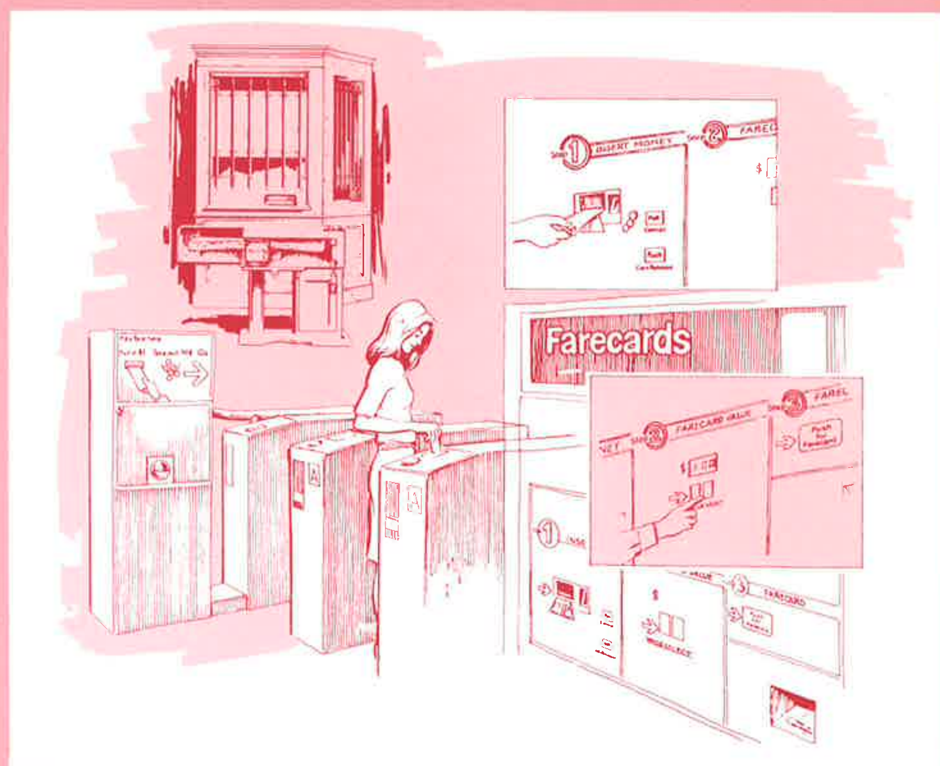
UMTA-MA-06-0153-85-1
DOT-TSC-UMTA-84-12

U.S. Department
of Transportation
**Urban Mass
Transportation
Administration**

A Reliability-Based Model to Analyze the Performance and Cost of a Transit Fare Collection System

Transportation Systems Center
Cambridge MA 02142

June 1985
Final Report



UMTA Technical Assistance Program

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
UMTA-MA-06-0153-85-1		PB85 248672/AS

4. Title and Subtitle
A RELIABILITY-BASED MODEL TO ANALYZE THE PERFORMANCE AND COST OF A TRANSIT FARE COLLECTION SYSTEM

5. Report Date
June 1985

6. Performing Organization Code
DTS-65

7. Author(s)
David I. Heimann

8. Performing Organization Report No.
DOT-TSC-UMTA-84-12

9. Performing Organization Name and Address
U.S. Department of Transportation
Research and Special Programs Administration
Transportation Systems Center
Cambridge MA 02142

10. Work Unit No. (TRAIS)
UM476/R4638

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
U.S. Department of Transportation
Urban Mass Transportation Administration
Office of Technical Assistance
Washington DC 20590

13. Type of Report and Period Covered
Final Report
July 1982 - Jan 1984

14. Sponsoring Agency Code
URT-10

15. Supplementary Notes

16. Abstract

The collection of transit system fares has become more sophisticated in recent years, with more flexible structures requiring more sophisticated fare collection equipment to process tickets and admit passengers. However, this new and complex equipment has often been plagued by reliability problems, frequently resulting in significant passenger congestion and delay. Development efforts are underway to improve reliability. It is uncertain, however, to what extent reliability needs to be improved and how much the improvement will cost. Attempting either too small or too large an improvement in equipment reliability or maintainability may waste valuable transit funds, and may not even solve the underlying problem. A way is needed to determine the dependability (i.e., the passenger congestion and delay arising from the system) and cost of a fare collection system, given the passenger demand, the equipment capacity, reliability, and maintainability, and the various system costs.

This report discusses fare collection dependability analysis, and how transit systems can use it to make more effective investment decisions in selecting fare collection systems which best fit their needs, minimize costs, and provide effective service to passengers. Software implementing such analysis is available in the form of user-friendly computer models, also described in this report. Various types of analyses are described: evaluation (how well is the given system doing?), sensitivity analysis (how would system performance be affected if a change were made?), and tradeoff analysis (what tradeoffs can be made among system values without affecting overall performance?).

The simulation and analytical models to carry out fare collection system dependability and cost analysis are presented, including their technical approach and data requirements. Sample fare collection dependability and cost analyses using data based on actual transit systems are shown, and results and conclusions are discussed.

17. Key Words
Fare Collection, Modeling, Simulation, Transit, Transit Planning, Cost Analysis, Performance Analysis, Cost/Benefit, Reliability, Queuing Analysis, Passenger Flow, Passenger Delay

18. Distribution Statement
DOCUMENT IS AVAILABLE TO THE PUBLIC THROUGH THE NATIONAL TECHNICAL INFORMATION SERVICE, SPRINGFIELD, VIRGINIA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
UNCLASSIFIED	UNCLASSIFIED	120	$16.95

PREFACE

This report and the model it describes are part of the ongoing program in automated fare collection and transit revenue research supported by the Transportation Systems Center (TSC) and the Urban Mass Transportation Administration (UMTA). The work was sponsored by the Office of Systems Engineering, within UMTA's Office of Technical Assistance.

The need for developing a reliability-based performance and cost analysis technique was identified in conjunction with the American Public Transit Association (APTA) Fare Collection Reliability Liaison Board, which consists of representatives from the rail transit systems within APTA and provides guidance to the TSC Transit Revenue program. The objective of this study is to develop a technique which can be used by transit systems in specifying and procuring fare collection equipment.

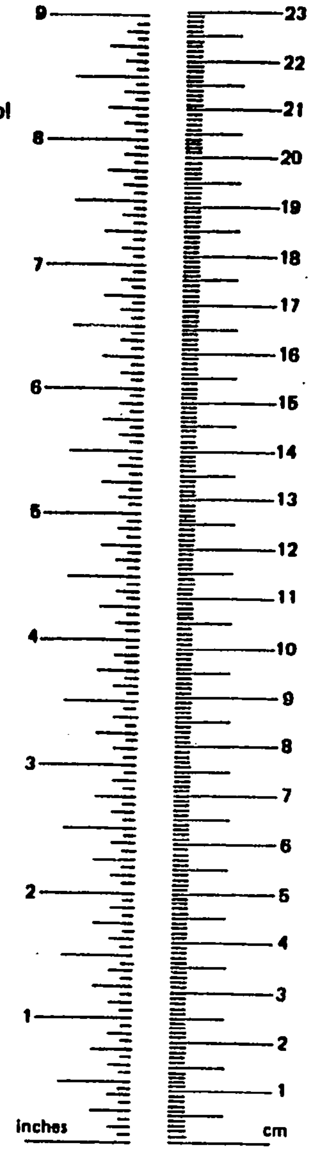
The author wishes to acknowledge the generosity of the Miami Dade County Transit Authority and the Metropolitan Atlanta Rapid Transit Authority in making available specification, performance, and cost data for the sample analyses in this report. While the analyses themselves do not necessarily represent either MDCTA or MARTA, they are designed to be, thanks to the supplied data, representative of situations which arise in actual practice.

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in. = 2.54 cm (exactly). For other exact conversions and more detail tables see NBS Misc. Publ. 286, Units of Weight and Measures. Price $2.25 SD Catalog No. C13 10 286.



Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	36	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

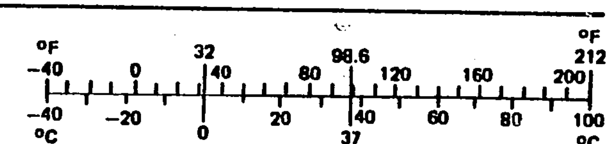


TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

EXECUTIVE SUMMARY

The collection of transit system fares has been receiving increased attention as fares rise and Federal operating subsidies decrease. Transit authorities are becoming more concerned about ways to maximize revenue and minimize costs while providing equitable fares and reliable, convenient service for passengers. Fare collection methods have a significant impact on total transit costs, amount of revenue generated, and passenger service. One study showed that fare collection costs range from 7% to 31% of passenger revenue at rail transit systems. Moreover, revenues generated from fares can vary from 40% to 90% of total transit costs. Fare collection systems must therefore be selected only after careful examination of their cost, revenue, and service effects.

Automated fare collection offers the potential for reducing costs by minimizing the need for personnel to perform cumbersome, repetitive functions. However, the newer and more complex a piece of equipment, the more likely it is to have frequent failures, which can lead to significant passenger delay, lower throughput capacity, and general frustration. Efforts are underway to increase the reliability of automated fare collection equipment. However, it becomes imperative to know just how much of a reliability increase is required, as the amount of reliability increase contemplated makes a significant difference in the cost of such an effort, as well as its likelihood of success.

Furthermore, because of the significant cost of fare collection equipment, the number of equipment units to acquire for a given station also becomes quite important. Too many units can increase total system cost considerably, while too few can lead to significant passenger congestion and delay problems. In addition, it is important to reasonably assess total system costs, and thus control them by comparing the costs and passenger performance for various possible system specifications.

This report describes an analysis technique designed to help transit systems to make more effective investment decisions in selecting fare collection methods, systems, and equipment to best fit their needs, minimize costs, and provide equitable and convenient service to passengers. Software implementing this analysis technique is available in the form of a user-friendly computer model. The model is designed to be used by existing transit systems, as they enhance or expand their fare collection systems, as well as by new transit systems, as they consider their fare collection needs. In either case, the model can uncover important cost-saving or service-enhancing implications in station design.

The model allows transit systems to assess the performance and cost of various fare collection systems, including those with entry processing only, those with both entry and exit processing, and barrier-free systems. A transit authority, consultant, or analyst can use this software to examine the fare collection system in various useful ways:

- (Fare collection system evaluation) -- An existing or proposed fare collection system can be evaluated to see if current or proposed specifications on sizing, reliability, and other features will lead to acceptable performance and cost.

- (Sensitivity analysis) -- Proposed changes in specifications can be analyzed to determine their impact on cost and performance, in particular whether the changes will bring (or keep) the cost and performance within acceptable levels.

- (Specification determination) -- Specifications necessary to meet given cost and performance constraints can be determined.

- (Tradeoff analysis) -- Tradeoffs can be made between different specifications, tightening one specification while relaxing another; between various types of costs, decreasing one cost while increasing another; or between cost and performance.

Such analyses can uncover very useful findings. For example, sample analyses shown in this report, conducted on a derived fare collection system based on data from actual systems, found the following:

- The equipment reliability specifications could be sharply reduced, by nearly an order of magnitude, without significantly affecting performance.

- A proposed minor reliability improvement at some additional acquisition cost turned out to decrease corrective maintenance costs to the extent that the total costs decreased, while performance meanwhile improved.

- o Unless the equipment reliability is very poor, a spares margin of only one additional unit (above the minimum number necessary to process the average peak-hour arrival rate of passengers in the absence of failures) was enough to yield acceptable performance.

The fare collection analysis software can thus provide useful insights to a variety of purposes, such as:

Determination of the number of machine units to deploy at a station.
Determination of reliability and maintainability specifications for fare collection equipment.
Assessing the impact of changes in passenger demand.
Evaluating changes in maintenance policies.
Evaluating changes in fare collection procedures.

The technical approach of the software is to model the operation of the fare collection system as a multiple-server queue, with passengers as customers, machine units as servers, and on a first-come-first-served service discipline. A key facet of the approach, and the reason that off-the-shelf queuing models cannot be used, is that the number of servers (machine units) changes as the machine units fail and are repaired.

The performance measures consist of passenger congestion (queue length) and passenger delay time. These are expressed as average values, as well as probability frequency distributions. The cost measures consist of equipment acquisition costs, spares costs, equipment operating costs, scheduled-maintenance costs, and corrective-maintenance costs, all of which are computed on an annualized basis.

Three kinds of input data are required for the fare-collection analysis; hardware data, passenger flow data, and cost data (if cost analyses are desired). The hardware and passenger-flow data required are:

Passenger arrival rate
Group size (optional)
Passenger processing rates
Failure rates

Repair times
Number of machine units
Division of passenger flow (for multiple service areas only)

For cost analyses, the required input data are:

Acquisition cost per unit
Useful life of the unit
Discount rate
Spares ratio
Operating cost per unit
Annual hours of scheduled maintenance
Hourly pay rate for repair personnel
Annual passenger volume at the fare collection area

To demonstrate the use of the software, sample fare collection system assessments are described. These analyses are based on passenger demand, equipment performance, reliability and maintainability, and cost data from actual transit systems. Passenger-performance and cost results are obtained, and fare collection system evaluations, sensitivity analyses, specification determinations, and tradeoff analyses based on these results are described. Conclusions of these analyses are given. Among the outputs of the analysis are a sensitivity graph showing the effect of equipment reliability on passenger delay, and an equivalent-cost tradeoff graph, which shows, for various increases in equipment reliability, by how much the resulting equipment acquistion cost can increase without increasing the overall annual system cost.

Further information on the analysis software, including source code, is available from the author or the Transportation Systems Center upon request.

1. INTRODUCTION

The collection of transit system fares has become more sophisticated in recent years, as transit authorities turn to more flexible fare structures. Instead of a single flat fare to use the system, the fare now often depends on the passenger's origin and destination points. This is done for various reasons, such as to make the fares more equitable in relation to the actual distance travelled on the system, to allow for fare advantages to certain areas or to certain types of passengers (such as senior citizens or students), or to allow for accommodation among the various governmental bodies and other groups which support the system (1,2,3,4).

In order to collect the fares on a system with such a complex fare structure, a transit property can no longer use a simple coin- or token-activated turnstyle. More sophisticated equipment is necessary to ascertain boarding and exiting information for each passenger and to encode it on tickets for the necessary processing, to vend tickets and collect money, etc. This more sophisticated function could conceivably be done by human beings. However, in most systems to use personnel to the extent that would be necessary would be too costly. Therefore, transit systems have turned to more sophisticated fare collection machinery, using data processing and electronics, to operate the fare collection system (1,2,4).

However, a major problem arises in the use of this equipment, that of reliability. Quite often, the more complex a piece of equipment, and the newer its technology, the more likely it is to have relatively frequent failures (due to some extent to electronic complexity, but mostly due to mechanical aspects such as ticket transport, coin and bill detection and transport, etc.). Transit systems using this equipment have indeed often experienced high failure rates, leading to significant passenger delay, passenger pass-throughs (i.e., the emergency gates are opened and passengers are allowed into the system with the fare either collected manually or not at all), lower throughput capacity, and general frustration (5,6). Efforts are underway to increase the reliability of fare collection equipment, through better off-the-shelf components, more reliable components, redundancy techniques, or improved maintenance procedures (5,6,7,8). The question that arises, however, is by just how much should the reliability be improved? While improvements in reliability are necessary, they are also expensive, both in terms of money and time. Under some circumstances, such as the ones shown in the following examples, equipment improvements may not be cost-effective:

- The wrong service area is being improved: Quite clearly, one should not go to any expense to improve the reliability of a subsystem whose failures are not strongly impeding the system (e.g., if most of the delay occurs at the ticket vendors, don't improve the gates). However, the process leading to a delay can be subtle. For example, suppose most of the delay in a fare collection system presently occurs at the ticket vendors, and occurs there because the ticket-vendor reliability is poor. One might expect that system delay could be reduced by improving the ticket-vendor reliability. Suppose, however, that all passengers must pass through the gates, and that the gates, while not showing significant delays, are presently processing just about as many passengers as they can handle. If the reliability of the vendors is improved, what may happen is that the passenger flow to the gates may increase beyond their capacity to process it, causing large delays at the gate area. In other words, the effort and expense to improve the vendor reliability would be ineffective in this case, since it would just shift the delay from the ticket vendors to the gates.

- The reliability is improved too much: As the reliability of a subsystem improves, its frequency of failure of course decreases. After some point, it no longer fails often enough for the failures to significantly affect system operation. Any expense spent to further improve reliability beyond this point, even if successful, is a wasted expense.

- Measures other than reliability improvement may be more efficient: Improving the equipment reliability is not the only way to reduce the impact of failures on system performance. Faster recovery times in case of failures (i.e., maintainability or recoverability) or having more units available for service (i.e., redundancy) may improve system performance even with the same reliability. While these measures also have their costs, they may be less expensive than reliability improvements, particularly if less of an improvement is needed through a maintainability or redundancy approach.

o System failure may not be the main problem: Though there may be congestion and delay in a system, and though there may be failures, the failures may not be the primary cause of the delay. For example, if a station is served by a major feeder bus line which periodically sends large numbers of passengers simultaneously into the fare collection system, delays will occur which will not be primarily due to failures. If so, reducing the failures that exist will not really help the problem.

In order to properly answer the question, "By how much should reliability be improved?", one needs some way to find out the passenger delay in a fare collection system, given information on its reliability, maintainability, the number of machine units (redundancy), nominal processing rate, and passenger demand. With such a method one can investigate the effect on delay of not only reliability improvements, but improvements in maintainability, operating policy, and the number of machine units as well, and derive the proper mix and extent of improvements necessary.

This report describes a method we have developed to analyze this interrelationship among reliability, maintainability, the number of machine units, and passenger delay. The basic approach is a model to simulate the flow of passengers through the fare collection system. This is done by treating the system as a network of queues, with a queue at each service area and the passengers moving from one service area to the next (a "service area" is a specific set of machine units, such as coin and bill changers, ticket vendors, gates, etc.). Superimposed on this network is the failure/recovery process, by which units fail at a rate according to their reliability and are repaired according to the failure response and repair times.

In addition, we have developed an analytical (mathematical) model to evaluate passenger congestion and delay at a single service area, given failures and repairs. The analytical model obtains the congestion and delay values not by simulation, but rather by solving the underlying queue length probability equations directly. The model investigates a single service area, with certain assumptions on the probability distributions being needed in order to solve the resulting equations, which somewhat limits the scope of the model. However, within that scope, the model requires much less computer time for a large-scale analysis than does a simulation, and also provides the actual underlying queue length distribution, which a simulation provides only after running for a large amount of simulated time. With the analytical model, therefore, many analyses can be carried out with the computer resources necessary for one simulation run, and so the analytical model should be used in the situations for which it is applicable. One such manner is to use the analytical model to conduct the initial investigations, in which many runs may bemade, which many different parameter values. From the results of these investigations, if desired, a smaller number of scenarios of interst can be identified and analyzed further by means of the simulation model.

Section 2 describes the types of results which the models provide, both direct results such as passenger congestion and delay, and indirect results such as sensitivity and tradeoff analyses. It also describes how transit systems can make use of these model results for planning, procurement, maintenance, or analyses of operating policies.

Section 3 develops the detailed technical approach of the models. The concept of the fare collection system as a network of queues is outlined. The simulation model is then described, both for a single service area (e.g., the ticket vendors) and for entire fare collection systems. This is followed by a description of the analytical model.

Section 4 describes the data required for fare collection system analysis, procedures to collect them, and possible problems and cautions. To demonstrate how the modeling process operates and the sensitivity and tradeoff analyses which result, several examples are provided in Section 5, including one based on an actual transit system currently in development. The final section, Section 6, outlines further work to verify and improve the models and to produce a user-oriented package for transit systems.

2. THE ANALYSIS PROCESS

2.1 Types of Dependability Analyses

The direct output of the fare collection dependability model is information on the congestion (queue length) and passenger delay in the fare collection system, given the system configuration and passenger demand. This is produced for each of the service areas (i.e., ticket vendors, gates, etc.) in the system, as well as delay for the overall system. The information is given in terms of probability frequency distributions for congestion and for delay at each service area, from which means and variances are obtained.

This output can give rise to four different kinds of analyses:

Evaluation
Sensitivity Analysis
Specification Determination
Tradeoff Analysis

In evaluation, a given fare collection system is examined. The required information about the system is collected and entered into the model as input data. The results are an estimate as to how well the system performs under the given passenger demand, reliability, maintainability, and number of machine units.

Once a system has been evaluated, one may naturally wish to know how the results would change if one or more of the input parameters were different from their indicated values (this is particularly true if there is some doubt as to the values of some of the input parameters). One can then make several runs of the model with differing values for a given input parameter, and see what changes occur to system congestion and delay. This is called sensitivity analysis, as it measures the sensitivity of congestion and delay to changes in input parameters such as reliability, maintainability, or number of machine units.

Specification determination is a process which determines the values of given input parameters (such as reliability or maintainability) necessary to achieve a desired level of performance. These values then become specifications for those parameters. Specification determination is the reverse of sensitivity analysis, in that while sensitivity analysis takes given values of the input parameters and determines the impact of these values on system performance, specification determination takes a given level of system performance and determines the input parameter values which are necessary to achieve it.

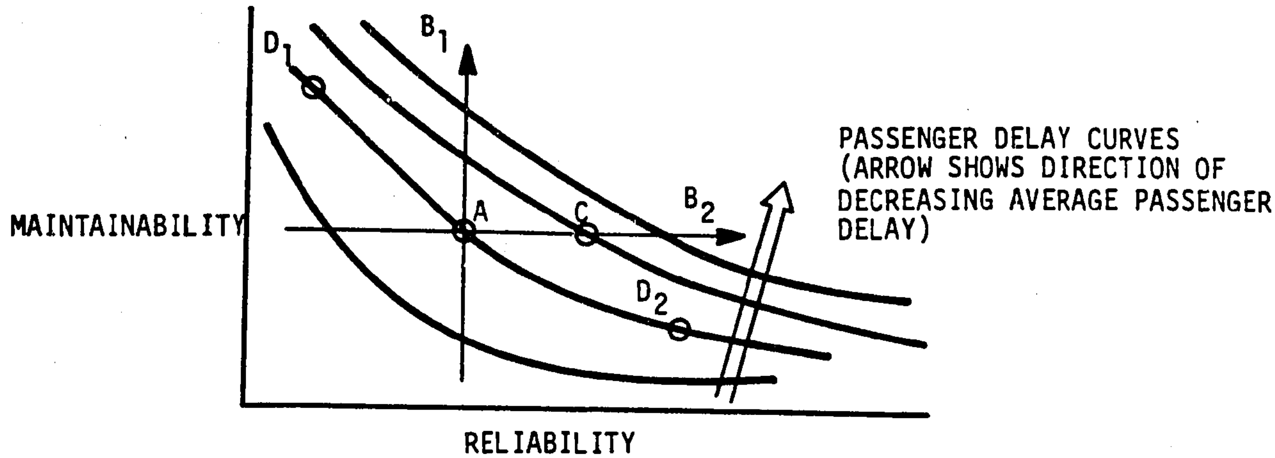
Another aspect one may wish to know about the system is tradeoffs, i.e., how could one trade off between two input parameters while achieving the same overall result. For example, if the equipment reliability declined, by how much would maintainability have to improve to obtain the same overall performance? This type of analysis, called tradeoff analysis, differs from sensitivity analysis in that tradeoff analysis examines the interaction between two input parameters, whereas sensitivity analysis examines the interaction between an input and an output parameter.

The relationship among these four different types of analyses is shown in the conceptual graph given in Figure 1.

2.2 Possible Uses of Dependability Analyses by Transit Systems

The results produced by the fare collection model can be used by transit systems for a number of different purposes. Examples of these are the following:

- Determination of required number of machine units
- Reliability and maintainability specifications
- Impact of changes in passenger demand
- Effect of maintenance policy changes
- Effect of changes in fare collection method



A. Evaluation -- How does the system perform for a given reliability and maintainability?

B_1. Sensitivity Analysis -- What happens to the average delay when maintainability improves?

B_2. Sensitivity Analysis -- What happens to the average delay when reliability improves?

C. Specification Determination -- How high must the reliability be to obtain the desired level of average delay?

D_1. Tradeoff Analysis -- If reliability declines, by how much must maintainability improve in order to retain current performance?

D_2. Tradeoff Analysis -- If reliability improves, by how much can maintainability requirements be relaxed while still retaining current performance?

FIGURE 1. TYPES OF FARE COLLECTION DEPENDABILITY ANALYSES

o Determination of Required Number of Machine Units

An important consideration in designing a fare collection system is determining how many machine units are required in each service area, i.e., how many coin changers, ticket vendors, gates, etc., are needed. This is an important consideration, since these machines are expensive (note -- a cost module is described in Chapter 6 which can be used to estimate these expenses). While an insufficient number will adversely affect station performance, an excessive number will cause a large unnecessary expense, particularly if this happens for a large number of stations.

By conducting a sensitivity analysis between passenger delay and the number of machine units in place at a particular station, one can choose the minimum number of units such that the anticipated passenger delay remains within acceptable limits.

o Reliability and Maintainability Specifications

The level of specification for the reliability and maintainability of the fare collection system units is very important. If the specifications are too low, the resulting reliability and maintainability will be too low, producing unacceptable passenger delays and, hence, expensive retrofits or even more expensive reprocurements. However, if the specifications are too high, the units will become unnecessarily expensive to obtain, or even infeasible to produce at all, leading to unreasonable expectations followed by failure to meet specifications, cost overruns, and possible lawsuits. Either overspecification or underspecification thus gets the procurement process off to a poor start.

With the model, one can evaluate a fare collection system under a set of proposed reliability and maintainability specifications. If the resulting congestion and delay is unacceptable, one can then, through sensitivity and tradeoff analyses, increase one or both specifications until an acceptable performance is reached. If, on the other hand, the congestion and delay is acceptable, one can then, again by means of sensitivity and tradeoff analyses, decrease the specifications, as long as the performance remains acceptable, in order to decrease system costs.

o Impact of Changes in Passenger Demand

Another use of the model results, particularly for new systems, is to determine the accuracy required in predicting passenger demand. If a significant increase in passenger demand does not cause a large increase in passenger delay, then the accuracy of the demand estimate is not that critical. However, should a small increase in demand cause a large rise in passenger delay, then the demand estimate must be much more accurate. Knowing this sensitivity of system delay to passenger demand thus allows the demand
prediction to be carried out at an appropriate level of accuracy.

An additional use of the sensitivity of the fare collection system to increase (or decrease) demand is to find out in advance what changes need to be made in the system in the future due to changes in demand. These changes could arise fromincreased use of the system or from significant changes either in the transit system itself or its feeder systems (buses or parking lots).

o Effect of Maintenance Policy Changes

Maintenance of fare collection equipment can be carried out in a number of ways. One can have an attendant present at every station (or only at heavily-used ones) to correct minor failures and to call in immediate aid for more significant ones. One can make repairs immediately as failures occur, or let failures accumulate until enough occur at a given location to justify sending a repairman. One can forego preventive maintenance, or have such maintenance, with either shorter or longer periods between successive preventive maintenance actions. Since maintenance can significantly affect system performance, it is important to be able to verify the impact of changing the manner in which it is conducted. By running the model under various maintenance schemes, one can compare them in terms of resulting system performance, thus allowing, for example, a comparison of the minimum reliability or number of machine units needed for adequate performance under different candidate maintenance policies.

o Effect of Change in Fare Collection Method

The model may be used to evaluate changes in fare collection media, i.e., ticket-only, token-only, cash-only, cash-or-ticket, cash-or-ticket-or-pass, etc. The choice of media dictate the type of arrangement of machinery used in the system. For example, one can compare a system in which the gates will only accept tickets, thus simplifying the gate design in return for requiring heavier use of the ticket vendors and coin changers, with a system in which the gates will accept cash, thus increasing the cost and possibly decreasing the reliability of the gates, but reducing the use of the ticket vendors and coin changers. In this way, different fare collection methods can be evaluated before expensive machinery is ordered or installed.

The model can evaluate systems with either entry control only (i.e., no exit processing of passengers necessary), both entry and exit control (i.e., passengers must use a ticket or other medium to enter the system at their origin stations, as well as to leave the transit system at their destination stations), or barrier-free control (i.e., passengers may enter and exit the system without impedance by gates, but must obtain and validate a ticket, and must produce such a validated ticket on demand).

3. TECHNICAL APPROACH

The basic approach to the model is to investigate the operation of the fare collection system as a multiple-server queue, with passengers as customers, machine units as servers, and a first-come-first-served service discipline. In addition to the normal queue features, there is the additional aspect that the number of servers (machine units) itself decreases and increases as the units fail and are repaired, respectively.

Two types of simulation models have been developed; a basic model which covers a single service area (such as a gate or ticket vendor), and an extended model which covers several service areas in tandem. In addition, an analytical model has been developed which covers the single service area without the use of simulation.

Note: The random variables in the models as presented are assumed to be exponentially distributed. This is done for tractability and to provide a reasonable generic probability distribution if detailed data for such a distribution is not available. The simulation models can be routinely changed to different distributions, however, if such data is in fact available.

3.1 The Single-Service-Area Simulation Model

Figure 2 shows a situation in which there is a single service area. Several machine units make up the service area, each of which can serve one passenger provided they are not otherwise busy or out of service because of failure. The model is an event-oriented simulation in which the next event to be processed is the earliest-occurring of the four basic events: passenger arrivals, passenger departures, equipment breakdown, and equipment repair. Events are processed until the time of the prospective next event is no longer within the time period being simulated. Each of the four events is described in detail in Sections 3.1.1 - 3.1.4.

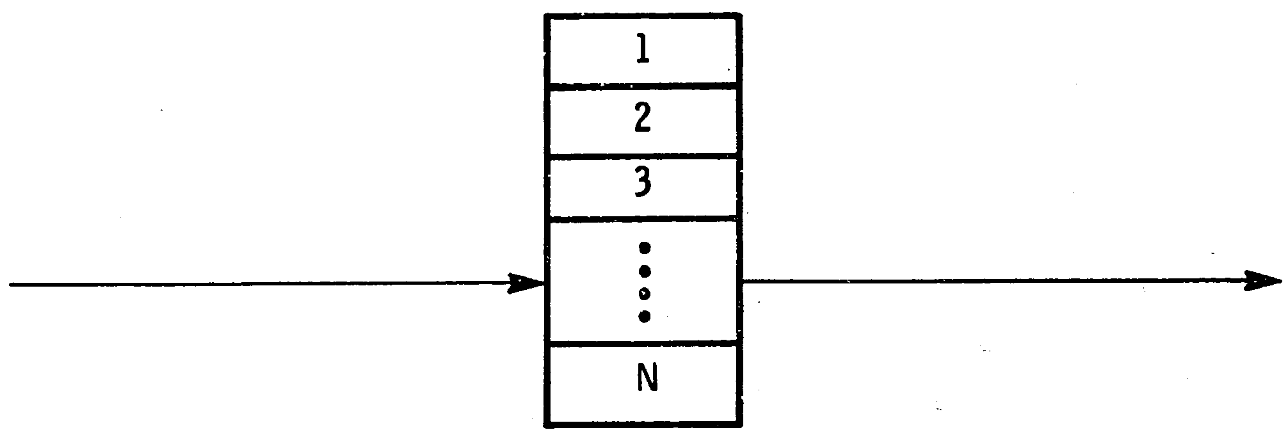


FIGURE 2. A SINGLE SERVICE AREA WITH N MACHINE UNITS

3.1.1 Passenger Arrivals

The arrival of passengers in the system is assumed to be a sequence of independent events, in which the time between successive arrivals is an exponentially-distributed random variable. (While the fact of an exponential distribution is not essential to the model, the independence of successive arrivals is.) Passengers are assumed to arrive in groups. For passengers exiting the station after arriving on a train or for those entering the station after arriving on a feeder bus, the group size can be large. However, for passengers entering the station after arriving on foot, or by taxi, or by private car, the group size can be small, possibly consisting of only one passenger.

Upon arrival, a passenger immediately begins service if a machine unit is available. If so, a departure time is calculated and put into the departure stack, which is an array of departure times of passengers in service, sorted in ascending order, that is used to determine the time of the next departure. If no machine is available, however, the passenger enters a queue, which is maintained on a first-in-first-out basis.

Passengers who find an available unit upon arrival and so immediately enter service have delay times of zero. Passengers entering service from the queue have their (non-zero) delay times calculated at the time they enter service. The delay time is the interval between the arrival time and the time of entry into service.

3.1.2 Passenger Departures

The service (or processing) time of passengers is also assumed to be an independent random variable with an exponential distribution. The departure time, calculated upon entry into service, is equal to the time of arrival plus the service time. Upon departure, the record of the departure time is removed from the departure stack.

3.1.3 Equipment Breakdown

The reliability of the equipment being modeled is given in terms of the MCBF, or mean cycles between failures. The reliability is given in terms of cycles rather than time because the equipment (especially the mechanical parts) is exposed to failure by actual use rather than simply by the passage of time. Upon the departure of a passenger from service, a random draw is made with probability 1/MCBF that the machine unit just used breaks down. If it does not break down, it becomes available for service and, if a queue exists, the first passenger in the queue enters service. If it does break down, however, a repair time for it is calculated (see "Equipment Repair" below) and it is placed in the repair stack,which is an array of return-to-service times of failed units, arranged in ascending order. The number of machine units available is decreased by 1 if a breakdown occurs, and no passengers are brought in from the queue.

3.1.4 Equipment Repair

The time necessary to repair a failed unit is also assumed to be an independent random variable with an exponential distribution. Note that because a failed unit impedes system operation from the time it fails until the time it returns to service, "repair time" includes not only the time to actually do the repair, but also the administrative downtime, i.e., the time necessary to detect a failure, have a repair crew get to the unit, and diagnose the failure. The return-to-service time is calculated upon breakdown as the sum of the time of breakdown and the total repair time, and entered into the repair stack. Upon repair, the unit is removed from the repair stack, and the number of units available is increased by 1. If a queue exists when a unit returns to service, the first passenger in the queue enters service.

3.2 The Multiple-Service-Area Simulation Model

An actual fare collection system at a station will have not one, but several service areas which passengers may use during the fare collection process. For example, a station may have coin changers, ticket vendors, and gates. An arriving passenger may: 1) have the proper ticket and proceed directly to the gate; 2) may need to purchase a ticket and therefore proceed first to the ticket vendors and then to the gate area; or 3) may need the proper change to buy a ticket and so proceed through all three service areas. The multiple-service-area model (Figure 3) investigates this type of situation. This model contains three service areas (a change to a different number of areas is easy to implement): service area #1 (the coin changers, in the above example), service area #2 (the ticket vendors), and service area #3 (the gates). The probability is p_1 that an arriving passenger begins service at area #1, p_2 that he begins service at area #2, and p_3 that he begins service at area #3. (The probabilities p_1, p_2, and p_3 are specified parameters whose sum is 1.) After completing service at his original service area, the passenger continues to the next service area downstream (i.e., from area #1 to area #2, and from area #2 to area #3), until he completes service at area #3 (the gate), upon which he departs the system. Congestion (queue length) and delay times are tabulated at each service area, and the delay time for a passenger is the sum of the delay times at each service area the passenger uses.

A departure from service area #1 creates a simultaneous arrival at service area #2. Similarly, a departure from service area #2 creates a simultaneous arrival at service area #3. A departure from service area #3 (the gates) is a pure departure, as the passenger then leaves the system.

3.3 The Analytical Model

A simulation model is a reasonably straightforward way to investigate a fare collection system. However, simulation does have drawbacks. The underlying probability distributions for congestion and delay are not specifically determined by a simulation. Rather, random draws on these distributions are taken for each passenger simulated and, after some number of passengers have been processed, the frequency distributions of congestion and delay of these passengers are taken as an approximation to the actual distributions. However, these are only approximations, and are subject to a number of statistical sensitivities. Furthermore, because each passenger must be generated and processed individually, a simulation of this type will require a substantial amount of computer time if a large number of passengers need to

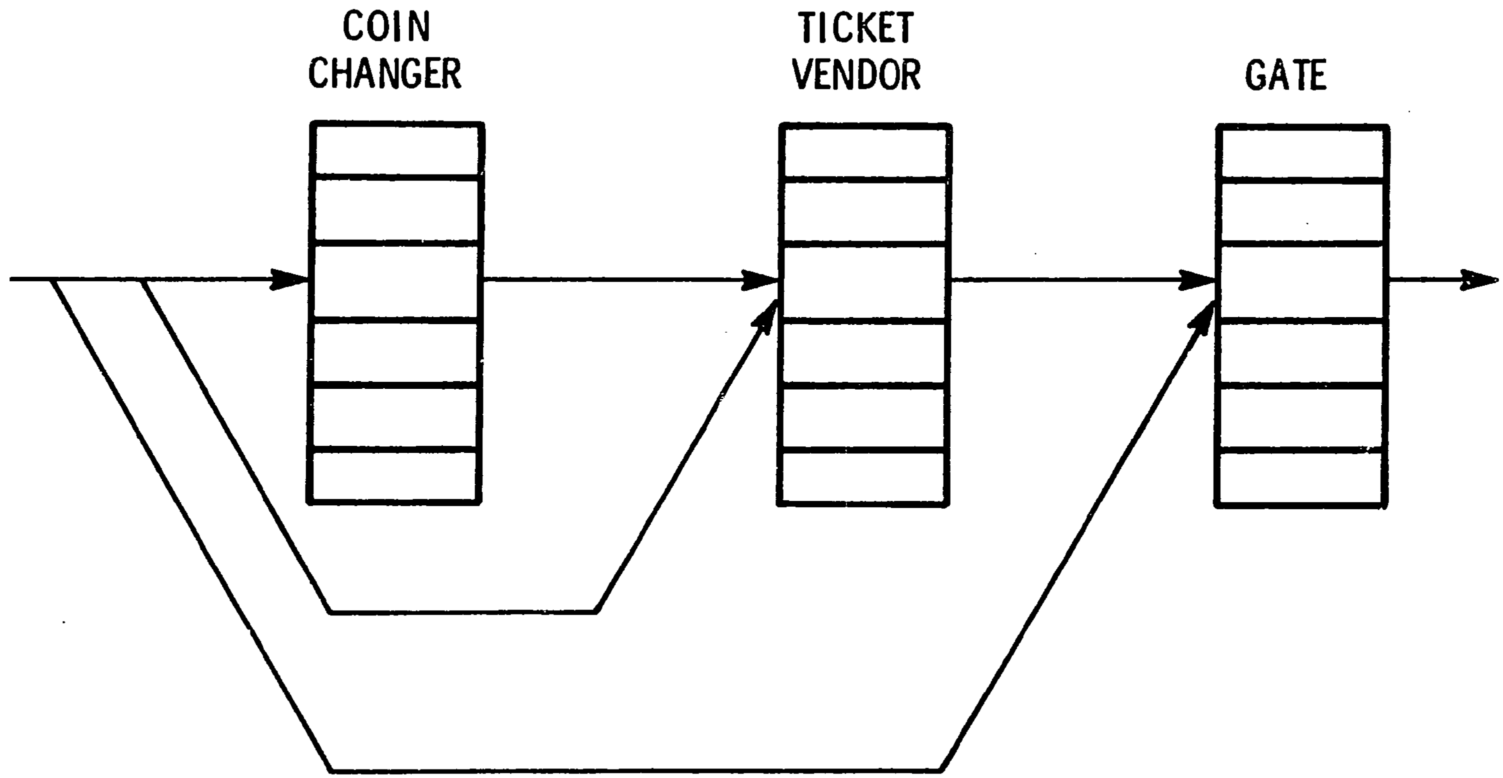


FIGURE 3. PASSENGER FLOW THROUGH A FARE COLLECTION SYSTEM WITH THREE SERVICE AREAS (COIN CHANGERS, TICKET VENDORS, AND GATES)

be processed. This would be necessary not only for stations with a high passenger demand, but also for stations which need to be simulated for a long period of time due to the statistical sensitivities mentioned above.

One example of the latter is a station where the flow of incoming passengers is at or near processing capacity. In such a station, large queues will develop when, due to random fluctuations, a surge of passenger arrivals occur. Furthermore, these large queues will take a long time to dissipate, since the station has very little, if any, extra processing capacity. These surges thus cause considerable congestion and delay; in fact they cause most of the total congestion and delay in this case. Because these random passenger arrival surges do not occur very often, and because the model needs a reasonably large sample of surges to provide accurate statistical results, the simulation must be run for a long period of time.

Another example is a station with machine units which have low failure rates, but long repair times. Failures therefore do not happen often, but significantly affect the system when they do. Because, similar to the above case, the model needs to have a large enough sample of these failures in order to provide statistical results of sufficient accuracy, we need to run the simulation for a long period of time in this case as well.

We therefore have developed, in addition to the simulation model, an analytical model to analyze a service area for a fare collection system. The analytical model directly formulates and solves the mathematical equations for the underlying queue length probabilities, rather than obtaining an approximate solution by simulation. As mentioned in Section 1, while the model only considers a single service area (with no grouping of passengers), rather than the entire fare collection system, and requires certain assumptions on the arrival, processing, and repair rate distributions (i.e., that they are independent and exponentially distributed), it does obtain the congestion and delay distributions directly. Hence, the computer time required by the model does not depend on the passenger arrival rates, degree of saturation (passenger arrival rate compared to service capacity), or failure and repair rates. The analytical model can therefore be used as the initial investigating tool for large problems, allowing many runs to be made with many different parameter values, without using an inordinate amount of computer time. From these runs, a far smaller number of scenarios of interest can be identified and investigated with the simulation model.

The approach to such a model is a modification of that derived by Neuts and Lucantoni (9) for the multiple-server exponential queue with a randomly-varying number of servers. As with the simulation model, the service area is represented as a queue with as many servers as there are machine units. Passengers are assumed to arrive according to a Poisson process (i.e., the interarrival time between any two successive passengers is independent of any other passenger interarrival time and is exponentially distributed), and their processing times are also assumed to be independent and exponentially distributed. Similarly, times to failure and repair times for the machine units are independent and exponentially distributed.

The modification to the Neuts-Lucantoni model is that the rate at which failures occur in a service area is proportional not to the number of functioning machine units, as specified in (9), but rather the minimum of the number of functioning machine units and the queue length. In other words, machine units cannot fail while they are idle. The modification results in a straightforward change in the arguments in (9), which are then used to directly calculate the probabilities (x_{ij}), where

x_{ij} = Probability that the queue length (number of customers either being served or waiting for service) is i and that j machines are functioning.

The queue length probabilities (l_i) are:

l_i = Probability that there are i customers either being served or waiting for service

$$= \sum_j x_{ij} \tag{1}$$

and the mean queue length (congestion) is

$$L = \sum_{i=0}^{\infty} i l_i \tag{2}$$

The mean time in the system (i.e., the time spent both in service and waiting for service) is given by Little's Formula (10, P. 60) as

$$W = L/\lambda \ , \tag{3}$$

and the mean delay time as

$$W_d = (L/\lambda) - (1/\mu) = W - (1/\mu) \ , \tag{4}$$

where λ is the arrival rate and μ the processing rate of customers.

By the above process the analytical model computes the probability distribution of the queue length (congestion), and the mean congestion and delay.

4. USING THE MODEL

4.1 Data Requirements

The data requirements of the model are of two kinds: hardware and passenger flow. The hardware data include reliability and maintainability data, as well as the passenger processing rate per machine unit and the number of units provided, for each of the service areas in the station. The passenger flow data include the passenger arrival rate, group size, and division of passenger flow to the various service areas. While average values of these data are acceptable (one can assume, as the present models do, an exponential distribution with the given average value as the mean), for simulation models actual frequency distribution information is preferable if available, so that the variability of the data from the average can be taken into account.

Specifically, the data requirements are as follows:

Passenger arrival rate (one parameter for entire system) -- The hourly rate at which passengers arrive at the fare collection system during the peak period. This may be established by <u>a priori</u> demand estimation techniques or by an actual passenger count. Assuming a stable (i.e., steady-state) mean flow during the peak period, a good estimate of the actual frequency distribution of flow can be obtained by taking successive passenger counts over many brief periods of time (such as five minutes).

Group size (one parameter for the entire system) -- The size of a group of arriving passengers (this can be given either as a single deterministic value or as a random variable defined by a probability distribution). Passengers exiting from the transit system and those arriving in large vehicles such as commuter vans or buses will generally come or go in large groups, while passengers arriving on foot or in small vehicles such as cars will usually arrive in small groups (frequently consisting of just one passenger). This information is not easy to obtain, as it requires direct observation of passengers, and careful observation at that, since groups may form, merge, divide, or dissipate rapidly.

Passenger processing rate (one parameter for each service area) -- The hourly rate at which a machine unit in the service area can process passengers, and hence the unit's capacity to handle passenger flow. This is a difficult value to obtain under actual operating conditions. While design capacities are normally given for a machine unit, this merely represents the theoretical maximum rate at which passengers can be processed. The actual rate in the field, even assuming the unit is never idle, will be significantly less, because of passenger time lost accessing, fumbling for, or dropping coins or tickets, reading and comprehending (or misunderstanding) instructions, making decisions while using the unit, making errors and correcting them, and simply using the unit at a slower rate than its capacity. The actual processing rate is thus difficult to specify and estimate beforehand, and so can only be obtained by an actual count which simultaneously obtains for a given time period the number of passengers processed and the amount of time the unit was actually busy. The latter data are not routinely collected, so that a special effort must usually be made to obtain the passenger processing rates. Furthermore, in a new or significantly upgraded fare collection system such data by definition does not exist, and so can only be estimated beforehand, as difficult and imprecise as that process is to accomplish.

Failure rates, or reliability (one parameter for each service area)-- The rate at which failures occur to a machine unit which make it unable to process passengers. Because of the mechanical nature of the equipment, and the resulting fact that the basic measure of stress to the unit, and hence exposure to failure, is the use of the unit by an individual passenger, i.e., one cycle, the measure of failure rate is best given in terms of Mean Cycles Between Failures (MCBF). If necessary, a Mean Time Between Failures (MTBF) measure can be used (where the time used is hours in actual use, rather than elapsed clock time), from which MCBF can be obtained by dividing MTBF by the mean passenger processing time.

For an existing system, the MCBF can be obtained by direct observation, or by combining passenger use records and failure logs. For new or significantly upgraded systems, since such data does not exist, estimates, predictions, specifications, or manufacturer's descriptions of reliability must be used.

Repair times, or maintainability (one parameter for each service area) -- The elapsed time, in hours, between the failure of a machine unit and its return to service. Notice that there are other concepts of repair times, such as the time it takes to carry out the actual repair on the unit, the time between the discovery of the failed unit and its return to service, etc. However, the repair time defined above is the appropriate one for our purposes, as it denotes the length of time the system will function at a reduced capacity due to the failure.

For an existing system, repair time can be obtained by direct observation, or by consulting repair logs. If the latter, care must be taken to include an estimate of the time necessary to detect a failure which has occurred, as the logs will at best only give the time from detection to return-to-service. (In addition, if the logs only give the times to actually repair the units, estimates must also be made of the time required for a repairman to travel to the failed unit, as well as any additional travel time which may be necessary). For new or significantly upgraded systems, the repair times must be estimated.

Number of machine units (one parameter for each service area) -- The number of machine units nominally available for passenger use, in the absence of failures. In a small station, space considerations may restrict the number of machine units possible in a given service area, even if a greater number would be warranted by dependability considerations.

Division of passenger flow to service areas (one parameter for each service area) -- The proportion of arriving passengers who begin their use of the fare collection system in that particular service area (see Figure 3). In an aggregate sense, passenger flow division can be obtained by determining flow rate counts for each of the service areas and comparing them to the overall passenger arrival rate (or the total passenger flow through the gates, which should be equivalent unless the system is overloaded).

4.2 Suggested Precautions

Any model, including this one, is not an exact replication of reality, but rather an approximation. As such, it exhibits differences from reality which can mislead an unwary user. The following are some precautions which a user should take into account.

A Monte Carlo simulation model depends on a random number generator to produce the random variation in input data that is so important in putting the simulated system "through its paces." Virtually all computer systems have such a generator, which produces numbers which vary randomly between 0 and 1. However, some of these generators can be defective. In the early stages of this effort, inexplicable discrepancies occurred between expected and actual results. It turned out that the random number generator being used by the computer tended to generate high numbers (values near 1) significantly more often than low numbers (values near 0). A different random number generator was substituted, and the discrepancies disappeared. This is no isolated occurrence, since such problems have been described in the literature. One must therefore make sure that the random number generator being used is an accurate one.

The collection of data may also introduce some inaccuracies. For example, to obtain the data for passenger processing rates, one must simultaneously count the number of passengers using the machine units and measure the total time that these units are busy (the processing rate is the quotient of these two values). If arrivals occur frequently but intermittently, the busy periods of the machine units will be numerous but brief. It is often difficult to tell exactly when a busy period begins or ends, and errors of several seconds may be introduced, either because of this or merely through observer inattention (this particular data collection effort requires much concentration by the observer). Because of this difficultly, the data collection process may introduce significant inaccuracies into the data when the busy periods of the machine units are numerous but brief.

The actual probability distribution of the data (i.e., whether it is normal, exponential, or some other shape) may be quite important to system performance, as different distributions have different probabilities of generating extreme values (i.e., values which are extremely different from the mean, and especially values which are extremely larger than the mean). It is often these extreme values (such as a random overwhelming surge of passenger arrivals, or a random rash of failures) which cause system delay. Nevertheless, quite often only the mean value is available for the desired data, and so some particular type of distribution (such as the exponential one) must be assumed. If this is so, a possibly significant source of delay may be missed.

Also, quite often only estimates are available for much of the required data. While this is especially so for new or significantly upgraded systems, which practically by definition have not had opportunity to accumulate actual performance records, it also holds for some data, such as passenger processing rates for machine units, in existing systems as well. Estimates, however, may be inaccurate to a greater or lesser degree, which would then affect the results of the analysis.

The final precaution applies to the simulation models. Clearly, breakdowns and repairs of machine units occur far less frequently than do passenger arrivals and departures. If the simulation is run for only a short simulated time period, only a few breakdowns and repairs will have occurred. This will make the results quite volatile and unreliable, since they will be highly sensitive to the addition or subtraction of only a few breakdowns, or even a change in the time of occurrence (i.e., early or late in the simulated time period) of the breakdowns which do occur. These kinds of events can take place quite readily, simply due to the random fluctuations provided (intentionally) by the random number generator. Furthermore, if the simulation is run for a short simulated time period, the starting and ending effects can be significant (starting effects are what happens between the start of the simulation and the time it "settles down" to a steady-state situation, while ending effects take into account the delay of passengers still in the system when the simulation ends). Therefore, the simulation models must be run for a long enough time (or enough separate runs must be made) for a sufficient amount of breakdowns and repairs to have occurred, and for starting and ending effects to be no longer significant. This may require a large amount of computer time.

With these precautions in mind, the models can be used by transit systems as a useful tool in determining fare collection system requirements.

5. SAMPLE RUNS

To demonstrate the model and its application to fare collection system analysis, a number of sample runs were performed. For the single-service-area simulation model, hypothetical test data were generated to describe a ticket-vending facility under various circumstances. In addition, to demonstrate the use of the multiple-service-area model for actual fare collection system analysis, sample runs were also carried out based on preliminary data obtained from the Miami Dade County Transit Authority describing their stations and fare collection system (the sample runs, described below, thus use preliminary configurations and are not intended to reflect the final configuration of the actual Miami system). A base run was made using a representative station, followed by various sensitivity runs to assess the impact of possible changes in reliability, maintainability, and the number of machine units at the station. Additionally, a single service area of the representative station was used to provide a test example for the analytical model. In a similar manner as with the simulation model example, a base run was made, followed by various sensitivity and tradeoff analyses.

5.1 Ticket-Vendor Simulation (Single-Service-Area Model)

In this analysis it is assumed that a ticket vendor has low reliability, and that significant delays are occurring as a result. The aim is to assess the base case, then examine a number of other cases to see how delay is affected by various alternative approaches. These cases are defined as follows (see Table 1 for the data):

1. The base case, in which there is a problem with low reliability (note the MCBF of 120).
2. An extra ticket vendor is added.
3. The reliability of the ticket vendors is improved.
4. The maintainability of the ticket vendors is improved.
5. The reliability is improved as a tradeoff against worse maintainability.
6. The maintainability is improved as a tradeoff against worse reliability.

7. Passengers arrived in much larger group clusters than in the base case.
8. Case 7, but an extra ticket vendor is added.
9. There are twice as many ticket vendors, but also twice as many passengers.
10. An unusually large passenger flow occurs initially, then recedes. To handle this, an extra ticket vendor is added.

TABLE 1. DATA FOR SINGLE-SERVICE-AREA ANALYSES

Case	Group Size (per hr.)	Group Arrival Rate (per hr.)	Passenger Arrival Rate (per hr.)	Number of Units	Service Rate (per hr.)	Reliability (MCBF)	Maintainability (MTTR; seconds)
1.	2	250	500	3	300	120	720
2.	2	250	500	4	300	120	720
3.	2	250	500	3	300	480	720
4.	2	250	500	3	300	120	180
5.	2	250	500	3	300	480	2880
6.	2	250	500	3	300	30	180
7.	20	25	500	3	300	120	720
8.	20	25	500	4	300	120	720
9.	2	500	1000	6	300	120	720
10.	2	800/250*	1600/500*	4	300	120	720

*Arrival rates are at first figure for first 360 seconds of simulated time, then at second figure for the remaining time.

Underlining indicates changes from the base case (Case 1)

Cases 2, 3, and 4 are sensitivity analyses; Cases 5 and 6 tradeoff analyses; and Cases 7, 8, 9, and 10 are variants of the base case. The results of the model runs for the various cases are shown in Table 2.

TABLE 2. RESULTS OF SINGLE-SERVICE-AREA ANALYSIS

Case	1	2	3	4	5	6	7	8	9	10
Mean Queue Length	15.2	4.7	3.8	4.0	28.0	6.1	112.3	23.0	5.6	9.4
Mean Passenger Delay (Seconds)	65.5	17.2	11.9	13.6	185.8	34.4	348.5	77.9	9.0	31.3

In the base case (Case 1), the mean delay is about one minute per passenger, which is a large amount of delay, and indicates that there is indeed a problem at this service area. The three cases in which an improvement is made in the physical system (Cases 2, 3, and 4), all improve the situation equally well, so that in a choice among them, the least expensive alternative should be used. The two tradeoff cases and the base case (Cases 1, 5, and 6) demonstrate that, in this particular system, if any tradeoff is made, it should be in favor of maintainability. Case 7 demonstrates the adverse sensitivity of the system to heavily bunched arrivals, which is significantly improved by the addition of the extra unit in Case 8. Case 9 indicates that the improvement in adding extra units is more than linear, i.e., twice as many units can handle more than twice as many passengers. Finally, Case 10 indicates that if the system has an extra (fourth) unit, it is moderately capable of handling surges in passenger demand.

5.2 Full-System Performance Simulation (Multiple-Area Model)

The Miami Dade County Transit Authority is in the process of designing, procuring, and constructing a rapid transit system for the Greater Miami area. Specifications have been developed for the various fare collection facilities and contracts awarded to supply these facilities. Estimates have also been obtained on passenger flow and machine capacity. We shall use these estimates, along with the reliability and maintainability specifications, as input data for a sample model run based on data derived from an actual system (2,11).

The fare on the Miami system is collected at entry gates which accept either coins, passes, or (at some stations) transfers. Some gates are set for full-fare passengers, while others are set for reduced-fare passengers. Passes are sold off-site, while transfer dispensers and parking-lot-receipt machines operate in the station area.

The station analyzed in the model run is derived from the Dadeland North station during peak-hour operation (this station was selected because it is a relatively important station which has enough passenger demand to result in significant congestion and delay if enough machine units fail). For simplicity, the sample station uses gates which accept (no transfers), processes full-fare only, and includes only the gates and change machines. No exit processing of passengers is required, so only the entering passenger flow will be examined.

In order to provide a multiple-service-area example, the station design will be based on the preliminary design configuration for the Miami system, which provides for change machines to change bills and coins (the final design configuration, which does not include change machines, can be analyzed most efficiently by using the single-service-area simulation model or the analytical model). The station layout is shown in Figure 4. The estimated peak-hour passenger flow at the sample station is 5400 passengers per hour. Of these, 30% use the change machines, while 70% go directly to the gates. Passengers are assumed to arrive one at a time, not in groups.

There are five gates at the station. Each gate has a physical capacity to process 1800 passengers per hour. As mentioned previously, the actual field processing capacity is lower than this and is difficult to obtain. As the Miami system is not yet in revenue service, there is no actual field data available, nor are there detailed analyses of the field passenger-processing capacity of these types of gates. (There are data collection efforts underway at different transit systems to obtain such information. As discussed earlier, it is not straightforward to measure the field processing capacity). A rough rule-of-thumb for field processing capacity of 75% of the physical processing capacity is assumed for this analysis. Therefore, the gate processing rate used in the model runs is .75 x 1800, or 1350 passengers per hour.

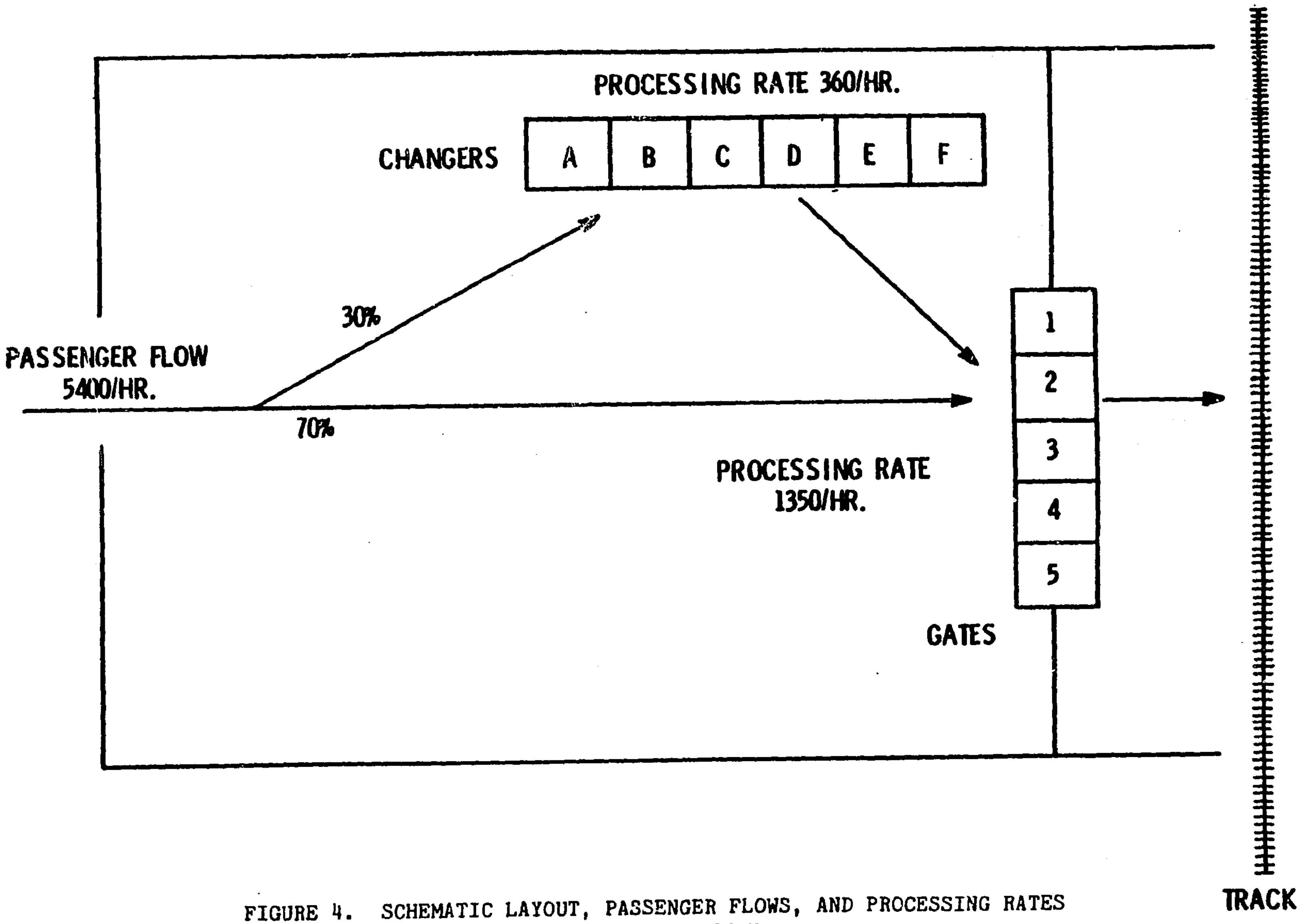


FIGURE 4. SCHEMATIC LAYOUT, PASSENGER FLOWS, AND PROCESSING RATES FOR EXAMPLE TRANSIT STATION

There are also six coin changers. Each can process bills at 12 transactions per 100 seconds, and coins at 20 transactions per 100 seconds. The transactions involve bills 85% of the time and coins 15% of the time. The mean transaction rate is therefore 12*(0.85) + 20*(0.15) = 13.2 per 100 seconds, or 7.92 per minute, or approximately 475 per hour. Using the rule-of-thumb field processing rate ratio of 75%, we obtain an estimated field transaction rate of 360 passengers per hour for the changers.

The reliability specifications are an MCBF of 60000 for the gates and 80000 for the changers. For maintainability, the specifications are that the time to carry out the actual repair on a unit shall not exceed 0.6 hours (36 minutes) for the gates and 0.3 hours (18 minutes) for the changers. We shall assume that the time to detect a failure and get a repairman to the failed unit averages 0.2 hours (12 minutes). Therefore, the specified mean total downtime due to a failure (MTTR) shall be 0.8 hours (48 minutes) for the gates and 0.5 hours (30 minutes) for the changers.

The analysis focuses on the gates, investigating the effects of changes in the number of gates and their reliability and maintainability. Ten cases are examined (the data for the gates are tabulated in Table 3):

1. The base case, with specifications as described above.
2. The gate reliability is 10000 MCBF.
3. The gate reliability is 3000 MCBF.
4. The gate reliability is 1000 MCBF.
5. Case 4, but with 6 gate units.
6. Case 4, but with a mean total downtime of 0.6 hours.
7. Case 4, but with a mean total downtime of 0.3 hours.
8. Case 4, but with a mean total downtime of 0.2 hours.
9. Case 4, but with a mean total downtime of 0.1 hours.
10. Case 3, but with only 4 gate units.

TABLE 3. DATA FOR MULTIPLE-SERVICE AREA ANALYSES

Case	Gate Arrival Rate (per hour)	Number of Gate Units	Reliability (MCBF)	Maintainability (mean total downtime) (in hours)	Gate Processing Rate (per) hours)
1	5400	5	60000	0.8	1350
2	5400	5	10000	0.8	1350
3	5400	5	3000	0.8	1350
4	5400	5	1000	0.8	1350
5	5400	6	1000	0.8	1350
6	5400	5	1000	0.6	1350
7	5400	5	1000	0.3	1350
8	5400	5	1000	0.2	1350
9	5400	5	1000	0.1	1350
10	5400	4	3000	0.8	1350

Underlining indicates changes from the base case (Case 1)

The results for the gates are as follows (Table 4):

TABLE 4. RESULTS OF MULTIPLE-SERVICE AREA ANALYSES

Case	1	2	3	4	5	6	7	8	9	10
Mean Queue Length (Congestion)	3.2	3.2	9.2	40.2	3.3	28.0	20.4	7.3	5.3	*
Mean Passenger Delay (in seconds, excluding processing time)	0.5	0.5	6.6	34.1	2.6	24.2	16.7	4.6	2.6	*

*Infinity (queue length exceeds 500)

(Note -- Except for Case 10, the mean congestion and delay at the coin changers (not the gates) remain more-or-less constant over all the cases at mean queue lengths and mean passenger delays of approximately 6.0 and 3.0, respectively.)

There are a number of conclusions which can be drawn from these results:

(Evaluation of given situation -- Case 1) No serious delay problems are expected from the fare collection system as specified.

(Sensitivity analysis of gate reliability -- Cases 1-4) (Figure 5) The specification for gate reliability can be significantly reduced from its original level of 60000 MCBF without seriously affecting delay. In fact, the reliability can decrease by almost an order of magnitude without serious impact. Delays start becoming significant when the MCBF reaches 3000, and become a problem when the MCBF reaches 1000.

(Sensitivity analysis of increased number of gates under conditions of low reliability -- Cases 4 and 5) Adding one additional gate, making six units in all, when the gate reliability is low (1000 MCBF), is equivalent, in terms of the resulting delay, to entirely solving the problem of low reliability, i.e., improving the reliability to effectively 60000 MCBF. (Note that with cost data showing how cost varies with reliability, an effective tradeoff analysis can be made to decide whether adding a gate or improving gate reliability would be more cost-effective.)

(Sensitivity analysis of maintainability under conditions of low reliability -- Cases 4 and 6-9) (Figure 6) A delay problem due to low reliability can be solved for this system by improving maintenance response, but the improvement must be consider-able (even an improvement from 0.8 hrs. to 0.1 hrs. does not completely restore the performance of the base case). Notice that the nominal detection and response time for the system is 0.2 hours, so that Cases 8 and 9 can only be brought about by an improvement in this time, not just by an improvement in the repair time itself.

(Sensitivity analysis of decreased number of gates under conditions of marginal reliability -- Cases 3 and 10) The system cannot operate with fewer than five gates. If failures occur under a four-gate operation, the

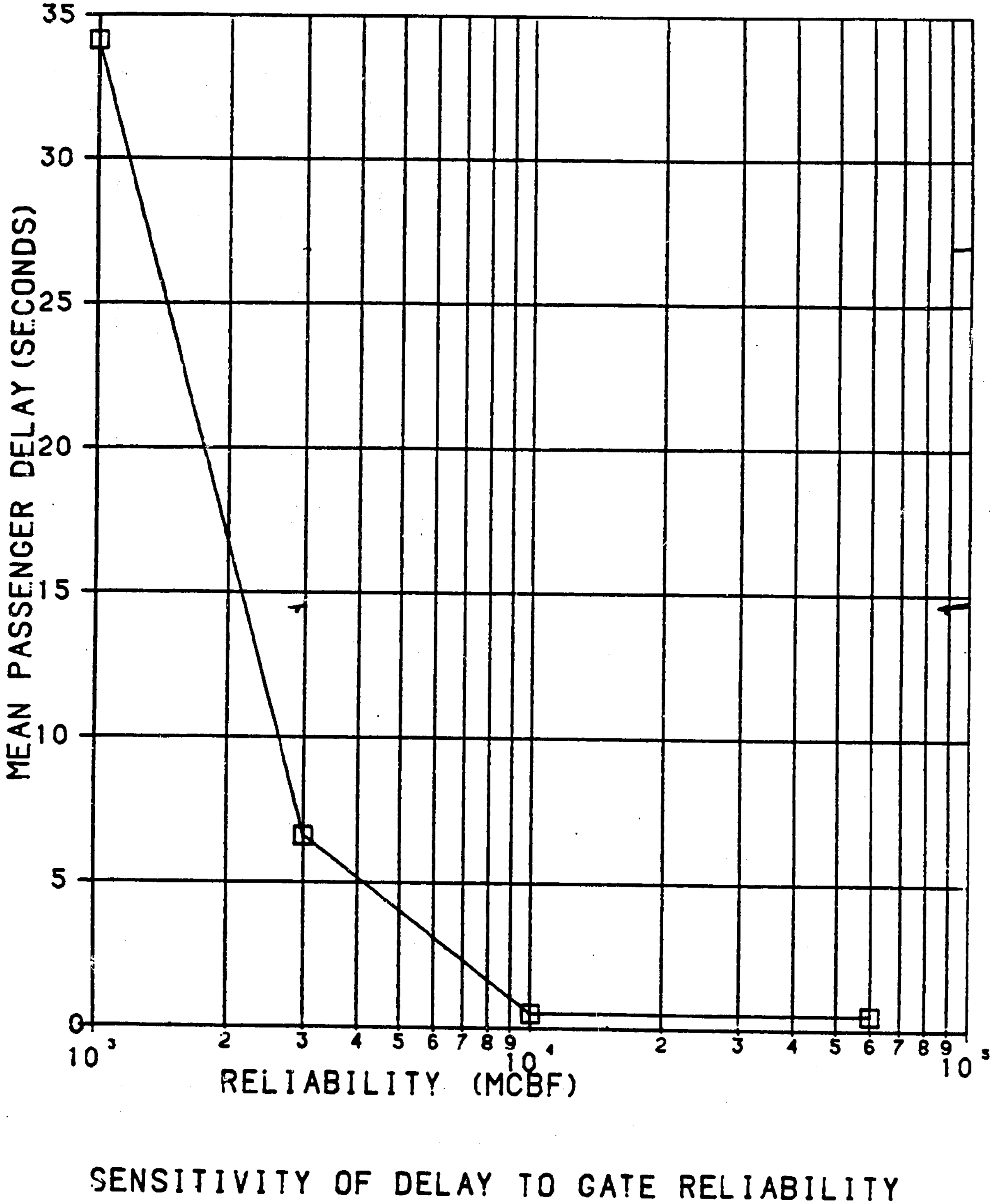


FIGURE 5. GRAPH OF GATE SENSITIVITY TO RELIABILITY

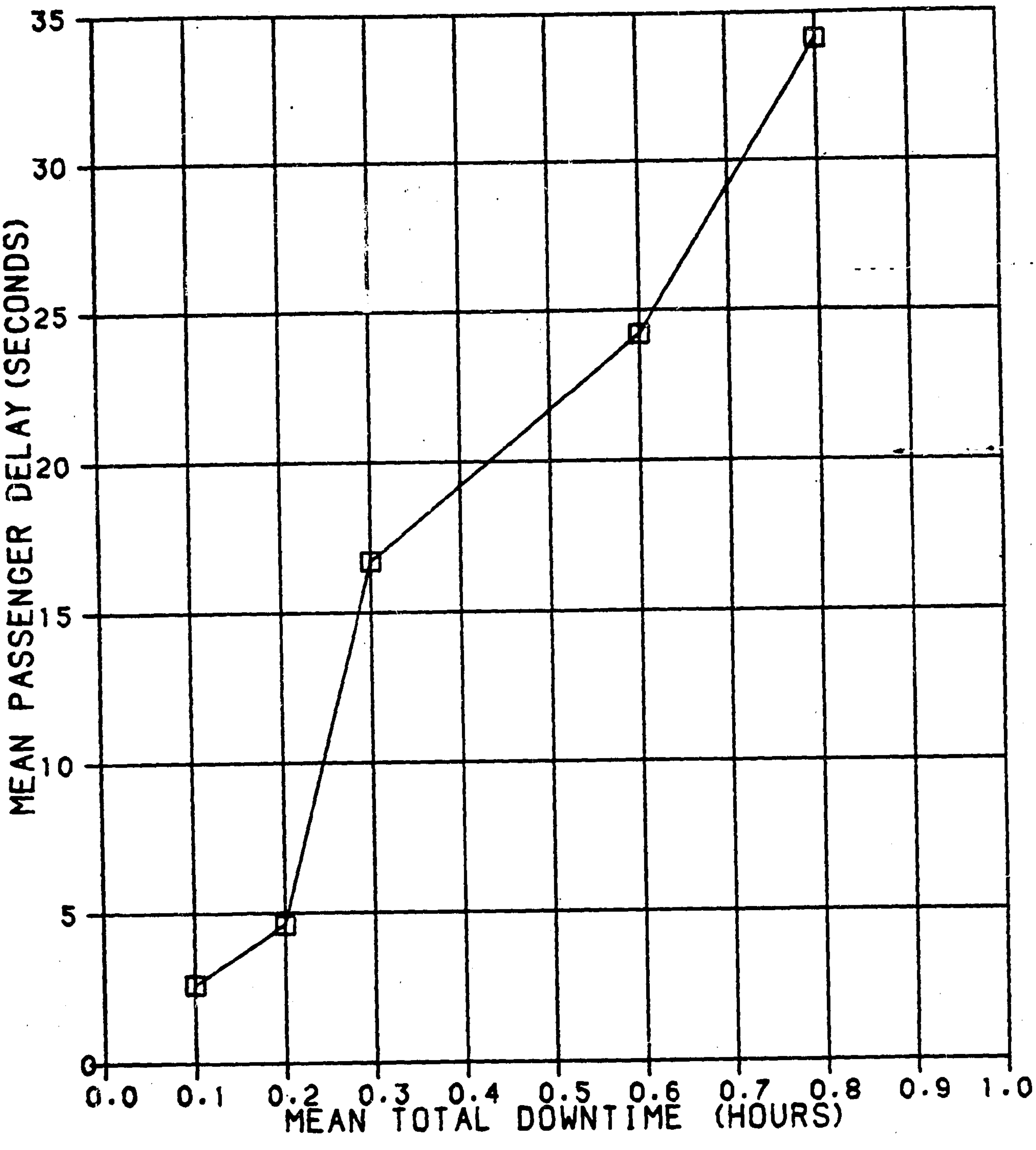


FIGURE 6. GRAPH OF GATE SENSITIVITY TO MAINTAINABILITY

system will sustain catastrophic congestion and delay. (In fact, even without the impact of failures, the four-gate system is operating at capacity, so that large congestion and delay can arise simply due to random surges of passenger arrivals.)

(Tradeoff analysis of reliability vs. maintainability under conditions of low reliability -- Cases 3, 4, and 8) An increase in the reliability (of Case 4) from 1000 to 3000 is approximately equivalent in delay impact to an improvement in the maintainability from 0.8 hours to 0.2 hours.

Remark -- The precautions described previously apply of course to the interpretation of this example, especially the precaution on the volatility of the results when only a few breakdowns have occurred. For example, with an MCBF of 3000, an average of 1.8 breakdowns would be expected in a peak hour in which 5400 passengers arrive (5400/3600=1.8). Of course, however, in an actual simulation run the number of breakdowns would be some (Poisson-distributed) integer-valued random variable of 0, 1, 2, 3, or more, depending not only on the underlying probability distribution for the breakdown process, but also on the particular sequence of random numbers on which the specific run is based. The most likely number of breakdowns to occur, 2, is indeed what happens in the actual sample run. However, in another run using the same input data, the number of breakdowns could be 1 or 3, since each run is based on a new sequence of random numbers. This would significantly change the results.

The way to make the results less volatile is to run a number of independent peak-hour runs and combine the results (or make a single run for a long time period). For example, if five independent runs were made of the peak-hour situation with an MCBF of 3000 (or if one run were made for a five-hour period under peak conditions), the expected number of breakdowns would be 9.0. To have 8 or 10 breakdowns instead of 9 would clearly affect the delay result much less than to have 1 or 3 breakdowns instead of 2. The results would therefore be less subject than before to random fluctuations in the simulation process. To make these five runs, however, would use five times as much computer time. For the simulation with an MCBF of 3000, this would mean over 22 minutes of CPU time instead of under 5 minutes. Since the cost of such amounts of computer time would have been prohibitive, only one 1-hour time run was made for each case examined. In a real situation, however, either longer or multiple runs must be carried out.

5.3 The Analytical Model (Single-Service-Area Analysis)

In this section the analytical model described in Section 3.3 is used to examine the coin changers in the Miami-system-based example discussed above. The analytical model considers a single service area (i.e., the coin changers) rather than the complete system. However, this is a reasonable approach in this example, since the coin changers receive customers directly from outside the system, not from any other service area within the fare collection system itself.

Since 5400 passengers per hour arrive at the station and 30% use the coin changers, the mean arrival rate at the changers is 1620 passengers per hour. As described in the previous section, there are six changer units which process passengers at a rate of 360 passengers per hour each, with a reliability of 80000 MCBF and a maintainability of 0.5 hours mean total downtime.

Eight cases are considered (the data are tabulated in Table 5):

1. The base case, with specifications as described above.
2. The reliability is 8000 MCBF.
3. The reliability is 4000 MCBF.
4. The reliability is 3000 MCBF.
5. Case 4, but with a mean total downtime of 0.4 hours.
6. Case 4, but with a mean total downtime of 0.3 hours.
7. Case 4, but with a mean total downtime of 0.2 hours.
8. Case 4, but with a seventh coin changer unit added.

TABLE 5. DATA FOR ANALYTICAL MODEL ANALYSES

Case	Changer Arrival Rate (passengers per hour)	Number of Changer Units	Reliability (MCBF)	Maintainability (mean total downtime; in hours)	Changer Processing Rate (passengers per hour)
1	1620	6	80000	0.5	360
2	1620	6	8000	0.5	360
3	1620	6	4000	0.5	360
4	1620	6	3000	0.5	360
5	1620	6	3000	0.4	360
6	1620	6	3000	0.3	360
7	1620	6	3000	0.2	360
8	1620	7	3000	0.5	360

The resulting mean congestion and delay for these examples are shown in Table 6 (the delay is obtained by using formula (4), Section 3.3):

TABLE 6. RESULTS OF ANALYTICAL MODEL ANALYSES

Case	1	2	3	4	5	6	7	8
Mean Queue Length (Congestion)	5.9	8.1	18.0	40.9	12.8	8.1	6.8	6.3
Mean Passenger Delay (in seconds, not including processing time)	3.1	8.0	30.0	80.9	18.4	8.0	5.1	4.0

Some of the conclusions which can be drawn from these results are:

(Evaluation of given situation -- Case 1) The coin changer service area as specified should not have serious delay problems.

(Sensitivity analysis of coin changer reliability -- Cases 1-4) (Figure 7) The coin-changer reliability specification can be significantly reduced from its original level of 80000 without significantly degrading the system. A decrease of an order of magnitude, to 8000, only marginally increases the delay. Delays start becoming significant when the MCBF reaches 4000, and become a problem when the MCBF reaches 3000.

(Sensitivity analysis of coin changer maintainability under conditions of low reliability -- Cases 4-7) (Figure 8) An improvement in mean maintainabilty from 0.5 hours to 0.4 hours will significantly reduce the mean delay resulting from low reliability (3000 MCBF). Further improvements beyond this occur more-or-less proportionally, i.e., decreasing the maintenance recovery time by one-half will decrease the mean delay by one-half.

(Sensitivity analysis of increased number of gates under conditions of low reliability -- Cases 4 and 8) Adding an additional unit to the system when the reliability is low (3000 MCBF) solves the reliability problem, i.e., the mean delay becomes similar to that of the original system (which has the specified reliability of 80000 MCBF).

(Tradeoff analysis of reliability vs. maintainability under conditions of low reliability -- Cases 1,2,4,6,and 7) Improving the maintenance recovery time (of Case 4) from 0.5 hours to 0.3 hours is equivalent to improving the reliability from 3000 MCBF to 8000 MCBF. Improving the recovery time from 0.5 hours to 0.2 hours is nearly equivalent to improving the reliability from 3000 MCBF to the full original specification of 80000 MCBF.

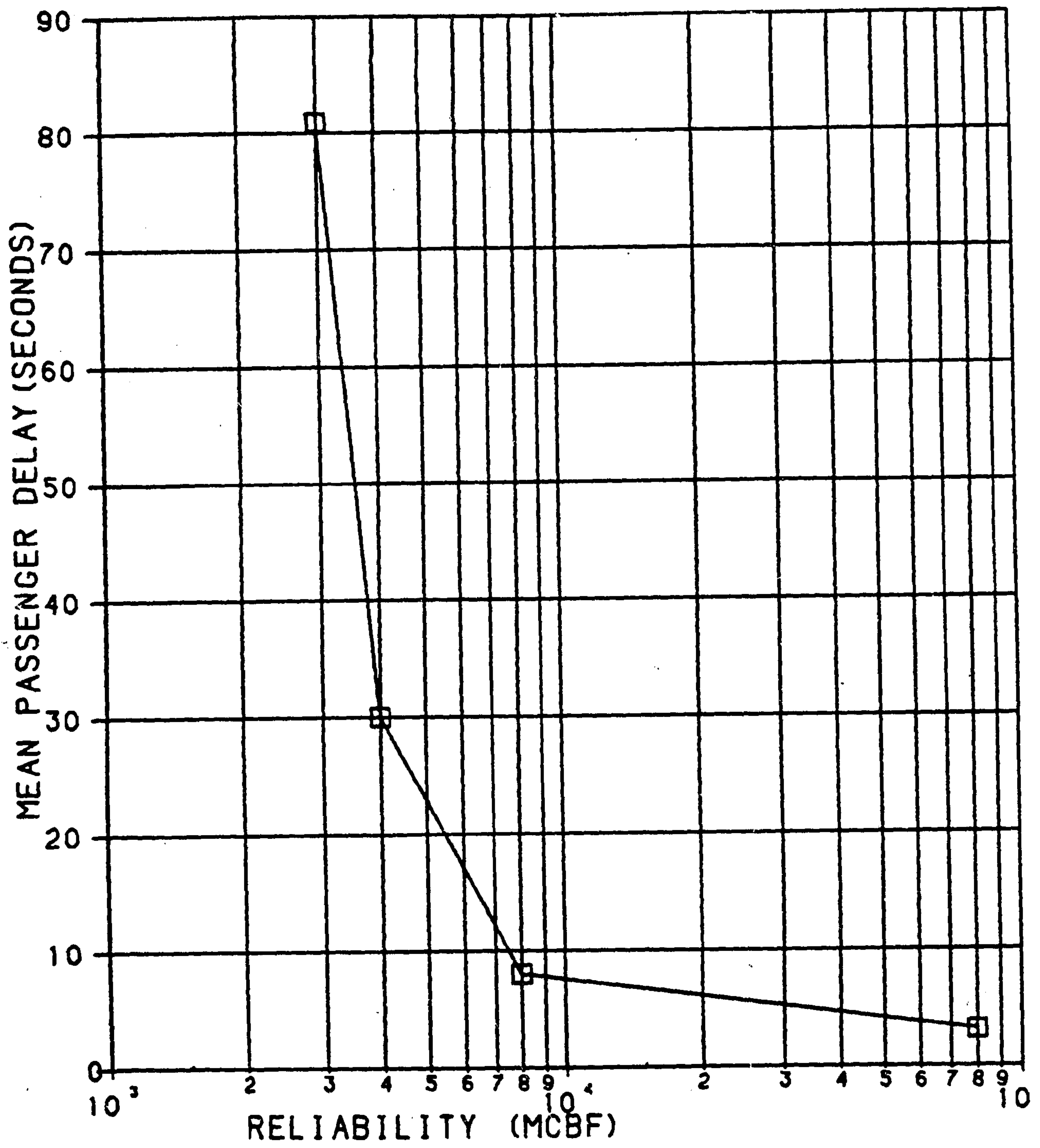


FIGURE 7. GRAPH OF CHANGER SENSITIVITY TO RELIABILITY

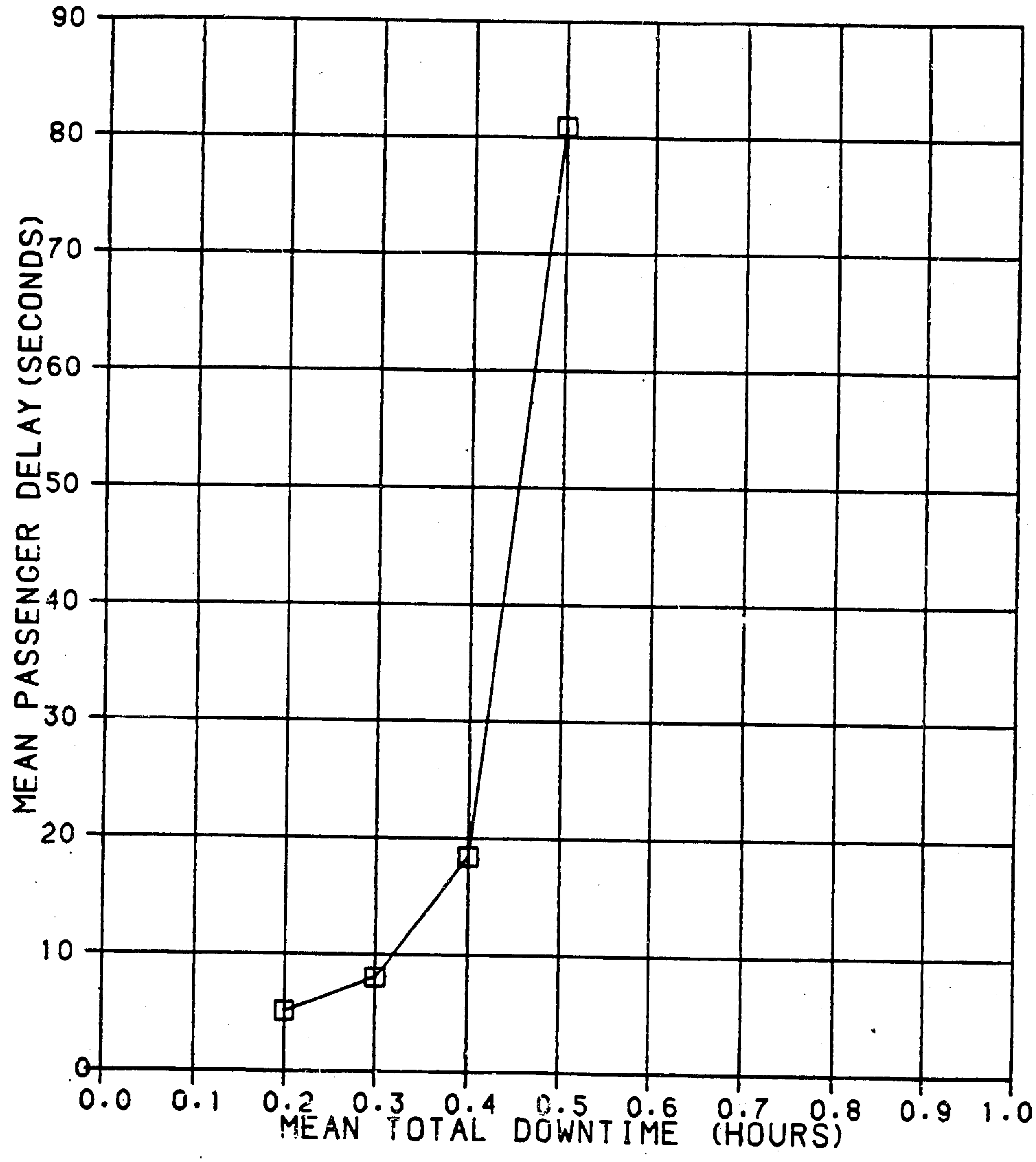


FIGURE 8. GRAPH OF CHANGER SENSITIVITY TO MAINTAINABILITY

6. THE COST MODULE

We have dealt so far with the analysis of the passenger congestion and delay in a fare collection system, given the configuration of the station, passenger demand, and the processing rates, reliability, and maintainability of the equipment. However, one would often like to have some idea of the cost of the various alternatives, since this would supplement the information on passenger dependability performance and thus provide an enhanced analysis of the comparative strengths and weaknesses of each alternative. To that end, we have developed a cost module which computes the annual costs relevant to fare collection dependability (these include equipment acquisition, spares provision, equipment operation, and scheduled and corrective maintenance). The module accepts input data on equipment acquisition costs, operation costs, discount rates, spares requirements, annual required scheduled maintenance, pay rates for repair personnel, and annual passenger volume, and uses these data, along with data previously supplied to the model, to calculate specific and total annualized costs at the station under considera-tion. The module thus makes possible such analyses as cost/performance evaluations, sensitivity assessments of costs to changes in specifications, tradeoffs between cost and performance, and tradeoffs between different types of costs.

6.1 Module Description

Among the large variety of costs incurred in operating a transit system, five types play a role in fare collection dependability analysis:

Capital costs -- This represents the price to acquire the fare collection equipment. It includes the direct cost of the equipment itself, as well as the costs of financing, specification development, procurement, delivery, installation, etc.

Spares costs -- In order to assure having enough functioning equipment available for adequate operation, one must have more equipment on hand than required for service, with the excess kept as spares to replace units which fail in the field. Spares cost is the cost of this extra equipment. The requirement for spares is stated as a given percentage of the number of field units, and so spares cost will be a similar percentage of the capital cost.

Operating cost -- This is the cost to operate the fare collection equipment. It consists of such costs as energy, media (such as tickets or tokens), routine service other than maintenance (such as ticket filling, ticket removal, coin-change filling, or revenue removal), etc.

Scheduled maintenance -- The equipment must be maintained periodically to keep it in good operating order. Often, such maintenance is required to keep valid the warranties on the equipment, and in any event it will tend to enhance equipment reliability, thus improving performance and decreasing repair costs.

Corrective maintenance -- This is the cost incurred to repair equipment which fails while in service. This cost, mainly for personnel, transportation, and parts, clearly depends on the reliability and maintainability of the equipment, as well as the number of units and passenger demand at the station.

Note that the above costs apply to a given type of fare collection equipment. If the analysis covers multiple service areas (coin changers, ticket vendors, and gates, for example), there will be several sets of these costs, one for each type of service area.

The cost module calculates annual costs of each of the above five types at the station under consideration. To obtain these costs, the following data are required (in addition to the information already provided to the dependability model itself):

1. Acquisition cost per unit:
   The total price to acquire one machine unit of the fare collection equipment.

2. Useful life of the unit:
   The number of years the unit will provide adequate service. After the useful life period has elapsed, one will expect to have to replace the unit, thereby incurring a new acquisition cost.

3. Discount rate (exclusive of inflation):

   The "time value of money"; the ratio of the value of one dollar now to the value of one dollar a year from now. The discount rate is used to convert acquisition cost, a single expense incurred at the start of the useful life of the equipment, to annualized capital cost, the equivalent annual expense. This conversion is necessary since the other costs in the module are provided and calculated on an annual basis. The annualized capital cost represents the annual repayment which would be required for a loan equal to the acquisition cost, with an interest rate equal to the discount rate and a repayment period equal to the useful life of the equipment.

4. Spares ratio:

   The number of additional units to be bought as spares, expressed as a percentage of the number of units required for active service.

5. Annual operating cost per unit:

   The cost for energy, media, routine service, etc., for a single machine unit for one year.

6. Annual hours of scheduled maintenance:

   The number of labor-hours needed to perform the required scheduled maintenance on a single unit for one year.

7. Hourly pay rate for repair personnel:

   The costs for repair personnel, including benefits, supplies, and overhead.

8. Annual passenger volume at the station:

   The total volume of passengers passing through the fare collection area during the year. This determines the usage of the equipment, and hence the number of failures and consequent corrective repair actions.

The five cost types are calculated from the above input data as follows (a dagger "†" indicates data from the dependability model itself, rather than the above data):

I. Annualized Capital Cost

$$ACAP = ACQ * (r / (1-(1+r)^{-t})) * (n)$$

where

ACQ = acquisition cost

r = discount rate

t = useful life

n = number of units at the station†

II. Annualized Spares Cost:

$$SPRS = s * (ACAP)$$

where

s = spares ratio

III. Operating Cost:

$$OPER = n * (UOPR)$$

where

UOPR = annual operating cost per unit

IV. Cost of Scheduled Maintenance:

$$SCHD = h * n * w$$

where

h = annual hours of scheduled maintenance per unit

w = pay rate for repair personnel

V. Cost of Corrective Maintenance:

$$CORR = (VOL) * (p) * (1/MCBF) * (MTTR) * (w)$$

where

VOL	=	annual passenger volume at station
p	=	passenger split; i.e., percentage of total passengers who use the service area containing the equipment under consideration†
MCBF	=	Mean Cycles Between Failures †
MTTR	=	Mean Time To Repair†

6.2 Sample Cost Analysis

To demonstrate the cost module, we have carried out sample runs based on the full-system example in Section 5.2. The cost and other additional data necessary for the cost module are based on cost information for the faregates at the Metropolitan Atlanta Rapid Transit Authority. (The numbers used in these examples, however, represent hypothetical situations and therefore do not reflect actual costs at Atlanta. In fact, it is because these are hypothetical situations that we can reasonably combine the Atlanta-based cost data with the Miami-based performance and demand data.)

The passenger arrival and processing rates, the number of gate units, and the gate reliability and maintainability are the same as those provided for the full-system example. To recapitulate, these values are:

Passenger arrival rate	=	5400 passengers/hour
Passenger processing rate	=	1350 passengers/hour
Number of machine units	=	5
Reliability	=	1000 MCBF
Maintainability	=	0.8 hours MTTR
Percentage of passengers using gates	=	100%

The cost and other additional data are:

Acquisition cost	=	$33000/unit
Useful life	=	10 years
Discount rate	=	10%

Spares ratio	= 5.5%
Annual operating cost	= $4100/unit
Annual hours of scheduled maintenance	= 36 hours/unit
Pay rate for repair personnel	= $15.66/hour
Annual station passenger volume	= 2,448,000 passengers/year

Eight cases are considered in these sample analyses:

1. The base case, with data as shown above (this case is equivalent to Case 4 in section 5.2).
2. An extra unit is added.
3. A minor improvement increases the reliability to 1667 MCBF, while increasing the acquisition cost by $1000.
4. The scheduled maintenance is doubled to 72 hours annually, and the equipment reliability improves so as to keep the total annual costs unchanged from the base case.

(Cases 5 - 8): The following reliability improvements occur, and the acquisition cost increases so as to leave total annual costs unchanged from the base case.

5. The reliability improves to 1667 MCBF (as in Case 3 above).
6. The reliability improves to 3000 MCBF.
7. The reliability improves to 10000 MCBF.
8. The reliability improves to 3000 MCBF, and the useful life improves to 15 years.

The results of the analysis are as follows (underlined values represent the answers sought in each particular case):

Spares ratio	= 5.5%
Annual operating cost	= $4100/unit
Annual hours of scheduled maintenance	= 36 hours/unit
Pay rate for repair personnel	= $15.66/hour
Annual station passenger volume	= 2,448,000 passengers/year

Eight cases are considered in these sample analyses:

1. The base case, with data as shown above (this case is equivalent to Case 4 in section 5.2).
2. An extra unit is added.
3. A minor improvement increases the reliability to 1667 MCBF, while increasing the acquisition cost by $1000.
4. The scheduled maintenance is doubled to 72 hours annually, and the equipment reliability improves so as to keep the total annual costs unchanged from the base case.

(Cases 5 - 8): The following reliability improvements occur, and the acquisition cost increases so as to leave total annual costs unchanged from the base case.

5. The reliability improves to 1667 MCBF (as in Case 3 above).
6. The reliability improves to 3000 MCBF.
7. The reliability improves to 10000 MCBF.
8. The reliability improves to 3000 MCBF, and the useful life improves to 15 years.

The results of the analysis are as follows (underlined values represent the answers sought in each particular case):

Case	1	2	3	4	5	6	7	8
Reliability (MCBF)	1000	1000	1667	1100	1667	3000	10000	3000
Number of gate units	5	6	5	5	5	5	5	5
Sched. maintenance hours	36	36	36	72	36	36	36	36
Useful life (years)	10	10	10	10	10	10	10	15
Acquisition cost	$33000	$33000	$34000	$33000	$47300	$56900	$65200	$70400
Annual Costs								
Capital	$26853	$32224	$27667	$26853	$38489	$46301	$53005	$46279
Spares	1477	1772	1521	1477	2116	2546	2918	2545
Operating	20500	24600	20500	20500	20500	20500	20500	20500
Scheduled Maintenance	2819	3383	2819	5638	2819	2819	2819	2819
Corrective maintenance	30688	30668	18398	27880	18398	10223	3067	10223
Total annual costs	$82317	$92647	$70905	$82348	$82322	$82389	$82359	$82366
Mean passenger delay (seconds)	34.1	2.6	14.2	25.6	14.2	6.6	0.5	6.6

From the above analyses, one can draw a number of conclusions:

(Case 1) The base case shows a very high mean passenger delay (this delay, of course, is the same as Case 4 of section 5.2). Some action needs to be taken to reduce delay. Furthermore, because of the low reliability, the corrective maintenance costs are quite high; in fact they are of the same magnitude as the capital and the operating costs.

(Case 2) By adding a sixth gate unit, the mean passenger delay drops off sharply, becoming virtually insignificant. However, the total annual costs rise by some $10000, or 12.5%, mostly due to increased capital and operating costs. Despite the greater number of machines, corrective maintenance costs remain the same, since each machine handles fewer passengers (the corrective maintenance cost is unchanged because it depends on total annual failures at the station, which in turn depends on machine reliability and total passenger volume, both of which are unchanged from Case 1).

(Case 3) The minor equipment improvement is definitely worthwhile at a cost of $1000. In fact, not only does the mean passenger delay decrease, but total annual cost decreases as well, since the savings in corrective maintenance cost due to fewer failures outweigh the increase in capital and spares cost. Case 3 thus represents not a tradeoff from Case 1, but in fact a clear improvement.

(Case 4) From a strictly cost standpoint, a doubling of the scheduled maintenance effort is worthwhile if it yields at least a 10% increase in reliability. However, such a small reliability increase still leaves a large mean passenger delay.

(Case 5) The minor improvement of Case 3 represents a clear advancement even at costs more expensive than $1000, up to a cost of $14,300. Therefore, the conclusion of Case 3 will hold up even against severe cost overruns.

(Cases 5,6, and 7) As the equipment reliability improves, the corrective maintenance costs decrease. One can therefore sustain increased acquisition costs in order to obtain this improved reliability and still have the situation of better passenger dependability for the same or lower cost. This situation of absolute improvement holds as long as the acquisition cost remains below a given cutoff figure, which is plotted in Figure 9 against the associated equipment reliability (for acquisition costs above this figure, the situation becomes a cost/performance tradeoff rather than an absolute improvement). For an improvement in reliability from 1000 MCBF to 1667 MCBF (a range of improvement possibly achievable by a small enhancement or retrofit), the acquisition costs can increase by up to 43% while still yielding an absolute improvement. For an improvement in reliability from 1000 MCBF to 3000 MCBF purchase of new or replacement equipment), the acquisition costs can increase by up to 72%. For an improvement in reliability from 1000 MCBF to 10000 MCBF (a range of improvement which likely would need an advance in the state of the art compared to the equipment used in the base case), the acquisition costs can increase by up to 97%.

(COMBINATIONS OF ACQUISITION COST AND EQUIPMENT RELIABILITY WHICH GIVE RISE TO EQUAL TOTAL ANNUAL COSTS)

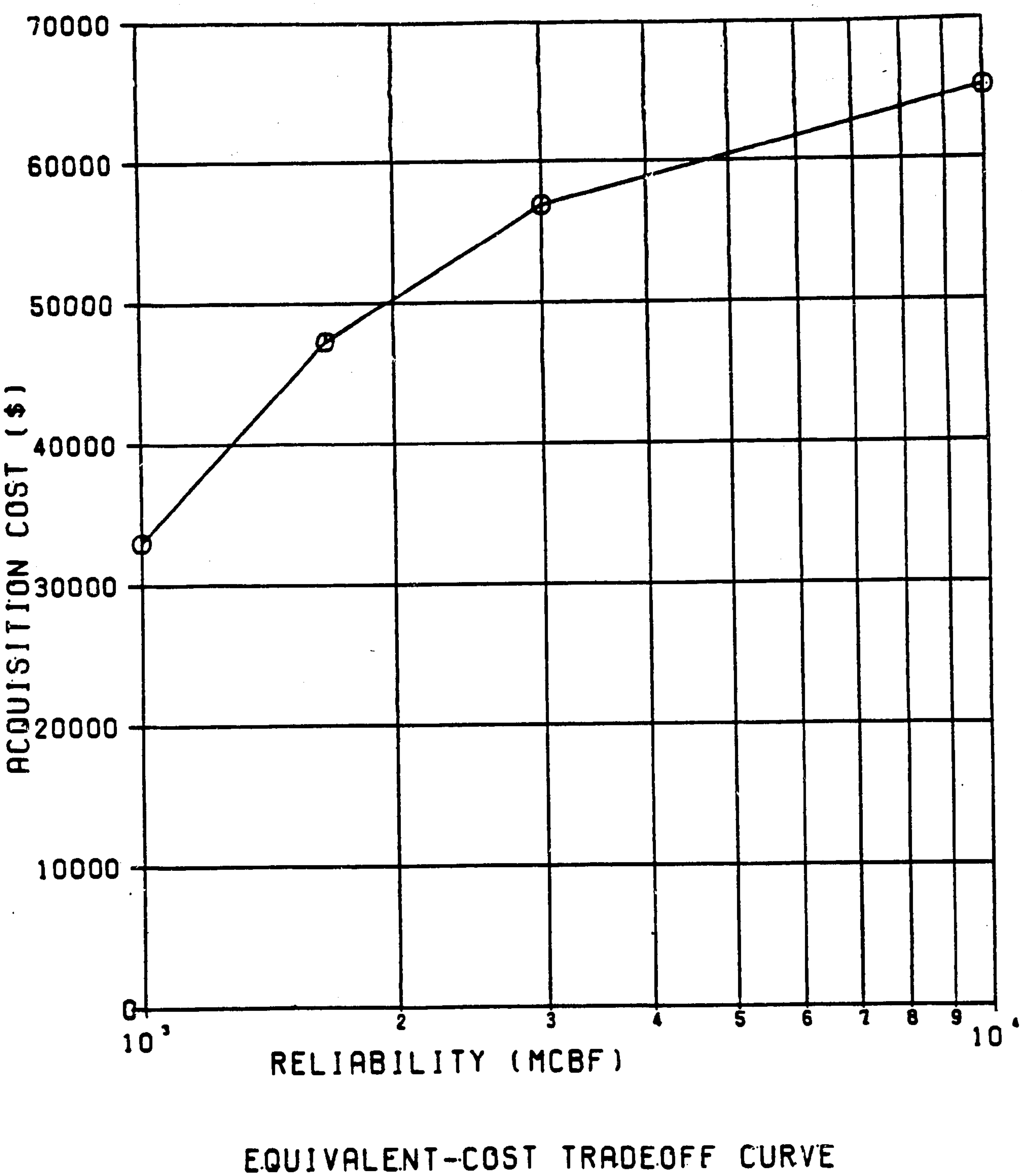


FIGURE 9. EQUIVALENT-COST TRADEOFF CURVE

Note that the acquisition costs increasing by more than the given limits for the specified reliability does not imply that the improved equipment would not be a better buy than the base-case equipment. Rather, it would simply imply that total annual costs would increase. Since the mean passenger delay is significantly better for the improved equipment (the decreases in delay are 67%, 85%, and 98%, respectively, for the three cases), the additional annual cost could in fact be worthwhile. The analysis in this case would show the tradeoff between annual cost and mean passenger delay, and thus facilitate the decision between the two choices of equipment.

(Cases 6 and 8) In order to obtain an increase in the useful equipment life from 10 years to 15 years (assuming an equipment reliability of 3000 MCBF), it is feasible to spend up to a 24% increase in the acquisition cost. Since the equipment reliability and hence the mean passenger delay is unchanged, this does not represent a tradeoff between cost and dependability, but rather a clear improvement or worsening of the current situation, according to whether the acquisition cost increases by less or more than 24%.

7. SUMMARY AND FURTHER DEVELOPMENT

In this report, we have discussed the concept and importance of fare collection system dependability analysis, as well as the types of such analyses and their uses to transit systems. Analysis software consisting of simulation and analytic queuing models has been developed to assess the performance and cost of fare collection systems. Sample analyses based on data derived from actual transit systems have been presented to demonstrate the use of this software for fare collection system analysis. (If these sample analyses had been those of an actual transit system, such a system would have realized that they could reduce their reliability specifications by nearly an order of magnitude with little degradation of performance, and also implement a reliability-improving retrofit while recouping the expenditure involved through lower corrective maintenance costs.)

This report and the accompanying software fulfills the development of the fare collection performance model. However, the modeling effort is really not complete until transit systems can actually use the model to carry out their fare collection system analyses. The next step, therefore, is to facilitate the use of the model by transit operators and planners. Towards this end, we shall enhance the model's efficiency and user-friendliness, and demonstrate the model by applying it to current fare collection planning efforts. This includes the following activities:

User-friendliness

- Improvements in model efficiency
- Interactive menu and query system for users
- Graphic output
- User's manual
- Obtaining of comments and feedback from potential transit users

Model Demonstration

- Support of fare collection system planning and design efforts
- Implementation of models at existing transit systems
- Holding workshops and conference presentations on the model and its use

APPENDIX

PROGRAM LISTINGS FOR FARE COLLECTION DEPENDABILITY ANALYSIS MODELS

A.1 LISTINGS FOR SIMULATION MODEL

Main Program MAIN.

Subroutines

DPTIME
QUEUE
BRAKDN
RANDU
BEGIN
INPUT
ENDING
CSTMOD

```
      COMMON/S1/DEP(3,500),NCUST(3),PASS(3,500)
     1 ,TDLAY(3,500),FUNIT(3,500)
      COMMON/S2/ARRTM(3,500),NQUEUE(3),PASQ(3,500)
     1 ,TDLAYQ(3,500),FUNITQ(3,500)
      COMMON/S3/REP(3,30),NDOWN(3)
      COMMON/S4/I5,IMULT(1000),IQUE(1000),IUNIT(1000),ILOC2(1000)
      COMMON/S5/NM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,TIMEND,DLYMAX
     1 ,PRTIM,ALT,PROB,ISEED,LAST,ICOST,IPRINT
     2 ,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR
      DIMENSION LABEL(5),LSTAR(51),QLSTAR(101)
      DIMENSION CUNIT(3),LIFE(3),SPARES(3),OPER(3),HRSMNT(3)
      DIMENSION WRATE(3),COST(3,6),QLPROB(500)
      INTEGER QLDIST(3,0:500)
      INTEGER PASSYR
      INTEGER PASS,PASQ,AUNIT,FUNIT,FUNITQ
      INTEGER QUE,QL(3),BLANK,STAR,QSTAR,QLSTAR,COUNT(3),NM(3),
     1 PRINT,MULT(3),ONE,FROMWH,UNIT, MCBF(3),NP2(3),NPASS(3)
      INTEGER PERIOD,RPT
      REAL LOSSPR(3), TOTQL(3), PROB(3),RPRBAR(3),TOTDLY(3)
      REAL QLBAR(3), SRVBAR(3), SRVRTE(3), DLYBAR(3)
      DATA LABEL/'ARRIV','DEPRT','RPAIR','BRKDN','CONTN'/
      DATA FROMWH/0/,ONE/1/
      DATA NMUNIT/3/,BLANK/' '/,STAR/'*'/,QSTAR/'#'/
C
C  READ IN THE INPUT THROUGH SUBROUTINES BEGIN AND INPUT
C
   20 CALL BEGIN(FROMWH,NM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,
     1 TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED,PRINT,NMUNIT,CUNIT
     2 ,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR,ICOST,IPRINT)
C-----------------------------------------------------------------
C
C  MOST OF THE INPUT VARIABLES (NM,ARRATE,NARR,SRVBAR,MCBF,
C  RPRBAR,TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED), ARE LISTED
C  IN SUBROUTINE 'BEGIN'.
C
C  OTHER MAIN VARIABLES AND CONSTANTS:
C
C  TYPE  NAME    DESCRIPTION
C
C   R    ARR     NEXT ARRIVAL
C   R    ARRBAR  MEAN ARRIVAL TIME IN SECONDS
C   R    ARRTM   ARRIVAL TIME
C   I    BLANK   BLANK VALUE FOR GRAPHS
C   R    BRK     2500 RANDOM NUMBERS FOR BREAKDOWNS
C   I    COUNT   IN CONJUNCTION WITH 'MULT', AIDS IN PRINTING THE
C                QUEUE LENGTH GRAPH EVERY 10 SECONDS
C   R    DELAY   DELAY TIME
C   R    DEP     DEPARTURE ARRAY
C   R    DEP1    NEXT DEPARTURE-IS USED ONLY FOR PRINTING PURPOSES
C                AND DOES NOT FIGURE INTO ANY CALCULATIONS
C   R    DLYBAR  MEAN DELAY
C   R    DTIME   DEPARTURE TIME FOR THAT PASSENGER
C   I    INDEX   DETERMINES IF AN ARRIVAL, DEPARTURE OR REPAIR
C                SHOULD BE PROCESSED NEXT (VALUES ARE 1, 2 OR 3
C                WITH 0 AT THE START)
C   I    LABEL   ALPHABETIC ARRAY OF TITLES FOR THE CURRENT
C                PROCEDURE
C   I    LEVENT  LAST EVENT
C   I    LOC     LOCATION OF '*' WITHIN GRAPH TO BE PRINTED ON UNIT 4
C   I    LOC1    LOCATION OF '#' WITHIN GRAPH TO BE PRINTED ON UNIT 7
C   I    LOC2    LOCATION OF '#' WITHIN GRAPH TO BE PRINTED ON UNIT 5
```

```
C    I    LOSS     LOSS OF SERVICE
C    R    LOSSPR   LOSS OF SERVICE PROBABILITY
C    I    LSTAR    GRAPH FOR UNIT 4
C    I    MULT     IN CONJUNCTION WITH 'COUNT', AIDS IN PRINTING THE
C                  QUEUE LENGTH GRAPH EVERY 10 SECONDS
C    I    NCUST    NUMBER OF CUSTOMERS-ONLY USED IN SUBROUTINE 'DPTIME'
C    I    NDOWN    NUMBER OF BREAKDOWNS-ONLY USED IN SUBROUTINE 'BRAKDN'
C    I    NMUNIT   NUMBER OF UNITS
C    I    NP2      PASSENGER NUMBER - USED IN MEAN QUEUE LENGTH
C    I    NPASS    PASSENGER NUMBER-INVOLVED IN MEAN DELAY
C    I    NQUEUE   NUMBER IN QUEUE-ONLY USED IN SUBROUTINE 'QUEUE'
C    I    NSWTCH   A SWITCH USED IN SETTING NEW ARRIVAL RATE
C    I    PERIOD   PERIOD FOR THE QUEUE LENGTH GRAPH (USUALLY 10 SECONDS)
C    I    QL       QUEUE LENGTH IN SERVICE OR WAITING
C    R    QLBAR    MEAN QUEUE LENGTH
C    I    QLSTAR   GRAPH FOR UNIT 5 AND UNIT 7
C    I    QSTAR    # VALUE FOR GRAPHS
C    I    QUE      USED FOR PLACEMENT OF # IN GRAPHS
C    R    REP      REPAIR ARRAY
C    R    REP1     NEXT REPAIR-THIS VARIABLE IS ONLY USED FOR PRINTING
C                  PURPOSES AND DOES NOT FIGURE INTO ANY CALCULATIONS
C    R    RPR      ARRAY OF RANDOM NUMBERS FOR BREAKDOWNS
C    R    RPTIME   REPAIR TIME
C    R    SERVTM   SERVICE TIME
C    R    SRVRTE   SERVICE RATE-EACH MACHINE SERVES X NO. OF PASSENGERS
C                  PER HOUR
C    I    STAR     * VALUE FOR GRAPHS
C    R    TMIN     TIME MINIMUM, TIME ARRIVAL
C    R    TOTDLY   TOTAL DELAY
C    R    TOTQL    TOTAL QUEUE LENGTH
C    I    UNIT     THE NUMBER OF UNITS (OR TYPES OF MACHINES);
C                  CURRENTLY DEFINED AT 3, BUT TO RETURN TO A SINGLE
C                  UNIT, JUST GIVE AN ARRIVAL PROBABILITY OF 1.0 FOR
C                  UNIT 3, AND THE OTHER 2 UNITS WILL NEVER BE USED
C
C
C    INITIALIZE ALL COUNTERS
C
      LAST=0
      NSWTCH=0
      NPT=0
      I5=0
      INDEX=0
      J=0
      K=0
      DO 45 UNIT=1,NMUNIT
      NCUST(UNIT)=0
      NQUEUE(UNIT)=0
      NDOWN(UNIT)=0
      NPASS(UNIT)=0
      QL(UNIT)=0
      NP2(UNIT)=0
      TOTDLY(UNIT)=0.
      TOTQL(UNIT)=0.
      MULT(UNIT)=0
      COUNT(UNIT)=0
      DEP(UNIT,1)=1.E9
      REP(UNIT,1)=1.E9
      LOSSPR(UNIT)=MCBF(UNIT)/FLOAT(MCBF(UNIT)-1)
      DO 45 I=0,500
      QLDIST(UNIT,I)=0
```

```
45        CONTINUE
C
C   GET RANDOM NUMBER GENERATOR STARTED
C
          IF(ISEED.EQ.99999)GO TO 51
          CALL RANDU(ISEED,IX1,R)
          CALL RANDU(ISEED+40,IX2,S)
          CALL RANDU(ISEED+80,IX3,S)
          CALL RANDU(ISEED+120,IX4,S)
          CALL RANDU(ISEED+160,IX5,S)
          GO TO 52
51        TYPE 512
512       FORMAT(' ENTER LAST RANDOM NUMBER SEEDS FROM PRIOR RUN'/
     1    7X'(FIVE INTEGER VALUES; ON ONE LINE SEPARATED BY SPACES)')
          ACCEPT 511,IX1,IX2,IX3,IX4,IX5
511       FORMAT(5I)
          CALL RANDU(IX1,IY,R)
          IX1=IY
C
C   DETERMINE FIRST ARRIVAL
C
52        ARRBAR=3600./ARRATE
          ARR=-ARRBAR*ALOG(1.-R)
C
C   COMPUTE SERVICE RATE
C
          DO 44 UNIT=1,NMUNIT
          SRVRTE(UNIT)=1.E9
          IF(SRVBAR(UNIT).EQ.0.)GO TO 44
          SRVRTE(UNIT)=3600./SRVBAR(UNIT)
   44     CONTINUE
C
C   INITIALIZE GRAPH ARRAYS
C
          DO 5 I=1,51
5         LSTAR(I)=BLANK
          DO 18 I=1,101
18        QLSTAR(I)=BLANK
          WRITE(4,302)
302       FORMAT(1H1,' PASSNGR',' AT   ARRIVAL',2X'SERVICE',3X'DELAY',
     1    /,9X,'UNIT',4X,'TIME',5X,'TIME',4X'TIME'/)
          WRITE(7,400)
400       FORMAT(1H1,' PASS',2X,'UNIT    ','ARRIVAL',2X'QUEUE'/15X'TIME',
     1    4X,'LENGTH'/)
480       FORMAT(1H1,' TIME',2X'QUEUE',4X,'AT',/7X'LENGTH  UNIT'//)
          WRITE(6,202)
202       FORMAT(1H1,2X'TIME',' EVENT',6X'NEXT',7X'NEXT',5X'NEXT'
     1    ,4X'QUEUE',3X'MACHINES    AT'/19X'ARRIVAL',2X'DEPARTURE'
     2    ,2X'REPAIR',2X'LENGTH',2X,'IN SERVICE  UNIT'/)
          WRITE(8,490)
490       FORMAT('  DEPARTURE',13X'INITIAL',3X'TOTAL'/
     1    3X'TIME',5X'PASSENGER',4X'UNIT',5X'DELAY'/)
          DO 19 LD=10,30,10
          LD4=LD+4
          LD7=LD+7
          LD8=LD+8
          WRITE(LD4,302)
          WRITE(LD7,400)
          WRITE(LD8,490)
19        CONTINUE
C
```

```
C  PRINT OUT GRAPH HEADINGS FOR INTERACTIVE GRAPHS
C
      IF(PRINT.EQ.4) TYPE 1302
 1302 FORMAT(1H1,' PASS AT ARRIVAL SERVICE  DELAY ',/,
     1 '      UNIT  TIME    TIME    TIME '/)
      IF(PRINT.EQ.7) TYPE 1400
 1400 FORMAT(1H1,' PASS UN ARRIVAL QUEUE',/,11X,'TIME  LENGTH'/)
      IF(PRINT.EQ.5) TYPE 1480
 1480 FORMAT(1H1,'  TIME QUEUE AT',/,7X,'LENGTH UNIT'/)
      IF(PRINT.EQ.6)TYPE 202
      IF(PRINT.EQ.8)TYPE 490
C
C  START SIMULATION
C
C     WHEN INDEX = 1, DO ARRIVAL
C     WHEN INDEX = 2, DO DEPARTURE
C     WHEN INDEX = 3, DO REPAIRS
C
9     LEVENT=INDEX
      INDEX=2
      D=AMIN1(DEP(1,1),DEP(2,1),DEP(3,1))
      R=AMIN1(REP(1,1),REP(2,1),REP(3,1))
      IF(ARR.LT.D)INDEX=1
      TMIN=AMIN1(ARR,D)
      IF(R.LE.TMIN)INDEX=3
      TMIN=AMIN1(R,TMIN)
C
C  DRAW A UNIT FOR THE FIRST PASSENGER TO ARRIVE AT
C  NOTE THAT THIS SECTION IS ONLY USED ONCE AT THE VERY BEGINNING
C
      IF(LEVENT.GT.0) GO TO 42
      CALL RANDU(IX2,IY,S)
      IX2=IY
      NUNIT=1
      IF(S.GT.PROB(1)) NUNIT=2
      IF(S.GT.PROB(1)+PROB(2)) NUNIT=3
      UNIT=NUNIT
      GO TO 43
C  THIS UNIT IS THE ONE ASSOCIATED WITH TMIN
42    UNIT=1
      IF(INDEX.EQ.1) UNIT=NUNIT
      IF(INDEX.EQ.2.AND.DEP(2,1).LT.DEP(1,1).AND.DEP(2,1).LT.DEP(3,1))
     1 UNIT=2
      IF(INDEX.EQ.2.AND.DEP(3,1).LT.DEP(1,1).AND.DEP(3,1).LT.DEP(2,1))
     1 UNIT=3
      IF(INDEX.EQ.3.AND.REP(2,1).LT.REP(1,1).AND.REP(2,1).LT.REP(3,1))
     1 UNIT=2
      IF(INDEX.EQ.3.AND.REP(3,1).LT.REP(1,1).AND.REP(3,1).LT.REP(2,1))
     1 UNIT=3
   43 DEP1=DEP(UNIT,1)
      IF(DEP(UNIT,1).GT.9.E8)DEP1=0.
      REP1=REP(UNIT,1)
      IF(REP(UNIT,1).GT.9.E8)REP1=0.
      IF(INDEX.EQ.1)TOTQL(UNIT)=TOTQL(UNIT)+.5*NARR*(NARR-1)
     1  +NARR*QL(UNIT)
      WRITE(6,201)TMIN,LABEL(INDEX),ARR,DEP1,REP1,QL(UNIT),NM(UNIT)
     1 ,UNIT
      IF(PRINT.EQ.6) TYPE 201,TMIN,LABEL(INDEX),ARR,DEP1,REP1,
     1 QL(UNIT), NM(UNIT),UNIT
201   FORMAT(1XF6.1,2XA5,3XF7.1,4XF7.1,2XF7.1,4XI4,5XI4,5X,I4)
C
```

```
C   IS IT TIME FOR THE SIMULATION TO END?
C
          IF(TMIN.GT.TIMEND)GO TO 4
C
C   IS IT TIME FOR PRINTING A 'COLUMN' OF THE QUEUE LENGTH GRAPH?
C   THIS IS A SAMPLING GRAPH, THAT IS, IT PRINTS EVERY 10 SECONDS
C
          PERIOD=10
          IF(TMIN .LT. MULT(UNIT))GO TO 500
          IF(LEVENT .EQ. 0) QUE=QL(UNIT)
          IF(LEVENT .EQ. 1) QUE=QL(UNIT)-1
          IF(QUE.EQ.-1)QUE=0
          IF(LEVENT .EQ. 2) QUE=QL(UNIT) + 1
          IF(LEVENT .EQ. 3) QUE=QL(UNIT)
          IF((TMIN-MULT(UNIT)) .LT. PERIOD) GO TO 510
515       LOC2=MIN0((QUE + 1),101)
          QLSTAR(LOC2)=QSTAR
C
C   THE MAXIMUM NUMBER OF CHARACTERS FOR THIS GRAPH IS 55
C
          IF(PRINT.EQ.5.AND.LOC2.GT.55) QLSTAR(55)=QSTAR
          II=MIN0(LOC2,55)
          IF(PRINT.EQ.5) TYPE 1520, MULT(UNIT),QUE,UNIT,
     1     (QLSTAR(IJ),IJ=1,II)
 1520     FORMAT(1X,I5,1X,I4,2X,I2,2X,55A1)
          IF(PRINT.EQ.5.AND.LOC2.GT.55) QLSTAR(55)=BLANK
          IF(NMUNIT.EQ.1)GO TO 91
          I5=I5+1
          IMULT(I5)=MULT(UNIT)
          IQUE(I5)=QUE
          IUNIT(I5)=UNIT
          ILOC2(I5)=LOC2
520       FORMAT(1XI4,2(3X,I4),3X101A1)
91        QLSTAR(LOC2)=BLANK
          COUNT(UNIT)=COUNT(UNIT) + 1
          MULT(UNIT)=COUNT(UNIT) * PERIOD
          IF((TMIN - MULT(UNIT)) .GE. PERIOD) GO TO 515
510       LOC2=MIN0((QUE + 1),101)
          IF(NMUNIT.EQ.1)GO TO 92
          I5=I5+1
          IMULT(I5)=MULT(UNIT)
          IQUE(I5)=QUE
          IUNIT(I5)=UNIT
          ILOC2(I5)=LOC2
92        QLSTAR(LOC2)=QSTAR
          IF(PRINT.EQ.5.AND.LOC2.GT.55) QLSTAR(55)=QSTAR
          II=MIN0(LOC2,55)
          IF(PRINT.EQ.5) TYPE 1520, MULT(UNIT),QUE,UNIT,
     1     (QLSTAR(IJ),IJ=1,II)
          IF(PRINT.EQ.5.AND.LOC2.GT.55) QLSTAR(55)=BLANK
          QLSTAR(LOC2)=BLANK
          COUNT(UNIT)=COUNT(UNIT) + 1
          MULT(UNIT)=COUNT(UNIT) * PERIOD
C
C   DETERMINE IF THE CHANGE IN ARRIVAL TIME IS TO BE MADE
C
500       IF(TMIN.LE.PRTIM)GO TO 8
          IF(NSWTCH.EQ.1)GO TO 8
          NSWTCH=1
          ARRATE=ALT
          ARRBAR=3600./ARRATE
```

```
      WRITE(4,310)PRTIM,NARR,ARRATE
      IF(PRINT.EQ.4) TYPE 310, PRTIM,NARR,ARRATE
310   FORMAT(//' THE SIMULATION HAS NOW COVERED',F5.0,'SECONDS.
    1 ARRIVALS ARE IN GROUPS OF',I4,'AT A RATE OF',F5.0,'PER HOUR')
C
8     GO TO (1,2,3),INDEX
C
C  GENERATE NEXT ARRIVAL
C
1     CALL RANDU(IX1,IY,S)
      IX1=IY
      ARR=ARR-ARRBAR*ALOG(1.-S)
C
C  GENERATE UNIT FOR NEW ARRIVALS AFTER CURRENT ARRIVALS HAVE
C  BEEN PROCESSED
C
C  NOTE THAT IF NARR (NUMBER OF ARRIVALS) IS GREATER THAN 1,
C  EACH ARRIVAL WILL STILL TAKE PLACE AT THE SAME SERVICE UNIT
C
      CALL RANDU(IX2,IY,S)
      IX2=IY
      NUNIT=1
      IF(S.GT.PROB(1)) NUNIT=2
      IF(S.GT.PROB(1)+PROB(2)) NUNIT=3
C
C  PROCESS ARRIVALS IN GROUPS
C
      DO 11 I=1,NARR
C
C  THIS IS ONE OF TWO PLACES WHERE 'NP2' IS INCREMENTED;
C  IT IS AN EXACT COUNT OF THE NUMBER OF ARRIVALS AT EACH UNIT.
C
C  WHEN THIS NUMBER IS PRINTED OUT AT THE END, REMEMBER
C  THAT ALL OF THEM MAY NOT BE PRINTED OUT IN THE DETAILED
C  SUMMARIES OF, FOR EXAMPLE, REPORT 3.  THIS IS BECAUSE
C  THEY GOT PUT IN THE ARRIVAL QUEUE (ARRTM) AND WHEN 'TMIN'
C  HIT THE CUTOFF TIME, THEY WERE LEFT HANGING IN THE QUEUE.
C
      NPT=NPT+1
      TDELAY=0.
      IF(NPT/500*500.EQ.NPT)TYPE 997,NPT
997   FORMAT(1XI4,' PASSENGERS HAVE ARRIVED')
      NP2(UNIT)=NP2(UNIT) + 1
      LOC1=MIN0((QL(UNIT)+1),101)
      QLSTAR(LOC1)=QSTAR
      WRITE(7,401)NP2(UNIT),UNIT,TMIN,QL(UNIT),(QLSTAR(II),II=1,LOC1)
      NQL=QL(UNIT)
      QLDIST(UNIT,NQL)=QLDIST(UNIT,NQL)+1
      IF(PRINT.EQ.7.AND.LOC1.GT.49) QLSTAR(49)=QSTAR
      II=MIN0(LOC1,49)
      IF(PRINT.EQ.7) TYPE 1401,NP2(UNIT),UNIT,TMIN,QL(UNIT),
    1  (QLSTAR(IJ),IJ=1,II)
1401  FORMAT(1X,I5,I2,1X,F7.1,1X,I4,1X,49A1)
      IF(PRINT.EQ.7.AND.LOC1.GT.49) QLSTAR(49)=BLANK
      LD=7+10*UNIT
      IF(NMUNIT.GT.1)WRITE(LD,401)NP2(UNIT),UNIT,TMIN,QL(UNIT)
    1 ,(QLSTAR(II),II=1,LOC1)
401   FORMAT(1XI4,1X,I4,3XF7.1,3XI4,3X101A1)
      QLSTAR(LOC1)=BLANK
      IF(QL(UNIT).GE.NM(UNIT))GO TO 12
C
```

```
C   GENERATE SERVICE TIME
C   NO DELAYS FOR THESE ARRIVALS
C
          CALL RANDU(IX3,IY,U)
          IX3=IY
          SERVTM=-SRVBAR(UNIT)*ALOG(1.-U)
C
C   NO DELAYS FOR THESE ARRIVALS
C
          DTIME=TMIN+SERVTM
          DELAY=0.
          NPASS(UNIT)=NPASS(UNIT)+1
          LSTAR(1)=STAR
          TOTDLY(UNIT)=TOTDLY(UNIT)+DELAY
          WRITE(4,301)NPASS(UNIT),UNTT,TMIN,SERVTM,DELAY,LSTAR(1)
          IF(PRINT.EQ.4) TYPE 1301, NPASS(UNIT), UNIT,TMIN,SERVTM,
      1   DELAY,(LSTAR(IJ),IJ=1,1)
 1301     FORMAT(1X,I5,I2,3F8.1,1X,39A1)
          LD=4+10*UNIT
          IF(NMUNIT.GT.1)WRITE(LD,301)NPASS(UNIT),UNIT,TMIN,SERVTM
      1   ,DELAY,LSTAR(1)
301       FORMAT(1XI5,2X,I4,2X,3F8.1,2X51A1)
          LSTAR(1)=BLANK
          CALL DPTIME(1,DTIME,NPT,TDELAY,UNIT,UNIT)
          GO TO 11
C
C   PUT AN ARRIVAL INTO THE QUEUE BECAUSE THE QUEUE EXCEEDS
C   THE NUMBER OF AVAILABLE MACHINES FOR THAT UNIT.
C
12        CALL QUEUE(1,TMIN,NPT,TDELAY,UNIT,UNIT)
11        QL(UNIT)=QL(UNIT)+1
          GO TO 9
C
C   PROCESS NEXT DEPARTURE
C
2         QL(UNIT)=QL(UNIT)-1
C
C   THIS IS THE ONLY DEPARTURE TIME THAT PASSENGERS LEAVE THE
C   DEPARTURE TIME QUEUE
C
          TDELAY=TDLAY(UNIT,1)
          JFUNIT=FUNIT(UNIT,1)
          NPS=PASS(UNIT,1)
          CALL DPTIME(2,0.,0,0.,0,UNIT)
C
C   CHECK WHETHER A BREAKDOWN HAS OCCURRED
C
          CALL RANDU(IX4,IY,BRK)
          IX4=IY
          LOSS=LOSSPR(UNIT)*BRK
          IF(LOSS.EQ.0)GO TO 21
          TYPE 899,UNIT,TMIN
899       FORMAT(5X'BREAKDOWN AT UNIT',I6,', AT TIME ',F8.1)
C
C   GENERATE REPAIR TIME
C
          CALL RANDU(IX5,IY,RPR)
          IX5=IY
          RPTIME=TMIN-RPRBAR(UNIT)*ALOG(1.-RPR)
C
C   A MACHINE ENTERS THE BREAKDOWN QUEUE
```

```
C
      CALL BRAKDN(1,RPTIME,UNIT)
      NM(UNIT)=NM(UNIT)-1
      IF(NM(UNIT).LT.0) TYPE 995
  995 FORMAT('  NM IS NEGATIVE.  STOP.')
      IF(NM(UNIT).LT.0) STOP 10
      DEP1=DEP(UNIT,1)
      IF(DEP(UNIT,1).GT.9.E8)DEP1=0.
      WRITE(6,201)TMIN,LABEL(4),ARR,DEP1,REP(UNIT,1),QL(UNIT),
     1 NM(UNIT),UNIT
      IF(PRINT.EQ.6) TYPE 201,TMIN,LABEL(4),ARR,DEP1,REP(UNIT,1),
     1 QL(UNIT),NM(UNIT),UNIT
      GO TO 49
21    IF(QL(UNIT).LT.NM(UNIT))GO TO 49
      CALL RANDU(IX3,IY,V)
      IX3=IY
      SERVTM=-SRVBAR(UNIT)*ALOG(1.-V)
      DTIME=TMIN+SERVTM
      DELAY=TMIN-ARRTM(UNIT,1)
      NPQ=PASQ(UNIT,1)
      TQDLY=TDLAYQ(UNIT,1)+DELAY
      NPASS(UNIT)=NPASS(UNIT)+1
      LOC=MIN1((DELAY/DLYMAX*50.)+1.99,51.)
      LSTAR(LOC)=STAR
      TOTDLY(UNIT)=TOTDLY(UNIT)+DELAY
      WRITE(4,301)NPASS(UNIT),UNIT,ARRTM(UNIT,1),SERVTM
     1 ,DELAY,(LSTAR(II),II=1,LOC)
      IF(PRINT.EQ.4.AND.LOC.GT.39) LSTAR(39)=STAR
      II=MIN0(LOC,39)
      IF(PRINT.EQ.4) TYPE 1301, NPASS(UNIT),UNIT,ARRTM(UNIT,1),
     1 SERVTM,DELAY,(LSTAR(IJ),IJ=1,II)
      IF(PRINT.EQ.4.AND.LOC.GT.39) LSTAR(39)=BLANK
      LD=4+10*UNIT
      IF(NMUNIT.GT.1)WRITE(LD,301)NPASS(UNIT),UNIT
     1 ,ARRTM(UNIT,1),SERVTM,DELAY,(LSTAR(II),II=1,LOC)
      LSTAR(LOC)=BLANK
      CALL DPTIME(1,DTIME,NPQ,TQDLY,FUNITQ(UNIT,1),UNIT)
      CALL QUEUE(2,0.,0,0.,0,UNIT)
C
C  REMOVE AN ARRIVAL FROM THE QUEUE
C
   49 IF(UNIT.EQ.NMUNIT) GO TO 50
      UNIT=UNIT+1
      DEP1=DEP(UNIT,1)
      IF(DEP(UNIT,1).GT.9.E8)DEP1=0.
      REP1=REP(UNIT,1)
      IF(REP(UNIT,1).GT.9.E8)REP1=0.
      WRITE(6,201) TMIN,LABEL(5),ARR,DEP1,REP1,QL(UNIT),
     1 NM(UNIT), UNIT
      IF(PRINT.EQ.6) TYPE 201, TMIN,LABEL(5),ARR,DEP1,REP1,QL(UNIT),
     1 NM(UNIT), UNIT
C
C  THERE HAS BEEN A DEPARTURE AND NOW IT IS TIME FOR THIS
C  CUSTOMER TO MOVE INTO THE NEXT UNIT.  HERE DEPARTURE TIME
C  IS THE NEXT ARRIVAL TIME.  THIS IS STILL VARIABLE TMIN,
C  NOT VARIABLE ARR.
C
      NP2(UNIT)=NP2(UNIT)+1
      LOC1=MIN0((QL(UNIT)+1),101)
      QLSTAR(LOC1)=QSTAR
      WRITE(7,401) NP2(UNIT),UNIT,TMIN,QL(UNIT),(QLSTAR(II),II=1,LOC1)
```

```
      NQL=QL(UNIT)
      QLDIST(UNIT,NQL)=QLDIST(UNIT,NQL)+1
      IF(PRINT.EQ.7.AND.LOC1.GT.49) QLSTAR(49) =QSTAR
      II=MIN0(LOC1,49)
      IF(PRINT.EQ.7)TYPE 1401, NP2(UNIT),UNIT,TMIN,QL(UNIT),
     1 (QLSTAR(IJ),IJ=1,II)
      LD=7+10*UNIT
      IF(NMUNIT.GT.1)WRITE(LD,401) NP2(UNIT),UNIT,TMIN,QL(UNIT)
     1 ,(QLSTAR(II),II=1,LOC1)
      IF(PRINT.EQ.7.AND.LOC1.GT.49) QLSTAR(49)=BLANK
      QLSTAR(LOC1)=BLANK
      IF(QL(UNIT).GE.NM(UNIT)) GO TO 112
      CALL RANDU(IX3,IY,Y)
      IX3=IY
      SERVTM=-SRVBAR(UNIT)*ALOG(1.-Y)
      DTIME=TMIN+SERVTM
      DELAY=0.
      NPASS(UNIT)=NPASS(UNIT)+1
      LSTAR(1)=STAR
      TOTDLY(UNIT)=TOTDLY(UNIT)+DELAY
      WRITE(4,301) NPASS(UNIT), UNIT,TMIN,SERVTM,DELAY,LSTAR(1)
      IF(PRINT.EQ.4) TYPE 1301,NPASS(UNIT),UNIT,TMIN,SERVTM,DELAY,
     1 (LSTAR(IJ),IJ=1,1)
      LD=4+10*UNIT
      IF(NMUNIT.GT.1)WRITE(LD,301) NPASS(UNIT), UNIT,TMIN,SERVTM
     1 ,DELAY,LSTAR(1)
      LSTAR(1)=BLANK
      CALL DPTIME(1,DTIME,NPS,TDELAY,JFUNIT,UNIT)
      QL(UNIT)=QL(UNIT)+1
      GO TO 9
C
C  PUT SOMEONE IN THE QUEUE
C
  112 CALL QUEUE(1,TMIN,NPS,TDELAY,JFUNIT,UNIT)
      QL(UNIT)=QL(UNIT)+1
      GO TO 9
50    LOC=MIN1((TDELAY/DLYMAX*50.)+1.99,51.)
      LSTAR(LOC)=STAR
      WRITE(8,308)TMIN,NPS,JFUNIT,TDELAY,(LSTAR(II),II=1,LOC)
308   FORMAT(1XF6.1,5XI5,9XI2,5XF7.1,2X51A1)
      LD=8+10*UNIT
      IF(NMUNIT.GT.1)WRITE(LD,308)TMIN,NPS,JFUNIT,TDELAY
     1 ,(LSTAR(II),II=1,LOC)
      IF(PRINT.EQ.8.AND.LOC.GT.39)LSTAR(39)=STAR
      II=MIN0(LOC,39)
      IF(PRINT.EQ.8)TYPE 308,TMIN,NPS,JFUNIT,TDELAY
     1 ,(LSTAR(IJ),IJ=1,II)
      LSTAR(39)=BLANK
      LSTAR(LOC)=BLANK
      GO TO 9
C
C  PROCESS NEXT REPAIR
C
3     NM(UNIT)=NM(UNIT)+1
      TYPE 898,UNIT,TMIN
898   FORMAT(5X'REPAIR AT UNIT',I6,', AT TIME ',F8.1)
      CALL BRAKDN(2,0.,UNIT)
C
C  A MACHINE LEAVES THE BREAKDOWN QUEUE
C
      IF(QL(UNIT).LT.NM(UNIT))GO TO 9
```

```
      CALL RANDU(IX3,IY,Y)
      IX3=IY
      SERVTM=-SRVBAR(UNIT)*ALOG(1.-Y)
      DTIME=TMIN+SERVTM
      DELAY=TMIN-ARRTM(UNIT,1)
      NPR=PASQ(UNIT,1)
      TDELAY=TDLAYQ(UNIT,1)+DELAY
      NPASS(UNIT)=NPASS(UNIT)+1
      LOC=MIN1((DELAY/DLYMAX*50.)+1.99,51.)
      LSTAR(LOC)=STAR
      TOTDLY(UNIT)=TOTDLY(UNIT)+DELAY
      WRITE(4,301)NPASS(UNIT),UNIT,ARRTM(UNIT,1),SERVTM
     1,DELAY,(LSTAR(II),II=1,LOC)
      IF(PRINT.EQ.4.AND.LOC.GT.39) LSTAR(39)=STAR
      I1=MIN0(LOC,39)
      IF(PRINT.EQ.4) TYPE 1301,NPASS(UNIT),UNIT,ARRTM(UNIT,1),SERVTM,
     1 DELAY,(LSTAR(IJ),IJ=1,II)
      IF(PRINT.EQ.4.AND.LOC.GT.39) LSTAR(39)=BLANK
      LD=4+10*UNIT
      IF(NMUNIT.GT.1)WRITE(LD,301)NPASS(UNIT),UNIT,ARRTM(UNIT,1)
     1,SERVTM,DELAY,(LSTAR(II),II=1,LOC)
      LSTAR(LOC)=BLANK
      CALL DPTIME(1,DTIME,NPR,TDELAY,FUNITQ(UNIT,1),UNIT)
C
C  REMOVE AN ARRIVAL FROM THE QUEUE
C
      CALL QUEUE(2,0.,0,0.,0,UNIT)
      GO TO 9
C
C  END-OF-JOB PROCESSING
C  (INCLUDING COMPUTATION OF MEAN QUEUE LENGTH AND MEAN DELAY TIME)
C
4     IF(ICOST.EQ.0)GO TO 41
      CALL CSTMOD(CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE
     1,PASSYR,NM,MCBF,RPRBAR,PROB,COST)
41    DO 46 UNIT=1,3
      IF(NM(UNIT).EQ.0)WRITE(9,904)UNIT
      IF(NM(UNIT).EQ.0.AND.PRINT.EQ.9)TYPE 904,UNIT
904   FORMAT(//' UNIT',I2,' IS UNASSIGNED')
      IF(NM(UNIT).EQ.0)GO TO 46
      QLBAR(UNIT)=0.
      IF(NP2(UNIT).NE.0)QLBAR(UNIT)=TOTQL(UNIT)/NP2(UNIT)
      DLYBAR(UNIT)=0.
      IF(NPASS(UNIT).NE.0)DLYBAR(UNIT)=TOTDLY(UNIT)/NPASS(UNIT)
      WRITE(9,303)UNIT,QLBAR(UNIT),DLYBAR(UNIT)
303   FORMAT(///' UNIT',I3///4X' MEAN QUEUE LENGTH IS ',F8.1
     1/4X' MEAN DELAY IS ',7XF8.1,'  SEC.')
      IF(IPRINT.EQ.1)TYPE 303,UNIT,QLBAR(UNIT),DLYBAR(UNIT)
      IMAX=-1
      DO 461 I=0,500
461   IF(QLDIST(UNIT,I).GT.0)IMAX=I
      IF(IMAX.EQ.-1)GO TO 462
      TPASS=NP2(UNIT)
      KMAX=MIN0(9,IMAX)
      WRITE(9,901)(K,K=0,KMAX)
      WRITE(9,905)
901   FORMAT(//1X' QUEUE LENGTH PROBABILITIES'//2X10I7)
905   FORMAT(' ')
      IF(IPRINT.EQ.1)TYPE 901,(K,K=0,KMAX)
      IF(IPRINT.EQ.1)TYPE 905
      DO 463 II=0,IMAX,10
```

```
      III=MIN0(II+9,IMAX)
      DO 464 JJ=II,III
      QLPROB(JJ)=QLDIST(UNIT,JJ)/TPASS
 464  CONTINUE
      WRITE(9,903)II,(QLPROB(JJ),JJ=II,III)
      IF(IPRINT.EQ.1)TYPE 903,II,(QLPROB(JJ),JJ=II,III)
 903  FORMAT(1XI3,10(1X2PF5.1,'%'))
 463  CONTINUE
      WRITE(9,906)(K,K=0,KMAX)
      IF(IPRINT.EQ.1)TYPE 906,(K,K=0,KMAX)
 906  FORMAT(//'  CUMULATIVE QUEUE LENGTH PROBABILITIES'//2X10I7)
      WRITE(9,905)
      IF(IPRINT.EQ.1)TYPE 905
      DO 465 II=0,IMAX,10
      III=MIN0(II+9,IMAX)
      DO 466 JJ=II,III
      IF(JJ.EQ.0)GO TO 466
      QLPROB(JJ)=QLPROB(JJ-1)+QLPROB(JJ)
 466  CONTINUE
      WRITE(9,903)II,(QLPROB(JJ),JJ=II,III)
      IF(IPPINT.EQ.1)TYPE 903,II,(QLPROB(JJ),JJ=II,III)
 465  CONTINUE
      NSERV=NP2(UNIT)-NPASS(UNIT)
      WRITE(9,203)NP2(UNIT),NPASS(UNIT),NSERV
      IF(IPRINT.EQ.1)TYPE 203,NP2(UNIT),NPASS(UNIT),NSERV
 203  FORMAT(/1XI5,' PASSENGERS ARRIVED AT SERVICE AREA'
     1 /1XI5,' PASSENGERS SERVED'/1XI5,' PASSENGERS '
     2 ,'NOT YET SERVED')
 462  IF(ICOST.EQ.0)GO TO 46
      IF(LIFE(UNIT).EQ.0)GO TO 46
      WRITE(9,902)UNIT,(COST(UNIT,I),I=1,6)
      IF(IPRINT.EQ.1)TYPE 902,UNIT,(COST(UNIT,I),I=1,6)
 902  FORMAT(//'  COSTS FOR UNIT ',I4
     1 /7X'ANNUALIZED CAPITAL COST',T35,'$',F9.2
     2 /7X'ANNUALIZED SPARES COST',T35,'$',F9.2
     3 /7X'OPERATING COST',T35,'$',F9.2
     4 /7X'SCHEDULED MAINTENANCE COST',T35,'$' [illegible]
     5 /7X'CURRECTIVE REPAIR COST',T35,'$',F9
     6 //7X'TOTAL COST',T34,'$',F10.2)
 46   CONTINUE
      TYPE 399,IX1,IX2,IX3,IX4,IX5
 399  FORMAT(//3X' LAST RANDOM NUMBER SEEDS'/5(4XI15/))
C
C  IS THERE ANOTHER RUN TO DO?
C
      CALL ENDING
      IF(LAST.EQ.1)STOP
      GO TO 20
      END
      SUBROUTINE DPTIME(INDEX,DTIME,PASS,DELAY,FUNIT,UNIT)
C
C  DTIME IS THE CURRENT TIME PLUS SERVICE TIME
C  DEPARTURE TIME.
C
      COMMON/S1/DEP(3,500),NCUST(3),P(3,500),D(3,500),U(3,500)
      INTEGER UNIT,P,PASS,U,FUNIT
      GO TO (1,2),INDEX
 1    IF(NCUST(UNIT).EQ.0)GO TO 14
      NC=NCUST(UNIT)
      DO 11 I=1,NC
      ISV=I
```

```
      IF(DEP(UNIT,I).LT.DTIME)GO TO 11
      GO TO 12
11    CONTINUE
14    NCUST(UNIT)=NCUST(UNIT)+1
      DEP(UNIT,NCUST(UNIT))=DTIME
      P(UNIT,NCUST(UNIT))=PASS
      D(UNIT,NCUST(UNIT))=DELAY
      U(UNIT,NCUST(UNIT))=FUNIT
      RETURN
12    NC=NCUST(UNIT)
      DO 13 JJ=ISV,NC
      J=NCUST(UNIT)+ISV-JJ
      DEP(UNIT,J+1)=DEP(UNIT,J)
      P(UNIT,J+1)=P(UNIT,J)
      D(UNIT,J+1)=D(UNIT,J)
      U(UNIT,J+1)=U(UNIT,J)
13    CONTINUE
      DEP(UNIT,ISV)=DTIME
      P(UNIT,ISV)=PASS
      D(UNIT,ISV)=DELAY
      U(UNIT,ISV)=FUNIT
      NCUST(UNIT)=NCUST(UNIT)+1
      RETURN
2     IF(NCUST(UNIT).EQ.1)DEP(UNIT,1)=1.E9
      IF(NCUST(UNIT).EQ.1)RETURN
      NC=NCUST(UNIT)
      DO 21 I=2,NC
      DEP(UNIT,I-1)=DEP(UNIT,I)
      P(UNIT,I-1)=P(UNIT,I)
      D(UNIT,I-1)=D(UNIT,I)
      U(UNIT,I-1)=U(UNIT,I)
21    CONTINUE
      NCUST(UNIT)=NCUST(UNIT)-1
      RETURN
      END
      SUBROUTINE QUEUE(INDEX,ARTM,PASS,DELAY,FUNIT,UNIT)
C
C  ARRIVAL QUEUE
C
      COMMON/S2/ARRTM(3,500),NQUEUE(3),P(3,500),D(3,500),U(3,500)
      INTEGER UNIT,P,PASS,U,FUNIT
      GO TO (1,2),INDEX
1     IF(NQUEUE(UNIT).EQ.0) GO TO 13
      NQ=NQUEUE(UNIT)
      DO 11 I=1,NQ
      IF(ARRTM(UNIT,I).LT.ARTM) GO TO 11
      DO 12 JJ=1,NQ
      J=NQ+I-JJ
      P(UNIT,J+1)=P(UNIT,J)
      D(UNIT,J+1)=D(UNIT,J)
      U(UNIT,J+1)=U(UNIT,J)
12    ARRTM(UNIT,J+1)=ARRTM(UNIT,J)
      P(UNIT,I)=PASS
      D(UNIT,I)=DELAY
      U(UNIT,I)=FUNIT
      ARRTM(UNIT,I)=ARTM
      NQUEUE(UNIT)=NQUEUE(UNIT)+1
      RETURN
11    CONTINUE
13    NQUEUE(UNIT)=NQUEUE(UNIT)+1
      ARRTM(UNIT,NQUEUE(UNIT))=ARTM
```

```
      P(UNIT,NQUEUE(UNIT))=PASS
      D(UNIT,NQUEUE(UNIT))=DELAY
      U(UNIT,NQUEUE(UNIT))=FUNIT
      RETURN
2     NQUEUE(UNIT)=NQUEUE(UNIT)-1
      IF(NQUEUE(UNIT).EQ.0)ARRTM(UNIT,1)=1.E9
      IF(NQUEUE(UNIT).EQ.0)RETURN
      NQ=NQUEUE(UNIT)
      DO 21 I=1,NQ
      P(UNIT,I)=P(UNIT,I+1)
      D(UNIT,I)=D(UNIT,I+1)
      U(UNIT,I)=U(UNIT,I+1)
21    ARRTM(UNIT,I)=ARRTM(UNIT,I+1)
      ARRTM(UNIT,NQUEUE(UNIT)+1)=0.
      RETURN
      END
      SUBROUTINE BRAKDN(INDEX,RPTIME,UNIT)
      COMMON/S3/REP(3,30),NDOWN(3)
      INTEGER UNIT
      GO TO (1,2),INDEX
1     IF(NDOWN(UNIT).EQ.0)GO TO 13
      ND=NDOWN(UNIT)
      DO 11 I=1,ND
      IF(REP(UNIT,I).LT.RPTIME)GO TO 11
      DO 12 JJ=I,NDOWN(UNIT)
      J=NDOWN(UNIT)+I-JJ
12    REP(UNIT,J+1)=REP(UNIT,J)
      REP(UNIT,I)=RPTIME
      NDOWN(UNIT)=NDOWN(UNIT)+1
      RETURN
11    CONTINUE
13    NDOWN(UNIT)=NDOWN(UNIT)+1
      REP(UNIT,NDOWN(UNIT))=RPTIME
      RETURN
2     IF(NDOWN(UNIT).EQ.1)REP(UNIT,1)=1.E9
      IF(NDOWN(UNIT).EQ.1)RETURN
      ND=NDOWN(UNIT)
      DO 21 I=1,ND-1
21    REP(UNIT,I)=REP(UNIT,I+1)
      NDOWN(UNIT)=NDOWN(UNIT)-1
      RETURN
      END
      SUBROUTINE RANDU(IX,IY,YFL)
C
C  RANDOM NUMBER GENERATOR
C
      IY=IX*262149
      IF(IY) 5,6,6
    5 IY=IY+34359738337+1
    6 YFL=IY
      YFL=YFL*.2910383E-10
      RETURN
      END
      SUBROUTINE BEGIN(FROMWH,NM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,
     1 TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED,PRINT,NMUNIT,CUNIT,LIFE
     2 ,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR,ICOST,IPRINT)
      DOUBLE PRECISION IFN1,IFN2,IFN3,IFN4,IFN5,IFN6
      INTEGER FROMWH, NM(3),MCBF(3), YES,NO,ANSR,PRINT
     1 , STORNM(3),UNIT,LIFE(3),PASSYR
      REAL SRVBAR(3), RPRBAR(3), PROB(3)
      REAL CUNIT(3),SPARES(3),OPER(3),HRSMNT(3),WRATE(3)
```

```
      DATA YES,NO/1HY,1HN/,STORNM/0,0,0/,FILE/1HF/
C-----------------------------------------------------------------------
C
C   STRUCTURE OF THE INPUT AND OUTPUT FILE QUESTIONS
C
C      OUTPUT
C
C         FIVE QUESTIONS ARE ASKED AT THE BEGINNING OF EACH RUN
C         REGARDING GIVING FIVE NAMES OF NEW DISK OUTPUT FILES.
C         THESE DISK OUTPUT FILES CONTAIN (IN UNFORMATTED WORDS),
C         ALL THE DATA THAT IS PRINTED IN THE GRAPHS, ETC., AT THE FIRST
C         PART OF EACH OF THE FIVE REPORTS. THE REASON FOR STORING THIS
C         IS SO THEY CAN BE SORTED LATER AND USED FOR PRODUCING
C         INDIVIDUAL REPORTS BY UNIT.  AT THE END OF A RUN, A QUESTION IS
C         ASKED ABOUT WHETHER THESE FILES SHOULD BE KEPT; UNLESS
C         THEY ARE BEING USED AS INPUT TO ANOTHER PROGRAM OR NEEDED FOR
C         MORE COPIES OR COMPARISONS, THEY CAN BE DELETED.
C
C
C      INPUT
C
C         FOR THE FIRST RUN, INPUT MAY BE READ IN AS ONE RECORD IN FREE
C         FIELD FORMAT FROM A DISK FILE, OR THE PROGRAM CAN ASK FOR DATA
C         TO BE TYPED IN AS A RESPONSE TO EACH OF ITS INPUT QUESTIONS.
C         WHEN THE RUN IS FINISHED, A QUESTION IS ASKED ABOUT WHETHER
C         ANOTHER RUN SHOULD FOLLOW,  THE USER HAS A CHOICE OF 'YES'
C         OR 'NO', AND IF 'YES' HAS ANOTHER CHOICE REGARDING INPUTTING
C         THE DATA.  IF THE DATA IS ON DISK, IT MUST BE THE NEXT
C         RECORD OF THE SAME INPUT FILE, OR THE USER CAN CALL FOR THE
C         INPUT PROMPT SEQUENCE TO CHANGE A FEW VARIABLES OR ALL VARIABLES.
C         ANY TIME THAT INPUT VARIABLES ARE CHANGED, A LIST OF THEIR
C         CURRENT VALUES IS PRINTED BEFORE EXECUTION.
C
C-----------------------------------------------------------------------
      DO 600 UNIT=1,NMUNIT
      NM(UNIT)=STORNM(UNIT)
  600 CONTINUE
      OPEN(UNIT=4,FILE='FOR04.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=6,FILE='FOR06.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=7,FILE='FOR07.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=8,FILE='FOR08.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=9,FILE='FOR09.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=14,FILE='U14.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=17,FILE='U17.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=18,FILE='U18.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=24,FILE='U24.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=27,FILE='U27.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=28,FILE='U28.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=34,FILE='U34.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=37,FILE='U37.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=38,FILE='U38.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      ICONT=0
      IF(FROMWH.GT.0)GO TO 241
      TYPE 20
20    FORMAT(/20X'FARE COLLECTION PASSENGER-DELAY MODEL')
      TYPE 294
294   FORMAT(//6X'THE MODEL PRODUCES OUTPUT FILES SHOWING MEAN '
     1 ,'QUEUE LENGTH AND'/6X'DELAY, AND QUEUE LENGTH '
     2 ,'DISTRIBUTION (AND COSTS IF INCLUDED).'
     3 //6X'SHOULD THESE BE DISPLAYED ON THE TERMINAL AS WELL?'
     4 /6X'(''Y'' OR ''N''): '$)
```

```
242     ACCEPT 7,ANSR
        IPRINT=-1
        IF(ANSR.EQ.YES)IPRINT=1
        IF(ANSR.EQ.NO)IPRINT=0
        IF(IPRINT.GE.0)GO TO 24
        ICONT=ICONT+1
        IF(ICONT.GT.3)STOP 77
        TYPE 15,ANSR
        GO TO 242
24      TYPE 1
1       FORMAT(//,6X'THE GRAPHS PRODUCED BY THE MODEL ARE'//
     1  9X'1 -- PASSENGER DELAYS BY SERVICE AREA'/
     2  9X'2 -- QUEUE LENGTHS AT INTERVALS OF 10 SECONDS'/
     3  9X'3 -- EVENT LOG'/
     4  9X'4 -- QUEUE LENGTHS AT PASSENGER ARRIVALS'/
     5  9X'5 -- TOTAL DELAY PER PASSENGER'/)
        TYPE 21
   21   FORMAT(/,'       ARE ANY OF THE GRAPHS TO BE ALSO PRINTED',
     1  ' DIRECTLY ON THE TERMINAL-',/,'       (ZERO FOR ''NO''',
     2    ' OR THE GRAPH NUMBER (1,2,3,4 OR 5) FOR ''YES''): '$)
   25   ACCEPT 203, PRINT
        PRINT=PRINT+3
  203   FORMAT(I)
        IF(PRINT.EQ.3.OR.PRINT.EQ.4.OR.PRINT.EQ.5.OR.PRINT.EQ.6.
     1   OR.PRINT.EQ.7.OR.PRINT.EQ.8) GO TO 241
        ICONT=ICONT+1
        IF(ICONT.GT.3) STOP 12
        TYPE 310
  310   FORMAT('       THE PREVIOUS RESPONSE WAS ',I5,
     1   /,'       PLEASE ANSWER WITH ONE INTEGER VALUE',
     2  'EITHER 0, 1, 2, 3, 4, 5: '$)
        GO TO 25
241     IF(FROMWH.EQ.2) GO TO 13
        IF(FROMWH.EQ.0)GO TO 19
        TYPE 27
27      FORMAT(6X'IS THE NEW DATA TO COME FROM A NEW INPUT FILE?'/
     1  7X'(THE ALTERNATIVE IS TO MAKE CHANGES IN THE CURRENT INPUT'
     2  ,' DATA)'/6X'(''Y'' OR ''N''): '$)
271     ACCEPT 7,ANSR
        IF(ANSR.EQ.YES)FROMWH=0
        IF(ANSR.EQ.YES)GO TO 19
        IF(ANSR.EQ.NO)CALL INPUT(FROMWH,NM,ARRATE,NARR,SRVBAR,MCBF,
     1  RPRBAR,TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED,PRINT,NMUNIT,
     2  STORNM,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT
     3  ,WRATE,PASSYR,ICOST,IPRINT)
        IF(ANSR.EQ.NO)RETURN
        ICONT=ICONT+1
        IF(ICONT.GT.3)STOP 1
        TYPE 15,ANSR
        GO TO 271
19      ICONT=0
        TYPE 6
    6   FORMAT(/'       ACCEPT INPUT FROM THE TERMINAL-',/,
     1          (''Y'' OR ''N''): '$)
  119   ACCEPT 7, ANSR
    7   FORMAT(A1)
        IF(ANSR.EQ.YES) GO TO 14
        IF(ANSR.EQ.NO) GO TO 30
        ICONT=ICONT+1
        IF(ICONT.GT.3) STOP 1
        TYPE 15,ANSR
```

```
15    FORMAT(/'       THE PREVIOUS RESPONSE WAS ''',
   1  A1,'''.  PLEASE ANSWER ''Y'' OR ''N'': '$)
      GO TO 119
14    CALL INPUT(FROMWH,NM,ARRATE,NARR,SRVBAR,MCBF,
   1  RPRBAR,TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED,PRINT,NMUNIT,
   2  STORNM,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT
   3  ,WRATE,PASSYR,ICOST,IPRINT)
      FROMWH=1
      RETURN
13    ICONT=0
      IF(FROMWH.EQ.0) GO TO 30
      TYPE 28,IFN5
28    FORMAT(/,'       IS THIS NEW DATA EITHER A NEW FILE OR'
   1  ,' THE NEXT RECORD OF FILE ',A10/6X'(''F'' FOR NEW FILE',
   2  ', ''Y'' FOR NEW RECORD, OR ''N'' FOR NEITHER): '$)
40    ACCEPT 7, ANSR
      IF(ANSR.EQ.YES) GO TO 29
      IF(ANSR.EQ.NO) CALL INPUT(FROMWH,NM,ARRATE,NARR,SRVBAR,MCBF,
   .  RPRBAR,TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED,PRINT,NMUNIT,
   2  STORNM,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT
   3  ,WRATE,PASSYR,ICOST,IPRINT)
      IF(ANSR.EQ.NO) RETURN
      IF(ANSR.EQ.FILE)FROMWH=0
      IF(ANSR.EQ.FILE)GO TO 19
      ICONT=ICONT+1
      IF(ICONT.GT.3) STOP 4
      TYPE 15, ANSR
      GO TO 40
C
C  NOTE THAT IF A BAD RESPONSE IS GIVEN TO THIS REQUEST, THAT IS,
C  IF THE PROGRAM CANNOT FIND THE SPECIFIC DATA FILE ON DISK, THE
C  SYSTEM WILL GENERATE A REQUEST FOR A NEW INPUT FILE.  HOWEVER,
C  THE INCORRECT FILE NAME WILL STILL BE PRINTED IN THE SECTION
C  REGARDING THE CURRENT INPUT FILES, BECAUSE THE PROGRAM
C  DIDN'T GENERATE THE NEW REQUEST, THE SYSTEM DID.
C
30    TYPE 22
22    FORMAT(/'       ENTER INPUT FILE NAME  ',/,
   1  '       (UP TO 10 CHARACTERS): ',$)
      ACCEPT 23,IFN5
23    FORMAT(A10)
      IF(FROMWH.EQ.2)CLOSE(UNIT=55)
      OPEN(UNIT=55,FILE=IFN5,DEVICE='DSK',ACCESS='SEQIN')
      TYPE 3, IFN5
3     FORMAT(6X'THE INPUT FILE IS: ',A10)
29    ICOST=0
      ICONT1=0
      TYPE 401
401   FORMAT(//6X'ARE COSTS INCLUDED IN THIS RECORD?'
   1  /6X'(''Y'' OR ''N''): '$)
132   ACCEPT 7,ANSR
      IF(ANSR.EQ.NO)GO TO 291
      IF(ANSR.EQ.YES)GO TO 292
      ICONT1=ICONT1+1
      IF(ICONT1.GT.3)STOP 2
      TYPE 15,ANSR
      GO TO 132
291   READ(55,*)STORNM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,TIMEND,
   1  DLYMAX,PRTIM,ALT,PROB,ISEED
      GO TO 293
292   READ(55,*)STORNM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,TIMEND,
```

```
     1  DLYMAX,PRTIM,ALT,PROB,ISEED,CUNIT,LIFE,DISCRT,SPARES,OPER,
     2  HRSMNT,WRATE,PASSYR
        ICOST=1
293     CLOSE(UNIT=55)
        DO 18 UNIT=1,NMUNIT
        NM(UNIT)=STORNM(UNIT)
   18   CONTINUE
        FROMWH=2
C
C  ARRATE (MEAN ARRIVAL RATE) IS NOT AN ATTRIBUTE OF EACH
C  MACHINE OR SERVICE UNIT, THEREFORE IT IS NOT
C  DIMENSIONED BY UNIT
C
C  INPUT IS FREE FORMAT
C
C  VARIABLE       NAME      TYPE      DESCRIPTION
C-----------------------------------------------------------------
C      1-3         NM        INT      THE NUMBER OF MACHINES FOR
C                                     EACH SERVICE UNIT
C
C       4        ARRATE      REAL     MEAN ARRIVAL RATE PER HOUR
C
C       5         NARR       INT      THE NUMBER OF ARRIVALS IN A
C                                     GROUP
C
C      6-8       SRVBAR      REAL     MEAN SERVICE TIME IN SECONDS
C
C      9-11       MCBF       INT      MEAN CYCLES BETWEEN FAILURES
C                                     FOR EACH SERVICE UNIT
C
C     12-14      RPRBAR      REAL     MEAN TIME TO REPAIR IN SECONDS
C
C      15        TIMEND      REAL     TIME THE SIMULATION ENDS IN
C                                     SECONDS
C
C      16        DLYMAX      REAL     THE MAXIMUM DELAY WHICH CAN BE
C                                     REPRESENTED BY THE GRAPH
C
C      17         PRTIM      REAL     THE TIME IN SECONDS WHEN THE
C                                     NEW ARRIVAL RATE WILL OCCUR
C
C      18          ALT       REAL     THE ALTERNATE MEAN ARRIVAL
C                                     TIME IN SECONDS
C
C     19-21       PROB       REAL     THREE PROBABILITIES OF ARRIVAL
C                                     FOR THE THREE SERVICE UNITS
C
C      22         ISEED      INT      RANDOM NUMBER SEED
C-----------------------------------------------------------------
        CALL INPUT(FROMWH,NM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,
     1  TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED,PRINT,NMUNIT,STORNM,CUNIT
     2  ,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR,ICOST,IPRINT)
        RETURN
        END
        SUBROUTINE INPUT(FROMWH,NM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,
     1  TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED,PRINT,NMUNIT,STORNM,
     2  CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR,ICOST,IPRINT)
        INTEGER PRINT,FROMWH,CHANGE,YES,NO,ANSR,NM(3),MCBF(3)
     1  ,STORNM(3),UNIT,LIFE(3),PASSYR
        REAL SRVBAR(3),RPRBAR(3),PROB(3),CUNIT(3),SPARES(3),OPER(3)
     1  ,HRSMNT(3),WRATE(3)
```

```
      DATA YES, NO/1HY,1HN/
      CHANGE=0
      ICONT=0
      IF(FROMWH.GT.0) GO TO 667
      ICONT=0
  1   TYPE 101
101   FORMAT(/'  1.  ENTER NUMBER OF MACHINES FOR EACH',
   1   ' SERVICE UNIT- '/'       (THREE INTEGER VALUES): '$)
      ACCEPT 201, STORNM
      DO 50 UNIT=1,NMUNIT
      NM(UNIT)=STORNM(UNIT)
 50   CONTINUE
      IF(ICONT.GT.0) GO TO 300
  2   TYPE 102
102   FORMAT(/'  2.  ENTER MEAN ARRIVAL RATE PER HOUR-',/,
   1  '       (ONE REAL VALUE): '$)
      ACCEPT 202,ARRATE
      IF(ICONT.GT.0) GO TO 300
  3   TYPE 103
103   FORMAT(/'  3.  ENTER NUMBER OF ARRIVALS IN EACH ',
   1   'GROUP-',/,'       (ONE INTEGER VALUE): ',$)
      ACCEPT 203, NARR
      IF(ICONT.GT.0) GO TO 300
  4   TYPE 104
104   FORMAT(/'  4.  ENTER MEAN SERVICE TIME IN SECONDS-',
   1  /,'       (THREE REAL VALUES): '$)
      ACCEPT 202, SRVBAR
      IF(ICONT.GT.0) GO TO 300
  5   TYPE 105
105   FORMAT(/'  5.  ENTER MEAN CYCLES BETWEEN FAILURES-',
   1   /,'       (THREE INTEGER VALUES): ',$)
      ACCEPT 201,MCBF
      IF(ICONT.GT.0) GO TO 300
  6   TYPE 106
106   FORMAT(/'  6.  ENTER MEAN TIME TO REPAIR IN SECONDS-',
   1  /,'       (THREE REAL VALUES): ',$)
      ACCEPT 202, RPRBAR
      IF(ICONT.GT.0) GO TO 300
  7   TYPE 107
107   FORMAT(/'  7.  ENTER TIME TO END THE SIMULATION IN ',
   1  'SECONDS-',/,'       (ONE REAL VALUE): ',$)
      ACCEPT 204,TIMEND
      IF(ICONT.GT.0) GO TO 300
  8   TYPE 108
108   FORMAT(/'  8.  ENTER THE MAXIMUM DELAY TIME WHICH CAN BE '
   2  ,'REPRESENTED'/6X'IN THE DELAY-TIME GRAPH, IN SECONDS-'
   1  ,/, '       (ONE REAL VALUE): ',$)
      ACCEPT 204, DLYMAX
      IF(ICONT.GT.0) GO TO 300
  9   TYPE 109
109   FORMAT(/'  9.  ENTER THE TIME IN SECONDS WHEN THE NEW',
   1  ' MEAN ARRIVAL RATE OCCURS'/
   2  6X'(IF THE MEAN ARRIVAL RATE DOES NOT CHANGE, ENTER 99999.)-'/
   3  '       (ONE REAL VALUE): ',$)
      ACCEPT 204, PRTIM
      IF(ICONT.GT.0) GO TO 300
 10   TYPE 110
110   FORMAT(/' 10.  ENTER THE NEW MEAN ARRIVAL RATE',
   1  ' IN SECONDS'/6X'(IF THE MEAN ARRIVAL RATE DOES NOT CHANGE, '
   2  ,'PRESS RETURN)-'/,'       (ONE REAL VALUE): ',$)
      ACCEPT 204,ALT
```

```
      IF(ICONT.GT.0) GO TO 300
  11  TYPE 111
 111  FORMAT(/' 11.  ENTER THE ARRIVAL-SPLIT PROBABILITIES-',
     1 /,'       (THREE REAL VALUES): ',$)
      ACCEPT 202,PROB
      IF(ICONT.GT.0) GO TO 300
  12  TYPE 112
 112  FORMAT(/' 12.  ENTER THE RANDOM NUMBER SEED-',
     A /6X'(USE ''99999'' TO INDICATE LAST SEEDS FROM PRIOR RUN)',
     1 /,'       (ONE INTEGER VALUE): ',$)
      ACCEPT 203, ISEED
      IF(ICONT.GT.0) GO TO 300
      TYPE 401
401   FORMAT(//7X'DO YOU WISH TO INCLUDE COSTS IN THIS RUN?'
     1 /7X'(''Y'' OR ''N''): '$)
132   ACCEPT 302,ANSR
      ICOST=0
      IF(ANSR.EQ.NO)GO TO 667
      IF(ANSR.EQ.YES)GO TO 131
      ICONT1=ICONT1+1
      IF(ICONT1.GT.3)STOP 2
      TYPE 303,ANSR
      GO TO 132
131   ICOST=1
13    TYPE 113
113   FORMAT(/' 13.  ENTER THE CAPITAL COST PER UNIT'
     1 /7X'(THREE INTEGER VALUES): ',$)
      ACCEPT 202,CUNIT
      IF(ICONT.GT.0)GO TO 300
14    TYPE 114
114   FORMAT(/' 14.  ENTER THE USEFUL LIFE OF THE UNIT'
     1 /7X'(THREE INTEGER VALUES): ',$)
      ACCEPT 201,LIFE
      IF(ICONT.GT.0)GO TO 300
15    TYPE 115
115   FORMAT(/'.15.  ENTER THE DISCOUNT RATE, IN PERCENTAGE TERMS'
     1 /7X'(ONE REAL VALUE): ',$)
      ACCEPT 204,DISCRT
      IF(ICONT.GT.0)GO TO 300
16    TYPE 116
116   FORMAT(/' 16.  ENTER THE SPARES RATIO, IN PERCENTAGE TERMS'
     1 /7X'(THREE REAL VALUES:) ',$)
      ACCEPT 202,SPARES
      IF(ICONT.GT.0)GO TO 300
17    TYPE 117
117   FORMAT(/' 17.  ENTER THE ANNUAL OPERATING COST PER UNIT'
     1 /7X'(THREE REAL VALUES:) ',$)
      ACCEPT 202,OPER
      IF(ICONT.GT.0)GO TO 300
18    TYPE 118
118   FORMAT(/' 18.  ENTER THE ANNUAL SCHEDULED MAINTENANCE HOURS '
     1 ,'PER UNIT'/7X'(THREE REAL VALUES): ',$)
      ACCEPT 202,HRSMNT
      IF(ICONT.GT.0)GO TO 300
19    TYPE 119
119   FORMAT(/' 19.  ENTER THE REPAIR WAGE RATE'
     1 /7X'(THREE REAL VALUES): ',$)
      ACCEPT 202,WRATE
      IF(ICONT.GT.0)GO TO 300
20    TYPE 120
120   FORMAT(/' 20.  ENTER ANNUAL PASSENGER VOLUME AT STATION'
```

```
     1  /7X'(ONE INTEGER VALUE): ',$)
        ACCEPT 203,PASSYR
        IF(ICONT.GT.0)GO TO 300
        GO TO 667
21      ICONT3=0
        TYPE 22
22      FORMAT(/6X'ARE ANY OF THE GRAPHS TO BE ALSO PRINTED',
     1  ' DIRECTLY ON THE TERMINAL-'/6X'(ZERO FOR ''NO''',
     2  ' OR THE GRAPH NUMBER (1,2,3,4 OR 5) FOR ''YES''): '$)
25      ACCEPT 403,PRINT
403     FORMAT (I)
        IF(PRINT.EQ.0.OR.PRINT.EQ.4.OR.PRINT.EQ.5.OR.PRINT.EQ.6
     1  .OR.PRINT.EQ.7.OR.PRINT.EQ.8) GO TO 24
        ICONT3=ICONT3+1
        IF(ICONT3.GT.3) STOP 12
        TYPE 410
410     FORMAT(6X'THE PREVIOUS RESPONSE WAS ',I5,
     1  /6X'PLEASE ANSWER WITH ONE INTEGER VALUE',
     2  'EITHER 0, 1, 2, 3, 4, 5: '$)
        GO TO 25
24      TYPE 294
294     FORMAT(6X'THE MODEL PRODUCES OUTPUT FILES SHOWING MEAN '
     1  ,'QUEUE LENGTH AND DELAY,'/6X'AND QUEUE LENGTH '
     2  ,'DISTRIBUTION (AND COSTS IF INCLUDED).'
     3  /6X'SHOULD THESE BE DISPLAYED ON THE TERMINAL AS WELL?'
     4  /6X'(''Y'' OR ''N''): '$)
242     ACCEPT 302,ANSR
        IPRINT=-1
        IF(ANSR.EQ.YES)IPRINT=1
        IF(ANSR.EQ.NO)IPRINT=0
        IF(IPRINT.GE.0)GO TO 241
        ICONT=ICONT+1
        IF(ICONT.GT.3)STOP 77
        TYPE 303,ANSR
        GO TO 242
241     ICONT3=0
        IF(ICONT.GT.0)GO TO 300
201     FORMAT(3I)
202     FORMAT(3F)
203     FORMAT(I)
204     FORMAT(F)
667     TYPE 671
671     FORMAT(6X'WOULD YOU LIKE TO SEE THE CURRENT INPUT FILE?'/
     1  6X'(''Y'' OR ''N''): '$)
        ICONT3=0
672     ACCEPT 302,ANSR
        IF(ANSR.EQ.NO)GO TO 674
        IF(ANSR.EQ.YES)GO TO 673
        ICONT3=ICONT3+1
        IF(ICONT3.GT.3)STOP 16
        TYPE 303,ANSR
        GO TO 672
673     TYPE 666, NM, ARRATE,NARR,SRVBAR,MCBF,RPRBAR,TIMEND,
     1  DLYMAX,PRTIM,ALT,PROB,ISEED
666     FORMAT(/,'   1.  NUMBER OF MACHINES IS ', 3I8,
     1  /,'   2.  MEAN ARRIVAL RATE IS ', F10.2,
     2  /,'   3.  NUMBER OF ARRIVALS IN GROUPS IS ', I8,
     3  /,'   4.  MEAN SERVICE TIME IS ',3F10.2,
     4  /,'   5.  MEAN CYCLES BETWEEN FAILURES IS ',3I8,
     5  /,'   6.  MEAN TIME TO REPAIR IS ',3F10.2,
     6  /,'   7.  TIME TO END THE SIMULATION ',F10.2,
```

```
     7  /,'   8.  MAXIMUM DELAY TIME IS ',F10.2,
     8  /,'   9.  TIME AT WHICH ARRIVAL RATE CHANGES ',F15.2,
     9  /,'  10.  NEW MEAN ARRIVAL RATE ',F10.2,
     A  /,'  11.  ARRIVAL PROBABILITIES ARE ',3F10.2,
     B  /,'  12.  RANDOM NUMBER SEED IS ',I11/)
        IF(ICOST.EQ.0)GO TO 669
        TYPE 670,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR
670     FORMAT(/'  13.  CAPITAL COST PER UNIT IS ',3F10.2
     1  /'  14.  USEFUL LIFE IS ',3I8
     2  /'  15.  DISCOUNT RATE IS ',F10.2,'%'
     3  /'  16.  SPARES RATIO IS ',3(F10.2,'%')
     4  /'  17.  ANNUAL UNIT OPERATING COST IS ',3F10.2
     5  /'  18.  ANNUAL HOURS OF SCHEDULED MAINTENANCE IS ',3F10.2
     6  /'  19.  REPAIR WAGE RATE IS ',3F10.2
     7  /'  20.  ANNUAL STATION PASSENGER VOLUME IS ',I9)
669     IGRAPH=PRINT-3
        IF(IPRINT.EQ.0)TYPE 664
        IF(IPRINT.EQ.1)TYPE 665
664     FORMAT(/6X'THE OUTPUT FILES WILL NOT BE DISPLAYED ON THE '
     1  ,'TERMINAL')
665     FORMAT(/6X'THE OUTPUT FILES WILL BE DISPLAYED ON THE TERMINAL')
        TYPE 668, IGRAPH
668     FORMAT(6X,'THE GRAPH DISPLAYED ON THE ',
     1  'TERMINAL IS (''0'' IF NONE): ',I5)
        IF(CHANGE.EQ.1)CHANGE=0
674     IF(CHANGE.EQ.1)GO TO 888
        ICONT=1
        ICONT1=0
300     TYPE 301
301     FORMAT(//'        DO YOU WISH TO MAKE ANY CHANGES ?',
     1  /,'        (''Y'' OR ''N''): '$)
309     ACCEPT 302, ANSR
302     FORMAT(A1)
        IF(ANSR.EQ.NO.AND.CHANGE.EQ.1)GO TO 667
        IF(ANSR.EQ.NO) GO TO 888
        IF(ANSR.EQ.YES) GO TO 306
        ICONT1=ICONT1+1
        IF(ICONT1.GT.3) STOP 2
        TYPE 303, ANSR
303     FORMAT(/,'        THE PREVIOUS RESPONSE WAS ''',
     1  A1,'''.  PLEASE ANSWER ''Y'' OR ''N'': '$)
        GO TO 309
306     ICONT2=0
        TYPE 304
304     FORMAT(6X'ENTER THE NUMBER OF THE QUESTION YOU WISH '
     1  ,'TO CHANGE')
        IF(CHANGE.EQ.0)TYPE 311
311     FORMAT(6X'(USE ''21'' TO CHANGE REPORTS DISPLAYED ON '
     1  ,'TERMINAL)-')
        TYPE 312
312     FORMAT('        (ONE INTEGER VALUE): '$)
        CHANGE=1
308     ACCEPT 203,I
        IF(I.LT.1.OR.I.GT.21) GO TO 305
        IF(.NOT.((I.GE.13 .AND. I.LT.21) .AND. ICOST.EQ.0))GO TO 678
        CHANGE=0
        TYPE 675
675     FORMAT(' COSTS ARE PRESENTLY NOT INCLUDED IN THIS MODEL.'
     1  ,'  DO YOU WISH TO ADD COST DATA?'/7X'(''Y'' OR ''N''):',$)
        ICONT3=0
676     ACCEPT 302,ANSR
```

```
          IF(ANSR.EQ.NO)GO TO 300
          IF(ANSR.EQ.YES)GO TO 677
          ICONT3=ICONT3+1
          IF(ICONT3.GT.3)STOP 15
          TYPE 303,ANSR
          GO TO 676
677       ICONT=0
          ICOST=1
          CHANGE=0
          GO TO 13
678       GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21),I
  305     ICONT2=ICONT2+1
          IF(ICONT2.GT.3) STOP 3
          TYPE 310
  310     FORMAT('       THE PREVIOUS RESPONSE WAS ',I5,
     1    /,'      PLEASE ANSWER WITH ONE INTEGER VALUE',
     2    'FROM 1 TO 12: '$)
          GO TO 308
888       WRITE(9,666)NM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR
     1    ,TIMEND,DLYMAX,PRTIM,ALT,PROB,ISEED
          IF( ICOST.EQ.1)WRITE(9,670)CUNIT,LIFE,DISCRT,SPARES,OPER
     1    ,HRSMNT,WRATE,PASSYR
          RETURN
          END
          SUBROUTINE ENDING
          COMMON/S4/I5,IMULT(1000),IQUE(1000),IUNIT(1000),LOC2(1000)
          COMMON/S5/NM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,TIMEND,DLYMAX
     1    ,PRTIM,ALT,PROB,ISEED,LAST,ICOST,IPRINT
     2    ,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR
          DIMENSION NM(3),SRVBAR(3),MCBF(3),RPRBAR(3),PROB(3),CUNIT(3)
     1    ,LIFE(3),OPER(3),HRSMNT(3),WRATE(3)
          DOUBLE PRECISION IFNX
          INTEGER ANSR,NO,YES,QUE,QL,PRINT,PUNIT
          INTEGER LSTAR(51),STAR,BLANK,QSTAR,QLSTAR(101), UNIT
          DATA YES,NO/1HY,1HN/
          DATA STAR/'*'/, BLANK/' '/, QSTAR/'#'/
          INP2=0
C-----------------------------------------------------------------------
C
C   WHEN THE PROGRAM ENDS, THE STATEMENT 'STOP' APPEARS AT THE
C   END OF THE TERMINAL PRINTOUT.  IF THE PROGRAM ENDS
C   NORMALLY, THE 'STOP' WILL BE UNACCOMPANIED BY ANY
C   NUMBER.  HOWEVER, AFTER SOME TYPES OF ABNORMAL ENDINGS, THE 'STOP'
C   STATEMENTS WILL HAVE AN OCTAL NUMBER AFTER THEM.
C   THESE NUMBERS ARE AS FOLLOWS:
C
C      (STOP     - NORMAL END-OF-JOB)
C
C       STOP   1 - MORE THAN THREE BAD RESPONSES TO THE QUESTION
C                  REGARDING ACCEPTING INPUT FROM THE TERMINAL
C
C       STOP   2 - MORE THAN THREE BAD RESPONSES TO THE QUESTION
C                  REGARDING MAKING CHANGES TO THE INPUT DATA
C
C       STOP   3 - MORE THAN THREE BAD RESPONSES TO THE QUESTION
C                  REGARDING THE NUMBER OF THE QUESTION TO BE
C                  CHANGED
C
C       STOP   4 - MORE THAN THREE BAD RESPONSES TO QUESTION
C                  REGARDING NEXT RECORD OF CURRENT DISK
C                  DATA FILE
```

```
C
C      STOP  5 - MORE THAN THREE BAD RESPONSES TO THE QUESTION
C                REGARDING MAKING ANOTHER RUN
C
C      STOP  6 - THE ARRAY 'BRK' CONTAINING 2500 RANDOM
C                NUMBERS IS BEING EXCEEDED
C
C      STOP  7 - THE ARRAY 'RPR' CONTAINING 90 RANDOM NUMBERS
C                IS BEING EXCEEDED
C
C      STOP 10 - THE NUMBER OF MACHINES FOR A SERVICE UNIT
C                (NM(UNIT)) HAS GONE NEGATIVE
C
C      STOP 11 - MORE THAN THREE BAD RESPONSES TO THE QUESTION
C                REGARDING DELETION OF DISK STORAGE FILES
C
C      STOP 12 - MORE THAN THREE BAD RESPONSES TO THE QUESTION
C                REGARDING CHOICE OF OUTPUT GRAPHING ROUTINES
C
C      STOP 13 - MORE THAN THREE BAD RESPONSES TO THE QUESTION
C                REGARDING ADDITIONAL PRINTOUT ON THE TERMINAL
C
C      STOP 14 - MORE THAN THREE BAD RESPONSES TO THE QUESTION
C                REGARDING PRINT NUMBER REPORT AND/OR UNIT NUMBER
C
C-----------------------------------------------------------------------
      LAST=0
      DO 2 I=1,51
      LSTAR(I)=BLANK
    2 CONTINUE
      DO 3 I=1,101
      QLSTAR(I)=BLANK
    3 CONTINUE
      ICONT=1
      ICONTO=0
      GO TO 33
302   FORMAT(1H1,' PASSNGR',' AT   ARRIVAL  SERVICE',3X'DELAY',
    1 /,9X,'UNIT',4X,'TIME',5X,'TIME',4X'TIME'/)
301   FORMAT(1XI5,2X,I4,2X,3F8.1,2X51A1)
480   FORMAT(1H1,' TIME',2X'QUEUE',4X,'AT',/7X'LENGTH  UNIT'//)
520   FORMAT(1XI4,2(3X,I4),3X101A1)
202   FORMAT(1H1,2X'TIME','  EVENT',6X'NEXT',7X'NEXT',5X'NEXT'
    1 ,4X'QUEUE',3X'MACHINES    AT'/19X'ARRIVAL',2X'DEPARTURE'
    2 ,2X'REPAIR',2X'LENGTH',2X,'IN SERVICE  UNIT'/)
201   FORMAT(1XF6.1,2XA5,3XF7.1,4XF7.1,2XF7.1,4XI4,5XI4,5X,I4)
400   FORMAT(1H1,' PASS  UNIT   ','ARRIVAL',2X'QUEUE'/15X'TIME',
    1 4X,'LENGTH'/)
401   FORMAT(1XI4,1X,I4,3XF7.1,3XI4,3X101A1)
C
C  SEE IF ANY OTHER REPORTS ARE TO BE PRINTED ON THE TERMINAL
C
   33 ICONT1=0
      IF(ICONT.EQ.0)TYPE 2000,PUNIT
2000  FORMAT(3X'UNIT ',I1,' INACTIVE OR NON-EXISTENT')
      TYPE 10
   10 FORMAT(/,'       DO YOU WISH TO SEE ANY OTHER GRAPHS-',/,
    1 6X,'(ANSWER ''Y'' OR ''N''): ',$)
   12 ACCEPT 1,ANSR
1     FORMAT(A1)
      IF(ANSR.EQ.YES) GO TO 19
      IF(ANSR.EQ.NO) GO TO 9
```

```
      ICONT1=ICONT1+1
      IF(ICONT1.GT.3) STOP 13
      TYPE 308,ANSR
      GO TO 12
   19 ICONT2=0
      IF(ICONT0.EQ.0)TYPE 901
901   FORMAT(' GRAPHS CAN BE DISPLAYED ON THE TERMINAL IN '
    1 ,'THEIR ENTIRETY'/' OR IN A SHORT FORM CONSISTING OF EVERY'
    2 ,' TENTH ENTRY.  TO REQUEST'/' THE SHORT FORM, ADD 10 TO THE'
    3 ,' GRAPH NUMBER (E.G., FOR THE SHORT FORM'/' OF GRAPH 4,'
    4 ,' TYPE ''14'')'//
    5 ' TO STOP DISPLAY OF ANY PRINTOUT, TYPE CONTROL-O'/
    6 ' (''O'', NOT ''ZERO''), THEN RESPOND ''Y'' OR ''N'' TO THE'/
    7 ,' QUESTION ''DO YOU WISH TO SEE ANY FURTHER PRINTOUTS?'''/
    8 ' (THIS QUESTION WILL NOT BE DISPLAYED ON THE TERMINAL)')
      ICONT0=1
      ICONT=0
      TYPE 20
   20  FORMAT(/,'       ENTER THE GRAPH NUMBER (1,2,3,4 OR 5)'
    1  ,/,'       AND THE UNIT NUMBER (0=ALL,1,2 OR 3)-',/,
    2  '       (TWO INTEGER VALUES): ',$)
C
C  NOTE THAT THE VARIABLE 'PRINT' DOESN'T GET TRANSFERRED BACK TO
C  THE MAIN ROUTINE SO THERE IS NO WORRY ABOUT LOSING ORIGINAL
C  VALUES
C
  255 ACCEPT 21,PRINT,PUNIT
   21 FORMAT(2I)
      IGRAPH=PRINT
      PRINT=PRINT+3
      IF(((PRINT.GE.4 .AND. PRINT.LE.8) .OR.
    A   (PRINT.GE.14 .AND. PRINT.LE.18)) .AND.
    1 (PUNIT.EQ.0.OR.PUNIT.EQ.1.OR.PUNIT.EQ.2.OR.PUNIT.EQ.3))
    2 GO TO 23
      ICONT2=ICONT2+1
      IF(ICONT2.GT.3) STOP 14
      TYPE 29,IGRAPH,PUNIT
   29 FORMAT(/,'       THE PREVIOUS RESPONSE FOR PRINT WAS ',I3,
    1  ' AND UNIT WAS ',I3,/,'       PLEASE ANSWER 1,2,3,4',
    2  ' OR 5 FOR PRINT (OR 11,12,13,14, OR 15 FOR THE '
    3 ,'SHORT-FORM PRINTOUTS) '/7X'AND 0,1,2,3 FOR UNIT: ',$)
      GO TO 255
23    ISHORT=0
      IF(PRINT.GT.10)ISHORT=1
      IF(PRINT.GT.10)PRINT=PRINT-10
      PRINT=PRINT-3
      GO TO (24,25,26,27,28) PRINT
C
C  REPORT 1 ON THE TERMINAL
C
24    LD=4+10*PUNIT
      REWIND LD
      READ(LD,302)
 1302 FORMAT(1H1,' PASS AT  ARRIVAL SERVICE  DELAY ',/,
    1 '       UNIT  TIME     TIME    TIME ')
      K=-1
  342 READ(LD,301,END=33)NPASS,UNIT,TIME,SERVTM,DELAY,LSTAR
      IF(ICONT.EQ.0)TYPE 1302
      ICONT=1
      IF(ISHORT.EQ.0)GO TO 241
      K=K+1
```

```
      IF(K/10*10.NE.K)GO TO 342
241   DO 34 I=1,38
      IF(LSTAR(I).EQ.STAR)GO TO 341
34    CONTINUE
      I=38
      LSTAR(38)=STAR
341   TYPE 1301, NPASS,UNIT,TIME,SERVTM,DELAY,(LSTAR(IJ),IJ=1,I)
 1301 FORMAT(1XI5,1XI2,3F8.1,1X,38A1)
      GO TO 342
C
C  REPORT 2 ON THE TERMINAL
C
25    IF(I5.EQ.0)GO TO 33
      K=-1
      DO 30 I=1,I5
      IF(.NOT.(PUNIT.EQ.0 .OR. PUNIT.EQ.IUNIT(I)))GO TO 30
      IF(ICONT.EQ.0)TYPE 1480
      ICONT=1
      IF(ISHORT.EQ.0)GO TO 251
      K=K+1
      IF(K/10*10.NE.K)GO TO 30
251   IJDEX=MIN0(LOC2(I),55)
      QLSTAR(IJDEX)=QSTAR
      TYPE 1520,IMULT(I),IQUE(I),IUNIT(I),(QLSTAR(II),II=1,IJDEX)
      QLSTAR(IJDEX)=BLANK
30    CONTINUE
      GO TO 33
 1480 FORMAT(1H1,'  TIME QUEUE AT',/,7X,'LENGTH UNIT'/)
 1520 FORMAT(1X,I5,1X,I4,2X,I2,2X,55A1)
C
C  REPORT 3 ON THE TERMINAL
C
   26 REWIND 6
      TYPE 903
903   FORMAT(' WARNING -- GRAPH 3 CAN BE VERY LONG.  TO STOP'
     1,' DISPLAY, TYPE CONTROL-O')
      READ(6,202)
      K=-1
   31 READ(6,201,END=33) TIME,LABEL,AR,DP,RP,QL,NMACH,UNIT
      IF(.NOT.(PUNIT.EQ.0 .OR. PUNIT.EQ.UNIT))GO TO 31
      IF(ICONT.EQ.0)TYPE 202
      ICONT=1
      IF(ISHORT.EQ.0)GO TO 261
      K=K+1
      IF(K/10*10.NE.K)GO TO 31
261   TYPE 201,TIME,LABEL,AR,DP,RP,QL,NMACH,UNIT
      GO TO 31
C
C  REPORT 4 ON THE TERMINAL
C
27    LD=7+10*PUNIT
      REWIND LD
      K=-1
      READ(LD,400)
32    READ(LD,401,END=33)NP2,UNIT,TMIN,QL,QLSTAR
      IF(ICONT.EQ.0)TYPE 1400
      ICONT=1
      IF(ISHORT.EQ.0)GO TO 271
      K=K+1
      IF(K/10*10.NE.K)GO TO 32
271   DO 37 I=1,49
```

```
      IF(QLSTAR(I).EQ.QSTAR)GO TO 371
37    CONTINUE
      I=49
      QLSTAR(49)=QSTAR
371   TYPE 1401,NP2,UNIT,TMIN,QL,(QLSTAR(IJ),IJ=1,I)
      GO TO 32
C
C  REPORT 5 ON THE TERMINAL
C
28    LD=8+10*PUNIT
      REWIND LD
      K=-1
      READ(LD,490)
490   FORMAT('  DEPARTURE',13X'INITIAL',3X'TOTAL'/
     1 ,3X'TIME',5X'PASSENGER',4X'UNIT',5X'DELAY'/)
291   READ(LD,309,END=33)TMIN,NPS,JFUNIT,TDELAY,LSTAR
309   FORMAT(1XF6.1,5XI5,9XI2,5XF7.1,2X51A1)
      IF(ICONT.EQ.0)TYPE 490
      ICONT=1
      IF(ISHORT.EQ.0)GO TO 281
      K=K+1
      IF(K/10*10.NE.K)GO TO 291
281   DO 292 I=1,30
      IF(LSTAR(I).EQ.STAR)GO TO 293
292   CONTINUE
      I=30
      LSTAR(30)=STAR
293   TYPE 1308,TMIN,NPS,JFUNIT,TDELAY,(LSTAR(IJ),IJ=1,I)
1308  FORMAT(1XF6.1,5XI5,9XI2,5XF7.1,2X30A1)
      GO TO 291
 1400 FORMAT(1H1,' PASS ON ARRIVAL QUEUE',/7X'IT',3X,'TIME  LENGTH'/)
 1401 FORMAT(1X,I5,1X,I2,F7.1,1X,I4,1X,49A1)
    9 ICONT2=0
      TYPE 4
4     FORMAT(/'       DO YOU WISH TO MAKE ANOTHER RUN?'/
     1 7X'(THIS WOULD DELETE ALL REMAINING CURRENT OUTPUT REPORTS)'/
     2 6X'(ANSWER ''Y'' OR ''N''): '$)
      ISURE=IPRINT
   98 ACCEPT 1,ANSR
      IF(ANSR.EQ.YES) GO TO 5
      IF(ANSR.EQ.NO) GO TO 88
      ICONT2=ICONT2+1
      IF(ICONT2.GT.3) STOP 11
      TYPE 308,ANSR
308   FORMAT(/,'       THE PREVIOUS RESPONSE WAS ''',
     1 A1,'''.  PLEASE ANSWER ''Y'' OR ''N'': '$)
      GO TO 98
5     IF(ISURE.EQ.1)GO TO 51
      TYPE 294
294   FORMAT(6X'IF YOU MAKE ANOTHER RUN, YOU WILL HAVE NO '
     1 ,'QUEUE LENGTH OR '/6X'COST OUTPUTS FROM '
     2 ,'THIS RUN (APART FROM THE GRAPHS).'/6X'DO YOU STILL '
     3 ,'WISH TO MAKE ANOTHER RUN?'/7X'(''Y'' OR ''N''): '$)
      ISURE=1
      GO TO 98
51    CLOSE(UNIT=4,DISPOSE='DELETE')
      CLOSE(UNIT=6,DISPOSE='DELETE')
      CLOSE(UNIT=7,DISPOSE='DELETE')
      CLOSE(UNIT=8,DISPOSE='DELETE')
      CLOSE(UNIT=9,DISPOSE='DELETE')
      CLOSE(UNIT=14,DISPOSE='DELETE')
```

```
      CLOSE(UNIT=17,DISPOSE='DELETE')
      CLOSE(UNIT=18,DISPOSE='DELETE')
      CLOSE(UNIT=24,DISPOSE='DELETE')
      CLOSE(UNIT=27,DISPOSE='DELETE')
      CLOSE(UNIT=28,DISPOSE='DELETE')
      CLOSE(UNIT=34,DISPOSE='DELETE')
      CLOSE(UNIT=37,DISPOSE='DELETE')
      CLOSE(UNIT=38,DISPOSE='DELETE')
      RETURN
 88   LAST=1
      CLOSE(UNIT=4)
      CLOSE(UNIT=6)
      CLOSE(UNIT=7)
      CLOSE(UNIT=8)
      CLOSE(UNIT=9)
      CLOSE(UNIT=14)
      CLOSE(UNIT=17)
      CLOSE(UNIT=18)
      CLOSE(UNIT=24)
      CLOSE(UNIT=27)
      CLOSE(UNIT=28)
      CLOSE(UNIT=34)
      CLOSE(UNIT=37)
      CLOSE(UNIT=38)
      ICONT1=0
      IF(I5.EQ.0)GO TO 40
      OPEN(UNIT=5,FILE='FOR05.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=15,FILE='U15.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=25,FILE='U25.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=35,FILE='U35.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      DO 90 LD=5,35,10
 90   WRITE(LD,480)
      DO 91 I=1,I5
      LD=5+10*IUNIT(I)
      QLSTAR(LOC2(I))=STAR
      WRITE(LD,520)IMULT(I),IQUE(I),IUNIT(I),
     1 (QLSTAR(II),II=1,LOC2(I))
      WRITE(5,520)IMULT(I),IQUE(I),IUNIT(I),
     1 (QLSTAR(II),II=1,LOC2(I))
      QLSTAR(LOC2(I))=BLANK
 91   CONTINUE
      CLOSE(UNIT=5)
      CLOSE(UNIT=15)
      CLOSE(UNIT=25)
      CLOSE(UNIT=35)
 40   ICONT1=0
      TYPE 940
 940  FORMAT(6X'DO YOU WISH TO SAVE THE CURRENT INPUT DATA?'/
     1 6X'(ANSWER ''Y'' OR ''N''): ',$)
 41   ACCEPT 1,ANSR
      IF(ANSR.EQ.NO)RETURN
      IF(ANSR.EQ.YES)GO TO 42
      ICONT1=ICONT1+1
      IF(ICONT1.GT.3)STOP 15
      TYPE 308,ANSR
      GO TO 41
 42   TYPE 942
 942  FORMAT(6X'ENTER FILE NAME (UP TO 10 CHARACTERS): ',$)
      ACCEPT 943,IFNX
 943  FORMAT(A10)
      OPEN(UNIT=56,FILE=IFNX,DEVICE='DSKS',ACCESS='SEQINOUT')
```

```
      WRITE(56,944)NM,ARRATE,NARR,SRVBAR,MCBF,RPRBAR,TIMEND
     1 ,DLYMAX,PRTIM,ALT,PROB,ISEED
944   FORMAT(1X3I3/1XF6.2/1XI3/1X3F8.2/1X3I6/1X3F8.2/1XF8.1
     1 /1XF8.1/1XF8.1/1XF8.2/1X3F6.2/1XI11)
      IF(ICOST.EQ.1)WRITE(56,945)CUNIT,LIFE,DISCRT,SPARES,OPER
     1 ,HRSMNT,WRATE,PASSYR
945   FORMAT(1X3F9.1/1X3I4/1XF4.1/1X3F5.2/1X3F9.1/1X3F5.1
     1 /1X3F6.2/1XI12)
      CLOSE(UNIT=56)
      RETURN
      END
      SUBROUTINE CSTMOD(CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE
     1 ,PASSYR,NM,MCBF,RPRBAR,PROB,COST)
      DIMENSION COST(3,6),CUNIT(3),LIFE(3),SPARES(3),OPER(3),HRSMNT(3)
     1 ,WRATE(3),NM(3),MCBF(3),RPRBAR(3),PROB(3)
      INTEGER PASSYR
      RATE=DISCRT/100.
      PFRAC=0.
      DO 1 I=1,3
      PFRAC=PFRAC+PROB(I)
      IF(LIFE(I).EQ.0)GO TO 1
      SPRPCT=SPARES(I)/100.
      CAPINV=NM(I)*CUNIT(I)
      SPRINV=CAPINV*SPRPCT
      PV=1.
      IF(RATE.LT..01)GO TO 2
      PV=RATE/(1.-(1./(1.+RATE)**LIFE(I)))
2     COST(I,1)=CAPINV*PV
      COST(I,2)=SPRINV*PV
      COST(I,3)=OPER(I)*NM(I)
      COST(I,4)=HRSMNT(I)*WRATE(I)*NM(I)
      COST(I,5)=WRATE(I)*RPRBAR(I)/3600.*(PASSYR*PFRAC/MCBF(I))
      COST(I,6)=0.
      DO 3 J=1,5
3     COST(I,6)=COST(I,6)+COST(I,J)
1     CONTINUE
      RETURN
      END
```

A.2 LISTINGS FOR ANALYTICAL MODEL

Main Program
MAIN.

Subroutines

QMATRX
RMATRX
CMTRX1
CMTRX2
PIVECT
MATMLT
VECMLT
BEGIN
INPUT
ENDING
CSTMOD
MINV
GELG

```
      DIMENSION Q(0:2,0:2),R(0:2,0:2),LAMDA(0:2),PIX(0:2),XX(0:500)
     1 ,TEMP(0:2,0:2),ZERO(0:2,0:2),C1(0:2,0:2),C2(0:2,0:2),SUM(0:2)
     2 ,TEMP1(0:2,0:2),R1INV(0:2,0:2),LWV(3),MWV(3),A(0:2),X(0:500,0:2)
     3 ,U1(0:2),U2(0:2),PI(0:2),XC(0:500,0:2),QBARJ(0:5)
      DIMENSION CC1(0:2,0:2),CC2(0:2,0:2),F1(0:2),F2(0:2),H(0:2)
     1 ,RMU(0:2),R2INV(0:2,0:2),TEMPA(0:2),TEMPB(0:2),TEMPC(0:2)
     2 ,TEMPZ(0:2,0:2),UU(0:2,0:2),UU1(0:2,0:2),UU1MU(0:2)
     3 ,Y(0:2,0:2),Y1(0:2,0:2)
      DIMENSION COST(6)
      COMMON IFILNM,FROMWH,N,ARR,SERVTM,MCBF,MTTR
     1 ,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR,ICOST
      DOUBLE PRECISION IFILNM
      REAL LAMDA,MCBF,MTTR
      INTEGER C,FROMWH,PASSYR
      DATA TOL,XTOL/.00001,.0005/
      DATA LIMIT,MAXIT/500,50000/
      FROMWH=0
      CALL BEGIN
201   FORMAT(1XI3,4F6.0)
202   FORMAT(1X2F9.6,2I5)
34    WRITE(6,201)N,ARR,SERVTM,MCBF,MTTR
      WRITE(6,202)XTOL,TOL,LIMIT,MAXIT
      SERV=3600./SERVTM
      C=N
      SIGMA=(3600./MTTR)/SERV
      THETA=1./MCBF
      DO 1 I=0,N
1     LAMDA(I)=ARR/SERV
      WRITE(6,110)SIGMA,THETA,LAMDA
110   FORMAT(1X2F9.5,5X(F9.5))
      DO 31 IC=1,C
      JC=IC
      CALL PIVECT(N,JC,SIGMA,THETA,PI)
      RHONUM=0.
      RHODEM=0.
      DO 32 J=0,N
      RHONUM=RHONUM+PI(J)*LAMDA(J)
32    RHODEM=RHODEM+PI(J)*J
      RHO=RHONUM/RHODEM
31    WRITE(6,108)IC,RHO,(PI(J),J=0,N)
108   FORMAT(I5,F7.4,5X7F7.4)
      WRITE(4,301)N,ARR,SERV,MCBF,MTTR
301   FORMAT('1FARE COLLECTION QUEUING ANALYSIS'//
     * ' NUMBER OF UNITS = ',I2/' ARRIVAL RATE = '
     1 ,F6.0,' /HR'/' SERVICE RATE = ',F6.0,' /HR'/' MCBF = '
     2 ,F6.0/' MTTR = ',F6.0,' SEC')
      WRITE(7,302)
      DO 351 J=0,N
351   WRITE(7,303)J,PI(J)
      WRITE(7,304)RHO
304   FORMAT(//' THE TRAFFIC INTENSITY IS ',F7.4)
      IF(RHO.GE.1.0)GO TO 30
      CALL QMATRX(Q,N,C,SIGMA,THETA)
      DO 2 I=0,N
2     WRITE(6,101)(Q(I,J),J=0,N)
101   FORMAT((1X7F7.4))
      CALL RMATRX(R,LAMDA,N,TOL,MAXIT,Q,H,F1,F2,UU,UU1,UU1MU,RMU)
      WRITE(6,102)
102   FORMAT('0')
      DO 3 I=0,N
3     WRITE(6,101)(R(I,J),J=0,N)
```

```
      DO 4 I=0,N
      DO 5 J=0,N
5     TEMP(I,J)=R(I,J)*J+Q(I,J)
4     TEMP(I,I)=TEMP(I,I)-I-LAMDA(I)
      CALL MATMLT(TEMP,N,N,R,N,ZERO)
      DO 6 I=0,N
6     ZERO(I,I)=ZERO(I,I)+LAMDA(I)
      WRITE(6,102)
      DO 7 I=0,N
7     WRITE(6,101)(ZERO(I,J),J=0,N)
      CALL CMTRX1(C1,C2,N,Q,R,SUM,LAMDA,Y,Y1)
      WRITE(6,102)
      DO 26 I=0,N
26    WRITE(6,101)(C1(I,J),J=0,N)
      WRITE(6,102)
      WRITE(6,101)SUM
      DO 8 I=0,N
      C2(I,0)=0.
      DO 8 J=0,N
      TEMP(I,J)=0.
      IF(J.EQ.I .OR. J.EQ.I+1)TEMP(I,J)=Q(I,J)
      IF(J.EQ.I .AND. I.GT.0)TEMP(I,J)=TEMP(I,J)+Q(I,I-1)
8     IF(I.EQ.J)TEMP(I,J)=TEMP(I,J)-LAMDA(I)
      CALL MATMLT(C1,N,N,TEMP,N,TEMP1)
      DO 12 I=0,N
      DO 12 J=0,N
12    TEMP(I,J)=C2(I,J)+TEMP1(I,J)
      DO 9 I=0,N
      DO 10 J=0,N
10    R1INV(I,J)=-R(I,J)
9     R1INV(I,I)=R1INV(I,I)+1.
      WRITE(6,102)
      DO 27 I=0,N
27    WRITE(6,101)(R1INV(I,J),J=0,N)
      CALL MINV(R1INV,N+1,D,LWV,MWV)
      WRITE(6,102)
      DO 99 I=0,N
99    WRITE(6,999)(R1INV(I,J),J=0,N)
999   FORMAT(1X7E11.4)
      WRITE(6,102)
      DO 28 I=0,N
28    WRITE(6,101)(R1INV(I,J),J=0,N)
      DO 11 I=0,N
      TEMP(I,N)=SUM(I)
      A(I)=0.
      DO 11 J=0,N
11    TEMP(I,N)=TEMP(I,N)+R1INV(I,J)
      DO 13 I=0,N
      DO 13 J=0,N
13    TEMP1(I,J)=TEMP(J,I)
      A(N)=1.
      EPS=5.E-4
      WRITE(6,102)
103   FORMAT(1X7F9.4)
      WRITE(6,102)
      DO 98 I=0,N
98    WRITE(6,999)(TEMP(I,J),J=0,N)
      WRITE(6,102)
      DO 29 I=0,N
29    WRITE(6,103)(TEMP(I,J),J=0,N)
      CALL GELG(A,TEMP1,N+1,1,EPS,IER)
```

```
      DO 14 I=0,N
14    X(N-1,I)=A(I)
      WRITE(6,102)
      WRITE(6,101)(X(N-1,I),I=0,N)
      CALL CMTRX2(N,Q,R,R11NV,X,U1,U2,LAMDA,LIMIT,CC1,CC2,Y,Y1
     1 ,TEMPZ,TEMPA,TEMPB,TEMPC,R2INV)
      DO 15 JJ=2,N
15    WRITE(6,101)(X(N-JJ,I),I=0,N)
      WRITE(6,102)
      WRITE(6,104)(U1(I),I=0,N)
      WRITE(6,104)(U2(I),I=0,N)
104   FORMAT(1X7F9.4)
      XTOT=0.
      DO 16 J=0,N
      PIX(J)=0.
      DO 16 JJ=0,N-1
      PIX(J)=PIX(J)+X(JJ,J)
16    XTOT=XTOT+X(JJ,J)
      JJ=N-1
17    IF(1-XTOT.LT.XTOL)GO TO 19
      IF(JJ.GE.LIMIT)GO TO 18
      JJ=JJ+1
      DO 20 J=0,N
      X(JJ,J)=0.
      DO 21 I=0,N
21    X(JJ,J)=X(JJ,J)+X(JJ-1,I)*R(I,J)
      PIX(J)=PIX(J)+X(JJ,J)
20    XTOT=XTOT+X(JJ,J)
      GO TO 17
18    WRITE(7,105)LIMIT
105   FORMAT(' X DOES NOT SUM TO ONE AFTER',I5,' QUEUE LENGTH POINTS')
19    WRITE(7,305)JJ,XTOT
305   FORMAT(///1XI4,' POINTS OF THE QUEUE LENGTH DISTRIBUTION'
     1 /' HAVE BEEN COMPUTED, FOR A TOTAL PROBABILITY OF ',F7.4)
106   FORMAT(1XI4,3XF7.4)
      QLBAR=0.
      QLSTDV=0.
      DO 36 I=0,N
      QLBAR=QLBAR+U1(I)
      IF(PIX(I).GE.XTOL)QLSTDV=QLSTDV+U2(I)-(U1(I)**2/PIX(I))
36    CONTINUE
      QLSTDV=SQRT(QLSTDV)
      WRITE(4,306)QLBAR,QLSTDV
306   FORMAT(/' THE MEAN QUEUE LENGTH IS ',F7.2/
     1 5X'WITH A STANDARD DEVIATION OF ',F7.2)
      DLYBAR=QLBAR/ARR*3600.
      WRITE(4,310)DLYBAR
310   FORMAT(/' THE MEAN DELAY IS ',F7.2,' SEC.')
      WRITE(7,101)(PIX(I),I=0,N)
      WRITE(7,102)
      WRITE(4,302)
302   FORMAT(/'0PROBABILITY OF NUMBER OF UNITS AVAILABLE'/5X'NUMBER'
     1 ,' OF UNITS AVAILABLE',10X'PROBABILITY')
      DO 35 J=0,N
35    WRITE(4,303)J,PIX(J)
303   FORMAT(13XI2,25XF7.4)
      WRITE(4,102)
      RHONUM=0.
      RHODEN=0.
      DO 37 J=0,N
      RHONUM=RHONUM+PIX(J)*LAMDA(J)
```

```
      DO 14 I=0,N
14    X(N-1,I)=A(I)
      WRITE(6,102)
      WRITE(6,101)(X(N-1,I),I=0,N)
      CALL CMTRX2(N,Q,R,R11NV,X,U1,U2,LAMDA,LIMIT,CC1,CC2,Y,Y1
     1 ,TEMPZ,TEMPA,TEMPB,TEMPC,R2INV)
      DO 15 JJ=2,N
15    WRITE(6,101)(X(N-JJ,I),I=0,N)
      WRITE(6,102)
      WRITE(6,104)(U1(I),I=0,N)
      WRITE(6,104)(U2(I),I=0,N)
104   FORMAT(1X7F9.4)
      XTOT=0.
      DO 16 J=0,N
      PIX(J)=0.
      DO 16 JJ=0,N-1
      PIX(J)=PIX(J)+X(JJ,J)
16    XTOT=XTOT+X(JJ,J)
      JJ=N-1
17    IF(1-XTOT.LT.XTOL)GO TO 19
      IF(JJ.GE.LIMIT)GO TO 18
      JJ=JJ+1
      DO 20 J=0,N
      X(JJ,J)=0.
      DO 21 I=0,N
21    X(JJ,J)=X(JJ,J)+X(JJ-1,I)*R(I,J)
      PIX(J)=PIX(J)+X(JJ,J)
20    XTOT=XTOT+X(JJ,J)
      GO TO 17
18    WRITE(7,105)LIMIT
105   FORMAT(' X DOES NOT SUM TO ONE AFTER',I5,' QUEUE LENGTH POINTS')
19    WRITE(7,305)JJ,XTOT
305   FORMAT(///1XI4,' POINTS OF THE QUEUE LENGTH DISTRIBUTION'
     1 /' HAVE BEEN COMPUTED, FOR A TOTAL PROBABILITY OF ',F7.4)
106   FORMAT(1XI4,3XF7.4)
      QLBAR=0.
      QLSTDV=0.
      DO 36 I=0,N
      QLBAR=QLBAR+U1(I)
      IF(PIX(I).GE.XTOL)QLSTDV=QLSTI V+U2(I)-(U1(I)**2/PIX(I))
36    CONTINUE
      QLSTDV=SQRT(QLSTDV)
      WRITE(4,306)QLBAR,QLSTDV
306   FORMAT(/' THE MEAN QUEUE LENGTH IS ',F7.2/
     1 5X'WITH A STANDARD DEVIATION OF ',F7.2)
      DLYBAR=QLBAR/ARR*3600.
      WRITE(4,310)DLYBAR
310   FORMAT(/' THE MEAN DELAY IS ',F7.2,' SEC.')
      WRITE(7,101)(PIX(I),I=0,N)
      WRITE(7,102)
      WRITE(4,302)
302   FORMAT(/'OPROBABILITY OF NUMBER OF UNITS AVAILABLE'/5X'NUMBER'
     1 ,' OF UNITS AVAILABLE',10X'PROBABILITY')
      DO 35 J=0,N
35    WRITE(4,303)J,PIX(J)
303   FORMAT(13XI2,25XF7.4)
      WRITE(4,102)
      RHONUM=0.
      RHODEN=0.
      DO 37 J=0,N
      RHONUM=RHONUM+PIX(J)*LAMDA(J)
```

```
37        RHODEN=RHODEN+PIX(J)*J
          RHO=RHONUM/RHODEN
          WRITE(4,304)RHO
          JMAX=MIN0(9,JJ)
          WRITE(4,307)(J,J=0,JMAX)
307       FORMAT(//20X' QUEUE LENGTH -- PROBABILITY DENSITY'//4X10(5XI2))
          WRITE(4,311)
311       FORMAT(' ')
          DO 22 I=0,JJ
          XX(I)=0.
          DO 23 J=0,N
          XC(I,J)=0.
          IF(PIX(J).LT.XTOL)GO TO 23
          XC(I,J)=X(I,J)/PIX(J)
          XX(I)=XX(I)+X(I,J)
23        CONTINUE
          IF((I+1)/10*10.NE.(I+1))GO TO 22
          IM9=I-9
          WRITE(4,112)IM9,(XX(J),J=IM9,I)
112       FORMAT(3XI3,10(1X2PF5.1,'%'))
308       FORMAT(1XI5,7XF6.4)
22        WRITE(7,107)I,XX(I),(X(I,J),J=0,N)
          IP1=IM9+10
          IF(JJ-IM9.GT.9)WRITE(4,112)IP1,(XX(J),J=IP1,JJ)
107       FORMAT(1XI4,4X2PF6.2,'%',5X7(1X2PF6.2,'%'))
          WRITE(4,309)(J,J=0,JMAX)
309      FORMAT(//21X'QUEUE LENGTH -- CUMULATIVE PROBABILITY'//4X10(5XI2))
          WRITE(4,311)
          WRITE(7,102)
          NULL=0
          WRITE(7,107)NULL,XX(0),(XC(0,J),J=0,N)
          QBAR=1.-XX(0)
          DO 39 J=0,N
39        QBARJ(J)=1.-XC(0,J)
          DO 24 I=1,JJ
          DO 25 J=0,N
          XC(I,J)=XC(I,J)+XC(I-1,J)
25        QBARJ(J)=QBARJ(J)+(1.-XC(I,J))
          XX(I)=XX(I)+XX(I-1)
          QBAR=QBAR+(1.0-XX(I))
          IF((I+1)/10*10.NE.(I+1))GO TO 24
          IM9=I-9
          WRITE(4,112)IM9,(XX(J),J=IM9,I)
24        WRITE(7,107)I,XX(I),(XC(I,J),J=0,N)
          IP1=IM9+10
          IF(JJ-IM9.GT.9)WRITE(4,112)IP1,(XX(J),J=IP1,JJ)
          WRITE(7,401)QBAR,QBARJ
401       FORMAT(' THE COMPUTED MEAN QUEUE LENGTH IS',F7.2/5X7F8.2)
          GO TO 33
30        WRITE(6,109)
          WRITE(4,109)
109       FORMAT(' THE QUEUE IS NOT STABLE')
          WRITE(4,304)RHO
33        IF(ICOST.EQ.0)GO TO 38
          CALL CSTMOD(CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE
     1    ,PASSYR,N,MCBF,MTTR,COST)
          WRITE(4,111)COST
111       FORMAT(/15X'-- COSTS --'/
     1    /7X'ANNUALIZED CAPITAL COST',T35,'$',F9.2
     2    /7X'ANNUALIZED SPARES COST',T35,'$',F9.2
     3    /7X'OPERATING COST',T35,'$',F9.2
```

```
     4  /7X'SCHEDULED MAINTENANCE COST',T35,'$',F9.2
     5  /7X'CORRECTIVE REPAIR COST',T35,'$',F9.2
     6  //7X'TOTAL COST',T34,'$',F10.2)
 38     CALL ENDING
        IF(FROMWH.LT.2)GO TO 34
        STOP
        END
        SUBROUTINE QMATRX(Q,N,C,SIGMA,THETA)
        DIMENSION Q(0:N,0:N)
        INTEGER C
        DO 1 I=0,N-1
        Q(I,I+1)=MIN0(C,N-I)*SIGMA
 1      Q(I+1,I)=(I+1)*THETA
        DO 2 I=1,N-1
 2      Q(I,I)=-(Q(I,I-1)+Q(I,I+1))
        Q(0,0)=-Q(0,1)
        Q(N,N)=-Q(N,N-1)
        RETURN
        END
        SUBROUTINE RMATRX(R,LAMDA,N,TOL,MAXIT,Q,H,F1,F2,U,U1,U1MU,RMU)
        DIMENSION R(0:N,0:N),LAMDA(0:N),H(0:N),F1(0:N),F2(0:N)
     1  ,U(0:N,0:N),U1(0:N,0:N),U1MU(0:N),RMU(0:N),Q(0:N,0:N)
        REAL LAMDA
        ITS=0
        DO 16 I=0,N
        DO 16 J=0,N
 16     R(I,J)=0.
        DO 1 I=0,N
 1      H(I)=LAMDA(I)+I-Q(I,I)
        DO 2 I=1,N
        F1(I)=Q(I,I-1)/H(I-1)
 2      F2(I)=Q(I-1,I)/H(I)
 3      DO 4 I=0,N
        DO 4 J=0,N
 4      U(I,J)=R(I,J)*J/H(J)
        DO 5 I=1,N
        U(I,I-1)=U(I,I-1)+F1(I)
 5      U(I-1,I)=U(I-1,I)+F2(I)
        DO 6 I=0,N
        DO 6 J=0,N
        U1(I,J)=0.
        DO 6 K=0,N
 6      U1(I,J)=U1(I,J)+R(I,K)*U(K,J)
        DO 7 I=0,N
 7      U1(I,I)=U1(I,I)+LAMDA(I)/H(I)
        ITS=ITS+1
        ERR=0.
        DO 8 I=0,N
        DO 8 J=0,N
 8      ERR=AMAX1(ERR,ABS(U1(I,J)-R(I,J)))
        IF(ERR.LE.TOL)GO TO 11
        IF(ITS.LT.MAXIT)GO TO 9
        WRITE(6,101)MAXIT
 101    FORMAT(' NO CONVERGENCE AFTER',I4,' ITERATIONS.')
        GO TO 11
 9      DO 10 I=0,N
        DO 10 J=0,N
 10     R(I,J)=U1(I,J)
        GO TO 3
 11     WRITE(6,102)ITS
 102    FORMAT('0',10XI5)
```

```
      GO TO 14
      DO 20 I=0,N
20    WRITE(6,103)(R(I,J),J=0,N)
103   FORMAT(1X7F7.4)
      DO 12 I=0,N
      RMU(I)=0.
      U1MU(I)=0.
      DO 12 J=1,N
      RMU(I)=RMU(I)+R(I,J)*J
12    U1MU(I)=U1MU(I)+U1(I,J)*J
      ISSAME=0
      DO 13 I=0,N
      IF(U1MU(I).EQ.RMU(I))GO TO 13
      ISSAME=1
      TEMP=(LAMDA(I)-U1MU(I))/(RMU(I)-U1MU(I))
      DO 13 J=0,N
      R(I,J)=U1(I,J)+TEMP*(R(I,J)-U1(I,J))
13    CONTINUE
      IF(ISSAME.EQ.0)GO TO 14
      RETURN
14    DO 15 I=0,N
      DO 15 J=0,N
15    R(I,J)=U1(I,J)
      RETURN
      END
      SUBROUTINE CMTRX1(C1,C2,N,Q,R,SUM,LAMDA,Y,Y1)
      DIMENSION C1(0:N,0:N),C2(0:N,0:N),Q(0:N,0:N),R(0:N,0:N),SUM(0:N)
     1 ,Y1(0:N,0:N),Y(0:N,0:N)
      REAL LAMDA(0:N)
      DO 1 I=0,N
      SUM(I)=0.
      DO 1 J=0,N
      C2(I,J)=0.
      QIJ=Q(I,J)
      IF(I.EQ.N .AND. J.EQ.N-1)QIJ=QIJ-THETA
      IF(I.EQ.N .AND. J.EQ.N)QIJ=QIJ+THETA
      C1(I,J)=-(QIJ+R(I,J)*J)/LAMDA(J)
      IF(I.NE.J)GO TO 1
      C2(I,J)=1.
      C1(I,J)=C1(I,J)+(1.+AMINO(N-1,J)/LAMDA(J))
1     SUM(I)=SUM(I)+C1(I,J)
      IF(N.EQ.2)RETURN
      JJ=3
2     WRITE(6,101)JJ
101   FORMAT(1XI3)
      DO 6 I=0,N
6     WRITE(6,102)(C1(I,J),J=0,N)
102   FORMAT((1X6F7.4))
      DO 7 I=0,N
      DO 7 K=0,N
      Y(I,K)=0.
      DO 7 J=0,N
      QJK=Q(J,K)
      IF(J.LE.N-JJ+1)GO TO 7
      IF(J.EQ.K)QJK=QJK+(J-(N-JJ+1))*THETA
      IF(J.EQ.K+1)QJK=QJK-(J-(N-JJ+1))*THETA
7     Y(I,K)=Y(I,K)+C1(I,J)*QJK
      DO 3 I=0,N
      DO 3 J=0,N
3     Y(I,J)=Y(I,J)-C1(I,J)*(LAMDA(J)+AMINO(N-JJ+1,J))
      DO 4 I=0,N
```

```
      DO 4 J=0,N
      Y1(I,J)=(-Y(I,J)-C2(I,J)*AMINO(N-JJ+2,J))/LAMDA(J)
 4    SUM(I)=SUM(I)+Y1(I,J)
      DO 5 I=0,N
      DO 5 J=0,N
      C2(I,J)=C1(I,J)
 5    C1(I,J)=Y1(I,J)
      JJ=JJ+1
      IF(JJ.LE.N)GO TO 2
      RETURN
      END
      SUBROUTINE CMTRX2(N,Q,R,R1INV,X,U1,U2,LAMDA,LIMIT,C1,C2,Y,Y1
     1 ,TEMP1,TEMPA,TEMPB,TEMPC,R2INV)
      DIMENSION Q(0:N,0:N),R(0:N,0:N),R1INV(0:N,0:N),U1(0:N)
     1 ,U2(0:N),X(0:LIMIT,0:N),LAMDA(0:N)
      DIMENSION C1(0:N,0:N),Y(0:N,0:N),Y1(0:N,0:N)
     1 ,TEMP1(0:N,0:N),TEMPA(0:N),TEMPB(0:N),TEMPC(0:N)
      DIMENSION R2INV(0:N,0:N),C2(0:N,0:N)
      REAL LAMDA
      DO 1 I=0,N
      DO 1 J=0,N
      C2(I,J)=0.
      QIJ=Q(I,J)
      IF(I.EQ.N .AND. J.EQ.N-1)QIJ=QIJ-THETA
      IF(I.EQ.N .AND. J.EQ.N)QIJ=QIJ+THETA
      C1(I,J)=-(QIJ+R(I,J)*J)/LAMDA(J)
      IF(I.NE.J)GO TO 1
      C2(I,J)=1.
      C1(I,J)=C1(I,J)+(1.+AMINO(N-1,J)/LAMDA(J))
 1    CONTINUE
      DO 2 J=0,N
      X(N-2,J)=0.
      DO 3 I=0,N
 3    X(N-2,J)=X(N-2,J)+X(N-1,I)*C1(I,J)
      U1(J)=(N-2)*X(N-2,J)
 2    U2(J)=(N-2)*U1(J)
      IF(N.EQ.2)GO TO 9
      JJ=3
 4    DO 16 I=0,N
      DO 16 K=0,N
      Y(I,K)=0.
      DO 16 J=0,N
      QJK=Q(J,K)
      IF(J.LE.N-JJ+1)GO TO 16
      IF(J.EQ.K)QJK=QJK+(J-(N-JJ+1))*THETA
      IF(J.EQ.K+1)QJK=QJK-(J-(N-JJ+1))*THETA
 16   Y(I,K)=Y(I,K)+C1(I,J)*QJK
      DO 5 I=0,N
      DO 5 J=0,N
 5    Y(I,J)=Y(I,J)-C1(I,J)*(LAMDA(J)+AMINO(N-JJ+1,J))
      DO 6 J=0,N
      X(N-JJ,J)=0.
      DO 7 I=0,N
      Y1(I,J)=(-Y(I,J)-C2(I,J)*AMINO(N-JJ+2,J))/LAMDA(J)
 7    X(N-JJ,J)=X(N-JJ,J)+X(N-1,I)*Y1(I,J)
      U1(J)=U1(J)+(N-JJ)*X(N-JJ,J)
 6    U2(J)=U2(J)+(N-JJ)*U1(J)
      DO 8 I=0,N
      DO 8 J=0,N
      C2(I,J)=C1(I,J)
 8    C1(I,J)=Y1(I,J)
```

```
      JJ=JJ+1
      IF(JJ.LE.N)GO TO 4
9     CALL MATMLT(R1INV,N,N,R1INV,N,R2INV)
      DO 10 I=0,N
      DO 11 J=0,N
11    TEMP1(I,J)=-(N-2)*R(I,J)
      TEMP1(I,I)=TEMP1(I,I)+(N-1)
10    TEMPA(I)=X(N-1,I)
      CALL VECMLT(TEMPA,N,R2INV,N,TEMPB)
      CALL VECMLT(TEMPB,N,TEMP1,N,TEMPA)
      DO 12 I=0,N
12    U1(I)=U1(I)+TEMPA(I)
      DO 13 I=0,N
      DO 14 J=0,N
14    TEMP1(I,J)=(N-2)**2*R(I,J)
13    TEMP1(I,I)=TEMP1(I,I)-((2.*N-6.)*N+3.)
      CALL VECMLT(TEMPB,N,R1INV,N,TEMPA)
      CALL VECMLT(TEMPA,N,TEMP1,N,TEMPB)
      CALL VECMLT(TEMPB,N,R,N,TEMPC)
      DO 15 I=0,N
15    U2(I)=U2(I)+(TEMPC(I)+(N-1)**2*TEMPA(I))
      RETURN
      END
      SUBROUTINE PIVECT(N,C,SIGMA,THETA,PI)
      DIMENSION PI(0:N)
      INTEGER C
      PI(0)=1.
      SUM=1.
      CST=C*SIGMA/THETA
      DO 1 J=1,N-C+1
      PI(J)=PI(J-1)*CST/J
1     SUM=SUM+PI(J)
      IF(C.LE.1)GO TO 3
      DO 2 J=N-C+2,N
      PI(J)=PI(J-1)*CST*(1.-(J-(N-C+1))/FLOAT(C))/J
2     SUM=SUM+PI(J)
3     DO 4 J=0,N
4     PI(J)=PI(J)/SUM
      RETURN
      END
      SUBROUTINE MATMLT(A,I,J,B,K,C)
      DIMENSION A(0:I,0:J),B(0:J,0:K),C(0:I,0:K)
      DO 1 II=0,I
      DO 1 KK=0,K
      C(II,KK)=0.
      DO 1 JJ=0,J
1     C(II,KK)=C(II,KK)+A(II,JJ)*B(JJ,KK)
      RETURN
      END
      SUBROUTINE VECMLT(A,I,B,J,C)
      DIMENSION A(0:I),B(0:I,0:J),C(0:J)
      DO 1 JJ=0,J
      C(JJ)=0.
      DO 1 II=0,I
1     C(JJ)=C(JJ)+A(II)*B(II,JJ)
      RETURN
      END
      SUBROUTINE BEGIN
      COMMON IFILNM,FROMWH,NM,ARRATE,SRVBAR,MCBF,MTTR
     1,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR,ICOST
      REAL MCBF,MTTR
```

```
      INTEGER FROMWH,YES,NO,ANS,PASSYR
      DOUBLE PRECISION IFILNM
      DATA YES,NO/1HY,1HN/
      OPEN(UNIT=4,FILE='FOR04.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=6,FILE='FOR06.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      OPEN(UNIT=7,FILE='FOR07.DAT',DEVICE='DSKS',ACCESS='SEQINOUT')
      IF(FROMWH.EQ.1)GO TO 7
      ICOST=0
      TYPE 101
101   FORMAT(' ACCEPT INPUT FROM TERMINAL?'/6X'(''Y'' OR ''N'') ;'$)
1     ACCEPT 102,ANS
102   FORMAT(A1)
      IF(ANS.EQ.YES)GO TO 5
      IF(ANS.EQ.NO) GO TO 2
      TYPE 103
103   FORMAT('0TYPE EITHER Y OR N.  TRY AGAIN. '$)
      GO TO 1
2     TYPE 104
104   FORMAT(/' ENTER INPUT FILE NAME (UP TO 10 CHARACTERS): '$)
      ACCEPT 105,IFILNM
105   FORMAT(A10)
      OPEN(UNIT=55,FILE=IFILNM,DEVICE='DSKS',ACCESS='SEQIN')
      TYPE 106,IFILNM
106   FORMAT(/' THE INPUT FILE IS: ',A10)
7     READ(55,*)NM,ARRATE,SRVBAR,MCBF,MTTR
      TYPE 107
107   FORMAT(/' ARE COSTS INCLUDED IN THIS RECORD?'
     1 /6X' (''Y'' OR ''N''): '$)
3     ACCEPT 102,ANS
      IF(ANS.EQ.NO)GO TO 6
      IF(ANS.EQ.YES)GO TO 4
      TYPE 103
      GO TO 3
4     READ(55,*)CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR
      ICOST=1
      GO TO 6
5     CALL INPUT
6     FROMWH=1
      RETURN
      END
      SUBROUTINE INPUT
      COMMON IFILNM,FROMWH,NM,ARRATE,SRVBAR,MCBF,MTTR
     1 ,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR,ICOST
      INTEGER FROMWH,CHANGE,YES,NO,ANSR,PASSYR
      REAL MCBF,MTTR
      DOUBLE PRECISION IFILNM
      DATA YES,NO/1HY,1HN/
      CHANGE=0
      ICONT=0
      IF(FROMWH.GT.0) GO TO 667
    1 TYPE 101
  101 FORMAT(/'  1.  ENTER NUMBER OF MACHINES',
     1 /'       (ONE INTEGER VALUE): '$)
      ACCEPT 201, NM
      IF(ICONT.GT.0) GO TO 300
    2 TYPE 102
  102 FORMAT(/'  2.  ENTER MEAN ARRIVAL RATE PER HOUR-',/,
     1 '       (ONE REAL VALUE): '$)
      ACCEPT 202,ARRATE
      IF(ICONT.GT.0) GO TO 300
    3 TYPE 103
```

```
  103    FORMAT(/'  3.  ENTER MEAN SERVICE TIME IN SECONDS-',
     1   /,'       (ONE REAL VALUE): '$)
         ACCEPT 202, SRVBAR
         IF(ICONT.GT.0) GO TO 300
    4    TYPE 104
  104    FORMAT(/'  4.  ENTER MEAN CYCLES BETWEEN FAILURES-',
     1    /,'       (ONE REAL VALUE): ',$)
         ACCEPT 202,MCBF
         IF(ICONT.GT.0) GO TO 300
    5    TYPE 105
  105    FORMAT(/'  5.  ENTER MEAN TIME TO REPAIR IN SECONDS-',
     1   /,'       (ONE REAL VALUE): ',$)
         ACCEPT 202, MTTR
         IF(ICONT.GT.0) GO TO 300
         TYPE 401
401      FORMAT(//7X'DO YOU WISH TO INCLUDE COSTS IN THIS RUN?'
     1   /7X'(''Y'' OR ''N''): '$)
132      ACCEPT 302,ANSR
         ICOST=0
         IF(ANSR.EQ.NO)GO TO 667
         IF(ANSR.EQ.YES)GO TO 131
         ICONT1=ICONT1+1
         IF(ICONT1.GT.3)STOP 2
         TYPE 303,ANSR
         GO TO 132
131      ICOST=1
6        TYPE 106
106      FORMAT(/' 6.  ENTER THE CAPITAL COST PER UNIT'
     1   /7X'(ONE REAL VALUE): ',$)
         ACCEPT 202,CUNIT
         IF(ICONT.GT.0)GO TO 300
7        TYPE 107
107      FORMAT(/' 7.  ENTER THE USEFUL LIFE OF THE UNIT'
     1   /7X'(ONE INTEGER VALUE): ',$)
         ACCEPT 201,LIFE
         IF(ICONT.GT.0)GO TO 300
8        TYPE 108
108      FORMAT(/' 8.  ENTER THE DISCOUNT RATE, IN PERCENTAGE TERMS'
     1   /7X'(ONE REAL VALUE): ',$)
         ACCEPT 204,DISCRT
         IF(ICONT.GT.0)GO TO 300
9        TYPE 109
109      FORMAT(/' 9.  ENTER THE SPARES RATIO, IN PERCENTAGE TERMS'
     1   /7X'(ONE REAL VALUE:) ',$)
         ACCEPT 202,SPARES
         IF(ICONT.GT.0)GO TO 300
10       TYPE 110
110      FORMAT(/' 10.  ENTER THE ANNUAL OPERATING COST PER UNIT'
     1   /7X'(ONE REAL VALUE:) ',$)
         ACCEPT 202,OPER
         IF(ICONT.GT.0)GO TO 300
11       TYPE 111
111      FORMAT(/' 11.  ENTER THE ANNUAL SCHEDULED MAINTENANCE HOURS '
     1   ,'PER UNIT'/7X'(ONE REAL VALUE): ',$)
         ACCEPT 202,HRSMNT
         IF(ICONT.GT.0)GO TO 300
12       TYPE 112
112      FORMAT(/' 12.  ENTER THE REPAIR WAGE RATE'
     1   /7X'(ONE REAL VALUE): ',$)
         ACCEPT 202,WRATE
         IF(ICONT.GT.0)GO TO 300
```

```
13      TYPE 113
113     FORMAT(/' 13.   ENTER ANNUAL PASSENGER VOLUME AT STATION'
   1    /7X'(ONE INTEGER VALUE): ',$)
        ACCEPT 203,PASSYR
        IF(ICONT.GT.0)GO TO 300
        GO TO 667
 201    FORMAT(I)
 202    FORMAT(F)
 203    FORMAT(I)
 204    FORMAT(F)
 667    TYPE 671
671     FORMAT(6X'WOULD YOU LIKE TO SEE THE CURRENT INPUT FILE?'/
   1    6X'(''Y'' OR ''N''): '$)
        ICONT3=0
672     ACCEPT 302,ANSR
        IF(ANSR.EQ.NO)GO TO 674
        IF(ANSR.EQ.YES)GO TO 673
        ICONT3=ICONT3+1
        IF(ICONT3.GT.3)STOP 16
        TYPE 303,ANSR
        GO TO 672
673     TYPE 666, NM, ARRATE,SRVBAR,MCBF,MTTR
 666    FORMAT(/,'  1.  NUMBER OF MACHINES IS ', I8,
   1    /,'  2.  MEAN ARRIVAL RATE IS ', F10.2,
   3    /,'  3.  MEAN SERVICE TIME IS ',F10.2,
   4    /,'  4.  MEAN CYCLES BETWEEN FAILURES IS ',F9.0,
   5    /,'  5.  MEAN TIME TO REPAIR IS ',F10.2)
        IF(ICOST.EQ.0)GO TO 674
        TYPE 670,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR
670     FORMAT(/'  6.  CAPITAL COST PER UNIT IS ',F10.2
   1    /'  7.  USEFUL LIFE IS ',I8
   2    /'  8.  DISCOUNT RATE IS ',F10.2,'%'
   3    /'  9.  SPARES RATIO IS ',F10.2,'%'
   4    /' 10.  ANNUAL UNIT OPERATING COST IS ',F10.2
   5    /' 11.  ANNUAL HOURS OF SCHEDULED MAINTENANCE IS ',F10.2
   6    /' 12.  REPAIR WAGE RATE IS ',F10.2
   7    /' 13.  ANNUAL STATION PASSENGER VOLUME IS ',I9)
674     IF(CHANGE.EQ.1)RETURN
        ICONT=1
        ICONT1=0
 300    TYPE 301
 301    FORMAT(//'        DO YOU WISH TO MAKE ANY CHANGES ?',
   1    /,'        (''Y'' OR ''N''): '$)
 309    ACCEPT 302, ANSR
 302    FORMAT(A1)
        IF(ANSR.EQ.NO.AND.CHANGE.EQ.1)GO TO 667
        IF(ANSR.EQ.NO) RETURN
        IF(ANSR.EQ.YES) GO TO 306
        ICONT1=ICONT1+1
        IF(ICONT1.GT.3) STOP 2
        TYPE 303, ANSR
 303    FORMAT(/,'        THE PREVIOUS RESPONSE WAS ''',
   1    A1,'''.  PLEASE ANSWER ''Y'' OR ''N'': '$)
        GO TO 309
 306    ICONT2=0
        CHANGE=1
        TYPE 304
 304    FORMAT('        ENTER THE NUMBER OF THE QUESTION YOU WISH '
   1    ,'TO CHANGE'/
   2    '        (ONE INTEGER VALUE): '$)
 308    ACCEPT 203,I
```

```
      IF(.NOT.(I.GE.1.AND.I.LE.13))GO TO 305
      IF(.NOT.((I.GE.6 .AND. I.LE.13) .AND. ICOST.EQ.0))GO TO 678
      CHANGE=0
      TYPE 675
675   FORMAT(' COSTS ARE PRESENTLY NOT INCLUDED IN THIS MODEL.'
     1 ,'  DO YOU WISH TO ADD COST DATA?'/7X'(''Y'' OR ''N''):',$)
      ICONT3=0
676   ACCEPT 302,ANSR
      IF(ANSR.EQ.NO)GO TO 300
      IF(ANSR.EQ.YES)GO TO 677
      ICONT3=ICONT3+1
      IF(ICONT3.GT.3)STOP 15
      TYPE 303,ANSR
      GO TO 676
677   ICONT=0
      ICOST=1
      CHANGE=0
      GO TO 13
678   GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13),I
  305 ICONT2=ICONT2+1
      IF(ICONT2.GT.3) STOP 3
      TYPE 310
  310 FORMAT('       THE PREVIOUS RESPONSE WAS ',I5,
     1 /,'       PLEASE ANSWER WITH ONE INTEGER VALUE',
     2 'FROM 1 TO 12: '$)
      GO TC 308
      END
      SUBROUTINE ENDING
      COMMON IFILNM,FROMWH,NM,ARRATE,SRVBAR,MCBF,MTTR
     1 ,CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE,PASSYR,ICOST
      REAL MCBF,MTTR
      INTEGER FROMWH,YES,NO,ANS,PASSYR
      DOUBLE PRECISION IFILNM
      DATA YES,NO/1HY,1HN/
      FROMWH=2
      TYPE 101
101   FORMAT('ODO YOU WISH TO MAKE ANY FURTHER RUNS?'
     1 /6X' (ANSWER ''Y'' OR ''N'') ',$)
1     ACCEPT 102,ANS
102   FORMAT(A1)
      IF(ANS.EQ.YES)GO TO 2
      IF(ANS.EQ.NO)GO TO 9
      TYPE 103
103   FORMAT('OTYPE EITHER Y OR N.  TRY AGAIN. ',$)
      GO TO 1
2     FROMWH=1
      CLOSE(UNIT=4,DISPOSE='DELETE')
      CLOSE(UNIT=6,DISPOSE='DELETE')
      CLOSE(UNIT=7,DISPOSE='DELETE')
      TYPE 104
104   FORMAT('OIS THE NEW DATA THE SAME AS THE CURRENT DATA'
     1 ,' (EXCEPT FOR SOME CHANGES)?'/6X' (ANSWER ''Y'' OR ''N'') ',$)
3     ACCEPT 102,ANS
      IF(ANS.EQ.YES)GO TO 4
      IF(ANS.EQ.NO)GO TO 5
      TYPE 103
      GO TO 3
4     CALL INPUT
      RETURN
5     TYPE 105,IFILNM
105   FORMAT('OIS THE NEW DATA THE NEXT RECORD OF: ',A10,'?')
```

```
6        ACCEPT 102,ANS
         IF(ANS.EQ.YES)GO TO 8
         IF(ANS.EQ.NO)GO TO 7
         TYPE 103
         GO TO 6
7        FROMWH=0
         CLOSE(UNIT=55)
         TYPE 106
106      FORMAT('0THE NEW DATA IS FROM A NEW INPUT FILE.')
8        CALL BEGIN
         FROMWH=1
         RETURN
9        CLOSE(UNIT=4)
         CLOSE(UNIT=6)
         CLOSE(UNIT=7)
         RETURN
         END
         SUBROUTINE CSTMOD(CUNIT,LIFE,DISCRT,SPARES,OPER,HRSMNT,WRATE
     1   ,PASSYR,NM,MCBF,MTTR,COST)
         DIMENSION COST(6)
         INTEGER PASSYR
         REAL MCBF,MTTR
         RATE=DISCRT/100.
         IF(LIFE.EQ.0)GO TO 4
         SPRPCT=SPARES/100.
         CAPINV=NM*CUNIT
         SPRINV=CAPINV*SPRPCT
         PV=1.
         IF(RATE.LT..01)GO TO 2
         PV=RATE/(1.-(1./(1.+RATE)**LIFE))
2        COST(1)=CAPINV*PV
         COST(2)=SPRINV*PV
         COST(3)=OPER*NM
         COST(4)=HRSMNT*WRATE*NM
         COST(5)=WRATE*MTTR/3600.*(PASSYR/MCBF)
         COST(6)=0.
         DO 3 J=1,5
3        COST(6)=COST(6)+COST(J)
         GO TO 1
4        WRITE(4,101)I
101      FORMAT(' NO COST DATA FOR OR NON-EXISTENT UNIT',I3)
1        CONTINUE
         RETURN
         END
C
C     ..............................................................
C
C        SUBROUTINE MINV
C
C        PURPOSE
C           INVERT A MATRIX
C
C        USAGE
C           CALL MINV(A,N,D,L,M)
C
C        DESCRIPTION OF PARAMETERS
C           A - INPUT MATRIX, DESTROYED IN COMPUTATION AND REPLACED BY
C               RESULTANT INVERSE.
C           N - ORDER OF MATRIX A
C           D - RESULTANT DETERMINANT
C           L - WORK VECTOR OF LENGTH N
```

```
C           M - WORK VECTOR OF LENGTH N
C
C        REMARKS
C           MATRIX A MUST BE A GENERAL MATRIX
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           THE STANDARD GAUSS-JORDAN METHOD IS USED. THE DETERMINANT
C           IS ALSO CALCULATED. A DETERMINANT OF ZERO INDICATES THAT
C           THE MATRIX IS SINGULAR.
C
C     ..................................................................
C
      SUBROUTINE MINV(A,N,D,L,M)
      DIMENSION A(1),L(1),M(1)
C
C        ...............................................................
C
C        IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE
C        C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION
C        STATEMENT WHICH FOLLOWS.
C
C     DOUBLE PRECISION A,D,BIGA,HOLD
C
C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS
C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS
C        ROUTINE.
C
C        THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO
C        CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  ABS IN STATEMENT
C        10 MUST BE CHANGED TO DABS.
C
C        ...............................................................
C
C        SEARCH FOR LARGEST ELEMENT
C
      D=1.0
      NK=-N
      DO 80 K=1,N
      NK=NK+N
      L(K)=K
      M(K)=K
      KK=NK+K
      BIGA=A(KK)
      DO 20 J=K,N
      IZ=N*(J-1)
      DO 20 I=K,N
      IJ=IZ+I
   10 IF( ABS(BIGA)- ABS(A(IJ))) 15,20,20
   15 BIGA=A(IJ)
      L(K)=I
      M(K)=J
   20 CONTINUE
C
C        INTERCHANGE ROWS
C
      J=L(K)
      IF(J-K) 35,35,25
   25 KI=K-N
```

```
      DO 30 I=1,N
      KI=KI+N
      HOLD=-A(KI)
      JI=KI-K+J
      A(KI)=A(JI)
   30 A(JI) =HOLD
C
C        INTERCHANGE COLUMNS
C
   35 I=M(K)
      IF(I-K) 45,45,38
   38 JP=N*(I-1)
      DO 40 J=1,N
      JK=NK+J
      JI=JP+J
      HOLD=-A(JK)
      A(JK)=A(JI)
   40 A(JI) =HOLD
C
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS
C        CONTAINED IN BIGA)
C
   45 IF(BIGA) 48,46,48
   46 D=0.0
      RETURN
   48 DO 55 I=1,N
      IF(I-K) 50,55,50
   50 IK=NK+I
      A(IK)=A(IK)/(-BIGA)
   55 CONTINUE
C
C        REDUCE MATRIX
C
      DO 65 I=1,N
      IK=NK+I
      HOLD=A(IK)
      IJ=I-N
      DO 65 J=1,N
      IJ=IJ+N
      IF(I-K) 60,65,60
   60 IF(J-K) 62,65,62
   62 KJ=IJ-I+K
      A(IJ)=HOLD*A(KJ)+A(IJ)
   65 CONTINUE
C
C        DIVIDE ROW BY PIVOT
C
      KJ=K-N
      DO 75 J=1,N
      KJ=KJ+N
      IF(J-K) 70,75,70
   70 A(KJ)=A(KJ)/BIGA
   75 CONTINUE
C
C        PRODUCT OF PIVOTS
C
      D=D*BIGA
C
C        REPLACE PIVOT BY RECIPROCAL
C
      A(KK)=1.0/BIGA
```

```
   80 CONTINUE
C
C        FINAL ROW AND COLUMN INTERCHANGE
C
      K=N
  100 K=(K-1)
      IF(K) 150,150,105
  105 I=L(K)
      IF(I-K) 120,120,108
  108 JQ=N*(K-1)
      JR=N*(I-1)
      DO 110 J=1,N
      JK=JQ+J
      HOLD=A(JK)
      JI=JR+J
      A(JK)=-A(JI)
  110 A(JI) =HOLD
  120 J=M(K)
      IF(J-K) 100,100,125
  125 KI=K-N
      DO 130 I=1,N
      KI=KI+N
      HOLD=A(KI)
      JI=KI-K+J
      A(KI)=-A(JI)
  130 A(JI) =HOLD
      GO TO 100
  150 RETURN
      END
C
C     ..................................................................
C
C        SUBROUTINE GELG
C
C        PURPOSE
C           TO SOLVE A GENERAL SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS.
C
C        USAGE
C           CALL GELG(R,A,M,N,EPS,IER)
C
C        DESCRIPTION OF PARAMETERS
C           R      - THE M BY N MATRIX OF RIGHT HAND SIDES.  (DESTROYED)
C                    ON RETURN R CONTAINS THE SOLUTION OF THE EQUATIONS.
C           A      - THE M BY M COEFFICIENT MATRIX.  (DESTROYED)
C           M      - THE NUMBER OF EQUATIONS IN THE SYSTEM.
C           N      - THE NUMBER OF RIGHT HAND SIDE VECTORS.
C           EPS    - AN INPUT CONSTANT WHICH IS USED AS RELATIVE
C                    TOLERANCE FOR TEST ON LOSS OF SIGNIFICANCE.
C           IER    - RESULTING ERROR PARAMETER CODED AS FOLLOWS
C                    IER=0  - NO ERROR,
C                    IER=-1 - NO RESULT BECAUSE OF M LESS THAN 1 OR
C                             PIVOT ELEMENT AT ANY ELIMINATION STEP
C                             EQUAL TO 0,
C                    IER=K  - WARNING DUE TO POSSIBLE LOSS OF SIGNIFI-
C                             CANCE INDICATED AT ELIMINATION STEP K+1,
C                             WHERE PIVOT ELEMENT WAS LESS THAN OR
C                             EQUAL TO THE INTERNAL TOLERANCE EPS TIMES
C                             ABSOLUTELY GREATEST ELEMENT OF MATRIX A.
C
C        REMARKS
C           INPUT MATRICES R AND A ARE ASSUMED TO BE STORED COLUMNWISE
```

```
C           IN M*N RESP. M*M SUCCESSIVE STORAGE LOCATIONS. ON RETURN
C           SOLUTION MATRIX R IS STORED COLUMNWISE TOO.
C           THE PROCEDURE GIVES RESULTS IF THE NUMBER OF EQUATIONS M IS
C           GREATER THAN 0 AND PIVOT ELEMENTS AT ALL ELIMINATION STEPS
C           ARE DIFFERENT FROM 0. HOWEVER WARNING IER=K - IF GIVEN -
C           INDICATES POSSIBLE LOSS OF SIGNIFICANCE. IN CASE OF A WELL
C           SCALED MATRIX A AND APPROPRIATE TOLERANCE EPS, IER=K MAY BE
C           INTERPRETED THAT MATRIX A HAS THE RANK K. NO WARNING IS
C           GIVEN IN CASE M=1.
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           SOLUTION IS DONE BY MEANS OF GAUSS-ELIMINATION WITH
C           COMPLETE PIVOTING.
C
C     ..................................................................
C
      SUBROUTINE GELG(R,A,M,N,EPS,IER)
C
C
      DIMENSION A(1),R(1)
      IF(M)23,23,1
C
C     SEARCH FOR GREATEST ELEMENT IN MATRIX A
    1 IER=0
      PIV=0.
      MM=M*M
      NM=N*M
      DO 3 L=1,MM
      TB=ABS(A(L))
      IF(TB-PIV)3,3,2
    2 PIV=TB
      I=L
    3 CONTINUE
      TOL=EPS*PIV
C     A(I) IS PIVOT ELEMENT. PIV CONTAINS THE ABSOLUTE VALUE OF A(I).
C
C
C     START ELIMINATION LOOP
      LST=1
      DO 17 K=1,M
C
C     TEST ON SINGULARITY
      IF(PIV)23,23,4
    4 IF(IER)7,5,7
    5 IF(PIV-TOL)6,6,7
    6 IER=K-1
    7 PIVI=1./A(I)
      J=(I-1)/M
      I=I-J*M-K
      J=J+1-K
C     I+K IS ROW-INDEX, J+K COLUMN-INDEX OF PIVOT ELEMENT
C
C     PIVOT ROW REDUCTION AND ROW INTERCHANGE IN RIGHT HAND SIDE R
      DO 8 L=K,NM,M
      LL=L+I
      TB=PIVI*R(LL)
      R(LL)=R(L)
    8 R(L)=TB
```

```
C
C        IS ELIMINATION TERMINATED
         IF(K-M)9,18,18
C
C        COLUMN INTERCHANGE IN MATRIX A
       9 LEND=LST+M-K
         IF(J)12,12,10
      10 II=J*M
         DO 11 L=LST,LEND
         TB=A(L)
         LL=L+II
         A(L)=A(LL)
      11 A(LL)=TB
C
C        ROW INTERCHANGE AND PIVOT ROW REDUCTION IN MATRIX A
      12 DO 13 L=LST,MM,M
         LL=L+I
         TB=PIVI*A(LL)
         A(LL)=A(L)
      13 A(L)=TB
C
C        SAVE COLUMN INTERCHANGE INFORMATION
         A(LST)=J
C
C        ELEMENT REDUCTION AND NEXT PIVOT SEARCH
         PIV=0.
         LST=LST+1
         J=0
         DO 16 II=LST,LEND
         PIVI=-A(II)
         IST=II+M
         J=J+1
         DO 15 L=IST,MM,M
         LL=L-J
         A(L)=A(L)+PIVI*A(LL)
         TB=ABS(A(L))
         IF(TB-PIV)15,15,14
      14 PIV=TB
         I=L
      15 CONTINUE
         DO 16 L=K,NM,M
         LL=L+J
      16 R(LL)=R(LL)+PIVI*R(L)
      17 LST=LST+M
C        END OF ELIMINATION LOOP
C
C
C        BACK SUBSTITUTION AND BACK INTERCHANGE
      18 IF(M-1)23,22,19
      19 IST=MM+M
         LST=M+1
         DO 21 I=2,M
         II=LST-I
         IST=IST-LST
         L=IST-M
         L=A(L)+.5
         DO 21 J=II,NM,M
         TB=R(J)
         LL=J
         DO 20 K=IST,MM,M
         LL=LL+1
```

```
   20 TB=TB-A(K)*R(LL)                                                  GELG1550
      K=J+L                                                             GELG1560
      R(J)=R(K)                                                         GELG1570
   21 R(K)=TB                                                           GELG1580
   22 RETURN                                                            GELG1590
C                                                                       GELG1600
C                                                                       GELG1610
C     ERROR RETURN                                                      GELG1620
   23 IER=-1                                                            GELG1630
      RETURN                                                            GELG1640
      END                                                               GELG1650
```

REFERENCES

1. L. Rubenstein, J. Land, G. Desphande, B. Harrow; Overview of Rail Transit Fare Collection; Jet Propulsion Laboratory (Report #UMTA-CA-06-0116-80-1, JPL Publication 80-89); August, 1980.

2. G. Desphande, J. Cucchissi, R. Heft; Rail Transit Fare Collection Study: Policy and Technology Assessment, Volume 1; Jet Propulsion Laboratory (prepared for Urban Mass Transportation Administration / Transportation Systems Center); August, 1981.

3. Canadian Department of Transport; A Study of Transit Fare Policies, Fare Structures, and Fare Collection Methods; Urban Transportation Research Branch; April 1978.

4. G. Surabian; "Automated Magnetic Card Fare Collection Systems: An Overview With Emphasis on the Washington Metro Experience"; Urban Transportation Dept., Northeastern University; May, 1980.

5. J. Heisler, R. Stevens; Assessment of WMATA's Automatic Fare Collection Equipment Performance; Input-Output Computer Services (Report #UMTA-MA-06-0080-81-1); January, 1981.

6. Automated Services, Inc.; Automatic Fare Collection Equipment Reliability and Maintainability Assessment Plan for Urban Rail Properties; (Report #UMTA-MA-06-0025-81-1); March, 1981.

7. J. Koziol; "Summary Report of Fare Collection Workshop, March 11 & 12, 1980"; Report of workshop sponsored by the American Public Transit Association; U.S. Dept. of Transportation, Transportation Systems Center; March, 1980.

8. Input-Output Computer Services, Inc.; An Assessment of ICG's Automatic Fare Collection Equipment; Prepared for Urban Mass Transportation Administration/Transportation Systems Center; June, 1981.

9. M. Neuts, D. Lucantoni; "A Markovian Queue With N Servers Subject to Breakdowns and Repairs"; Management Science, Vol. 25, No. 9, Pp.849-861, September, 1979.

10. D. Gross, C. Harris; Fundamentals of Queueing Theory; John Wiley & Sons, New York, 1974.

11. Miami Dade County Transit Authority; letter, April, 1981.

☆ U. S. GOVERNMENT PRINTING OFFICE: 1985--501-933--20,081

200 copies