T9,'TOTAL TIME SIMULATED WAS ',F5.2,' HOURS.',/,
   1T9,'TOTAL NUMBER OF MISSED APPROACHES WAS ',I2,'.',/,
   1T9,'TOTAL NUMBER OF ABORTED TAKEOFFS WAS ',I2,'.',/,
   1T9,'TOTAL NUMBER OF LANDINGS WAS ',I3,'.',/,
   1T9,'TOTAL NUMBER OF TAKEOFFS WAS ',I3,'.',/,
   1T9,'TOTAL NUMBER OF OPERATIONS WAS ',I3,'.')
    DO 20 I=1,5
    WRITE(6,12)I,QLAND(I)
 12 FORMAT(T9,'TOTAL NUMBER OF TYPE',I2,' LANDINGS WAS',F4.0,'.')
    WRITE(6,13)I,QTO(I)
 13 FORMAT(T9,'TOTAL NUMBER OF TYPE',I2,' TAKEOFFS WAS',F4.0,'.')
 20 CONTINUE
    WRITE(6,11)VARTIM,VAR,VARTIM,SD
 11 FORMAT(T9,'VARIANCE OF OPERATIONS PER ',F5.0,' SECONDS WAS ',F12.3
   1,'.',/,T9,'STANDARD DEVIATION OF OPERATIONS PER ',F5.0,' SECONDS W
   1AS ',F12.3,'.')
    STOP
 15 RETURN
    END
```

The vita has been removed from
the scanned document

AN ANALYSIS OF RUNWAY CAPACITY

BY

Charles F. Booth

(ABSTRACT)

The objective of this study was to develop an airport simulator and present statistical techniques with which the output may be meaningfully interpreted. The system modeled includes the runway and its associated approach and departure path airspace, including a holding stack.

As an example of how the simulator may be used and its results interpreted, a regression analysis was performed on the simulator output in order to arrive at equations with which to predict runway capacity from a knowledge of the independent variables under study.

Before any analysis could take place it was necessary to know how much data should be collected through simulation to insure statistically sound results. The combinations of levels of the independent variables were selected at random and the number of replications of the simulation at each combination of levels was defined in accordance with prescribed α and β values.

A complete statistical analysis was also performed after data collection to ensure that the desired power of the regression equations was present.