Applying Heuristic Traffic Assignment in Natural Disaster Evacuation - A Decision Support System

by

Kuo-Ping Hwang

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in
Civil Engineering

APPROVED:

A. G. Hobeika, Chairman

D. R. Drew

J. W. Dickey

S. A. Ardekani

S. C. Sarin

May, 1986

Blacksburg, Virginia

Applying Heuristic Traffic Assignment in Natural Disaster Evacuation - A Decision Support System

by

Kuo-Ping Hwang

A. G. Hobeika, Chairman

Civil Engineering

(ABSTRACT)

The goal of this research is to develop a heuristic traffic assignment method to simulate the traffic flow of a transportation network at a real-time speed. The existing assignment methods are reviewed and a heuristic path-recording assignment method is proposed. Using the new heuristic assignment method, trips are loaded onto the network in a probabilistic approach for the first iteration; paths are recorded, and path impedance is computed as the basis for further assignment iteration. The real-time traffic-assignment model developed with the new assignment method is called HEUPRAE. The difference in link traffic between this new assignment and Dial's multipath assignment ranges from 10 to 25 percent. Saving in computer time is about 55 percent. The proposed heuristic path-recording assignment is believed to be an efficient and reliable method.

Successful development of this heuristic assignment method helps solve those transportation problems which need assignment results at a real time speed, and for which the assign-

ment process lasts a couple of hours. Evacuation planning and operation are well suited to the application of this real-time heuristic assignment method.

Evacuation planning and operations are major activities in emergency management. Evacuation planning instructs people where to go, which route to take, and the time needed to accomplish an evacuation. Evacuation operations help the execution of an evacuation plan in response to the changing nature of a disaster.

The Integrated Evacuation Decision Support System (IEDSS) is a computer system which employs the evacuation planning model, MASSVAC2, and the evacuation operation model, HEUPRAE, to deal with evacuations. The IEDSS uses computer graphics to prepare input and interpret output. It helps a decision maker analyze the evacuation system, review evacuation plans, and issue an evacuation order at a proper time. Users of the IEDSS can work on evacuation problems in a friendly interactive visual environment.

The application of the IEDSS to the hurricane and flood problems for the city of Virginia Beach shows how IEDSS is practically implemented. It proves the usefulness of the IEDSS in coping with disasters.

ACKNOWLEDGEMENTS

The author wishes to express his deep appreciation and gratitude to his advisor Dr. Antoine G. Hobeika for his continued guidance and support.

His acknowledgement is extended to Dr. Donald Drew, Dr. John Dickey, Dr. Siamak Ardekani, and Dr. Subhash Sarin. They helped him enrich his knowledge in Transportation and related field.

All his professors, colleagues, and friends need to be thanked. There are too many names worth mentioning; only a few can be accommodated here. , , , , , , and will always be remembered for their guidance and help.

A special acknowledgement is expressed to who acts as the author's "mother" in the United States.

Most importantly, thanks go to the author's parents and his wife for their intellectual, financial, and emotional support.

TABLE OF CONTENTS

1.0 Introduction 1

1.1 Background of the Problem 1

1.2 Research Objectives 3

1.3 Outline of the Remaining Chapters 4

2.0 Literature Review 6

2.1 Evacuation and Human Behavior 6

2.2 Evacuation Traffic Simulators 8

2.2.1 Traffic Simulation Model 9

2.2.2 Evacuation Simulation Model 10

2.2.3 Short-Cut Method to Estimate Evacuation Time 13

2.3 Evacuation Decision Support System 14

2.4 Traffic Assignment 16

2.4.1 Role of Capacity Restraint in Traffic Assignment 18

2.4.2 Assignment Approach 19

2.4.3 Assignment Method 20

2.4.3.1 Iterative Assignment Procedure 22

2.4.3.2 Fractional Assignment Procedure 22

2.4.4 Assignment Methods Review 25

3.0 Theoretical Aspects of the Model 32

3.1 Introduction 32

3.2 Evacuation Time Estimation 32

3.2.1 Vehicle-Loading Phenomenon 33
3.2.2 Vehicle-Loading Curve 35
3.2.3 Critical Loading Parameter 39
3.2.4 Critical Evacuation Time 39
3.2.5 Minimum Evacuation Time 41
3.3 A Path-Recording Heuristic Assignment Model . . . 42
3.3.1 Modify Existing Assignment Method 42
3.3.2 A Proposed Heuristic Assignment Model 45
3.3.2.1 Two Approaches to Build the Multiple Paths 46
3.3.2.2 The Multiple-Path Algorithm 49
3.3.3 A New Assignment Algorithm 51
3.3.3.1 Link Index, Route Index, and Assignment . 51

4.0 Technical Aspects of the HEUPRAE Model 55
4.1 Introduction 55
4.2 Model Structure 55
4.3 Model Input and Output 57
4.4 Program Structure 58
4.4.1 Main Program 59
4.4.2 Subroutine INITIAL 59
4.4.3 Subroutine ODTAB 61
4.4.4 Subroutine EVAC 61
4.4.5 Subroutine SYSIN 62
4.4.6 Subroutine STOCH 62
4.4.7 Subroutine SYSOUT 63
4.4.8 Subroutine DIPATH 64

4.4.9 Subroutine HEUR 64
4.4.10 Subroutine TIMEE 65
4.4.11 Subroutine DISCH 65
4.4.12 Subroutine TREE 66
4.4.13 Subroutine FPASS 67
4.4.14 Subroutine SORTD 67
4.5 Technical Section Summary 67

5.0 Model Testing and Comparison **69**
5.1 Introduction 69
5.2 Evaluation of the MACROVAC Model 69
5.3 Comparison of the Assignment Algorithm 73
5.4 Comparison of the HEUPRAE Model 78
5.4.1 Number of Links with Positive Volume 81
5.4.2 Comparison of Link Volumes 82
5.4.3 Number of Assignment Paths 85
5.4.4 Network Clearance Time 86
5.4.5 Total Vehicle-Travel Time 86
5.4.6 Computer Execution Time Comparison 88
5.4.7 Computer Memory Comparison 91
5.5 Model-Sensitivity Analysis 92
5.5.1 Loading Parameter Analysis 93
5.5.2 Critical Evacuation Time Analysis 94
5.5.3 Diversion Parameter Analysis 97

6.0 An Evacuation Decision Support System **101**

6.1 System Components 101
6.1.1 System Control Module 103
6.1.2 Data Base Module 103
6.1.3 Simulation Model Module 104
6.1.4 Graphic Display Module 105
6.2 A Case Study Application 105
6.2.1 Characteristics of Virginia Beach 106
6.2.2 Building the Demographic Data Base 110
6.2.3 Coding the Highway Network 113
6.2.4 Estimating System Evacuation Time 114
6.2.5 Providing a Value for the Diversion Parameter 116
6.2.6 Running the Simulation Model 117
6.2.7 Storing the Simulation Output 118
6.2.8 Display the Simulation Output 118

7.0 Discussion, Conclusion, and Recommendation 119
7.1 Discussion 119
7.2 Conclusions 121
7.3 Recommendation 122

Appendix A. References 124

Appendix B. Glossarys 129

Appendix C. Program Flowcharts 132

Appendix D. Program Variable Definition 149

Appendix E. Program Listing 153

Appendix F. Listing of IEDSS Control Module 184

Appendix G. Demographic Data of Virginia Beach 185

Vita 190

LIST OF ILLUSTRATIONS

Figure 1. Iterative Assignment Procedure 23

Figure 2. Fractional Assignment Procedure 24

Figure 3. Evacuation System Input-output Relationship 34

Figure 4. Cumulated Vehicle-loading Time Relationship 36

Figure 5. Vehicle-loading Rate Time Relationship . . 37

Figure 6. Path-recording Assignment Approach 1 . . . 47

Figure 7. Path-recording Assignment Approach 2 . . . 48

Figure 8. General Framework of the HEUPRAE Model . . 56

Figure 9. Structure of the HEUPRAE Model 60

Figure 10. Virginia Beach Region 2 Highway Network . . 74

Figure 11. A Radial Hypothetical Highway Network . . . 75

Figure 12. Link-volume Comparison 1 (Va. Beach) . . . 83

Figure 13. Link-volume Comparison 2 (Hypothetical) . . 84

Figure 14. Evacuation Time and Loading Parameter . . . 95

Figure 15. Evacuation Time, Demand, and Route Capacity 96

Figure 16. Diversion Parameter Sensitivity 98

Figure 17. Components of the IEDSS 102

Figure 18. Virginia Beach Topographic Map 108

Figure 19. Virginia Beach 100 Years Flood Plain . . 109

Figure 20. Virginia Beach Evacuation Zones 111

LIST OF TABLES

Table 1. Assignment Method Classification 25
Table 2. Evacuation Time Estimation 72
Table 3. Algorithm Performance Comparison 77
Table 4. Model Output Comparison 80
Table 5. Computer Execution Time 89
Table 6. Computer Memory Comparison 92
Table 7. Diversion Parameter Analysis 99
Table 8. Virginia Beach Evacuation Shelters . . . 113
Table 9. Evacuation Routes for Region 2 115
Table 10. Evacuation Demand of Region 2 116
Table 11. Estimated Evacuation Times of Region 2 . 116

1.0 INTRODUCTION

1.1 BACKGROUND OF THE PROBLEM

Traffic assignment is the process of determining a route or routes of travel and allocating the centroid-to-centroid trips to these routes. Existing assignment methods can be classified in two categories: non-capacitated assignment methods and capacity-restraint assignment methods. The traffic-bearing capacities of the links and the effects of congestion are ignored for the non-capacitated assignment. The basis for all capacity-restraint assignments is the relationship between the impedance of a link and the volume of traffic assigned to that link. Link impedance is assumed to increase as traffic on the link increases. Two techniques are practically used in assignment method. They are the "single minimum-impedance routing technique" and the "multiple-routing technique." All-or-nothing assignment is the most common single-routing technique; it is easy to understand and apply but has unstable assignment result. Dial's probability assignment is the most common multiple-routing technique. Dial's method is proclaimed to be an equilibrium assignment method that is close to human behavior. However, for an assignment in which travel routes are important information, Dial's assignment is not an appropriate method since it obviates path enumeration. And even though paths can be traced out after the assignment, the tracing process consumes great amounts of computer time and memory.

Evacuation planning and operation are major activities of the preparedness and response phases of emergency management. They are the most predominantly used strategies to protect people threatened by disasters, and most importantly, they are controllable ones. The provision of transportation evacuation models to calculate evacuation time, to determine evacuation routes, and to simulate evacuation traffic assignment is only a partial concern of the evacuation problem. The major concerns of the problem include: 1. evacuation data-base management, 2. the computer processing unit, and 3. the computer-based software. They can incorporate and synthesize model results and expert-knowledge in ways that assist a system user in his decision-making under various disaster conditions. The approach of a decision support system (DSS) is to provide an interactive support for the thought process of one or more users in their principal function of making decisions. A major difference between the DSS approach and the traditional approach of operations research is that a DSS may include an optimization model but still relies on judgement. The development of a Decision Support System for evacuation planning and operation, which includes all the above elements, helps build a comprehensive integrated emergency management system.

1.2 RESEARCH OBJECTIVES

One goal of this research is to develop, test, and evaluate a Decision Support System for evacuation planning and operations under natural or man-made disasters. The system, running at a real-time speed, would assist the user in making decisions relative to the evacuation operation such as the evacuation routes, the proper time to issue an evacuation order, the adoption of transportation management strategies to alleviate traffic bottlenecks and to reduce network-clearance time, the allocation of response personnel, and the deployment of resources to demands.

A real-time, quick-response, interactive simulation model that can work on a microcomputer and provide the best evacuation route and reliable estimates of evacuation time is required to complete the proposed decision support system. It is particularly useful in the post-disaster or short-time rescues that take place after earthquakes or dam failure, since the conditions of disaster cannot be accurately predicted before a disaster occurs.

To convert existing evacuation models into real-time simulation models is also an objective of this research. The assignment method in the current computer program needs to be restructured to accomplish the real-time assignment. An algorithm which is able to internally develop the evacuation routes for a medium or large sized networks, and which requires very short execution time and small computer-memory storage, is also needed. To overcome the shortcomings of the existing assignment methods, a heuristic traffic assignment is proposed. In this new method, trips are loaded onto the network in a probabilistic manner for the first iteration; paths are recorded, and impedance of each path is computed as the basis for further assignment iteration. At the second or further iteration, assignment is constrained in these paths and, trips are loaded according to path-conductance, of which travel time is not the only consideration.

1.3 OUTLINE OF THE REMAINING CHAPTERS

A brief introduction to the problems, objectives, and main goals of the research were discussed in this chapter. The next chapter is devoted to a broad review of literature relevant to traffic assignments, disaster behavior, evacuation models, and the decision support systems for emergency management. In the review, the shortcomings and inability of the existing assignments in dealing with evacuation operation are discussed. In Chapter 3, the theoretical aspects of the new assignment algorithm and other algorithms used to build the computer simulation model are discussed. A detailed description of each component and subroutines of the computer program, plus the interaction among them, are presented in Chapter 4. The validation of the assignment algorithm, the model, and a variable-sensitivity analysis are given in Chapter 5. The development of an integrated evacuation decision support system, IEDSS, and its application in a case study, the city of Virginia Beach are described in Chapter 6. The application results, along with the application process, are also discussed in Chapter 6. Finally, in Chapter 7, the conclusions of this research, plus recommendations and suggestions for further research, are presented.

2.0 LITERATURE REVIEW

2.1 EVACUATION AND HUMAN BEHAVIOR

<u>Mitigation</u>, <u>Preparedness</u>, <u>Response</u>, and <u>Recovery</u> are the four phases of Emergency Management (EM). Mitigation refers to activities which eliminate or reduce the probability of occurrence of a disaster. Preparedness planning and training refers to what to do in case a disaster occurs. Response activities follow a disaster, and are designed to provide emergency assistance and to reduce the likelihood of secondary damage. Recovery involves restoring affected areas to normalcy. In these four phases, evacuation planning and operation are the major activities in the preparedness and response phase of emergency management. Although different mitigation actions might prevent a disaster happening, evacuation in many cases is still unavoidable. The practice of evacuating people threatened by disaster has a long history. As early as the fifth century B.C., the Greek historian Herodotus described the annual evacuation undertaken by Egyptians to cope with the seasonal flooding of the Nile River[Perry]. Many disasters, which have happened recently, also show the importance of evacuation. For example, the two 1985 disasters in Central America could have been alleviated if an evacuation plan had been available. The earthquake

which happened in Mexico City, on Sept. 19, 1985, left as many as 20,000 dead and by some estimates, up to 150,000 homeless. The eruption of the volcano Nevado del Ruiz in Colombia, on Nov. 13, 1985, also left at least 20,000 dead or missing in a steaming, mile-wide avalanche of gray ash and mud within hours[TIME]. The Colombian town of Armero (pop. about 22,500) was virtually wiped off the map after that catastrophe.

To prepare for a disaster, the emergency manager certainly wants to know how to evacuate the threatened people in the shortest period. Evacuation planning helps answer the questions, such as where to evacuate the people, which route to choose, and how long an evacuation process might last. In addition, evacuation planning assists a decision-maker to define his disaster boundaries and to prepare disaster countermeasures.

It is important to understand human behavior in a disaster and evacuation environment. Quarantelli, in his literature review[Quarantelli, 1980], concludes that the "community context in combination with threat conditions and social process result in certain patterns of evacuation behavior." He divided the evacuation process into four possible patterns: warning, withdrawal movement, shelter, and return. Other findings by Quarantelli pertinent to evacuation behavior are:

1. The automobile is the prime transportation mode used to withdraw from danger.
2. The withdrawal movement is always orderly with no wild panic or disorderly flight.
3. Traffic accidents or automobile breakdowns are minimal.
4. In the withdrawal movement, several researchers concluded that there is no instant bolting into flight by masses of individuals upon perception of danger. People assess the emergency situation, obtain confirmation of immediate and personal danger, then usually leave with the members of their families--their most important social group.
5. Not everyone leaves except in the most catastrophic situations.

The analysis done by Carter, et al[Hall] also shows that the cumulative percentage of evacuees who had left home during each hourly period of the hurricane evacuation followed a logit curve. Therefore, it is believed that people leave the endangered area in an orderly fashion and the leaving rate follows a normal curve, which is the derivation of a logit curve.

2.2 EVACUATION TRAFFIC SIMULATORS

The traffic models used to simulate vehicular flow for an evacuation come from two sources: they can be modified from the existing models designed to simulate traffic flow under normal (non-disaster) conditions, or they can be developed exclusively for the purpose of evacuation.

The models used to simulate traffic flow for an evacuation have some basic structures in common. They use as input:

1. The area and population characteristics.
2. The highway network topology and traffic control conditions.
3. The safe areas to which the evacuees should be withdrawn.
4. The vehicle-loading curves on a highway network which represent the public response to evacuation.

Using these inputs and a capacity-restrained traffic assignment algorithm, an evacuation model can simulate the vehicular flow of an evacuation process.

2.2.1 TRAFFIC SIMULATION MODEL

A recent survey of traffic simulation models reveals that there are as many as 104 models, but only a few models are considered useful and operational in dealing with current traffic problems[Gibson and Ross]. Those existing models, which are operational and can be modified for evacuation applications, include:

1. EMME/2-an interactive graphic system for modeling multimodal transportation networks[Florian].
2. NETSIM-NETwork SIMulation model[KLD, 1977].
3. SIGOP3-a macroscopic signal timing design and analysis model for coordinated signal systems[Lieberman, 1982].

4. TRAFLO-TRAffic FLOw simulation model[Lieberman, 1980].
5. TRANSYT7-TRAffic Network StudY Tool[Wallace].
6. UTPS-Urban Transportation Planning System[FHWA, 1985].

The application of these general purpose traffic simulation models in evacuation planning has been rare, because they are not primarily designed to produce the results needed by emergency planners, such as the cumulative evacuation time for a particular zone, the number of people arriving at a safe shelter area, and the expected use of different evacuation routes. These models require extensive restructuring to apply to evacuation planning. In addition, they employ an extensive data base beyond the one used in evacuation. Unfortunately, they also consume a good amount of computer running time and memory.

2.2.2 EVACUATION SIMULATION MODEL

A limited number of traffic flow simulators have been designed for evacuation planning under disaster conditions. Among those reviewed are:

1. The CLEAR (Calculating Logical Evacuation And Response) model: a model developed by the Pacific Northwest Laboratories for the U.S. Nuclear Regulatory Commission. It estimates the time required for a specific population density to evacuate an area using a specific transportation network[Carroll].
2. The DYnamic Network EVacuation (DYNEV) model: a special purpose microscopic evacuation model, derived through the

enhancement of a submodel contained in TRAFLO. The model has the capabilities to provide detailed information, both in graphic and tabular forms, about the operational performance of each network link[KLD, 1982].

3. The PRC Voorhees EVAC PLAN PACK model: a computer program developed for simulating traffic flow during evacuation and for calculating the evacuation time for a given set of population, road network, evacuation rate, and control measures[PRC V].
4. The Emergency Evacuation Model: a microscopic simulation model developed to determine evacuation times of Emergency Planning Zones (EPZ) around nuclear power plants[Cosby].
5. The MASSVAC/MASSVAC2 model: a model developed by Hobeika/Radwan/Jamei/Hwang at Virginia Tech for MASS eVACuation of people in urban areas threatened by natural disasters[Hobeika]. The MASSVAC2 model is a new version of the MASSVAC model, which is updated and improved by Kuo-Ping Hwang.
6. The NETwork Emergency EVACuation (NETVAC1) Simulation model: a model developed by MIT to estimate the network clearance time for areas surrounding nuclear power plants[Sheffi].

Except for MASSVAC, all the above models are designed to estimate the vehicular traffic evacuation time and to develop evacuation plans around a nuclear power plant, in case of radioactive leaks. They have the flexibility to simulate traffic flows for evacuation planning in various types of emergencies. The shelters are those areas anywhere beyond the the boundary lines of a problem area with no capacity restraint. Choosing evacuation routes is solely based on the traffic conditions without consideration of the risk factors, the direction of the disaster propagation, and the route topography. All six models are applicable to point-source disasters; the MASSVAC, CLEAR, and DYNEV models can also cover the area-wide disasters.

The MASSVAC model has two levels of analysis: a macroscopic and a microscopic level. The macroscopic level simulates the evacuation process on a highway network by looking at the major network arteries as a complete and integrated system. The microscopic level deals with a selected subnetwork of the whole network, where congestion is taking place, and provides solution through transportation management strategies to alleviate the congestion. It also considers the effects of the risk factor and route topology, which might affect routes chosen during an evacuation process.

The evacuation simulation models reviewed above, are primarily evacuation planning models with little or no capability to work for real-time evacuation operations. They mainly address the development of evacuation plans around point-source disasters, such as nuclear power plant accident, and area-wide disasters, such as hurricane and flood. Since the computer-execution speed of these models is slow, especially for the large network size, they cannot simulate for a real-time evacuation the dynamic movements of vehicular flow in a highway network. For example, during a chemical spill emergency, the rescue routes or evacuation routes cannot be accurately predicted prior to the emergency, and consequently need to be provided in at short notice. To respond to the changing nature of a disaster, a real-time traffic simulation model for evacuation operations is thus needed.

2.2.3 SHORT-CUT METHOD TO ESTIMATE EVACUATION TIME

Beside using simulation models to estimate evacuation time, two short-cut approximation methods are available. It is not necessary to employ a simulation if only the evacuation time is needed and if it can be obtained through other methods. These two short-cut methods are the Hans Method and the Highway Capacity Manual Method. These are useful and quick-response methods to roughly estimate evacuation time. The descriptions of these two methods are followed:

Hans Method: This model was developed by Hans in 1974[Hans]. He used regression analysis to analyze the reported evacuation time and built his model using the following form:

$$\log(T)=1.30571 - 0.21243\log(\text{persons/square miles})$$

where T = evacuation time in hours

The highway capacity used in his model was 2,600 cars/lane/hour, and the average vehicle occupancy was four persons. About 10,000 persons can be evacuated per lane per hour. The shortcomings of this model are that the model does not consider the effect of a highway network and the highway capacity is independent of route standard.

Highway Capacity Manual Method: The Highway Capacity Manual Method consists in allocating a part of the evacuating population to each evacuation route. The capacity of each route is estimated, following the Highway Capacity Manual[TRB, HCM]. The time necessary to clear each route is then calculated. This approach has been applied by Wilbur Smith in an early study of Seabrook, New Hampshire[Wilbur], and by A. M. Voorhees and Assoc. in an evacuation study for the Diablo Canyon plant in California[Voorhees].

Both of these quick and primitive methods suffer from some fundamental drawbacks. They do not account for basic traffic-flow relationships that consider vehicle densities, speeds, queues, and spillbacks. They also do not have the capability to make dynamic route-choice decision, which can only be captured by a time-dependent and probabilistic computer simulation model.

2.3 EVACUATION DECISION SUPPORT SYSTEM

For a decision support system, knowledge means facts plus beliefs and heuristics. Knowledge in evacuation decision-making is an amalgam of judgement, experience, and intuition gained by the practitioners and experts in the field of evacuation, in addition to the results provided by evacuation simulation models. The knowledge-base of an evacuation decision support system is divided into two parts: 1) the traffic flow simulators, and 2) the expert-supplied rules. Evacuation knowledge plus the simulators increase the power and depth of understanding of evacuation phenomena under consideration. The expert-supplied rules act as functional models that can validate, calibrate, and restructure the simulator results. The knowledge-base as a whole provides the first-step decision in the DSS for evacuation. The second-step decision is exercised by the decision generator software in its dialogue with the user to provide the final decision.

The application of DSS to emergency management has been rare. A microcomputer-based emergency response system was developed by Belardo et al to provide information to assist emergency managers in responding to an incident at a nuclear power plant[Belardo]. An Automated Emergency Response System (AERS), developed by Petrie, was installed for several transit systems, such as BART, to provide controllers, dispatchers, and supervisors with a quick and accurate information retrieval system[Petrie]. The AERS helps the decision-making in case of an emergency. Another successful application is the "Emergency Management System for Chemical Spills" designed to aid the decision-maker in the management of inland oil and hazardous chemical spills[Johnson].

Although there are few other applications of decision support system or expert system in emergency management, none of the existing systems deal with the problem of evacuation planning and operation.

2.4 TRAFFIC ASSIGNMENT

"Traffic assignment" may be formally defined as the process of determining a route or routes of travel and allocating the centroid-to-centroid trips to these routes[FHWA, 1972].

The result of the traffic assignment is an estimate of user volumes on each segment of a transportation network[FHWA, 1973]. The uses of traffic assignment techniques include:

1. The development and testing of alternate transportation systems.
2. The establishment of short range priority programs for transportation facility development.
3. The detailed study of traffic generators and their effects on the transportation system.
4. Analysis of location for facilities and service within a transportation corridor.
5. Development of design volume.
6. Provision of necessary input and feedback to other planning tools.

Input to the traffic assignment process, regardless of type of network to be considered (transit, highway, rail, etc.) includes:

1. Network Geometry: A description of the interconnections and segments of the network representing the transportation system under consideration.

2. Network Parameters: The required network segment parameters (impedances) to allow selection of routes in an assignment through the network under study. Examples of the impedance are travel time, distance, travel cost, or a combination of these.

3. Interchange Value: The trip unit to be loaded onto the transportation network through the assignment process. Examples of the interchange are vehicles, persons, or tons of cargo.

The output of the traffic assignment process basically consists of loads on each of the segments of the transportation network. These may be 24-hour vehicular highway traffic volumes, peak hour transit volumes, or yearly volumes of freight flow. There are two major steps included in the traffic assignment process.

1. The calculation of the assignment routes through the network.

2. The allocation and accumulation of travel (O-D trips, generally) on the links comprising the assignment routes.

The first step is a critical one if one realizes that there are thousands of possible alternate routes between each pair of zones in even a network with moderate size. The second step has its focus on the algorithm which determines the allocation of trips among different routes. In addition, the second step records the accumulation of trips to individual facility segments.

2.4.1 ROLE OF CAPACITY RESTRAINT IN TRAFFIC ASSIGNMENT

All assignment methods use the relative impedances of the various alternative routes to allocate the centroid-to-centroid trips. The route impedance is just the sum of the impedances of the individual links which are traversed in each route throughout the network. The manner in which the link impedances are treated during the assignment process provides a basis for classifying traffic assignment methods into two general categories[Morris].

Free or Non-capacitated Assignment In this approach, the traffic-bearing capacities of the links and the effects of congestion are ignored and the link impedances are maintained throughout the assignment process.

Capacity-Restraint Assignment This approach recognizes that the impedance to flow on a link will increase as the number of vehicles utilizing the link per unit of time increases.

Of the two assignment approach, the capacity-restraint assignments are most useful in allocating trips in urban transportation networks, but for inter-city planning, where peak-period congestion on the links is of less concern, the non-capacitated assignment models generally provide satisfactory results[Morris].

In this research of the traffic assignment under emergency conditions, the traffic demand varies in according to a time-dependent evacuation process and the trips are loaded onto an urban network. The capacity-restraint assignment reflects better the behavior of route choosing in evacuation. It is chosen and employed for further investigation.

2.4.2 ASSIGNMENT APPROACH

According to Wardrop's first principle, every trip-maker wants to shorten his travel impedance if his condition allows[Wardrop]. As a result of employing this principle, a traffic assignment will have an equilibrium system, in which the summation of the total travel impedance is minimized. Otherwise, the total impedance can be further reduced by assigning trips to shorter routes and the system is not in equilibrium. An assignment thus should be able to provide an equilibrium result as well as to simulate the traffic on a network.

A mathematical approach and an iterational approach are commonly used to solve the equilibrium assignment problem. The mathematical approach has as its objective function to minimize the total travel impedance. Link volumes for the equilibrium system are then obtained by solving the constraint functions simultaneously. Using this approach, different models have been developed, such as the nonlinear convex optimization model developed by Sang Nguyen[Nguyen]. However, the relationship between the traffic flow and impedance on a link is not linear, and the impedance is also increased by traffic delay at intersections. The mathematical approach is usually difficult to apply in most assignments since it has too many and complicated constraints.

Another approach is to do assignment iteratively to reach as close to the equilibrium as possible. This is a heuristic approach. The heuristic approach has the benefit of being easy to understand and program, but has no global optimal except an acceptable solution. Since the number of iterations is related to the time and cost of program execution, the trade-off between the accuracy of output and its cost should be seriously considered. Different techniques together with procedures have been proposed to improve the efficiency and accuracy of the assignment.

Because of the difficulty of the mathematical approach, this research discusses and employes only the heuristic iteration approach.

2.4.3 ASSIGNMENT METHOD

Existing assignment techniques may be categorized as either single-routing techniques or as multiple-routing techniques. The single-routing techniques load all trips from an origin to a destination on the minimum-impedance route connecting the centroids, while the multiple routing techniques attempt to spread the centroid-to-centroid trips over additional feasible and likely routes in the network. The single-routing techniques assume that the trip-makers have the same perception of route impedance and always act in an optimal manner. In contrast, the multiple-routing techniques attempt to incorporate the behavioral nature of route selection into the traffic assignment process.

The basis for all capacity-restraint assignments is the relationship between the impedance of a link and the traffic volume assigned to this link. Link impedance is assumed to increase as traffic on the link increases, and, of course, it also depends on the capacity of the link. The usual approach in capacity-restraint assignment is to modify the link impedance during the assignment process to reflect the increased utilization of each link. Different equations which use the ratio of volume to capacity (V/C) have been used to adjust the link impedances. They are

$$Ia = Io[1+0.15(V/C)^4] \qquad \text{[FHWA,1972]}$$

$$Ia = Io \times (2)^{(V/C-1)} \qquad Ia \leq 4Io \qquad \text{[Schneider]}$$

$$Ia = Ic \times e^{(V/C-1)} \qquad Ia \leq 5Ic \qquad \text{[Smock]}$$

$$Ia = Io/[.5+.5(1-V/C)^{\frac{1}{2}}] \qquad \text{[Drew]}$$

where Ia = the adjusted impedance obtained as a result of the assigned link volume V
Io = the free-flow (non-congested) impedance
Ic = the impedance when volume equal to capacity
C = the practical capacity of the link.

The two most commonly used procedures for iteration assignment which apply volume-impedance relationship to adjust the link impedance are the iterative procedure and the fractional procedure.

2.4.3.1 Iterative Assignment Procedure

Iterative procedures, illustrated in Figure 1, involve a recursive process in which the volume-impedance relationship is used to adjust the link impedance between repeated applications of the assignment technique. The entire O-D trip matrix is assigned at each iteration, and the impedance is then adjusted according to the resulting link volume.

2.4.3.2 Fractional Assignment Procedure

Fractional procedures differ from iterative procedures in that they adjust link impedances after the assignment of successive fractions of the trip matrix to the network, while the iterative procedures alter the impedances at each iteration after assigning the entire trip matrix. Figure 2 illustrates the sequence of operations followed by the fractional procedures.

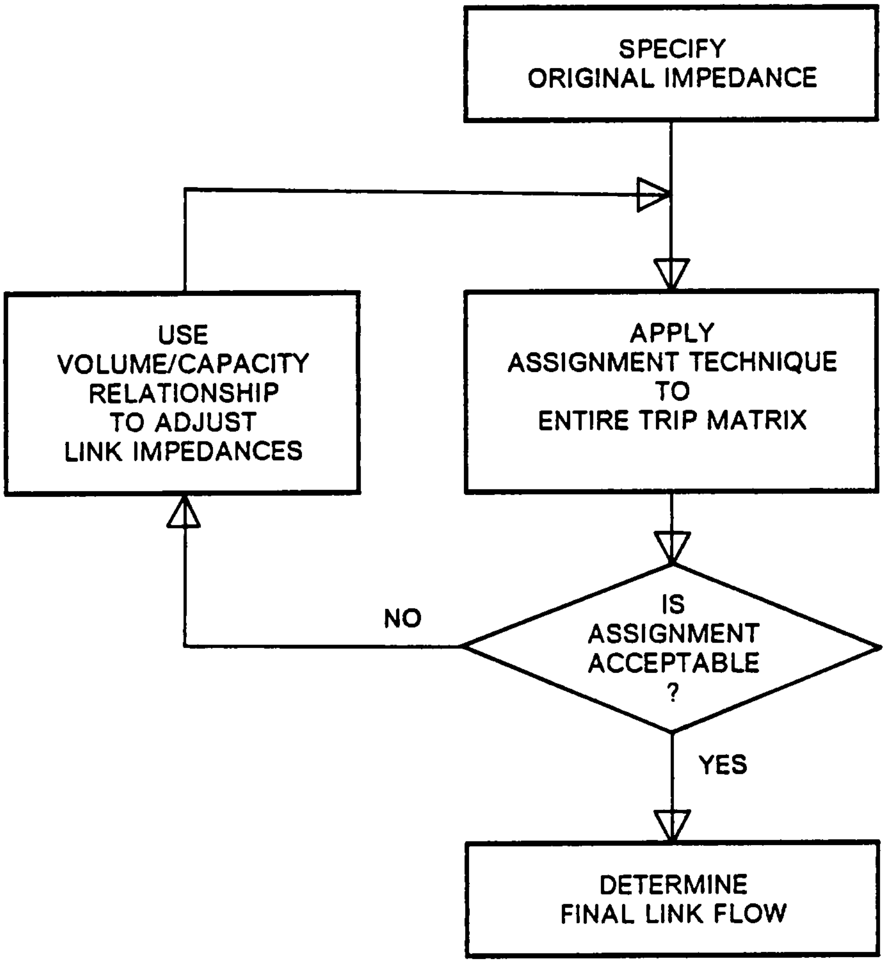


Figure 1. Iterative Assignment Procedure

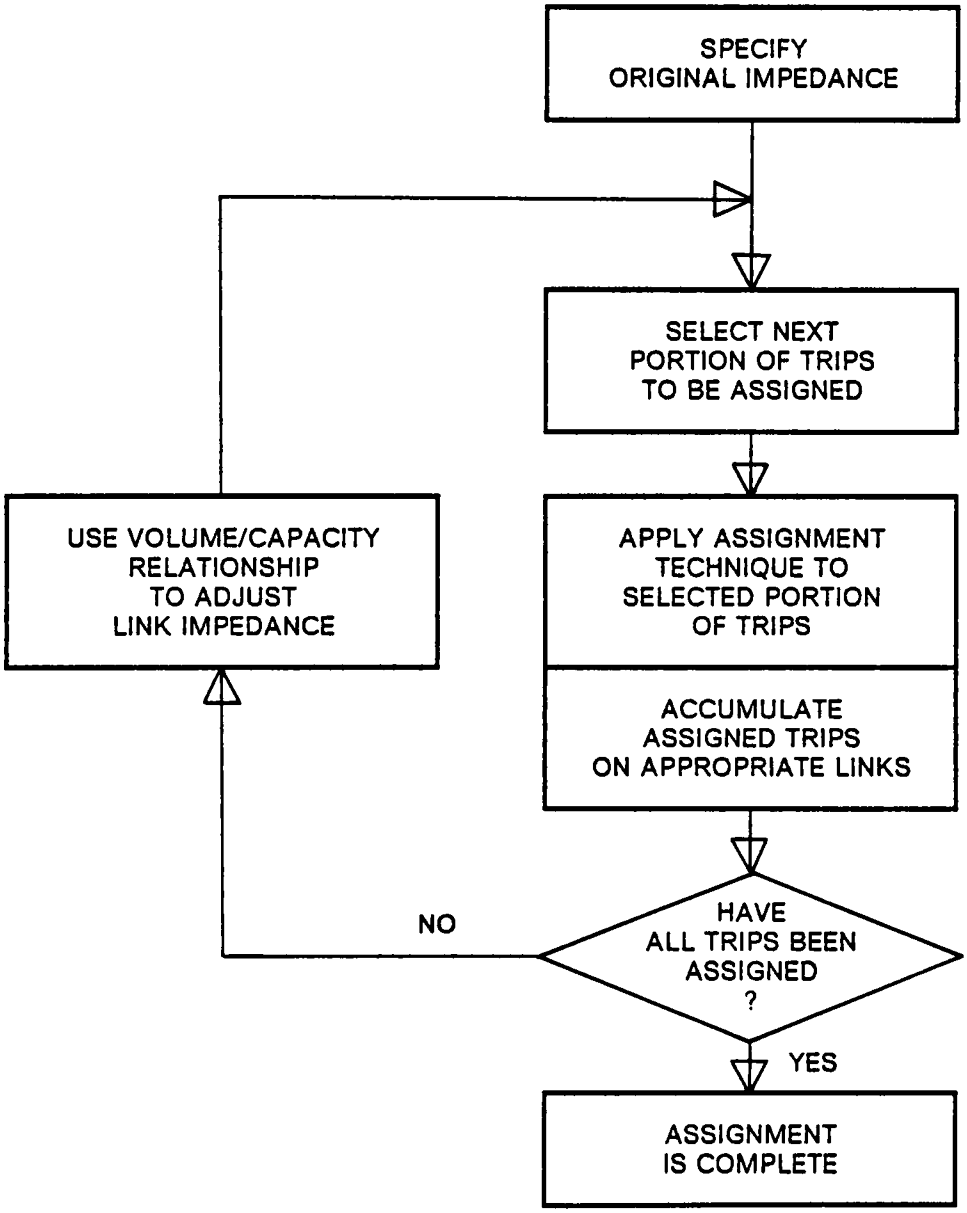


Figure 2. Fractional Assignment Procedure

Using the single or multiple routing technique and assignment procedure to classify assignment methods, different methods are identified as in Table 1.

Table 1. Assignment Method Classification

	ITERATIVE PROCEDURE	FRACTIONAL PROCEDURE
Single Minimum-impedance Routing	All-or-nothing assignment	All-or-nothing assignment
Multiple Routing	Diversion assignment, Probabilistic assignment	Diversion assignment, Probabilistic assignment

---

2.4.4 ASSIGNMENT METHODS REVIEW

All-or-nothing Assignment

The process of executing an all-or-nothing assignment consists of two basic steps:

1. The application of a shortest route algorithm to identify the minimum-impedance routes between each centroid pair.
2. The loading of the trips between each centroid pair onto the shortest-route connecting them.

The benefit of this method is that it is easy to understand and apply. But the iterative "all-or-nothing" assignment falls short at providing a convergent output among different iterations. The fractional "all-or-nothing" assignment is criticized for not incorporating human element in route selection although it does provide a better convergent result.

Diversion Assignment

This method assumes that a portion of trips, as determined by a diversion curve, is diverted from a major route to another competing route. Three types of diversion curves are usually used: 1. the time-ratio curve, 2. the distance and speed-ratio curve, and 3. the time and distance differential curve[Bureau of Public Roads]. Diversion assignment requires that there be two routings for each interzonal movement, usually one via the arterial street system and the other via the freeway system. Those two route systems are competing with each other and the difference between them is significant. In addition, these two routes should be known before vehicle-loading and a manual method is usually used to identify them instead of a computer program. Since the number of routes for each interzonal movement is usually more than two in an urban network, the diversion method is not an appropriate method for urban traffic assignment.

Probabilistic Assignment

The best-known, most commonly used, and most often cited model for probabilistic traffic assignment is Dial's multipath traffic assignment model[Dial]. Dial's model allocates trips to alternative "efficient" paths through a two-pass Markov approach. The efficient paths are defined as the paths which comprise links with positive assignment-likelihood. A link has a positive assignment-likelihood only when its initial node is closer to the origin node than its final node. Since it employs a probabilistic approach in its multipath assignment, Dial's model is claimed to be an equilibrium model which can reflect the actual human route-choosing behavior. In addition, Dial's model also obviates the path enumeration process. The efficient paths are not actually built but are pseudo ones; the traffic is directly assigned onto the links of which the efficient paths are composed.

The execution of Dial's model consists of four principal steps.

Step 1: Applying a shortest-route algorithm to build the minimum path tree from the origin centroid 'o' to each node 'i' in the network and determining their minimum impedances (Ioi).

Step 2: Calculating the "link assignment-likelihood" a(L) for each link = (i,j) where the initial node 'i' is closer than its final node 'j' to the origin:

$$a(L) = \begin{cases} e^{\theta(Ioj-Ioi-Lij)} & \text{if } Ioi < Ioj \\ 0 & \text{otherwise} \end{cases}$$

where Lij = impedance of link (i,j), and $Lij \geq (Ioj-Ioi)$

The diversion parameter θ is a constant which must be provided by the user. Users of the model use θ to control the "path diversion probability," the degree to which the assignment algorithm diverts trips from the shortest route onto longer alternative routes. The assignment-likelihood is a comparative number in which all links which fall on the shortest-path tree have their likelihood equal to one unit. Other links have assignment likelihood less than one unit. In most cases, a value of θ between 0 and 2 appears to be appropriate. For values greater than 10, almost all trips are assigned to the shortest route[Morris].

Step 3 (Forward Pass): Using a sort routine to order all nodes in an ascending sequence with respect to the node impedance (Ioi) from the origin centroid, then computing the "link weight" for all links with positive assignment likelihood. The link weight 'W(L)' is defined by this formula:

$$W(L) = \begin{cases} a(L) & \text{if } i = o \text{ (the origin centroid)} \\ a(L) \sum_{L' \text{in } Fi} w(L') & \text{otherwise} \end{cases}$$

where i = the starting node of link 1
Fi = the set of all links ending at node i.

Note that the weight 'W(L)' of each link 'L' depends upon the weights of the links which are on the efficient paths and precede link 'L'.

Step 4 (Backward Pass): After each link has been assigned a weight, trips are backwardly loaded onto the network through the most distant node to the origin. This is called "Backward Pass." For each link 'L' ending at node 'j', assign a link trip volume X(L) by proportion to the link weight w(L), that is

$$X(L) = \frac{Yj \times W(L)}{\sum_{L' \text{ in } Fj} W(L')}$$

where Yj = volume at node j
Fj = the set of all links ending at node j

After the trip-loading for node 'j', volume on link 'l' is increased by the amount X(L) and the loading process continues until the origin node is reached.

Merits and Faults of the Dial Multipath Assignment Model:

The primary advantage of Dial's algorithm is that it allows many alternative routes to be considered without having to enumerate them explicitly or examine them individually. Trips are assigned to all efficient paths simultaneously in the two-pass process, and the number of trips assigned to a specific path can be varied by changing the value of θ. The number of trips assigned to any given link depends upon the number of efficient paths in which the link appears, as well as upon the relative attractiveness of these paths: longer paths, and consequently the links comprising them, receive fewer trips. In this manner, the model appears to reflect well the behavior of a population of human trip-makers in choosing a travel route.

The principal difficulty of this technique lies in the determination of the value of θ. Even though the use of this parameter does extend the model's flexibility, it also provides an additional variable which must be adjusted and refined in the model calibration process. Few guidelines are currently available for the selection of proper values of this diversion parameter[Morris]. The nature of the link-likelihood function, in which the parameter is employed, may also lead to some difficulties in the use of the model. For example, decreasing the value of θ to divert trips from the shortest route to longer but competitive routes also results in assigning trips to very long routes. In addition, the model does not provide for separately specifying the portion of trips to be diverted from the shortest paths. All characteristics of the assignment are dependent upon the single value of θ utilized.

A final problem which might be encountered in the application of the model is that the model may preclude the assignment of trips to certain very attractive alternative routes. The model will not assign trips to an alternative path, if the shortest-route impedance from the origin centroid to the last node of the path before reaching the destination centroid is greater than the shortest-route impedance between the origin and destination centroids. According to the likelihood function, any node falling farther than the destination node to the origin node will have 0 unit assignment-likelihood and will not be included in an efficient path, even if the node is very close to the destination node.

3.0 THEORETICAL ASPECTS OF THE MODEL

3.1 INTRODUCTION

The goal of this research is to develop a heuristic traffic assignment method and apply it in an evacuation model which can run at a real-time speed. The inputs for the evacuation model are demographic characteristics, highway network geometry, and desired evacuation time. The final product of the model are network clearance time, congested links, traffic volume on links, average evacuation travel time, and average vehicle delay. However, the model involves many algorithms, techniques, and a new heuristic assignment method, which need to be explained in detail. In this chapter, the approach to developing the new heuristic assignment method, the theories for building the evacuation model, and the method for estimating parameters for an evacuation model are described. Also, the reason for applying any specific approach but not the alternative one is explained.

3.2 EVACUATION TIME ESTIMATION

To run a simulation model, the length of simulation period needs to be defined in advance. It is thus necessary to provide an estimated evacuation time for the evacuation model before simulation.

A macroscopic analytical model, MACROVAC--MACRO eVACuation, is thus developed to fulfill the requirement of estimating the evacuation time for an evacuation system. MACROVAC employs a logit vehicle-loading curve and the input-output relationship between evacuation demand and supply to estimate evacuation time, which is illustrated in Figure 3. The theories and techniques used to develop the MACROVAC model are described in the following section.

3.2.1 VEHICLE-LOADING PHENOMENON

During an evacuation, people leave the endangered areas for safety and protection. The time-distributed leaving rate of the evacuees is critical knowledge for the assumption of vehicle-loading in an evacuation-simulation model and its subsequent analysis. The literature review in Chapter 2 reveals that the evacuation process is more proactive than reactive; the evacuation withdrawal movement is always orderly with no wide panic, and the traffic accidents or automobile breakdowns are minimal. In addition, Carter's analysis also shows that the cumulative evacuation vehicle-loading follows a logit curve[Hall]. The leaving rate can thus be assumed to follow a normal curve, which is the derivation of a logit curve.

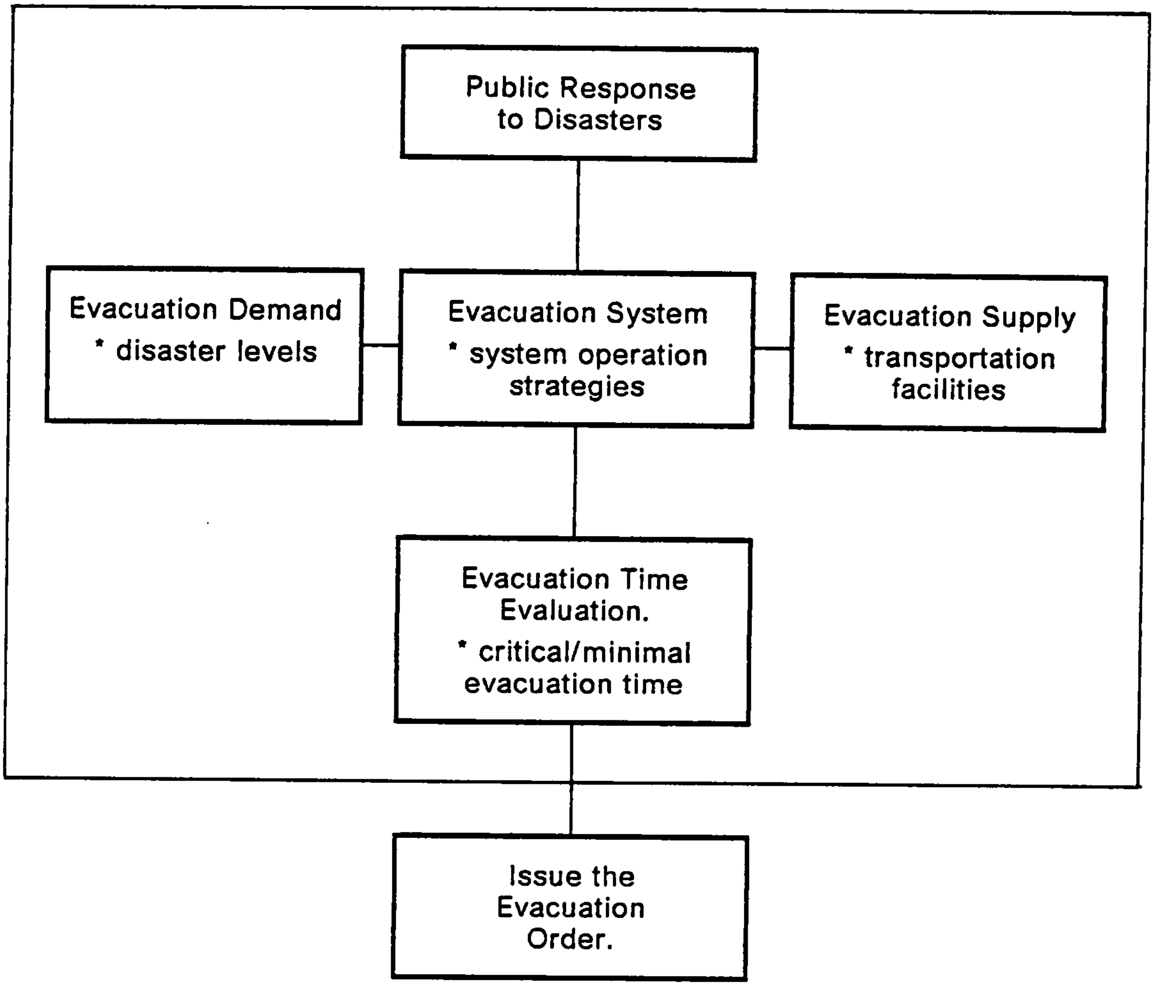


Figure 3. Evacuation System Input-output Relationship

VEHICLE-LOADING CURVE

The assumption about the normally-distributed human behavior during evacuation leads cumulated vehicle-loading follow the logit curve:

$$CP = \frac{1}{1 + e^{[LP \times (T-HLT)]}} \qquad 3.1$$

where CP = cumulative fraction of the vehicle loading
LP = loading parameter (1/minute)
T = time (minute)
HLT = half-loading time, the time 50% of the total evacuation demand is expected to have entered the network.

By taking the partial derivative of equation 3.1 to time 'T', and times it with the total evacuation demand, vehicle-loading rate can be derived:

$$VLR = \frac{TV \times [LP \times e^{(-LP \times (T-HLT))}]}{[1 + e^{(-LP \times (T-HLT))}]^2} \qquad 3.2$$

where VLR = vehicle-loading demand rate (veh/unit time)
TV = total evacuation demand (veh).

As can be seen in equations 3.1 and 3.2, TV, LP, and HLT are three variables which affect the evacuation vehicle loading. Figures 4 and 5 illustrate the curves of CP and VLR/TV for LP and HLT of 0.022 and 180 minutes respectively.

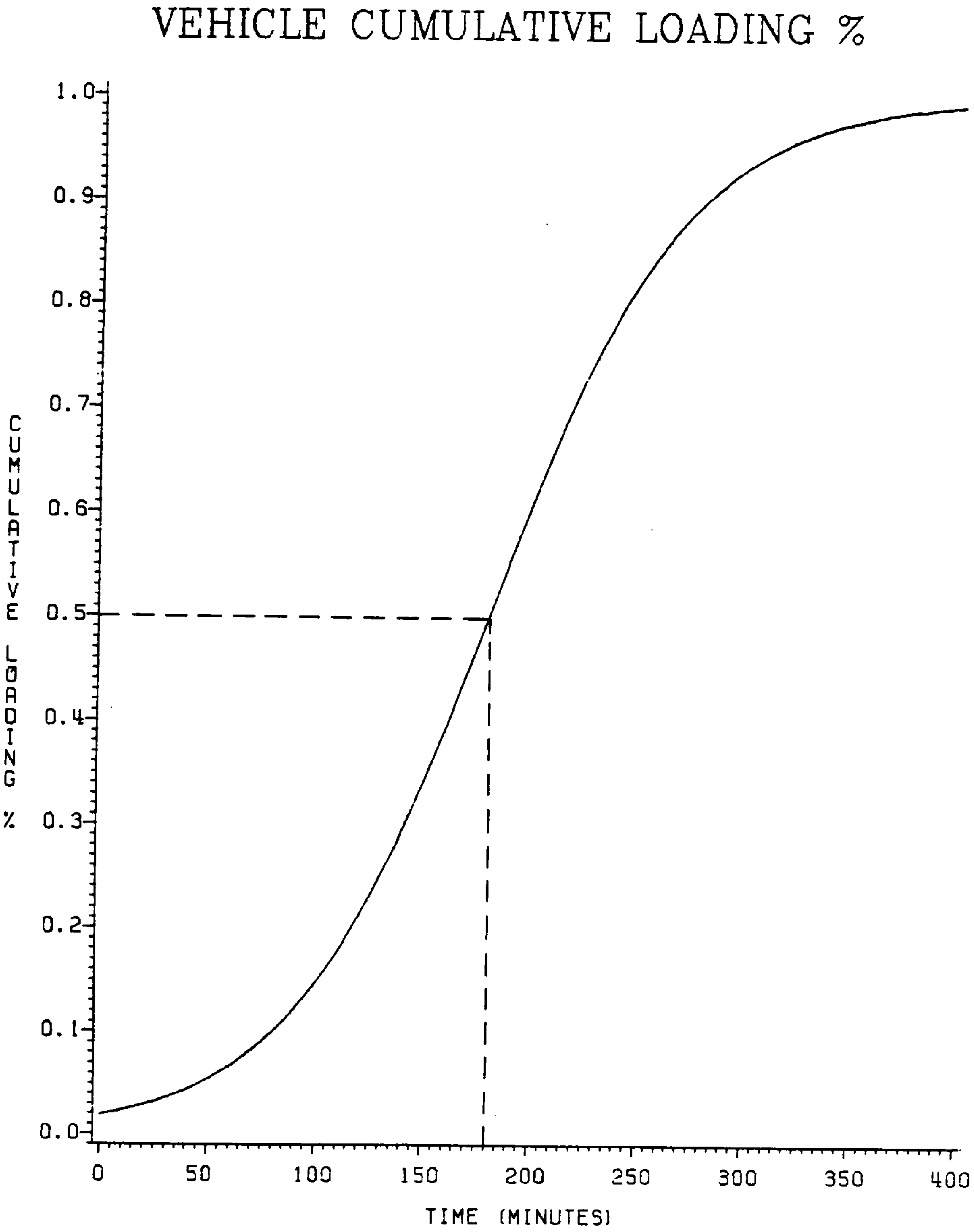


Figure 4. Cumulated Vehicle-loading Time Relationship

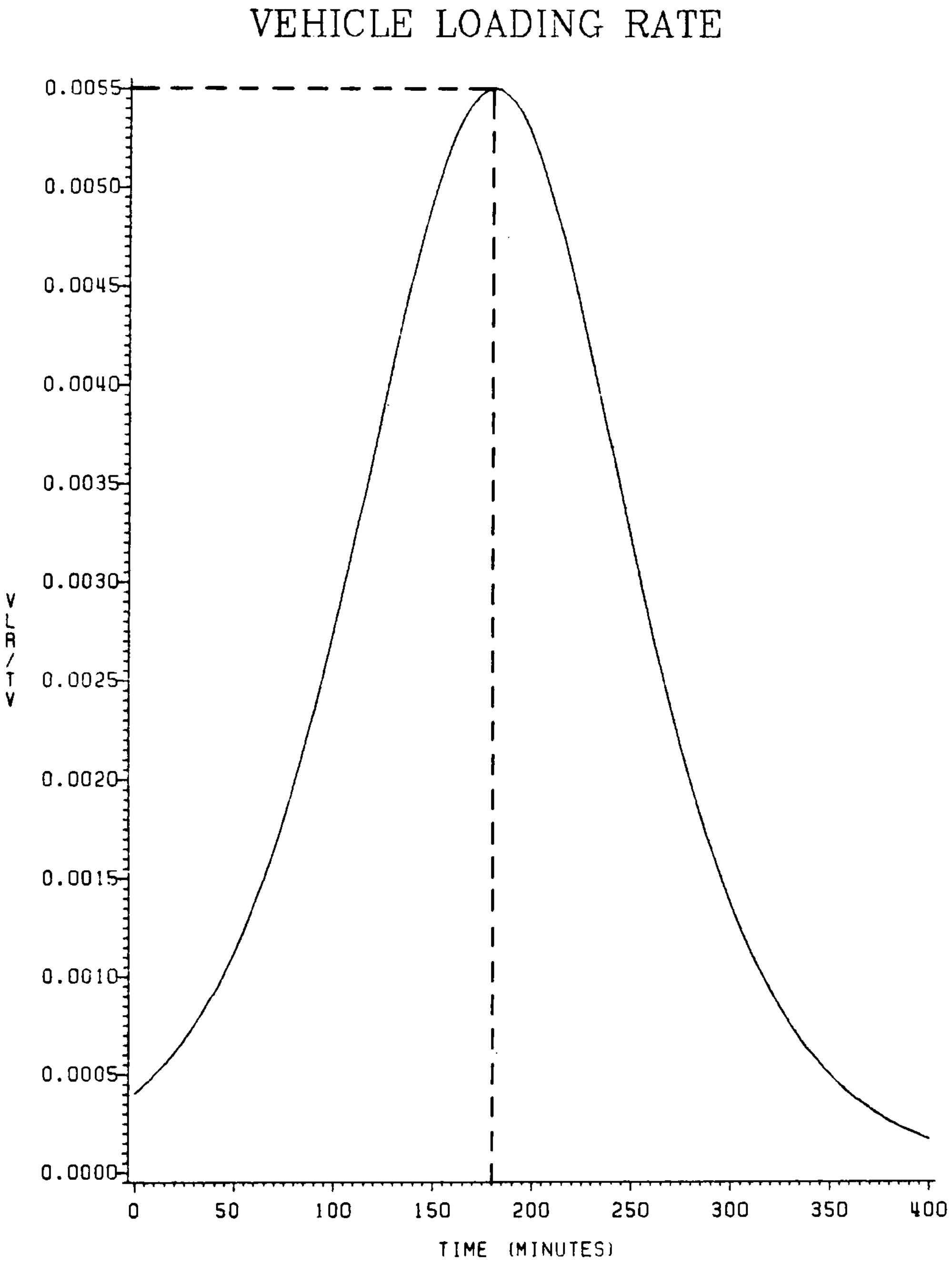


Figure 5. Vehicle-loading Rate Time Relationship

Regarding TV, in most evacuation cases, its value can be estimated accurately by analyzing the regional demographic characteristics, disaster type, and community boundary. TV can thus be viewed as a constant in a specific disaster evacuation. However, LP and HLT are two variables unknown until an evacuation happens.

During a disaster and before issuing an order to execute an evacuation plan, an emergency administrator can more or less control the time needed to evacuate the endangered people through the emergency warning system, e.g., radio, television, etc. Thus the variable HLT, which is half the value of the expected evacuation time (ET), can also more or less be viewed as a controllable one, and only variable LP is left unknown in depicting the evacuation vehicle-loading phenomenon.

The evacuation routes, through which evacuees leave the problem area, are in the supply side of the evacuation system. The evacuation route capacity can always be accurately computed for a specific highway network. With the evacuation supply and demand available, a theoretical minimum evacuation time (TMIN) is introduced by the followed equation:

$$\mathrm{TMIN} = \frac{\mathrm{TV}}{\mathrm{MDR}} \qquad 3.3$$

where MDR = maximum vehicle-dissipation rate (veh/min).

Here, the theoretical minimum evacuation time (TMIN) is the shortest time vehicles can dissipate under the assumption that the vehicle-loading rate is uniformly-distributed and equals to the evacuation route capacity. The value of the MDR is converted from the evacuation route capacity. Since TMIN is obtained under the assumption that vehicles enter the system at a constant rate, which is contradictory to the assumption of this research, that vehicle-loading rate is normal-distributed rather than uniform-distributed. The TMIN can only be used as an auxiliary reference in evacuation time estimation.

3.2.3 CRITICAL LOADING PARAMETER

Define the critical loading parameter (CLP) as the loading parameter when the maximum vehicle loading rate equals the maximum dissipation rate. A mathematical analysis[Hwang] shows that the variables CLP and TMIN have the following relationship:

$$\text{CLP} = 4 \times \text{MDR}/\text{TV} = 4/\text{TMIN}. \qquad 3.4$$

3.2.4 CRITICAL EVACUATION TIME

Since both tails of the logit loading curve extend to no end, the time required to load the whole 100% evacuation demand (evacuation time) is an infinite value. Being more realistic, it is convenient to define expected loading time (ET) as the time 98% or 99% of the total demand has been loaded onto the network. The value 98% is chosen in this research since to load this one percent (98% to 99%) evacuation demand into the system increases ET by as much as 18%. Substituting 1/2 ET to the HLT in equation 3.1, a relationship between ET and LP is found:

$$\text{ET} = 2 \times \ln(49)/\text{LP}. \qquad 3.5$$

Since the relationship between ET and LP is found, the only unknown variable 'LP' in equation 3.1 can be substituted by ET. The vehicle-loading phenomenon of equation 3.1 and 3.2 thus can be depicted by ET (2HLT) and TV only. By replacing LP with CLP, a critical expected loading time can also be achieved:

$$\begin{aligned} \text{CET} &= 1.95 \times \text{TMIN}. \qquad 3.6 \\ &= 1.95 \times \text{TV}/\text{MDR} \end{aligned}$$

Therefore, given the value of TV and MDR, information about CLP and CET can be computed analytically.

3.2.5 MINIMUM EVACUATION TIME

For any evacuation system, there exists a minimum evacuation time (MET) which is the shortest time an emergency manager can expect to clear the network successfully. To achieve this MET, the evacuation route should be fully operated at the capacity level from the time the vehicle-loading rate reaches the level of maximum dissipation rate; otherwise a smaller MET will be achieved. For this kind of evacuation phenomenon, vehicle queue length increases gradually to its maximum length, then decreases eventually until the last minute of the evacuation loading.

Since we have no exact formula to express MET, a hill-climbing algorithm is used in the MACROVAC model to search for the value of MET. The maximum loading parameter (MLP) coupled with this MET is also calculated to show how concentrated the vehicle-loading phenomenon is.

Referring to the value of CET and MET, and using MACROVAC to evaluate the evacuation system, an emergency manager can thus judge when the best time is to issue the evacuation order, and he can choose an appropriate ET as the estimated evacuation time to simulate an evacuation model if the network traffic condition is needed.

Since only very simple data are needed to run the MACROVAC model, no complicated demographic data manipulation is required; MACROVAC is a quick, responsive method to estimate evacuation time. It is especially useful for a local government which does not have a detailed evacuation plan but which is in a disaster prone area. In addition, compared with Hans' method or the HCM method, the MACROVAC model is more realistic since it considers the vehicle-loading as normally distributed rather than uniformly distributed.

3.3 A PATH-RECORDING HEURISTIC ASSIGNMENT MODEL

The development of a new heuristic assignment method and its application in a real-time assignment model are described in this section. The approach to improve the efficiency of the current assignment methods is carefully investigated; thus the new method can have good accuracy as well as efficiency in assignment result.

3.3.1 MODIFY EXISTING ASSIGNMENT METHOD

An assignment algorithm is a method used to assign trips to links of a network in an assignment model, such as the all-or-nothing algorithm or Dial's algorithm. Any assignment model must employ an assignment algorithm in its assignment procedure, whether the procedure is an iterative or fractional one. The approach to increase efficiency of the existing assignment model can thus be addressed in improving the assignment algorithm or the assignment iteration process.

As described in Chapter 2, an assignment process includes the route-finding step and the trip-loading step. Even though Dial's assignment model claims to be a path enumeration free method, its forward pass can be viewed as a step to build efficient paths.

The shortest-path algorithm is a necessary algorithm in almost all assignment methods. This algorithm helps find the shortest path for each O-D pair, as well as building the shortest-impedance tree for an origin to all nodes in a network. During each assignment iteration, the shortest-path algorithm must be executed as many times as necessary until each O-D pair has the shortest path in between. The execution of the shortest-path algorithm consumes a lot of computer time in an assignment iteration. Therefore, a good shortest-path algorithm, the avoidance of the shortest-path algorithm, or a decrease in the number of executions required by this algorithm, will improve the efficiency of an assignment model.

A better shortest-path algorithm certainly improves the efficiency of the assignment. A large number of different shortest-algorithms were developed in late 1950's and early 1960's. In 1969, Dreyfus reviewed these algorithms in his paper "An Appraisal of Some Shortest-Path Algorithms"[Dreyfus]. In 1976, Golden tested Dijkstra's algorithm and Bellman's algorithm at MIT and concluded that Bellman's algorithm outperformed Dijkstra's algorithm[Golden]. Those researches of shortest-path algorithms reveal that one algorithm might outperform the others in some network. However, no matter how good a shortest-path algorithm is, the number of its execution in an assignment cannot be reduced.

Another approach to improve efficiency of the assignment is trying to avoid execution of the shortest-path algorithm in different assignment iterations. This approach is not very important to a transportation planner since the assignment execution time is not an important factor but accuracy in planning work is. However, for an evacuation operation model, which needs to simulate the evacuation process for a few hours and provide the simulation result in a real-time speed, this approach shows a lot of potential. Since the success of this approach can reduce assignment time (simulation time) significantly, the question of how to do assignments without employing a shortest-path algorithm is raised.

For an assignment, if the assignment paths do not vary significantly in different iterations, the assignment paths for one iteration can be kept for the next iteration. The time to execute the shortest-path algorithm is then saved. For a case like an evacuation, not all trip-makers are familiar with their evacuation routes; choosing of the evacuation routes can be considered limited and constrained. Therefore, a path-recording approach may be useful in developing a real-time evacuation operation model.

Because of the volume-impedance relationship, the all-or-nothing algorithm, which uses a single-routing assignment technique, normally does not load trips on the same paths in different iterations. The path-recording approach thus needs a multiple-routing assignment technique to be successful in assignment accuracy as well as in cutting the assignment execution time.

3.3.2 A PROPOSED HEURISTIC ASSIGNMENT MODEL

As discussed in the previous section, the efficiency of an assignment can be increased by using the same paths in different iterations. This method can be called "a path-recording assignment method." A multiple-routing technique is necessary to provide enough paths between each O-D pair, and a path-recording mechanism is also needed to keep those paths for iterations. In addition, an assignment algorithm to apportion trips to those paths is also required. Since the path recording assignment method can be applied with two approaches to find multiple paths, those approaches will be discussed first. Details about building the multiple paths and the assignment algorithm will be discussed later.

3.3.2.1 Two Approaches to Build the Multiple Paths

Two approaches can be used to build the multiple-path tree for the path-recording assignment. The first approach uses a multiple-path algorithm to build a multiple-path tree for each O-D pair. The second approach traces and uses the multiple-path trees, which are built with a path-enumeration free assignment method in the first iteration, as the assignment paths for later iterations. The major differences between these two approaches lie in the computer memory and time required to execute the model. Figures 6 and 7 present the flow-diagrams of the path-recording assignment with these two approaches to build the assignment paths.

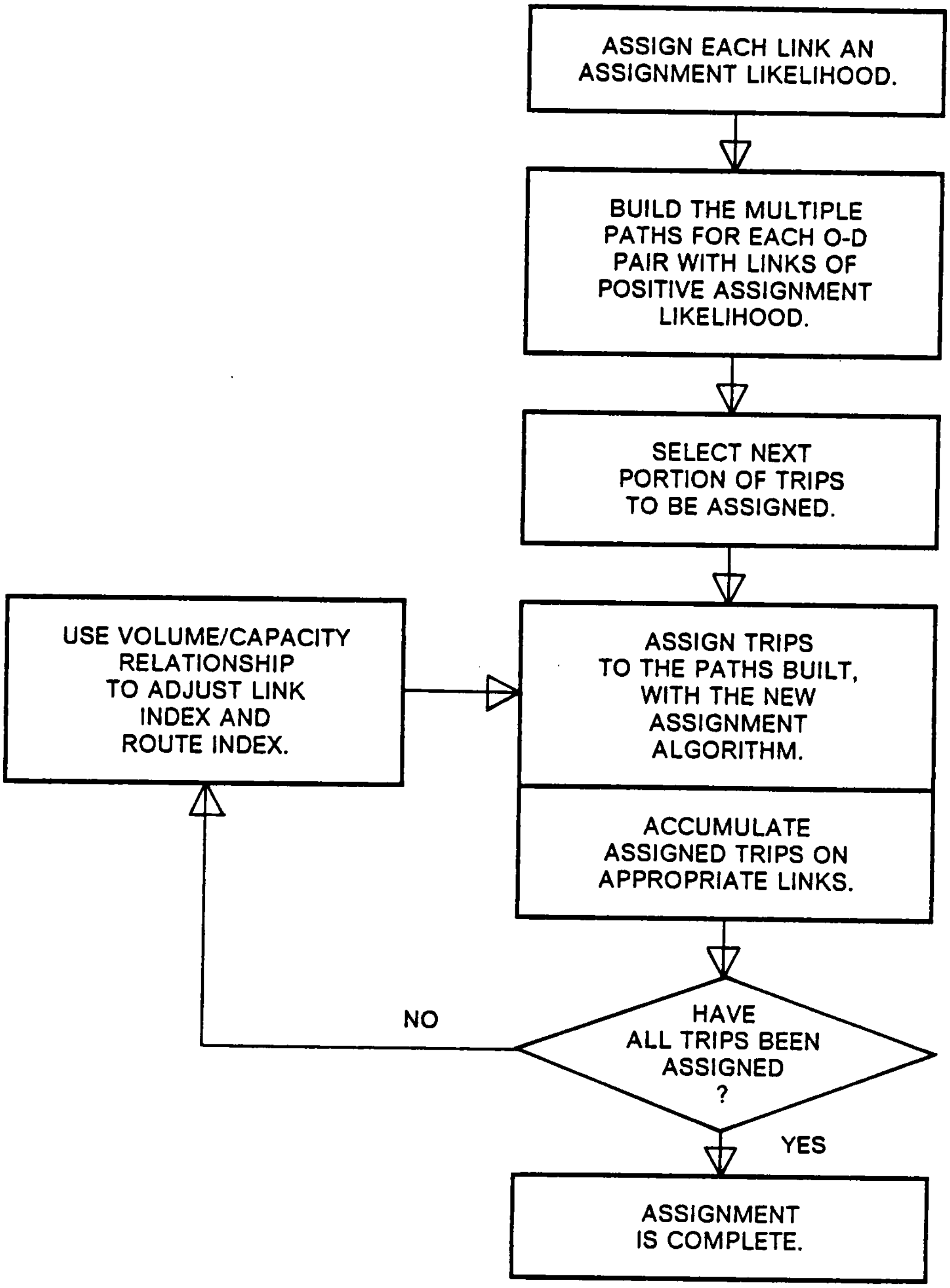


Figure 6. Path-recording Assignment Approach 1

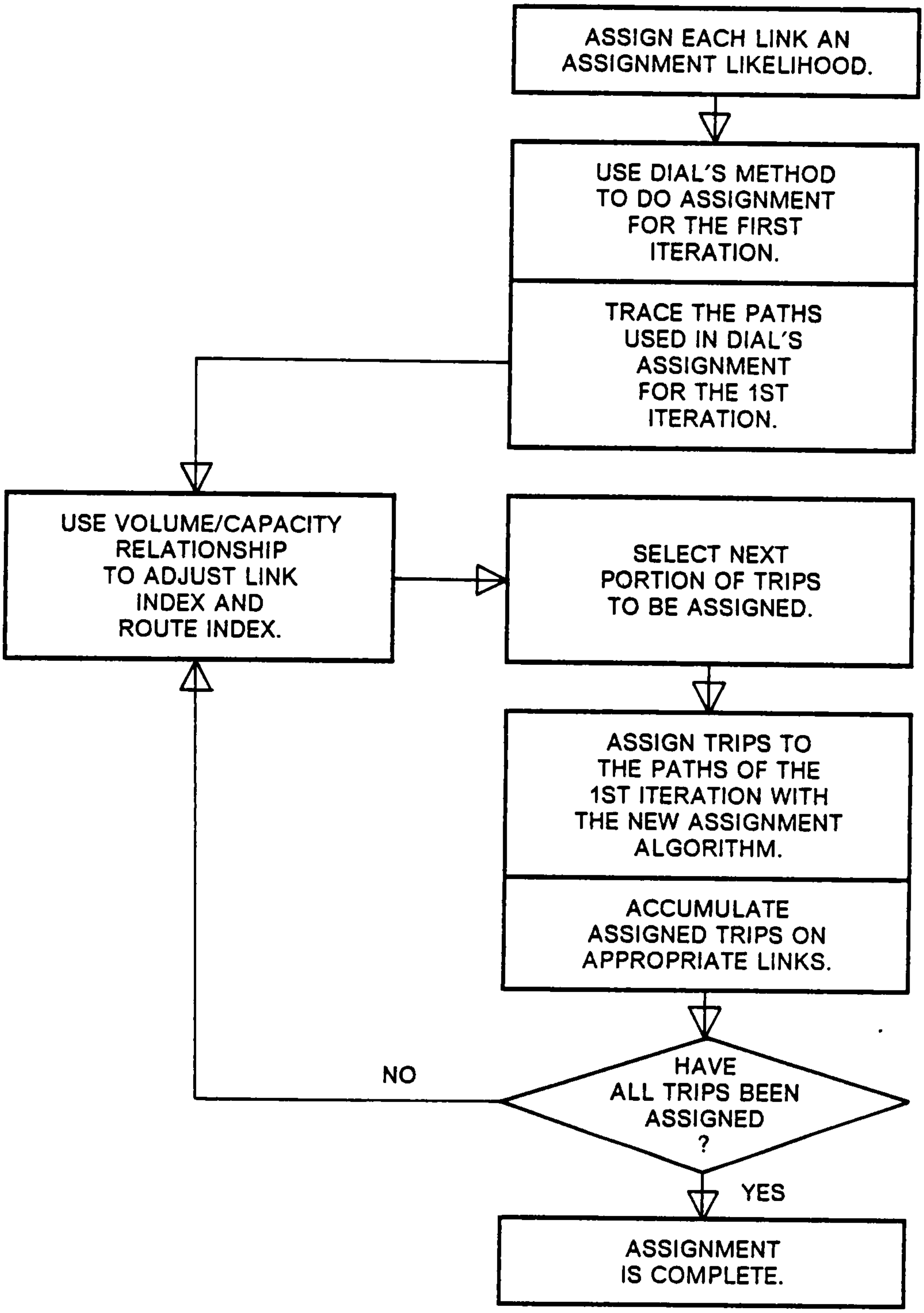


Figure 7. Path-recording Assignment Approach 2

3.3.2.2 The Multiple-Path Algorithm

A multiple-path tree is a collection of paths from one origin to one destination. A multiple-path algorithm is a method which helps build a multiple-path tree for an O-D pair. The multiple-path algorithm serves as the first approach in building the multiple-path tree for the path-recording assignment. The multiple-path algorithm, developed in this research, uses the branch-and-bound approach to build a multiple-path tree with the following iterative steps:

1. Sort network nodes in an ascending sequence according to their impedances to the origin node.
2. Employ Dial's efficient-path concept to assign each link in the network an assignment likelihood to identify the links used for path building.
3. Extend paths, through the links with positive assignment likelihoods, from the origin node (1st node) to the 2nd node.
4. Extend paths from the n node to the n+1 node with links of positive assignment likelihood.
5. If the destination node is reached, stop; otherwise continue from step 4.
6. Identify and keep record of the paths which reach the desired destination node.

The assignment-likelihood assigned to each link in the multiple-path algorithm is only used as an index to indicate if a link is going to be used in path-building. Its unit does not concern with the traffic assigned in the path-recording assignment.

The application of the multiple-path algorithm to build the assignment paths is one of the approaches to forming the path-recording assignment model. It is easy to understand and program. However, this approach requires a lot of computer memory to execute because the whole path-tree must be stored before a desired destination node is reached. Since only a few paths reaching the destination node are useful, the algorithm wastes a lot of memory in its execution. Therefore, this approach to forming the path-recording assignment model is not memory efficient.

Besides using a multiple-path algorithm for building the multiple paths, those paths can be traced out after the assignment of a path-enumeration free method, such as Dial's assignment. Dial's assignment does not need a lot of memory to execute since the trips are assigned by a two-pass Markov process instead of the path enumeration approach. Because tracing out the used paths requires less memory than to building the multiple paths, finding paths after the first assignment iteration and using them in later iterations is another approach to forming a path-recording assignment model.

In this research, both approaches to building the multiple paths are used to develop the path-recording assignment model, and the difference will be investigated in Chapter 5.

3.3.3 A NEW ASSIGNMENT ALGORITHM

With the multiple paths available for each O-D pair, trips are waiting to be loaded onto those paths. An assignment algorithm is thus required to apportion the trips onto those paths. Considering the effect of link volume, link capacity, and link travel time to the decision of path choosing, a new assignment algorithm is developed to reflect those factors and to apportion the trips. This assignment algorithm first assigns an index (link conductance) to each link of the network, then computes for each path a path index (weighted link conductance), and finally uses the path index as a weight to apportion the trips onto the multiple paths between an O-D pair. Details of the new assignment algorithm are described below.

3.3.3.1 Link Index, Route Index, and Assignment

The link index in this algorithm is a travel conductance which reflects the carrying capability of a link and its travel time. It represents the attraction of trip-maker to use the link. The link index has the form:

1. Link index:

$$Iij= \frac{(Cij - Vij) \times Sij}{Lij} = \frac{\text{Available Capacity x Speed}}{\text{Length}}$$

$$\text{i.e. } Iij = \frac{\text{Available Capacity}}{\text{Time}}$$

where

i,j = initial and the ending nodes of a link ij

Cij = highway capacity of link ij

Vij = total undischarged traffic on link ij after the previous assignment iteration

Sij = traffic speed on link ij at previous assignment iteration

Lij = length of link ij

Iij = index of link ij (its unit = available kinetic energy per mile).

2. Route index:

$$Rin = \frac{Iij \times Lij + Ijk \times Ljk + \ldots + Imn \times Lmn}{Lij + Ljk + \ldots + Lmn}$$

where

Iij = index of link ij on a route from node i to node n

Rin = route(i,n) index (its unit = average potential increase of kinetic energy per mile).

3. Traffic assignment:

$$tinr = Tin \frac{Rinr}{Rin1 + Rin2 + \ldots + Rinm}$$

where

tinr = traffic assigned from origin i to destination n using route r

Tin = total traffic assigned from origin i to destination n for each incremental assignment.

The route index is the weighted link-conductance of links which comprise a route. It represents the attraction of a route compared with other routes. A route which is composed of better performed links will have higher route index. Higher route index means higher attraction and thus more trips assigned.

In addition, the link index used in this algorithm can be thought of as the possible increase in vehicle kinetic energy[Drew] per unit length of the link. Kinetic energy is the energy of motion of the traffic stream; its opposite is internal energy. The internal energy is called "acceleration noise," which is the standard deviation of acceleration. Internal energy is a measure of discomfort and potential hazard condition. Increasing the kinetic energy means decreasing the internal energy and decreasing the internal energy implies the reduction of the accident potential. Route index is the weighted possible increase in vehicle kinetic energy per unit length of the route. Since a transportation system reaches its optimal efficiency when it operates at maximum kinetic energy, trips are apportioned to routes according to their potential to increase the total system energy.

With the two possible approaches for building the multiple paths and the assignment algorithm for apportioning trips onto paths, the path-recording assignment model can be developed. The multiple paths are built in the first iteration only and, in different iterations, link and path index are updated to apportion the trips entering the network at that period. The advantage of the new model will be great time saving in an assignment simulation, and the drawback of the path-recording assignment is the loss of path-choosing flexibility since no more path will be added to load trips after the first assignment iteration. If the model validation proves that the loss of path-choosing flexibility is insignificant, the path-recording assignment will be able used in any case which requires a real-time assignment simulation, such as the one for evacuation operation.

4.0 TECHNICAL ASPECTS OF THE HEUPRAE MODEL

4.1 INTRODUCTION

One goal of this research is to develop a real-time assignment model for evacuation operation. The path-recording heuristic assignment method, discussed in Chapter 3, is employed to build the model. HEUPRAE, HEUristic Path Recording Assignment for Evacuation, is the model developed to simulate network traffic flow during evacuation. Although the HEUPRAE model is an assignment model designed specifically for real time evacuation operation, it can also be used for "four stage transportation planning." The developed assignment algorithm can be applied in any assignment model with or without the approach of path-recording. The development of the HEUPRAE model is based on the evacuation planning model MASSVAC2, developed at the Virginia Tech Civil Engineering Department. Using the MASSVAC2 model as a prototype and modifying its structure, the HEUPRAE model is developed using the new assignment algorithm and the path-recording approach.

4.2 MODEL STRUCTURE

To explain the framework of the HEUPRAE model, Figure 8 illustrates its components and their relationship.

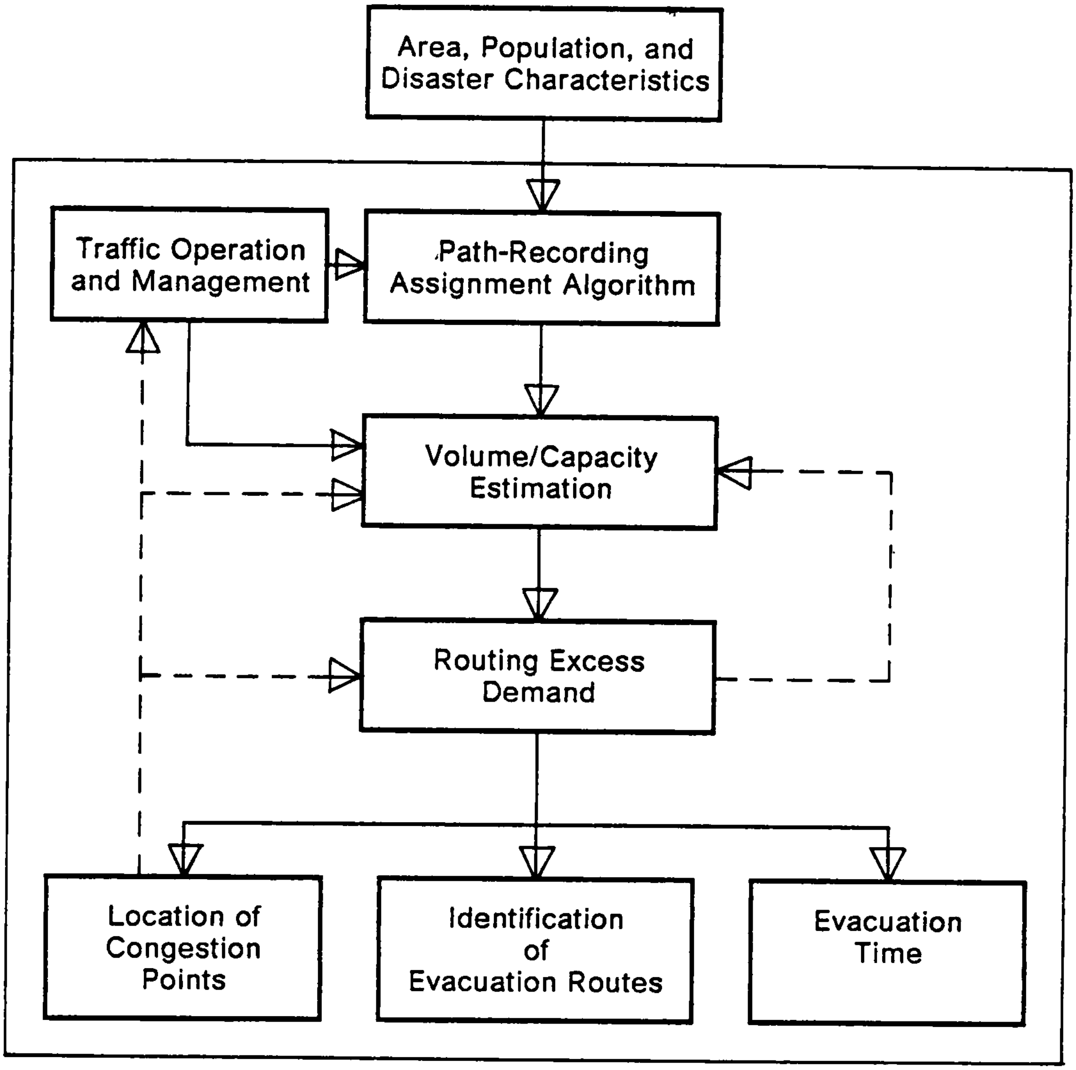


Figure 8. General Framework of the HEUPRAE Model

The model is basically a fractional assignment model. Evacuation demand is separated into different time-series intervals by the logit vehicle-loading curve discussed in Chapter 3. The trips in one interval are then loaded onto the network with one assignment iteration. For the first iteration, a warm up assignment period is used to provide link and route indices. For further iterations, the assignment is based on the link and route indices of the previous iteration. Travel time on each link and each evacuation route is updated and recorded after each iteration. The maximum evacuation time for vehicles at that interval is computed and provided. After that, link index and route index are updated before the assignment of next iteration. The iteration process then continues until no trips are left in the network.

4.3 MODEL INPUT AND OUTPUT

To meet the requirement of an evacuation model, input to and output of the model are carefully designed for the model's end user, an emergency officer, according to the information he can provide and needs. The required input for the HEUPRAE model includes two major sectors:

1. Community type and disaster characteristics: including the community type, such as urban or rural community; the population density classified by age and household; the car ownership; location of shelters and their capacities; the disaster type, such as flood, hurricane, or dam break; and the hazard area boundaries.

2. Highway network topology and traffic simulation strategies: including network geometry, characteristics, nodes, link distance, speed, existing volume, and capacity of each link. The traffic simulation strategies fed by the user include: expected evacuation time, macroscopic or microscopic simulation level, vehicle assignment parameters, assignment interval, risk factors, and traffic control management strategies.

The output of the HEUPRAE model varies depending on the options specified by the user. In general, the following data can be yielded:

1. Origin destination trip table: the trip table from origins to designated shelters, which is aimed to minimize the total vehicle-travel time and network evacuation time, is yielded at the end of each assignment loading interval and at the end of each simulation run.
2. Selected paths: the paths from any origin to the assigned shelter.
3. Congested links and network characteristics: volume/capacity ratio, link density, and link travel time are provided as network performance indices.
4. Network evacuation time: which is the evacuation time needed to evacuate people from the threatened area to shelters at each loading interval and for the total network to be cleared.

4.4 PROGRAM STRUCTURE

The HEUPRAE traffic simulation program consists of one main program and 13 subroutines. The structure of the whole model and the relationship of subroutines to the main program and to each other is shown in Figure 9. The flow charts of the main program and each subroutine are presented in Appendix C. A listing of the HEUPRAE model can also be found in Appendix E. The function of each subroutine is described in the following sections.

4.4.1 MAIN PROGRAM

The basic function of the main program is to call the execution of the subroutines INITIAL, ODTAB, and EVAC respectively, and to properly instruct the flow of the model. Each subroutine executes only once and the assignment iteration for each time interval is controlled by subroutine EVAC.

4.4.2 SUBROUTINE INITIAL

The INITIAL subroutine mainly reads the input for the program and sets the default variable value. The inputs are:

1. The required coding data for the highway network: the link data represented by initial node and ending node, link length, speed limit, capacity, number of origin and destination, etc.
2. The parameters for vehicle assignment: the expected evacuation time, interval of the iteration, value of the diversion parameter θ for computing link-likelihood, etc.
3. The program execution options: the option to run the model in macroscopic or microscopic level; the option about the O-D trip table, whether it is to be entered externally or to be calculated internally; the options about the traffic management strategies; and the options to produce output statistics.

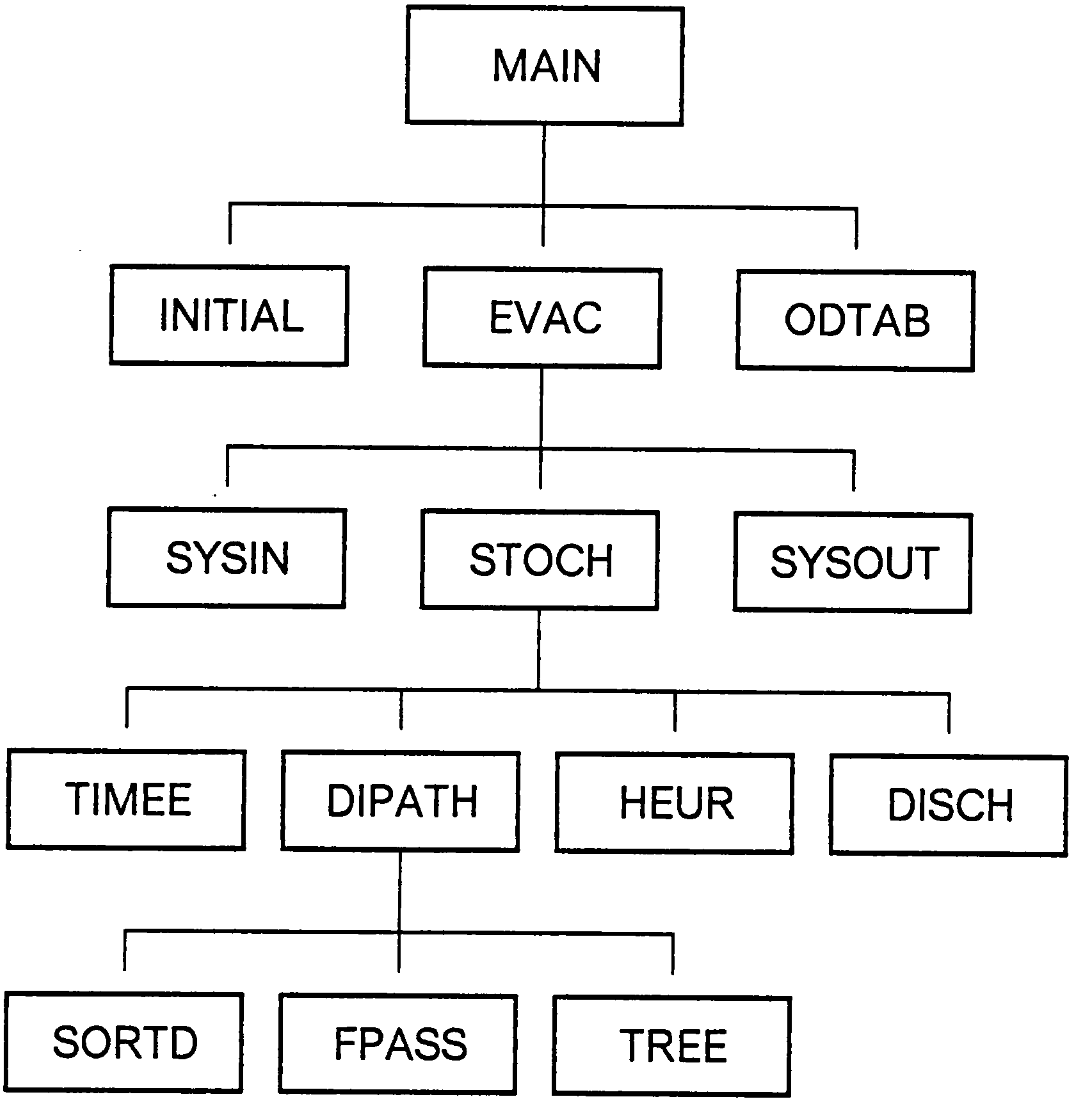


Figure 9. Structure of the HEUPRAE Model

4.4.3 SUBROUTINE ODTAB

According to the option given in the subroutine INITIAL, read the O-D trip table or determine the O-D table for further program execution. The O-D table may be internally determined according to the production of each origin and the capacity of shelters with the objective of minimizing the total vehicle travel time and the network evacuation time. The approach to determining the O-D table is a heuristic iterative procedure. Average travel time for each link during assignment process is estimated first. The O-D table travel time is computed and the minimum O-D travel time is chosen, then trips are filled into that O-D cell. After the trips are subtracted from the origin production and shelter capacity, the next minimum travel time is chosen and trips are allocated again. This procedure continues until all trips are allocated and the O-D table is decided.

4.4.4 SUBROUTINE EVAC

This subroutine controls the execution of the traffic assignment for evacuation demand during each time interval. It calls execution of the subroutines SYSIN, STOCH, and SYSOUT repeatedly until all trips have arrived at their destinations; the program then stops.

4.4.5 SUBROUTINE SYSIN

This subroutine has two major functions. Its first function is to prepare the evacuation demand for each time interval. The cumulative evacuation demand follows a logit curve. The part of the evacuation demand which cannot be loaded onto the network for the current time interval because of the congested routes will be stored and assigned at the next interval. The subroutine's second function is to check if all trips are loaded. If all trips have been loaded onto the network, SYSIN will clear the vehicles which are still left in the network and compute the network clearance time.

4.4.6 SUBROUTINE STOCH

This subroutine performs the actual simulation of the HEUPRAE model. It controls the execution of the four subroutines DIPATH, HEUR, TIMEE, and DISCH. Its flow is to build the multiple-path trees in the first assignment iteration, compute route index and assign trips to routes according to the heuristic assignment algorithm for the second and later iterations, update travel time on each link after trips are loaded, compute evacuation route travel time, and discharge link traffic. The detailed sequence of steps followed in the subroutine are described below:

1. It calls subroutine DIPATH to assign each link an assignment-likelihood and to build the multiple paths for each O-D pair with the links of positive assignment-likelihood. Subroutine DIPATH is executed for the first iteration only.
2. Subroutine HEUR is called to compute link index and route index; then it assigns trips for each interval onto the multiple paths.
3. After trips are assigned onto routes and traffic on each link is updated, subroutine TIMEE is called to update link travel time.
4. The travel time on each evacuation route is computed and compared to obtain the maximum evacuation time for vehicles leaving at this interval.
5. Vehicles on each link are removed from the system. If a link has traffic volume greater than its capacity, the amount greater than the capacity will be stored as the initial volume for the next assignment interval.

4.4.7 SUBROUTINE SYSOUT

This subroutine mainly functions to provide system information during the whole simulation process. It can provide system performance for each simulation interval or at the end of the simulation. The information includes:

1. The amount and percentage of the evacuation demand assigned at each interval.
2. The amount and cumulative percentage of the evacuation demand which has been assigned after each interval.
3. The total vehicle-travel time at each interval.
4. The link characteristics, such as V/C, travel time, and speed, at each interval.
5. The evacuation route and route travel time.

4.4.8 SUBROUTINE DIPATH

This subroutine applies Dial's concept of diversion parameter to assign each link an assignment-likelihood and builds each O-D pair multiple paths for assignment. It first calls subroutine TREE to build the minimum-path tree and to find the minimum travel time from each node to the origin node. Links in the minimum-path tree are assigned an assignment-likelihood of one unit. Links outside the minimum-path tree are given a likelihood less than one unit or zero. Subroutine FPASS is then called to build the multiple paths for each O-D pair with links of positive assignment likelihood. Subroutine DIPATH is executed for the first assignment iteration only.

4.4.9 SUBROUTINE HEUR

This subroutine uses the proposed heuristic assignment algorithm to compute link index and route index. Each path in the multiple-path tree is given an assignment weight (path index), and trips are loaded on the paths according to their comparative weights. The volume on each link assigned at each interval, as is its cumulative volume, is recorded for printing in subroutine SYSOUT.

4.4.10 SUBROUTINE TIMEE

This subroutine functions to update link travel time based on link type and traffic simulation option (macro or micro level). In the macroscopic level, the travel time is updated using appropriate curves which relate traffic flow to travel time of a link[Jamei]. The total travel time is made up of the intersection delay and the service time.

For microscopic level of simulation, major streets are divided into four classes depending primarily on prevailing speed and the number of signaled intersections per mile, which is discussed in Jamei's dissertation[Jamei]. At the microscopic level, the delays on links with congestion are determined using Webster's model[Webster], and then the travel times are updated.

4.4.11 SUBROUTINE DISCH

The main function of this subroutine is to dissipate traffic volume on each link. A detailed description of the subroutine is given below:

1. Traffic on noncongested links will all dissipate and leave the capacity space for assignment iteration at the next interval.
2. To discharge traffic at signaled intersections, this subroutine uses the relationship (V1/(V1+V2))*1800/60 to compute hourly dischargeable volume and multiplies this

by time interval, which is in minute unit. The saturation flow rate is assumed to be 1800 veh/hr in this research.

3. For freeways and expressways, the Laporte regression linear model for mixed vehicles modified by Kalevela (U=74.3-.75K) is used to determine the flow of traffic[Kalevala].

4. The vehicular volume which can not dissipate because of link or intersection capacity constraint will be stored on the link as the initial volume at next iteration.

The dissipation of vehicular volumes mentioned above is basically at the macroscopic level of simulation. For the microscopic level, the same procedure is applicable except for 2-way stop signs and 4-way stop signs. The number of vehicles to be discharged is determined using the gap acceptance technique for a 2-way stop sign. For a 4-way stop sign, up to 900 vehicles per hour are allowed to be discharged.

4.4.12 SUBROUTINE TREE

This subroutine uses Dijkstra's shortest-path algorithm[Dijkstra] to find the minimum impedance from an origin to all the nodes. It also helps determine the O-D travel time for subroutine ODTAB when determining the O-D trips table internally is required. For each origin node, this subroutine is executed once to help build the multiple-path tree for heuristic assignment.

4.4.13 SUBROUTINE FPASS

The subroutine uses the multiple-path algorithm, discussed in Chapter 3, to build each O-D pair their multiple paths for assignment. The smaller the diversion parameter, the more paths will be built and used for assignment.

4.4.14 SUBROUTINE SORTD

This subroutine is a completely technical one to help sort the links with the same ending node together in subroutine INITIAL. Since each link is identified by its two nodes, putting links with same ending node together will save computer time in searching for a near by link. In addition, this subroutine helps sort nodes in an ascending sequence according to their travel times to the origin in subroutine DIPATH. This sequence helps subroutine DIPATH compute link likelihood, beginning with the node closer to the origin first.

4.5 TECHNICAL SECTION SUMMARY

The model described in this chapter is written in FORTRAN 77. The model is structurized so that it is very easy to add or change for other purposes. It is compiled under VS FORTRAN with 1536 kilobytes of memory required to execute the

FORTRAN complier. The program also runs on an IBM personal computer. The computer memory needed to run the entire program in an IBM-PC is 640 RAM (random access memory).

In the next chapter, the HEUPRAE model is validated by comparing it with other existing models. Also, a sensitivity analysis is done on different variables to test their influence on an evacuation system. In addition, the efficiency of this heuristic model will be investigated.

5.0 MODEL TESTING AND COMPARISON

5.1 INTRODUCTION

In this chapter, the HEUPRAE model together with the proposed assignment algorithm will be tested and compared with other assignment models. Dial's multiple-path assignment algorithm is used for comparison with the new assignment algorithm and the path-recording assignment approach. The output from the evacuation planning model MASSVAC2, which employs Dial's assignment method, is used as a base to compare with the output from the HEUPRAE model. Testing of the assignment algorithm and the HEUPRAE model includes link flow, number of routes used for each O-D pair, vehicle-travel time, and the computer execution time.

In addition, before the HEUPRAE model is compared, the MACROVAC submodel should be evaluated, since the MACROVAC model provides the estimate of evacuation time as an input to the HEUPRAE model.

5.2 EVALUATION OF THE MACROVAC MODEL

Evaluation and testing of the MACROVAC model can be divided into two parts: 1. examining the mathematical deducing of the model and 2. checking the accuracy of the estimated evacuation time.

The mathematical deducing of the MACROVAC model has been investigated in Chapter 3. The input-output concept in MACROVAC for processing vehicle arrival and departure is commonly applied in traffic counting for a highway segment. Also, the system dissipation rate is computed according to the Highway Capacity Manual[HCM, 1985]. Regarding evacuation-time estimation, the MACROVAC model applies the same approach as the HCM Method reviewed in Chapter 2, except that the MACROVAC model involves a logit loading curve instead of the constant vehicle-loading in the HCM Method. Thus, there is no doubt about the concept and structure of the model.

To test the accuracy of the MACROVAC model, the evacuation time from the MASSVAC2 model is used as the base value for comparison since field data about evacuation time for a specific network are rare. Three real networks and a hypothetical grid network are chosen as comparison networks. The three networks are the city of Blacksburg, the region one, and region two of the city of Virginia Beach. For each real network, the evacuation demand and supply are estimated according to the social-economic characteristics and the highway geometry. The evacuation demand and supply are assumed for the hypothetical network.

Input and output from the MACROVAC and MASSVAC2 model are presented in Table 2.

Table 2. Evacuation Time Estimation

	BLACKSBURG NETWORK	HYPOTHETICAL NETWORK
Evacuation Demand	8,000 vehicle trips	20,000 vehicle trips
Dissipation Capacity	8,000 veh/hr	8,000 veh/hr
	Evacuation Time	
MACROVAC	121.5 Min	300.9 Min
MASSVAC2	134.5 Min	284.8 Min
Difference	-13.0 Min (-9.7%)	16.1 Min (5.7%)
	VIRGINIA BEACH REGION 2	VIRGINIA BEACH REGION 3
Evacuation Demand	10,977 vehicle trips	13,018 vehicle trips
Dissipation Capacity	3,600 veh/hr	8,900 veh/hr
	Evacuation Time	
MACROVAC	356.0 Min	170.8 Min
MASSVAC2	401.5 Min	187.3 Min
Difference	-45.5 Min (-11.3%)	-17.3 Min (-9.2%)

The output from the MACROVAC and MASSVAC2 models show the difference in estimating evacuation times for the four tested cases are -9.7%, 5.7%, -11.3%, and -9.2% respectively. They are all less than 12%.

Since the MACROVAC model employs the assumption that evacuees are equally dissipated through the evacuation routes according to route capacity, the evacuation-time estimation from the MACROVAC model is less than that from the MASSVAC2 model, which does not employ such an assumption. In addition, the MACROVAC model tends to underestimate evacuation time since it does not consider travel time between origin-destination pairs. However, Table 2 shows that the difference in evacuation-time estimation is insignificant. MACROVAC seems to be a reliable model in predicting evacuation time.

5.3 COMPARISON OF THE ASSIGNMENT ALGORITHM

Before comparing the HEUPRAE model, the new assignment algorithm must be examined to learn its assignment capability. Since field validation is difficult to achieve, Dial's algorithm is used as the comparison base and two networks are initially chosen for studying the difference between these two algorithms in assignment. One real network is part of the city of Virginia Beach, the city's second disaster region. The highway network of this studied region is illustrated on Figure 10. Another test network is a hypothetical network with a radial shape. Its highway geometry is shown in Figure 11.

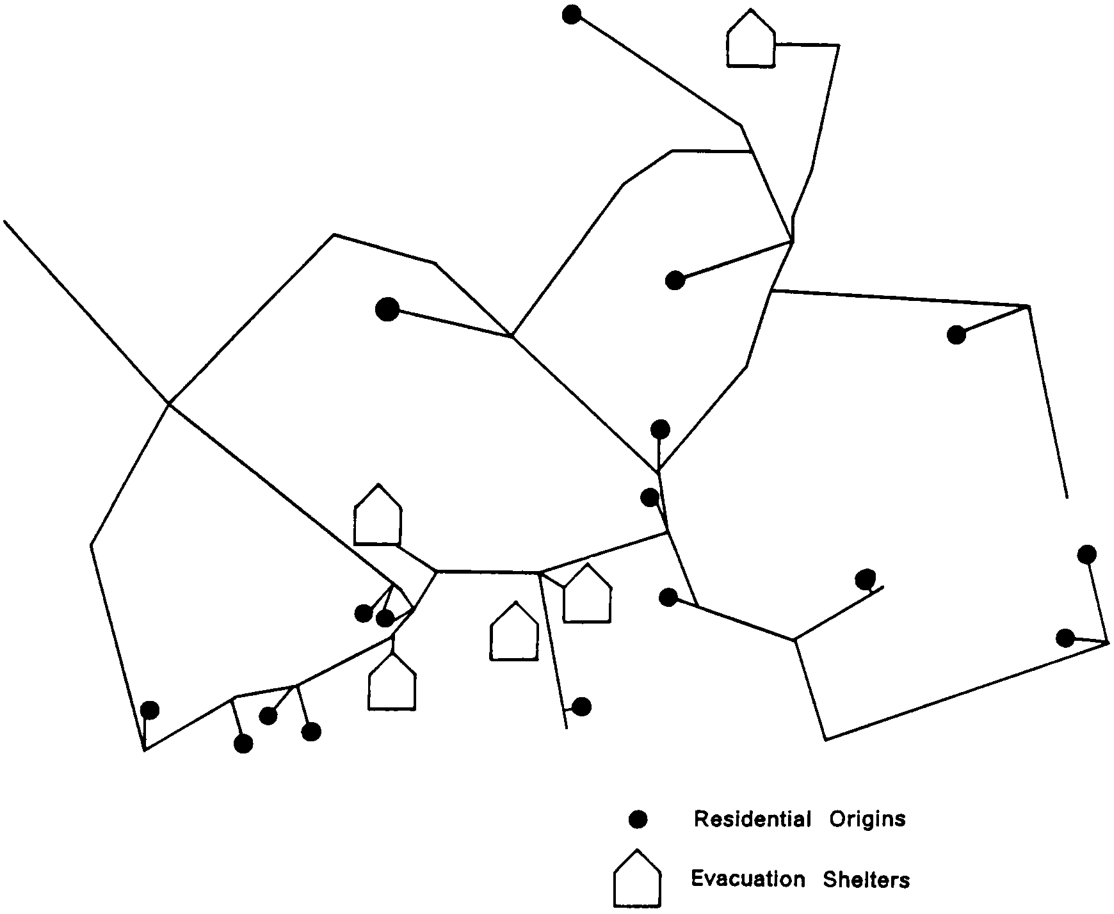


Figure 10. Virginia Beach Region 2 Highway Network

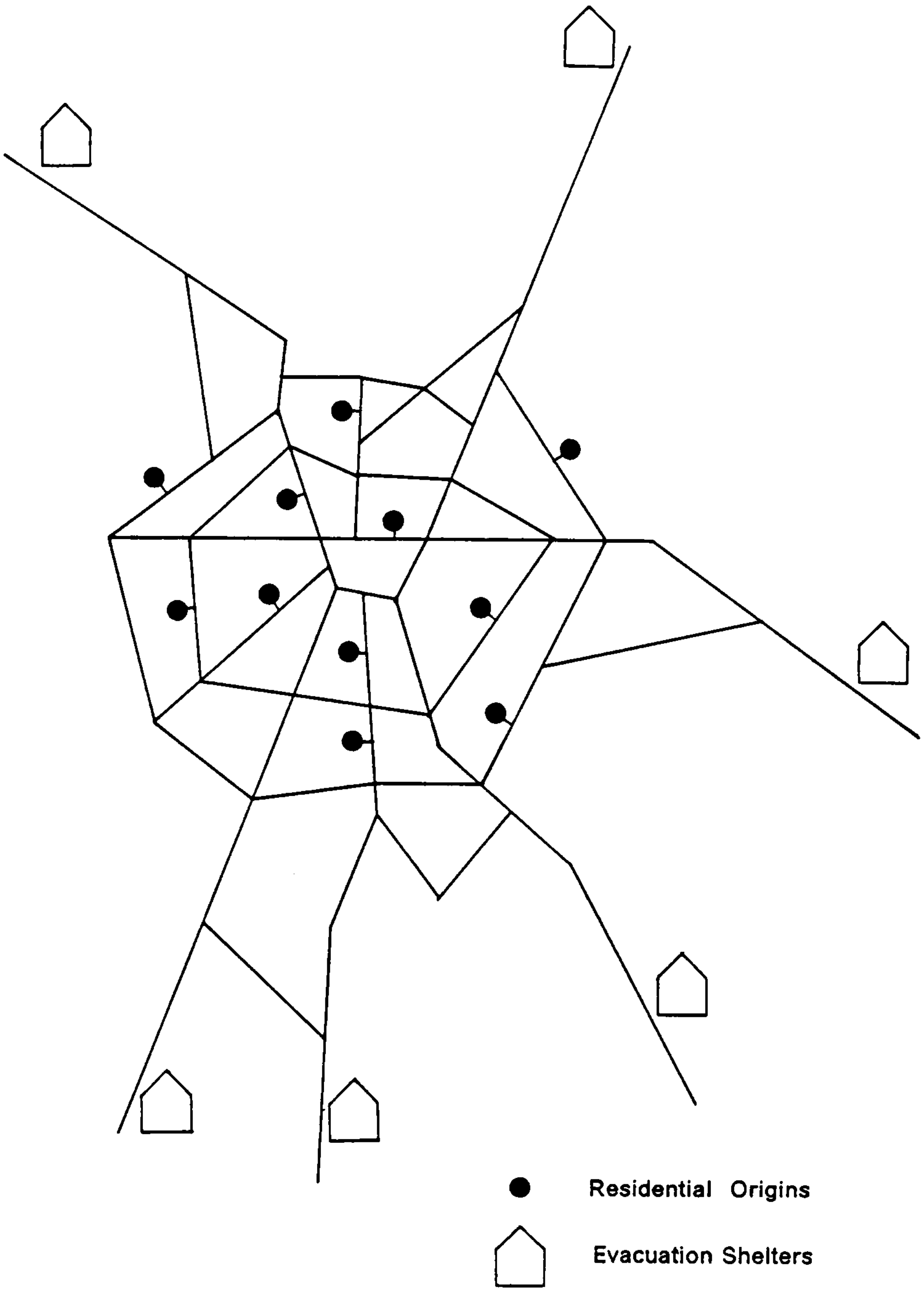


Figure 11. A Radial Hypothetical Highway Network

The number of links in these networks are 118 and 192 respectively. During the comparison, a number of traffic demands are loaded onto the network by Dial's algorithm. Then the same traffic demands are loaded by the new algorithm, which is described in Chapter 3, in another run. To test the difference in traffic loading by the algorithms, the assignment algorithm is executed in all iterations. In addition, the value 1.5 for the diversion parameter θ is given. Summary of the comparison is shown in Table 3.

It can be seen in Table 3 that both algorithms produce the same number of links with positive volume in the two networks. This is because Dial's forward pass is used to build the assignment paths in the model which employs the new assignment algorithm. The same paths are used to load trips in the comparison except by different assignment algorithms. Therefore, the difference in link volumes is caused by the difference of the algorithms in apportioning trips.

Table 3. Algorithm Performance Comparison

	VIRGINIA BEACH NETWORK	HYPOTHETICAL RADIAL NETWORK
Number of Links	118 links	192 links
Links with Positive Assigned Volume[1]	46 (46) links	114 (114) links
Average % of Volume Difference[2]	12.80% <18.39%>	2.35% <6.10%>
Range of Volume Difference %	-27.1% to 31.5%	-17.5% to 18.8%
Total Vehicle Travel Time	265,009.6 (276,664.3) Minutes	205,783.7 (208,717.8) Minutes
Difference in Total Travel Time	11,654.7 (4.40%)	2,934.1 (1.43%)

[1] The value in ( ) is from the new assignment algorithm.

[2] The value in < > does not consider dummy links.

The comparison shows that the difference in link volumes between two algorithms reaches as high as -30% to 30%. The difference is significant. The algorithms simply do not assign equal amount of trips to the same links. However, the differences in the total vehicle-travel times are only 4.40%

and 1.43%; Dial's algorithm results in lower total travel time, but the difference is not significant. It is thus concluded that the new assignment algorithm is different from Dial's algorithm in links using, but provides similar network travel time.

Since network equilibrium is evaluated by total vehicle-travel time, the new assignment algorithm appears to have the same capability as Dial's algorithm in apportioning trips to provide an equilibrium assignment. Dial's algorithm is not the only equilibrium assignment algorithm. However, Dial's algorithm tends to result in lower travel times since its objective is to minimize vehicle-travel time in assignment, but the new algorithm also considers link capacity.

5.4 COMPARISON OF THE HEUPRAE MODEL

After equilibrium comparison of the new assignment algorithm, the path-recording approach used to develop the HEUPRAE model is tested. The MASSVAC2 model, which employs Dial's algorithm to assign trips in each iteration, is borrowed to test the HEUPRAE model. The two networks used to test the assignment algorithm are continued as the test-base. The items to be compared are: links with positive volume, paths for O-D pairs, total vehicle-travel time, network-clearance time, and computer-execution time. Since the emergency manager of an evacuation area mainly wants to know the paths which can minimize network-clearance time and total vehicle-travel time, those are the major concerns in model comparison. After model comparison, computer-execution times are examined to prove that the HEUPRAE model with the path-recording assignment approach is more efficient than the MASSVAC2 model.

In Dial's assignment, value of the diversion parameter 'θ' affects the link assignment-likelihood and thus the number of paths for assignment. 1.5 is initially used as the value of θ.

Since there are two approaches to build the multiple paths in the HEUPRAE model, the model is called HEUPRAE-1 or HEUPRAE-2 to identify the approach employed. The HEUPRAE-1 model uses the multiple-path algorithm to build paths and assign trips. The HEUPRAE-2 model does Dial's assignment iteration in its first iteration and traces the paths for assignment in later iterations. The major difference in HEUPRAE-1 and HEUPRAE-2 is the assignment of the first iteration and the computer memory required to run the models.

Comparisons of the HEUPRAE model and MASSVAC2 model are presented in Table 4.

Table 4 (Part 1 of 2). Model Output Comparison

	VIRGINIA BEACH NETWORK	HYPOTHETICAL RADIAL NETWORK
Links with Positive Assigned Volume	46 (40) links	114 (98) links
Difference in Positive Volume Links	6 (13.04%)	16 (14.04%)
Average % of Volume Difference[1]	15.91% <20.07%>	12.11% <29.92%>
Range of Volume Difference %	-38.2% to 45.9%	-86.9% to 74.7%
Number of O-D pairs	11	15
Number of Paths[2] [3]	14 (14) [14]	45 (45) [45]
Total Vehicle Travel Time	265,009.6 (434,592.6) [434,722.1]	260,203.4 (257,102.7) [256,696.1]
Difference in Total Travel Time	169,583.0 (64.00%), 169,712.5 [64.04%]	-3,100.7 (-1.19%), -3,507.3 [-1.35%]
Network Clearance Time	401.5 (386.6) [386.6]	235.3 (214.9) [214.9]

Table 4 (Part 2 of 2). Model Output Comparison

	VIRGINIA BEACH NETWORK	HYPOTHETICAL RADIAL NETWORK
Difference in Network Clearance Time	-14.9 (-3.71%), -14.9 [-3.71%]	-20.4 (-8.67%), -20.4 [-8.67%]

[1] The value in < > does not consider dummy links.

[2] The value in ( ) is from the HEUPRAE-1 model.

[3] The value in [ ] is from the HEUPRAE-2 model.

5.4.1 NUMBER OF LINKS WITH POSITIVE VOLUME

It can be seen in both networks that the number of links with positive volume are higher in the MASSVAC2 model than in the HEUPRAE model. This is because the MASSVAC2 model employs Dial's assignment in each iteration and different paths may be used for assignment in different iterations. However, the differences of links with positive volume in both networks are only 6 links (13.04%) and 16 links (14.04%). Fewer than 15% of links are added to load trips in assignment with Dial's algorithm in each iteration. The difference is small and implies that the links kept for the path-recording assignment are very close to those in Dial's assignment.

It is thus concluded that the path-recording assignment in the HEUPRAE model fairly resembles the equilibrium assignment of Dial's in the test networks in terms of link usage. The lower the difference in number of links with positive volume, the more the path-recording assignment resembles a Dial's assignment.

5.4.2 COMPARISON OF LINK VOLUMES

In addition to Table 4, Figures 12 and 13 illustrate the link-volume comparisons of the two test networks respectively. It can been seen that the difference of the assigned link volumes between the MASSVAC2 and HEUPRAE models is significant. It ranges from -38.2% to 45.9% in one network and from -86.9% to 74.7% in another network. These numbers are higher than the associated numbers in Table 3, where assignment algorithms are compared. The average percentage of link-volume differences is also higher in the comparison of the models than in the comparison of the algorithms. This tendency of greater difference in the comparison of the models implies that the new links employed to load trips in the second or later iterations are crucial in Dial's assignment. New links are assigned with relatively higher assignment-likelihoods and thus more trips are assigned. Since the path-recording approach is the only difference between algorithm-comparison and model-comparison, the increased difference from algorithm-comparison to model-comparison is accredited to the path-recording approach.

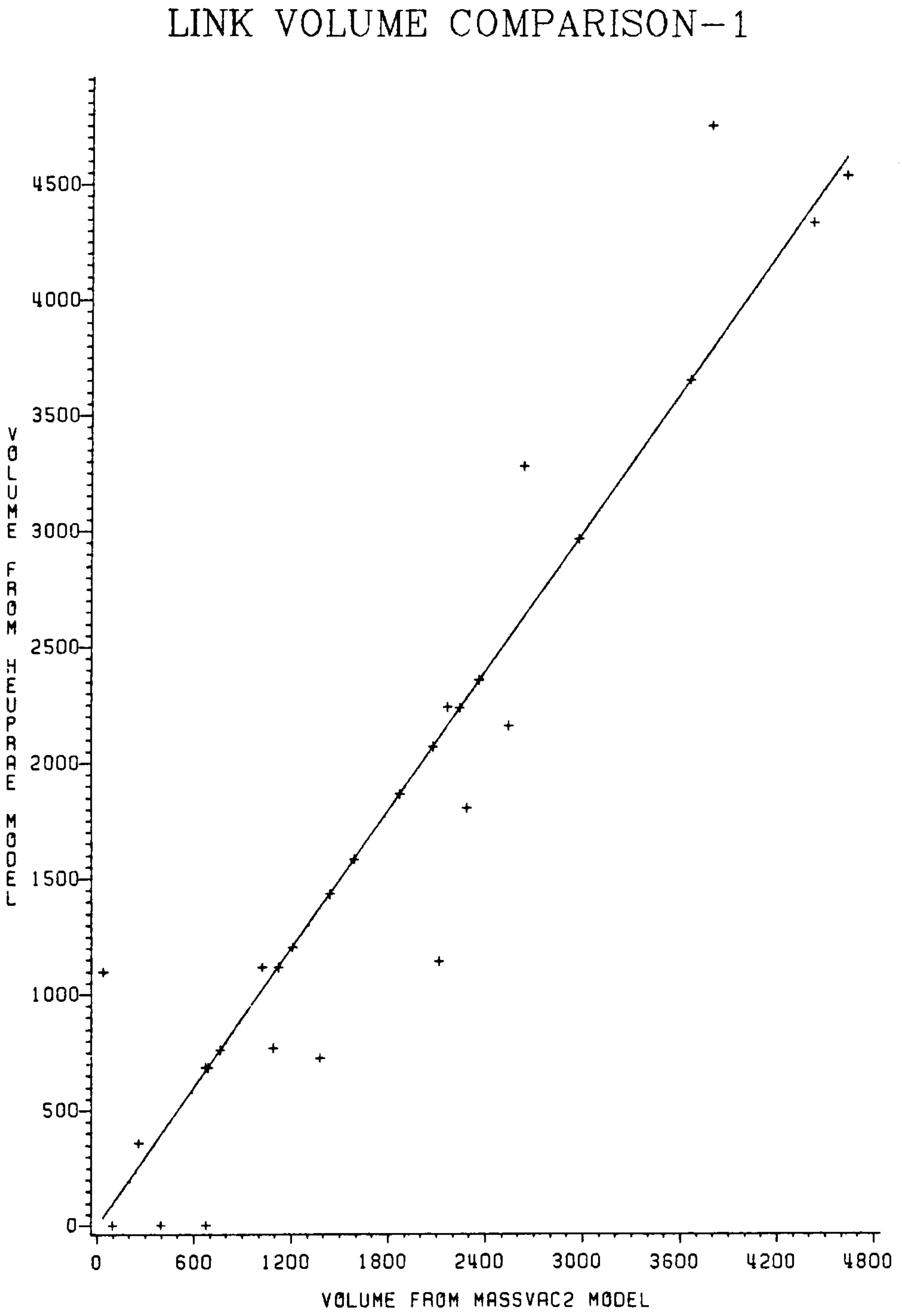


Figure 12. Link-volume Comparison 1 (Va. Beach)

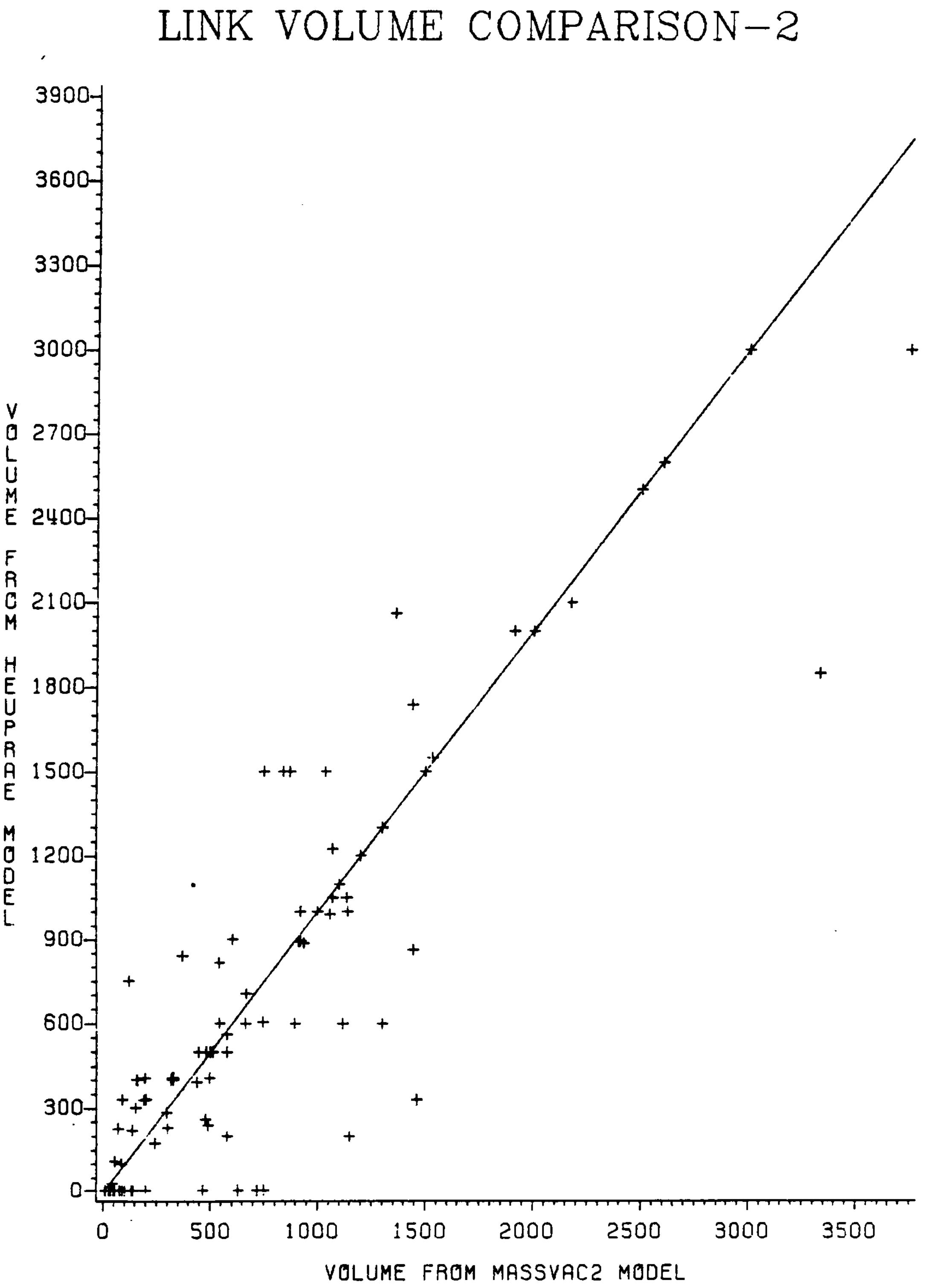


Figure 13. Link-volume Comparison 2 (Hypothetical)

In the path-recording assignment, since the number of links to load trips is a constant and no more links or paths will be added after the first iteration, the increased difference in link volumes using this approach is self-explanatory. The fewer links added, the better the results produced by this assignment. The more paths used in the assignment, the better an equilibrium with lower vehicle-travel time will be achieved. This conclusion is consistent with that in the previous section, which compares links with positive volume.

5.4.3 NUMBER OF ASSIGNMENT PATHS

Table 4. shows that the number of assignment paths used to connect the origin and destination centroids are 14 and 45 respectively in the test networks. In both networks, the HEUPRAE model uses as the MASSVAC2 model the same number of assignment paths for assignment. However, the same number of assignment paths does not imply the same paths; otherwise both models should have the same number of links with positive volume. The same number of paths only implies that Dial's assignment does not increase, or increases very little in some cases, the assignment paths for trips loading in different iterations. This implication helps prove that the approach of the path-recording assignment is a valid assignment approach.

5.4.4 NETWORK CLEARANCE TIME

The comparison of the network-clearance time in Table 4 shows that the HEUPRAE model provides close results to the MASSVAC2 model in both test networks. The differences are only -3.7% and -8.7%. Since network clearance time is the time of the last vehicle leaving the evacuation network, it is computed as adding the path-travel time of the last vehicle to the clock time when this vehicle leaves its origin. The fact that outputs are close to each other means that the last vehicle leaves its origin at the same iteration in the simulation of both models, and that the network has similar traffic conditions in both models before and at the last iteration. It also implies that path-travel times at the last iteration are very similar.

The HEUPRAE model is thus believed to be as reliable a model in providing network-clearance time as the MASSVAC2 model.

5.4.5 TOTAL VEHICLE-TRAVEL TIME

"Total vehicle-travel time" is the summation of the path-travel time of each individual evacuation vehicle. Table 4 shows that both models result in similar total vehicle-travel times for the hypothetical radial network. The difference in percentages are 1.19% and 1.35% for HEUPRAE-1 and

HEUPRAE-2 respectively. In addition, the total vehicle-travel time for the HEUPRAE model is shorter than that for the MASSVAC2 model. Although, the amount is insignificant, it implies that the path-recording assignment (the HEUPRAE model) can simulate an equilibrium assignment as well as Dial's assignment (the MASSVAC2 model).

However, in comparison, the Virginia Beach network shows a very unsatisfactory result. The difference is as large as 64.04% and implies that the path-recording approach assigns more trips to congested links than does Dial's assignment. This result can be explained partly by the difficulty of adding new links in the path-recording assignment and partly by the low path/pair ratio. The path/pair ratio is the average number of paths kept for each origin-destination pair in assignment iterations. Lower path/pair ratio implies fewer paths are used for trips loading in assignment. Comparison of Table 3 and 4 shows that the difference of total vehicle-travel time is only 4.40% in algorithm-comparison when new links can be involved to load trips, but as large as 64.04% in model-comparison when new links cannot be involved to load trips. The six extra links used to load trips in Dial's assignment in Table 4, therefore, are crucial to relieving the traffic congestion.

Another reason is the low path/pair ratio of the Virginia Beach network. Compared to the path number and O-D pair number in the hypothetical network, 45/15, the Virginia

Beach network shows a relatively low ratio, 14/11. This low ratio restrains the available paths in path-recording assignment; it is almost similar to an "all-or-nothing" assignment. However, this low ratio does not affect Dial's assignment, since the 14 paths may change in different iterations. In the path-recording assignment, trips are continuously loaded onto those 14 paths; total vehicle-travel time thus increases considerably. For the hypothetical network, each O-D centroid pair has an average of three paths for loading trips, the HEUPRAE model thus provides a satisfactory equilibrium result even though paths do not change in different iterations.

It is thus concluded from the comparison of total vehicle-travel time that the HEUPRAE model requires a substantial number of paths to efficiently simulate an equilibrium assignment. The more paths involved, the better equilibrium assignment may be achieved.

5.4.6 COMPUTER EXECUTION TIME COMPARISON

The main reason to apply the path-recording approach in an assignment is to develop a model for cases requiring real-time assignment simulation, such as an evacuation operation. The comparison of model-execution time for the path-recording assignment and Dial's assignment is well suited to prove that the path-recording assignment is more efficient.

Since the HEUPRAE model is going to be used in a decision support system on a personal computer for evacuation, the comparison of model-execution time is done on a PC. An IBM-PC XT with 640k random access memory (RAM) and a 8087 microprocessor is employed to perform this task. Both the MASSVAC2 and HEUPRAE models are fixed with the same input and output options to make the comparison as reliable as possible. The two test networks are used in this task again and their O-D trips tables are provided externally. The results are illustrated in Table 5.

Table 5 (Part 1 of 2). Computer Execution Time

	VIRGINIA BEACH NETWORK	HYPOTHETICAL RADIAL NETWORK
Number of Links	118 links	192 links
Number of Iterations	12	6
	Computer Execution Time[1] - seconds	
MASSVAC2[2]	307.5 (25.6)	316.5 (52.8)
HEUPRAE	139.5 (11.6)	135.5 (22.6)
Difference in Execution Time	168 (14.0)	181 (30.2)

Table 5 (Part 2 of 2). Computer Execution Time

	VIRGINIA BEACH NETWORK	HYPOTHETICAL RADIAL NETWORK
Difference %	54.6%	57.2%

[1] The computer execution time here includes 4 seconds program loading time.

[2] The value in ( ) is execution time per iteration.

The comparison of model-execution time shows that the path-recording assignment outperforms Dial's assignment significantly. The saving of computer-execution time reaches as high as 57.2%. Since loading the MASSVAC2 or HEUPRAE model into the computer system only takes four seconds and the O-D trip tables are provided externally, the computer-execution time in Table 5 can be considered used only for assignment iteration. The Virginia Beach network is simpler than the hypothetical network, 118 links compared to 192 links; thus its execution time per iteration is shorter than that of the hypothetical network.

Because both test networks are small networks, the computer-time savings are only 168 and 181 seconds respectively. These savings may seem unimportant. However, the saving percentage shows that a great amount of computer time can be saved by using the path-recording approach to deal with the assignment for a large network. In addition, the saving of the computer-execution time by using the path-recording approach implies that the execution of shortest-path algorithm in an assignment takes about 50% of the assignment execution time.

5.4.7 COMPUTER MEMORY COMPARISON

As discussed in Chapter 3, two approaches can be used to build the multiple-path tree for path-recording assignment. The first approach uses a multiple-path algorithm and requires greater memory in its execution. The HEUPRAE model built with this approach is named HEUPRAE-1. The second approach traces the paths used in Dial's assignment to form the multiple-path tree; it is more memory efficient. The HEUPRAE model built with this second approach is named HEUPRAE-2. These two approaches do not differ in their assignment, only in the memory required for model execution. Table 6 shows the memory required to program these two models with the capabilities of handling a network which has less than 350 links. The comparison is done on an IBM PC-XT with Microsoft 3.31 Fortran Compiler [Microsoft, FORTRAN]. Each real variable occupies four bytes of memory and each integer variable is assigned with only two bytes to reduce the memory requirement.

Table 6. Computer Memory Comparison

	HEUPRAE-1	HEUPRAE-2
Memory Required	240.8 kilobytes	325.2 kilobytes

The comparison shows that the HEUPRAE-2 model requires 84.4 kilobytes (26.0%) less than the HEUPRAE-1 model. The memory saving may be partially the result of programming skill, but the significant difference shows the second approach is more memory efficient. This conclusion is consistent with the discussion of building multiple-path in Chapter 3.

5.5 MODEL-SENSITIVITY ANALYSIS

To understand the degree to which each variable affects an evacuation system, a variable-sensitivity analysis is preformed. Some sensitivity analysis for the MASSVAC model can be found in Dr. Jamei's dissertation, since this research is an extension of his study[Jamei]. This section will mainly concentrate on the impact of the loading parameter to the evacuation time and the impact of the diversion parameter to the number of assignment paths. The loading parameter which affects the evacuation rate, together with the evacuation time will be investigated first. The diversion parameter will be studied second.

5.5.1 LOADING PARAMETER ANALYSIS

As can be seen in equations 3.1 and 3.5, the loading parameter (LP) affects the evacuation time and the distribution of the loading demand at each evacuation period. The loading rate, the rate of evacuation demand entering the evacuation system, has the relationship with LP described in equation 3.2. For a system with constant evacuation demand and evacuation route capacity, it is also found that the maximum loading rate has the following relationship with LP:

$$MVLR1 = MVLR2 \times LP1/LP2 \qquad 5.1$$

where MVL = maximum vehicle-loading rate.

Equation 5.1 shows that the larger the LP, the more the evacuation demand is concentrated. From equation 3.5, the relationship between evacuation time and LP is found to have the form:

$$ET1 = ET2 \times LP2/LP1 \qquad 5.2$$

where ET = evacuation time.

It can be seen from equation 5.2 that the evacuation time is inversely proportional to the LP. Except when the system gets overcongested, the smaller the LP, the longer is the evacuation time expected. Figure 14 illustrates the relationship between the evacuation time and the loading parameter. For a system with the evacuation demand and the evacuation route capacity equal to 15000 vehicle-trip and 4000 veh/hr respectively, the critical evacuation time (CET) and the minimum evacuation time (MET) are 437.8 minutes and 260.0 minutes respectively. According to the definition of the critical evacuation time in Chapter 3, if the issued evacuation time is smaller than the CET, Part B of the Figure 14, the evacuation system will get congested. If the issued evacuation time is greater than the CET, part A of the Figure 14, the evacuation will not result in congestion.

5.5.2 CRITICAL EVACUATION TIME ANALYSIS

As can be seen in equation 3.6, the critical evacuation time (CET) is proportional to the evacuation demand (TV) and is inversely proportional to the evacuation route capacity (MDR). The relationship between CET, TV, and MDR is illustrated in Figure 15. The greater the evacuation demand, the longer is the evacuation time expected. The greater the evacuation route capacity, the shorter is the evacuation time expected.

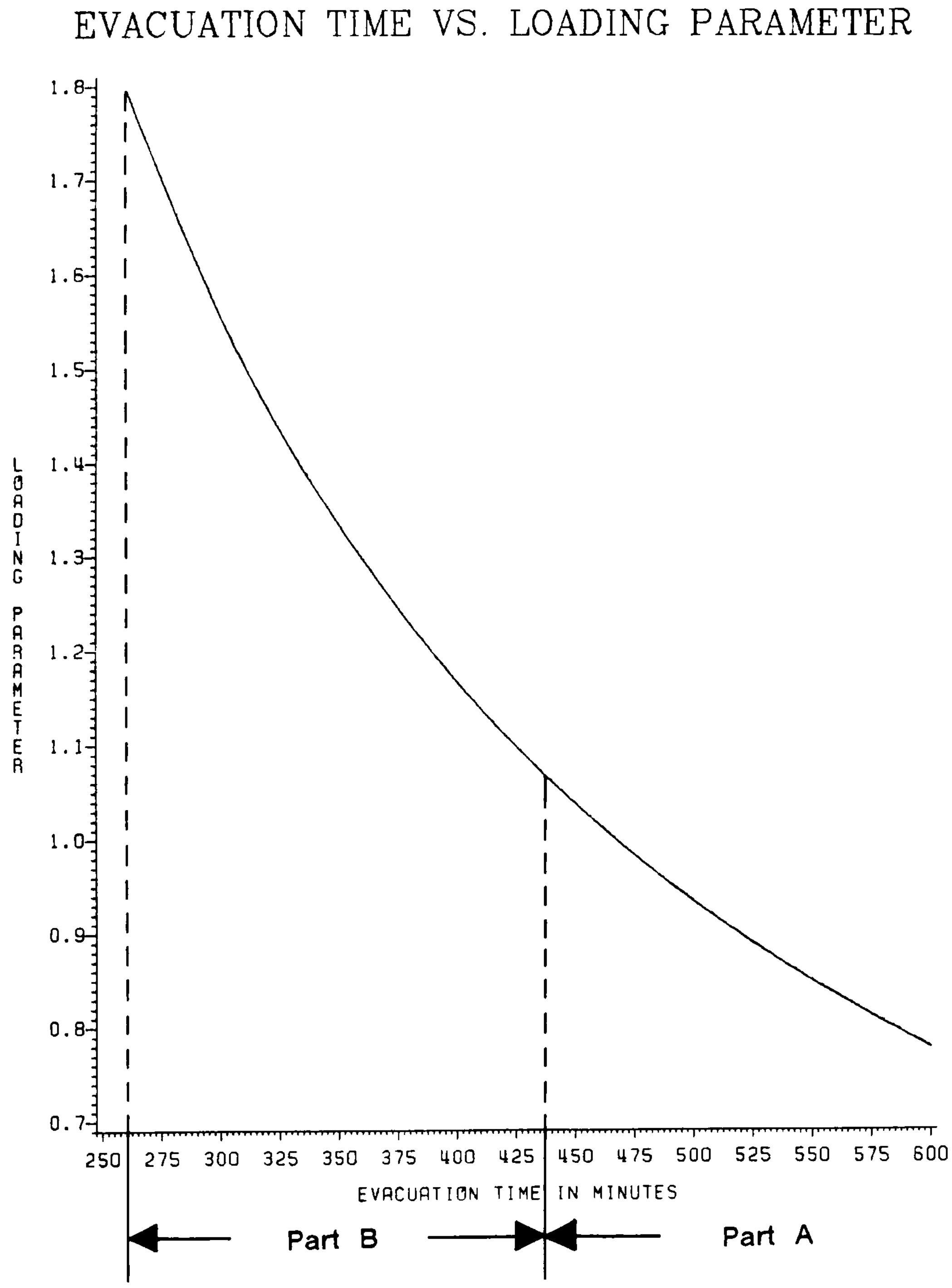


Figure 14. Evacuation Time and Loading Parameter

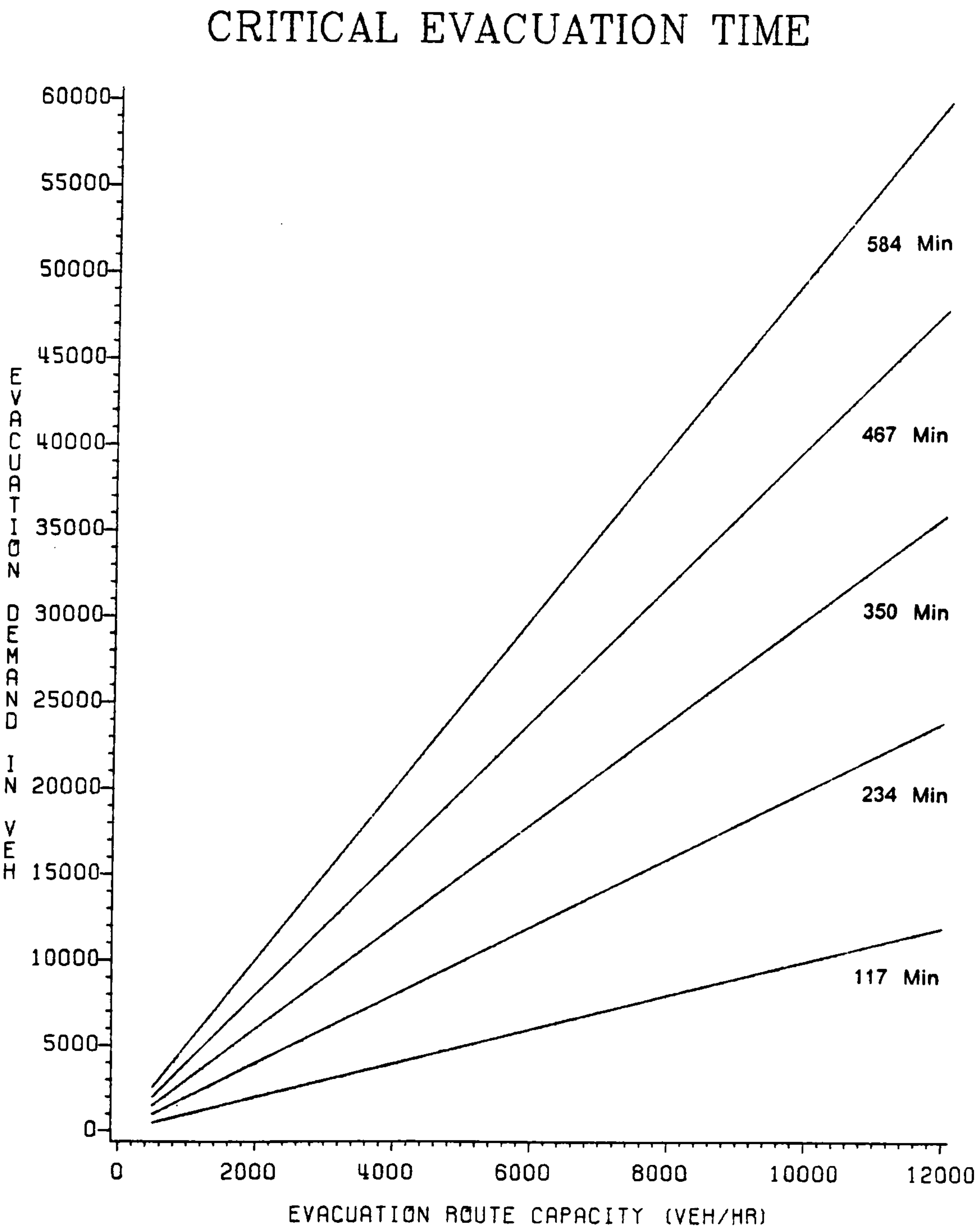


Figure 15. Evacuation Time, Demand, and Route Capacity

5.5.3 DIVERSION PARAMETER ANALYSIS

As discussed in the section "Number of Assignment Path", the value of the diversion parameter θ influences the link assignment-likelihood and the number of paths for assignment. It is thus valuable to know how θ affects the number of paths for assignment and to use it in determining an appropriate θ for model simulation. Both test networks continue to be used here for sensitivity analysis. The results are shown in Figure 16 and Table 7.

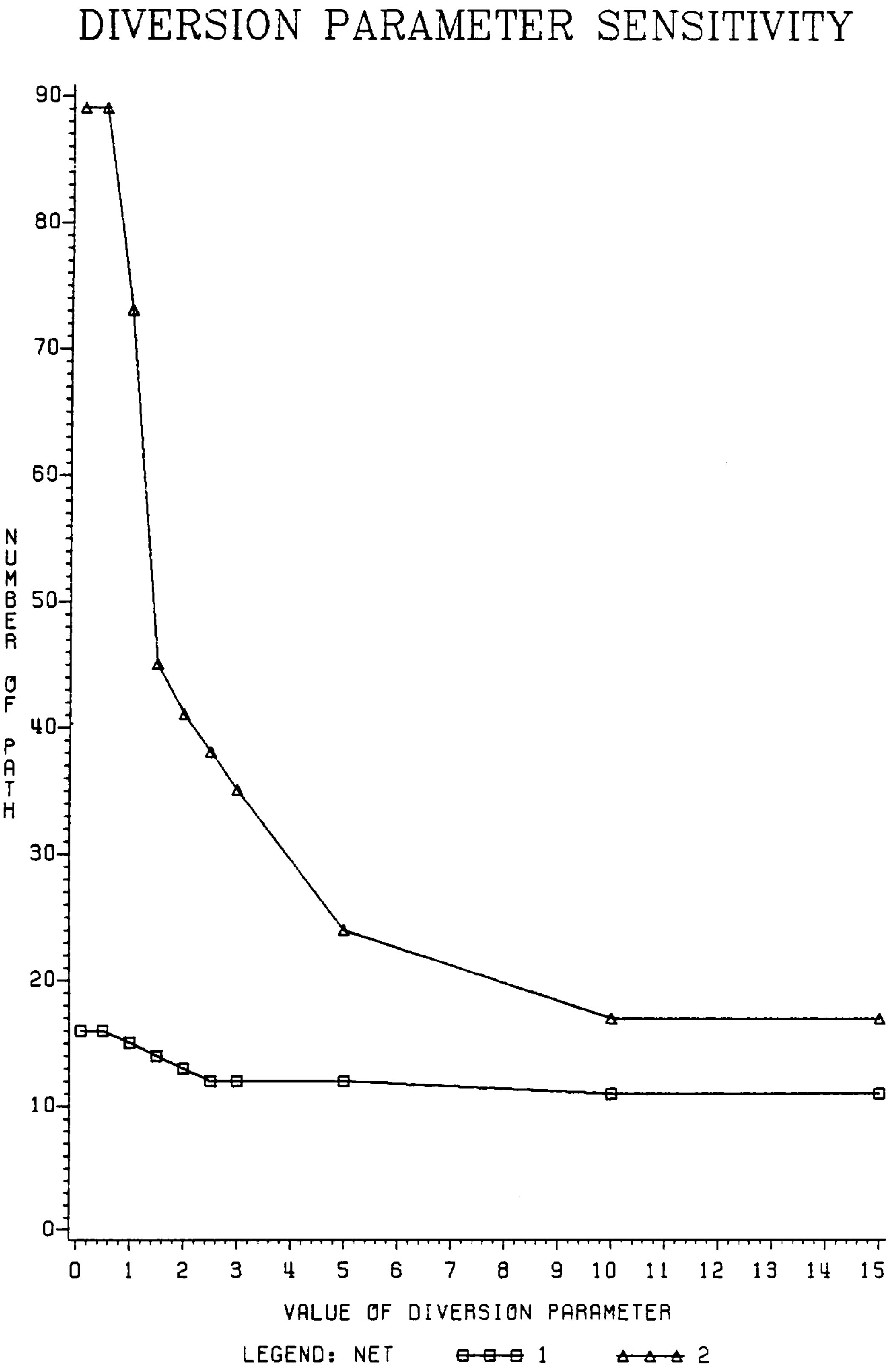


Figure 16. Diversion Parameter Sensitivity

Table 7. Diversion Parameter Analysis

	VIRGINIA BEACH NETWORK	HYPOTHETICAL RADIAL NETWORK
Value of θ	Number of Assignment Paths	
0.1	16	89
0.5	16	89
1.0	15	73
1.5	14	45
2.0	13	41
2.5	12	38
3.0	12	35
5.0	12	24
10.	11	17
15.	11	17
30.	11	16

It can be seen from Figure 16 and Table 7 that the assignment paths in the network of Virginia Beach are not sensitive to the changing of the diversion parameter. However, the diversion parameter does influence the path number in the hypothetical network. No relationship between the value of θ and the number of paths is observed. In addition, the sensitivity of the diversion parameter seems to be highly network dependent. It is also induced that if the path number of a network is not sensitive to the diversion parameter, the advantage of using path-recording approach to do equilibrium assignment might decrease. This is because the assignment path cannot be increased by providing smaller diversion parameters. The total vehicle-travel time of an assignment by using path-recording approach may be significantly higher than that by using non path-recording approach.

6.0 AN EVACUATION DECISION SUPPORT SYSTEM

6.1 SYSTEM COMPONENTS

The Integrated Evacuation Decision Support System (IEDSS) is a computer system developed to aid an emergency manager in his decision-making about evacuation planning and operation. The discussed MACROVAC, HEUPRAE, and MASSVAC2 models are employed in a simulation module to build the system; other features are added to enhance the models' operation. The developed IEDSS includes four modules:

1. the system control module
2. the data base module
3. the simulation model module
4. the graphic display module.

These four modules are not four subroutines of a super program but four decomposed subsystems. It is possible to combine these four subsystems (modules) into a super program, but the expandability, maintainability, and flexibility of the system will decrease. The super program will also not be easily adoptable in a personal computer. Figure 17 illustrates how these four modules are interrelated and how a user can operate the IEDSS. Details of each module of the IEDSS are described in the following sections.

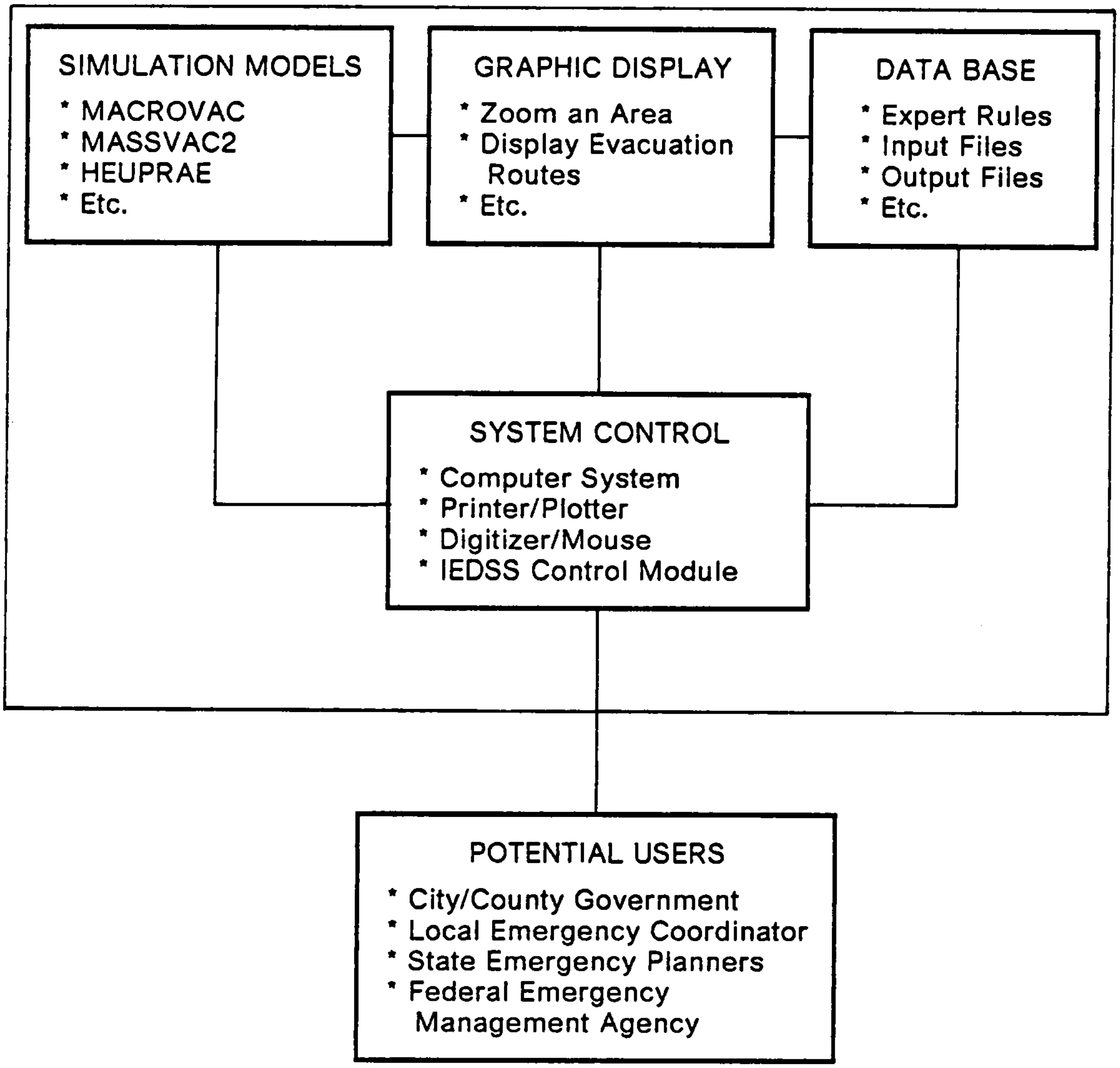


Figure 17. Components of the IEDSS

6.1.1 SYSTEM CONTROL MODULE

The system control module is the brain of the IEDSS; a user can operate the system only through the control module. The IEDSS can set a password in the control module to prevent another person's accessing the system or changing the built data base. Since the Microsoft DOS system[Microsoft, DOS] is used as the PC's operating system, the control module is designed as a nested batch file with different levels of execution options. A listing of the control module can be found in Appendix F.

6.1.2 DATA BASE MODULE

The data-base module functions to prepare an appropriate data file for the system's execution and to store a data file for further applications. The data files stored under this module can be categorized in three modes:

1. Input mode: input to the simulation-model module or to the graphic display module, such as the hazard type and characteristics, social-economic characteristics, transportation facilities, etc.
2. Output mode: output from the simulation-model module, either in descriptive form or in machine code (ASCII), such as evacuation trip allocation, evacuation routes, possible bottlenecks, etc.
3. Knowledge mode: decision rules and expert judgement to manage an evacuation system, such as maximum/minimum evacuation time, public response behavior, etc.

A detailed description of the functions of the data-base module follows:

1. Interactively creating or updating a demographic data file for the disaster prone areas, including the data of population, car ownership, dwelling unit per zone, and proportion of the land under each disaster level.
2. Interactively preparing, checking, or updating a data file for executing the simulation models, including network coding, traffic control strategies, etc.
3. Interactively preparing or updating a knowledge file for aiding decision making.
4. Storing and presenting the evacuation plans.

6.1.3 SIMULATION MODEL MODULE

The IEDSS currently has three simulation models, the MACROVAC model, the MASSVAC2 model, and the HEUPRAE model. The MACROVAC model is dissipation-rate model to estimate required system evacuation time. The MASSVAC2 model works as a planning tool for investigating different traffic control strategies to help prepare a master evacuation plan. Different evacuation scenarios under different disaster levels can be tested using MASSVAC2 and stored in the IEDSS before an emergency occurs. The HEUPRAE model works as a real-time operation tool for changing the evacuation master plan according to the changing nature of an oncoming disaster.

Although only three evacuation simulation models are available in IEDSS at the current stage, different models, such as NETVAC and DYNEV, can be added to the system in the future.

6.1.4 GRAPHIC DISPLAY MODULE

The graphic display module is an interface aid to the data-base module. It codes and edits the highway network in an interactive-graphic mode and displays the system's output in a graphic mode.

Currently, this module only has the capability of coding and editing the highway network for preparing an input file of the simulation model.

6.2 A CASE STUDY APPLICATION

To apply the developed IEDSS model to a real case, the city of Virginia Beach is selected for application in flood and hurricane evacuation.

The city of Virginia Beach has been hit by six hurricanes and several major storms since the turn of the century. The damage from these hazardous events ranged from beach erosion, heavy crop damage, extensive building and automobile damages, to loss of life in some extreme cases. One reason for the deaths and property damages was the lack of awareness by the population. Public awareness could be increased by planning ahead for disasters and by educating the people about the dangers and about the steps they should follow in disaster emergencies.

This section describes the implementation of the IEDSS in a real case by using Virginia Beach as a prototype. The required input data for operating the IEDSS and for preparing an evacuation plan are discussed.

6.2.1 CHARACTERISTICS OF VIRGINIA BEACH

The population of Virginia Beach was 272,900 persons in 1981, with nearly 7.1% of the total population above 60 years old. The total area of Virginia Beach is 310 square miles, out of which 258.7 square miles (83.5%) are land and the remaining 51.3 square miles (16.5%) is water. There are 74 public and private schools. Other facilities include five libraries, two hospitals, and 94 city parks.

It can be seen from a topographic map, shown in Figure 18, that the majority of the northern and eastern coastal strips of the city falls within the ground level equal to or less than 10 feet. In addition, a strip in the middle section that extends from between the eastern and western branches of Lynnhaven River in the north to the West Neck Creek in the south also has ground level equal to or less than 10 feet. In general, the area with elevation equal to or less than 10 feet is vulnerable to all levels of hurricanes and severe flooding conditions. Figure 19 shows the 100 year flood plain for the area. One can see that a close agreement exists between Figure 18 and 19, which is useful in defining the disaster prone area of the city.

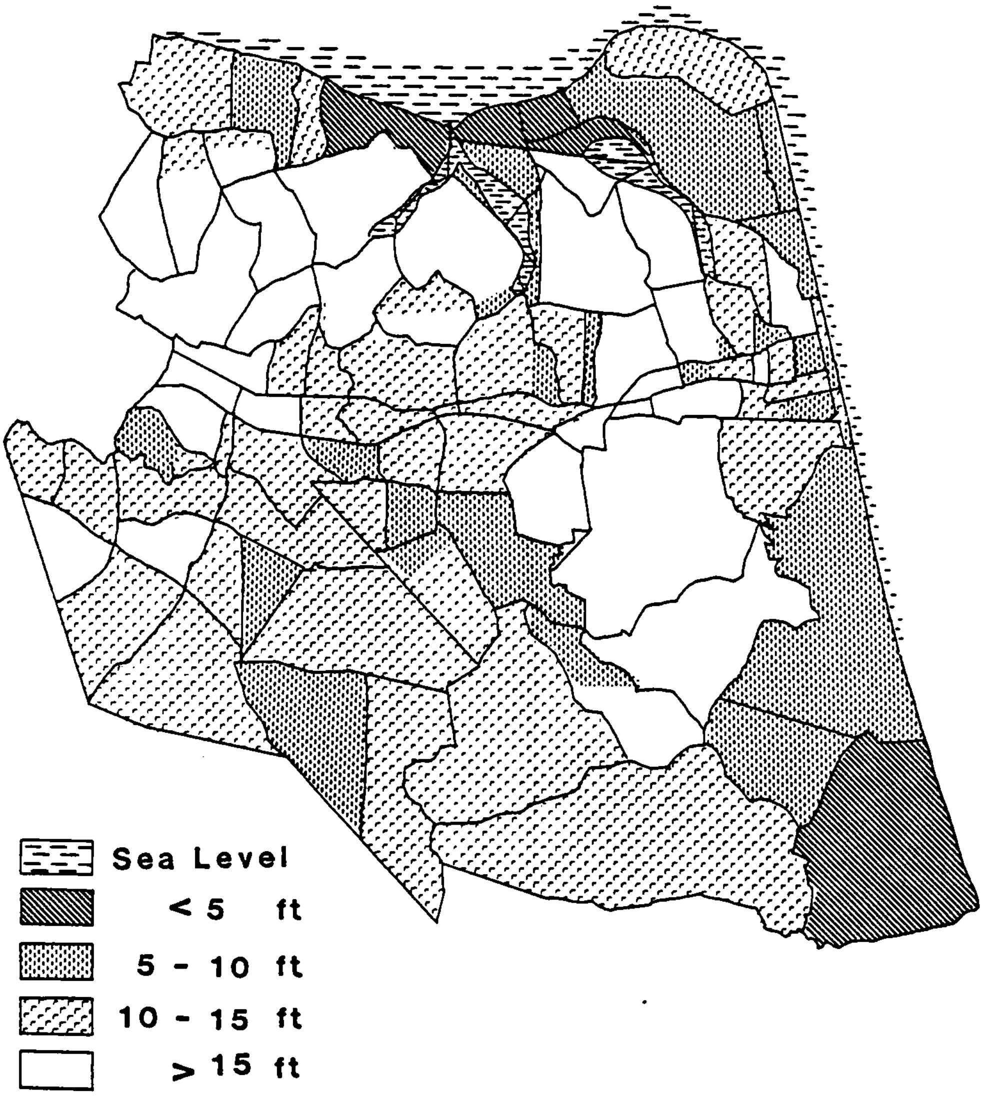


Figure 18. Virginia Beach Topographic Map

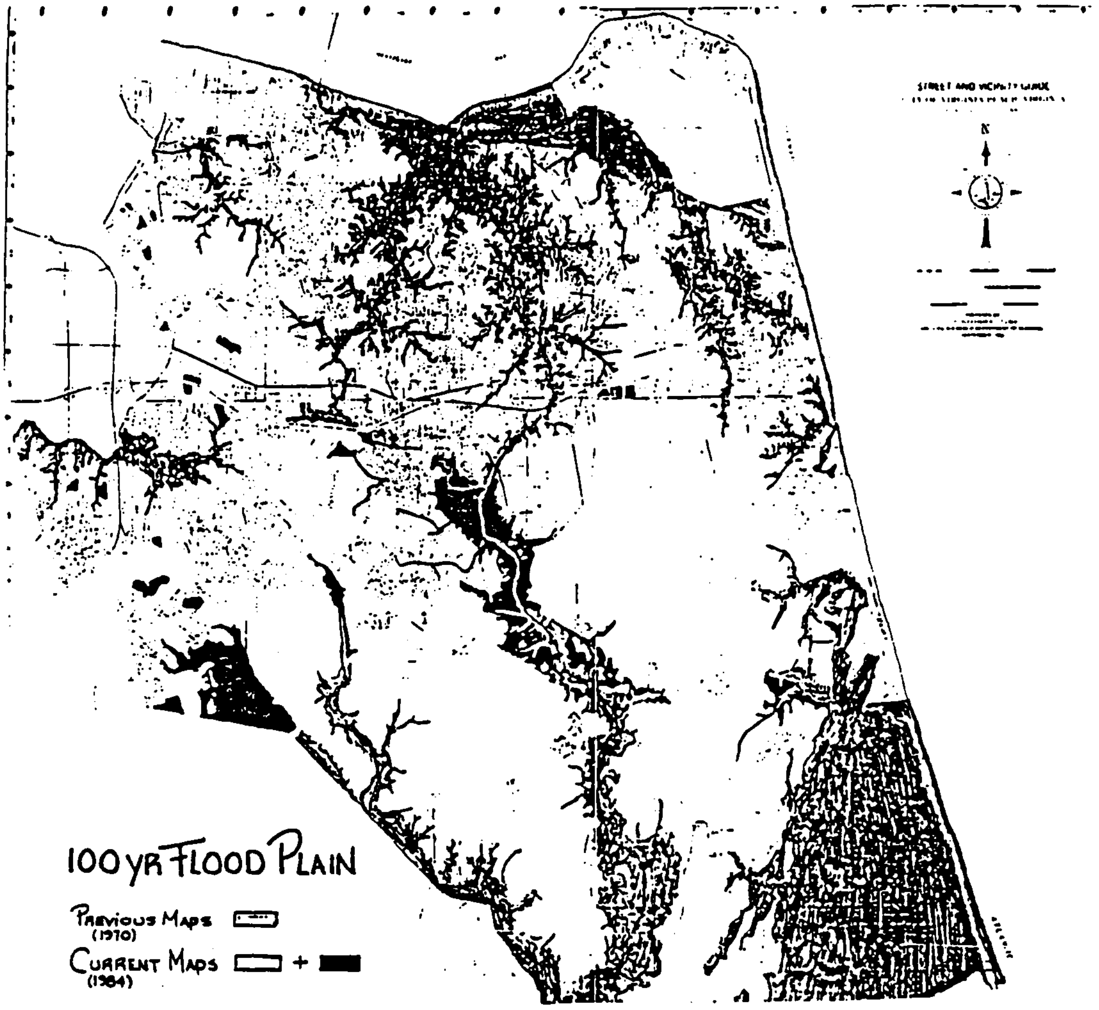


Figure 19. Virginia Beach 100 Years Flood Plain

6.2.2 BUILDING THE DEMOGRAPHIC DATA BASE

The statistics prepared by the City Planning Department contains 215 statistical areas and 143 transportation zones. Each transportation zone contains one or more statistical areas. In the process of finalizing the zones used for evacuation analysis, several transportation zones were combined and others were not considered. Overall, 89 zones were adopted for the analysis and shown in Figure 20.

To develop evacuation plans for Virginia Beach, the city was divided into four regions according to the flood plain, the social-economic characteristics, and the highway network characteristics. The first region included Oceanfront, Great Neck, and half of Bayfront. The second region contained Courthouse-sandbridge. The third region was composed of Little Creek, Bayside, Little Neck, and the other half of Bayfront. The fourth region incorporated Kempsville and Holland. The Pungo Blackwater area was ignored because of its low population.

To define the boundaries of the disaster area threatened by hurricane and flooding, Figure 18 and Figure 19 about the ground elevation and 100 year flood plain were employed.

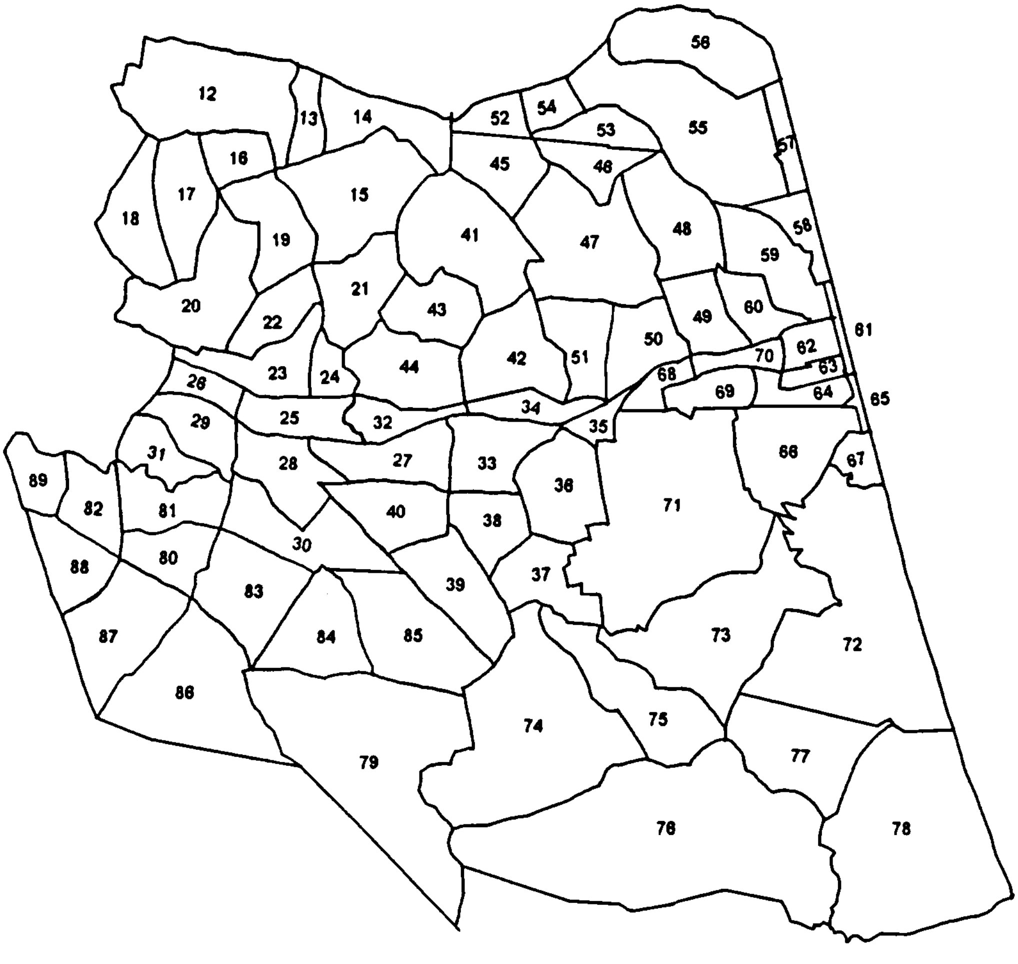


Figure 20. Virginia Beach Evacuation Zones

Four hurricane levels were chosen to identify the disaster level for the study area. In the level one hurricane, the height of water will not exceed 5 feet. The greatest height of water in level two and three is assumed to be 10 and 15 feet respectively. The level four hurricane is considered same as level three except that higher wind accompanying the water surge will threaten most of the ocean front area. In addition to the disaster boundaries, shelters for each region are selected and their capacities are estimated. Table 8 lists the names of these shelters together with their capacities.

To simplify demonstrating the application procedure of the IEDSS, only region 2 is used for further discussion in this chapter. A demographic data-base of region 2, including population, car ownership, and the percentage of land under each hurricane level are coded using the IEDSS data-base module. Appendix G illustrates the demographic data file for the whole Virginia Beach city which includes region 2. With the aid of the IEDSS, this data base can be easily updated when new demographic data is available.

Table 8. Virginia Beach Evacuation Shelters

REGION	SHELTER	CAPACITY[1]
1	Day's Elementary School	1000
1	First Colonial High School	2000
1	Seatack Elementary School	1000
1	Va. Beach Jr. High School	1500
1	Cox High School	2000
2	Princess Anne Elementary S.	1500
2	Princess Anne Jr. High S.	1500
3	Hermitage Elementary School	1000
3	Independence Jr. High S.	1500
3	Kings Grant Elementary S.	1000
3	Princess Anne High School	2000
4	Kempsville High School	2000
4	Plaza Jr. High School	1500

[1] unit: persons

6.2.3 CODING THE HIGHWAY NETWORK

There are two steps involved in highway network coding. First, a highway map is obtained and the coordinates of each node are determined. For the study case, a map with scale 1"=1600' is used[Va. Beach Map]; coordinates of each node, such as an intersection, origin, or destination, are measured directly from the map. Since the IEDSS currently does not interface with a digitizer, the node coordinates must be prepared manually. The graphic display module of the IEDSS then reads those nodes and displays them on screen; those links connecting nodes are also edited through an interactive procedure of the graphic-display module.

The second step in coding the highway network is to assign each link its characteristics for preparing a simulation input file. Link characteristics include road name, length, free flow speed, seeding volume, starting node, and ending node.

After coding the highway network, the coding file is sent to the data-base module to test its competence for running the simulation model. An incomplete file will be returned with an error message and will wait for correction. This checking process continues until a complete file is achieved. Finally, the complete data file is restructured to improve its efficiency for model execution and then saved. In this final file of network coding, each link is assigned a link number and the total link number is calculated.

6.2.4 ESTIMATING SYSTEM EVACUATION TIME

Besides highway network coding, the estimated system evacuation time is another required input to the simulation model. The MACROVAC model is used to provide an estimation of the system evacuation time. Table 9 shows the evacuation routes and the dissipation rate of the region 2. The evacuation demand under each hurricane level is also calculated and shown in Table 10. Table 11 presents the estimated evacuation times of region 2 under different hurricane levels by using the MACROVAC model.

Table 9. Evacuation Routes for Region 2

EVACUATION ROUTE	CAPACITY (VEH/HR)
Sandbridge Road	900
Ocean Boulevard	900
Princess Anne Road	4,000
London Bridge Road	900

Table 10. Evacuation Demand of Region 2

HURRICANE LEVEL	EVACUATION DEMAND (VEH)[1]
1	1,992
2	16,671
4	21,773

[1] vehicle trips = population/1.5

Table 11. Estimated Evacuation Times of Region 2

HURRICANE LEVEL	EVACUATION TIME (MINUTES)
1	35
2	291
4	380

6.2.5 PROVIDING A VALUE FOR THE DIVERSION PARAMETER

It is recommended that a small diversion parameter be used to run the simulation model. A small diversion parameter means more paths are used to load traffic, which results in a better equilibrium assignment and lower total vehicle-travel time. However, if the evacuees are not familiar with how to access the evacuation routes, a larger diversion parameter should be considered.

It is also recommended that the HEUPRAE model be tested for the diversion-parameter sensitivity to the number of paths used in the assignment. The appropriate value of the diversion parameter is then stored in the data-base module as a default. The suggested default value of the diversion parameter for region 2 is 0.1 according to the analysis done in Table 7.

6.2.6 RUNNING THE SIMULATION MODEL

With a complete input file representing the evacuation system of the region 2, the simulation model MASSVAC2 is run to provide an evacuation plan. The output from the MASSVAC2 model is then evaluated and discussed with the city emergency managers. Any unrealistical result is recorded and the MASSVAC2 model is executed again until an acceptable one is achieved.

6.2.7 STORING THE SIMULATION OUTPUT

The evacuation plan for region 2 is then stored in the data-base module of the IEDSS for future retrieval and application when a hurricane is nearing Virginia Beach.

6.2.8 DISPLAY THE SIMULATION OUTPUT

Currently, the simulation output can only be displayed on screen or through a hard copy. The capability of graphically displaying the output of the simulation model may be added to enhance the graphic-display module of the IEDSS in the future.

7.0 DISCUSSION, CONCLUSION, AND RECOMMENDATION

7.1 DISCUSSION

This research presents that a traffic assignment model which provides guidance for an emergency evacuation at a real-time speed has been successfully developed. Together with the new assignment model, a new equilibrium assignment algorithm has also been developed; Dial's algorithm is not the only equilibrium oriented algorithm. No criteria can be found to provide the value of Dial's diversion parameter 'θ' in determining the number of paths used for assignment. The sensitivity of the diversion parameter to the number of paths for trips loading is highly network dependent. In addition, Dial's algorithm only considers the difference in travel time in computing link assignment likelihood. A link with a constant travel time is given the same assignment likelihood without considering its relative distance to an origin. This is not reasonable, since the difference in travel time of between 50 and 60 minutes is apparently different from that of between 10 and 20 minutes in the perception of a tripmaker.

The new assignment method, however, still employs Dial's efficient-path concept to load trips. But instead of using the assignment likelihood as a weight to load trips, trips are apportioned to routes according to the available highway capacities, travel speeds, and travel times of the links forming those routes. The assignment algorithm in this research has as its objective to maximize the utilization of link capacity and to achieve an equilibrium assignment in term of total network travel time. The new assignment method is not a path-enumeration assignment. It shows the number of paths connecting each O-D centroids pair. Besides, its path-recording approach improves the efficiency of an assignment execution and allows the development of a real-time traffic assignment model.

The evacuation decision support system IEDSS developed in this research provides a decision aid for the emergency agency which must cope with the threat of a disaster. However, the IEDSS still has plenty of room to improve its operation. The evacuation-knowledge base, the graphic interface capability, and the computer hardware all could be upgraded to provide a better decision support system. Obtaining and programming the knowledge of evacuation experts need further study so as to have an automatic decision-screening capability. The IBM PC-XT is not a network-oriented system; its memory expansion ability is also limited. A better network-oriented PC, such as an IBM PC-AT, may be preferable to implement the IEDSS. Also, in order to provide better monitor resolution, the current IEDSS does not have as a function a color-graphic display. The problem of the trade-off between the resolution and the color of a monitor should be addressed in the future in order to provide a color network display to aid decision-making.

7.2 CONCLUSIONS

From this research, several conclusions are reached:

1. The IEDSS is a decision-support system for evacuation planning and operation with three levels of decision-aid:

    A dissipation-rate model: MACROVAC
    A Planning model: MASSVAC2
    An operation model: HEUPRAE

2. MACROVAC is a model for estimating evacuation time for a disaster area. MACROVAC's estimate of evacuation time is 12% less accurate than that of the MASSVAC2 model.

3. MASSVAC2 is an evacuation planning model restructured from the MASSVAC model. Both MASSVAC and MASSVAC2 models use Dial's algorithm in traffic assignment.

4. HEUPRAE is a model available to perform real-time traffic assignment simulation with the path-recording assignment method.

5. An assignment algorithm based on maximizing the utilization of link capacity of a system is a valid algorithm in equilibrium assignment. The difference between this algorithm and Dial's algorithm is less than 5% in terms of total travel time.

6. The path-recording approach is a valid assignment approach. The difference between path-recording and non path-recording assignment is less than 2% in terms of total travel time, if a substantial number of paths are used for loading trips.

7. In terms of network-clearance time, the HEUPRAE model is 91% as accurate as the MASSVAC2 model.

8. The more paths used for path-recording assignments, the better the equilibrium result achieved by the HEUPRAE model.

9. The saving of computer execution time by using the path-recording assignment reaches as high as 55% compared to the non path-recording Dial's assignment.

7.3 RECOMMENDATION

The study in this research is based on two assumptions:

1. The arrival of the evacuation demand following a logit curve.

2. The assignment algorithm fitting the route choosing behavior.

Research on these two assumptions certainly helps the continuing studies of evacuation. In addition, several theoretical studies are recommended:

1. Research of evacuation behavior concerning the relationship between the disaster site and resident location.

2. Research of human value perception in route choosing.

3. Use of a Kth shortest-path algorithm to replace the applied multiple-path algorithm in building paths and comparing the execution efficiency.

4. A Stochastic simulation of the evacuees' arrival in terms of location, which can help in preparing a guideline for evacuation sequence.

These practical improvements for operating the IEDSS are also recommended:

1. Different traffic-control strategies can be integrated with the IEDSS to provide enough alternatives in dealing with evacuation.
2. Expert judgements and rules may be added to complete the knowledge-base of the IEDSS.
3. Enhancement of the color-graphic capability could be added to provide better monitor resolution.
4. A digitizer may be added to interface with the IEDSS in network coding.
5. The graphic capability to directly interpret system output may be developed and added to the IEDSS.

APPENDIX A. REFERENCES

1. A. M. Voorhees and Assoc., "Evacuation Times Assessment Study for the Diablo Canyon Nuclear Power Plant," Report prepared for Pacific Gas and Electric Co., Berkeley, CA, April 1980.

2. Belardo, S., Howell, A., Ryan, R., and Wallace, W.A., "A Microcomputer-Based Emergency Response System," Disasters, July 1983.

3. Bureau of Public Roads, "Traffic Assignment Manual," V-13, U.S. Department of Commerce, June 1964.

4. Carroll, John M., "Computer Simulation in Emergency Planning," Society for Computer Simulation, Simulation Series, Volume II, No. 2, 1983.

5. Cosby, J.C. and Powers, D.L., "Simulation of Traffic in Emergency Evacuations," paper presented at the 61st TRB annual meeting, Washington, D.C., 1982.

6. Dial, Robert B., "A Probabilistic Multipath Traffic Assignment Model Which Obviates Path Enumeration," Transportation Research, Vol. 5, NO. 2, pp.83-111, 1971.

7. Dijkstra, E., "A Note on Two Problems in Connection with Graphs," Numerische Mathematik 1, 269-271, 1959.

8. Drew, Donald R., "Traffic Flow Theory and Control," McGraw Hill, Inc., 1968.

9. Dreyfus, S.E., "An Appraisal of Some Shortest-Path Algorithms," Operations Research, Vol. 17, No. 3, 1969.

10. Federal Highway Administration, "Urban Transportation Planning: General Information and Introduction to System 360," U.S. Department of Transportation, 1972.

11. Federal Highway Administration, "Traffic Assignment," III-15, U.S. Department of Transportation, Aug. 1973.

12. Federal Highway Administration, "UTPS - Urban Transportation Planning System, A Comprehensive Computer Package for Modeling Multimodal Transportation Network," U. S. Department of Transportation, 1985.

13. Florian, M. and Nguyen, S., "EMME/2 - An Interactive Graphic System for Modeling Multimodal Transportation

networks," Centre De Recherche Les Transports of the University of Montreal, Canada; distributed by INRO Systems, Inc., Jan. 1985.

14. Gibson, D.R. and Ross, P., "Simulation of Traffic Street Networks," Public Roads, A Journal of Highway Research and Development, Sept. 1977, Vol. 41, No. 2.

15. Golden, Bruce, "Shortes-Path Algorithms: A Comparison," Operations Research, Vol.24, No.6, 1976.

16. Hall, Richard A., Lewis, Donald C., "The Sensitivity of Evacuation Order Times to Behavioral Response," Working Paper, Post, Buckley, Schuh & Jernigan, Inc., Tallahassee, Florida.

17. Hans, J.M. and T.C. Sell, "Evacuation Risks - An Evaluation," U.S. Environmental Protection Agency, June 1974.

18. Highway Research Board, "Highway Capacity Manual - 1965," HRB Special Report No. 87, Washington, D.C. 1965.

19. Hobeika, A.G. and Jamei, B., "MASSVAC: A Model for Calculating Evacuation Times Under Natural Disasters," Proceedings of the Conference on Computer Simulation in Emergency Planning, Society of Computer Simulation, Volume 15, No. 1, La Jolla, California, Jan. 1985.

20. Hobeika, A.G., Radwan, A.E., and Jamei, B., "Virginia Beach Hurricane/Flood Transportation Evacuation Study," Final Report, Dept. of Civil Engineering, VPI&SU, for the Office of Emergency Services, City of Virginia Beach, Jan. 1985.

21. Hwang, K.P., "Vehicle Loading Phenomenon during Evacuation at Transportation Network," unpublished paper, Civil Engrg., VPI & SU, 1985.

22. Jamei, Bahram, "Transportation Actions to Reduce Highway Evacuation Times Under Natural Disasters," Dissertation submitted to Graduate School of Virginia Polytechnic Institute and State University, Nov. 1984.

23. Johnson, C., "Emergency Management for Chemical Spills," paper presented at First Symposium in the Application of Expert Systems in Emergency Management Operations, Washington, D.C., April 1985.

24. Kalevala, A.F. Sylvester, "Investigation of the Effect of Change in Vehicular Characteristics on Highway Capacity and Level of Service," Thesis submitted to Graduate School of Virginia Polytechinc Institute and State University, 1984.

25. KLD and Associates, "Network Flow Simulation for Urban Traffic Control System - Phase II," Volume 1. Technical Report, DOT-FH-118502, U. S. Department of Transportation, 1977.

26. KLD and Associates, "The DYNEV Model: An Effective Tool for Developing and Evaluating Evacuation Plans," Paper presented at the TRB 61st annual meeting, Washington, D.C., 1982.

27. Lieberman, E.B. and Andrews, B.J., "TRAFLO: A New Tool to Evaluate Transportation System Management Strategies," Transportation Research Record, No. 772, Washington, D.C., 1980.

28. Lieberman, E.B. and J.L. Woo, "SIGOP III - User's Manual," Federal Highway Administration, FHWA-IP-82-A, July 1982.

29. Microsoft, "Microsoft Disk Operating System," Version 2.10, Microsoft Corporation, Sept. 1983.

30. Microsoft, "Microsoft FORTRAN," a registered trademark of Microsoft Corporation, 1985.

31. Morris, Richard Jere, "A Comparative Analysis of Trip Distribution and Traffic Assignment Models for Transportation Planning in Developing Regions," pp.152-172, Standford University, DOT/UMTA University Research, Dec. 1973.

32. Nguyen, Sang, "An Algorithm for the Traffic Assignment Problem," Transportation Science, Vol.8, No.3, 1974.

33. Perry, Ronald W., Lindell, Michael K., Greene, Marjorie R., "Evacuation Planning in Emergency Management," pp.1-9, Battelle Human Affairs Research Centers, Lexington Books, D.C. Heath and Company, 1981.

34. Petrie, J.F. and Hathaway, W.T., "Development of An Automated Emergency Response System (AERS) for Rail Transit Systems," Interim Report prepared by Transportation Systems Centre, Cambridge, Mass., for U.S. DOT, Oct. 1984.

35. Potts, Renfrey B. and Oliver, Robert M., "Flows in Transportation Network," Academic Press, New York, 1972.

36. PRC Voorhees, "Evacuation Planning Package," Paper presented at the 61st TRB annual meeting, Washington, D.C., 1982.

37. Quarantelli, E. L., Editor, "Disaster Theory and Research," SAGE Publications Ltd., 1978.

38. Quarantelli, E. L., "Evacuation Behavior and Problems: Findings and Implications from the Research Literature," Miscellaneous Report No. 27, Disaster Research Center, The Ohio State University, 1980.

39. Schneider, Morton, "A Direct Approach to Traffic Assignment," Highway Research Record, NO. 6, 1963.

40. Sheffi, Y., Mahmassani, H. and Powell, W., "NETVAC: A Transportation Network Evacuation Model," Center of Transportation Studies, MIT, 1980.

41. Smock, Robert, "An Iterative Assignment Approach to Capacity Restraint on Arterial Networks," HRB Bull. 347, p.60, 1962.

42. TIME magazine, Nov. 25, pp.46-58, 1985.

43. Transportation Research Board, "Highway Capacity Manual - 1985," TRB Special Report No. 209, Washington, D.C. 1985.

44. Virginia Beach Planning Department, "Street and Vicinity Guide," Scale 1"=1600', Sept. 1982.

45. Wallace, C.E., K.G. Courage, D.P. Reaves, G.W. Schoene, and G.W. Euler, "TRANSYT-7F User's Manual," Office of Traffic Operations, Federal Highway Administration, Feb. 1981.

46. Wardrop, J.G., "Some Theoretical Aspects of Road Traffic Research," Proc. Inst. Civil Engineering, Part II, 325-378, 1952.

47. Webster, F.V., and Cobbe, B.M., "Traffic Signals," Road Research Technical Paper No. 56, Road Research Laboratory, London, 1976.

48. Wilbur Smith and Associates, "Roadway Network and Evacuation Study," Seabrook, New Hampshire, Report

prepared for the Public Service Company of New Hampshire, 1974

APPENDIX B. GLOSSARYS

Capacity Restraint: The process whereby assigned volumes are related to the capacity of the highway facilities in such a manner that overloaded routes become less attractive as minimum path candidates.

Centroid: The center of gravity of a zone, located at the mathematical center of the zone as determined by the cost functions of that zone.

Clearance Time: The time required to clear all evacuating vehicles from the roadways.

Connection Matrix: A matrix representation of all links interconnecting the nodes of a network.

Converted Flow: A component of the normal flow pattern which has made a change in its usual mode of transport.

Diverted Flow: A component of flow which has changed from its previous path of travel to another route without a change in origin, destination, or mode of transport.

Evacuation: The mass physical movement of people, of a temporary nature, that collectively emerges in coping with community threats, damages or disruptions.

Generated Flow: Transport flow that exists because of a particular land use.

Generator: An area that, due to its particular kind of land use, creates transport demand.

Heuristic Approach: A method to solve problem for a quick but non-optimal solution.

Heuristics: The science which studies the laws governing the design of new actions in new situation, aimed at achievement of the goal in a situation which is new for the system.

Hurricane: A form of tropical cyclone, pronounced rotary circulation, constant wind speed of 74 miles per hour (64 knots) or more.

Hurricane Watch: A warning issued for a coastal area when there is a threat of hurricane conditions within 36-48 hours.

Hurricane Warning: A warning issued when hurricane conditions are expected in a specified coastal area in approximately 18 to 2 hours. Actions for protection of life and property should begin immediately when this warning is issued.

Induced Flow: The added component of flow which did not previously exist in any form but which results when new or improved transport facilities are provided.

Land Use Feedback: When the location, size, costs, etc., of a proposed transport network have been fixed by virtue of transport assignment and other considerations, it is recognized that the building of the proposed system will in itself alter land use patterns which in turn affect the forecasts of interzonal movements. This definition implies that the whole transport planning process is a continuing iterative system.

Link: The one-way portion of the transport network connecting two nodes.

Link Flow: The total interzonal flow assigned to a link in the network. These are sometimes referred as leg volumes.

Link Impedance: A value assigned to each link in the network. This impedance may be some average value of travel time, it may be distance if minimum distance routes are desired, it may be cost for use of the link, or it may be any other parameter or combination of parameters so desired.

Link Cost Function: The mathematical relation describing the cost per unit of flow expended by using a given link.

Loading the Network: The process of assigning the interzonal flows to the network.

Network Description: The transport network under consideration described in tabular form as nodes, directional links, link impedances, link distances, turn restrictions, etc.

Node: A point of intersection in a transport network.

Path: The aggregate of all links in a route between any two nodes in a transport network.

Path Cost Function: The mathematical relation describing the cost per unit of flow expended in using a given path.

Potential Flow: The total flow that would in all probability move between two zones (on a given route), assuming ideal transmission facilities.

Real-Time System: A computer system wherein a computation is performed during the actual time that related physical process transpires, in order that results of the computation can be used in guiding the physical process.

Routing or Trace: A part of a tree. In the Moore algorithm, it is the minimum path through the network from one node to another.

Transit: The movement of people on mass public media, such as buses, streetcars, and subways, and thus a subclassification of "transportation."

Transport: Used in the most general sense to include the movement not only of people and goods, but also of other fluxes such as energy, information, water, and sewage.

Transportation: The movement of people and goods, and therefore, a subclassification of the more general "transport."

Transport Network: A network of links and nodes describing the possible flow paths for interzonal or inter-centroidal movements for any single transport flux.

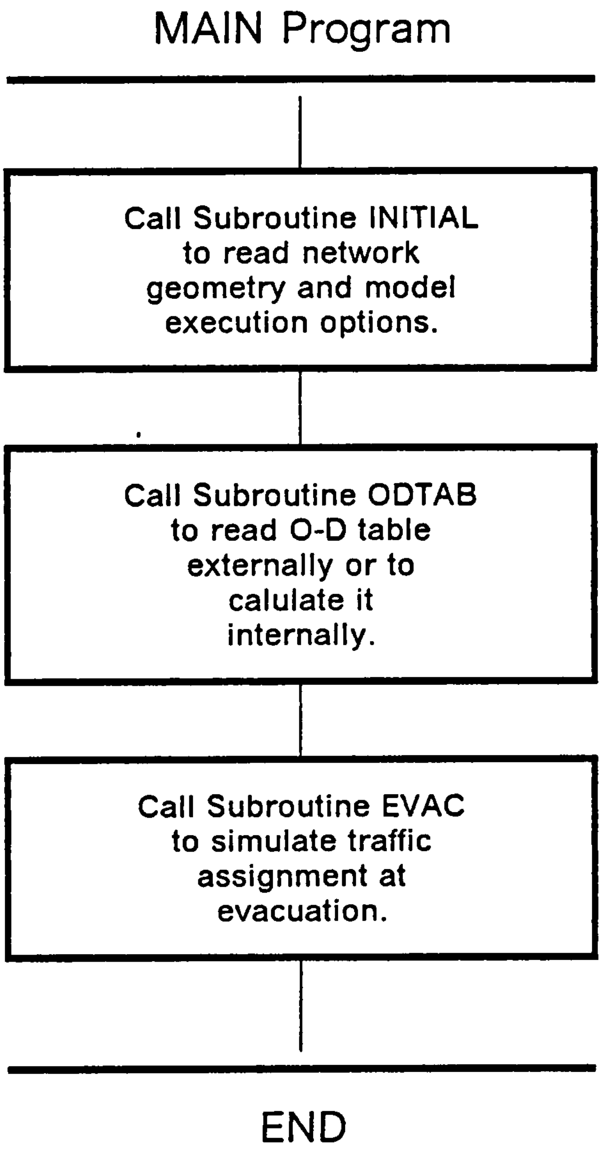
Tree: The aggregate of all the minimum path routings from a node to all other nodes in the network.

Tree Building: The use of an algorithm for computing minimum paths.

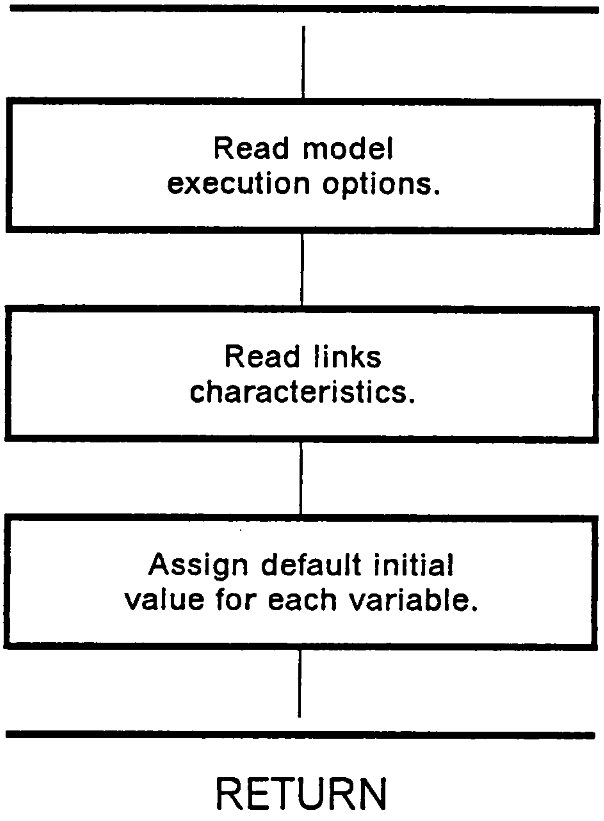
Tropical Cyclone: The general term for all cyclone circulations originating over water, classified by form and intensity.

Zone: An area of a system throughout which its describing parameters can be considered constant.

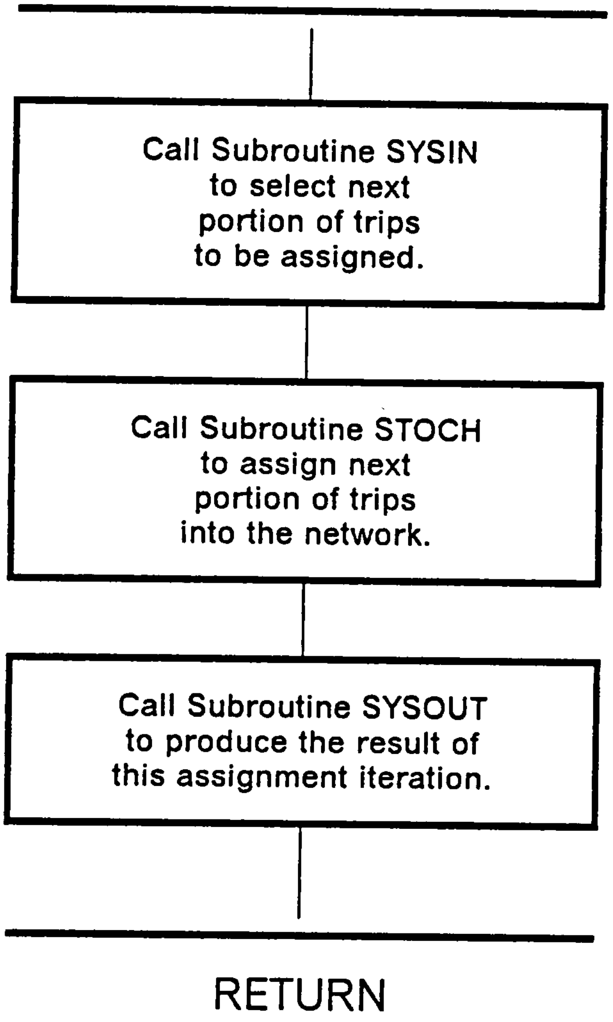
APPENDIX C. PROGRAM FLOWCHARTS

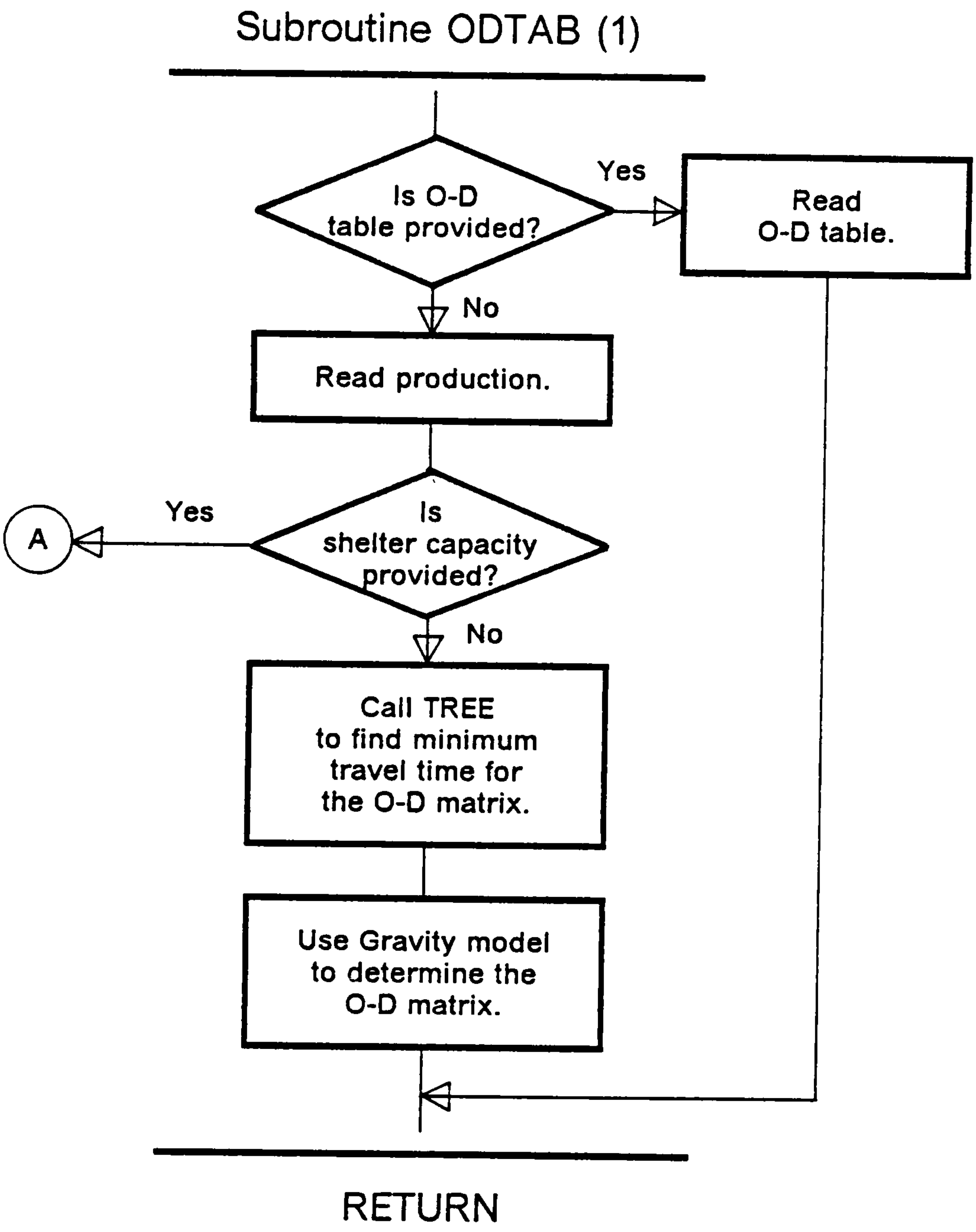


Subroutine INITIAL

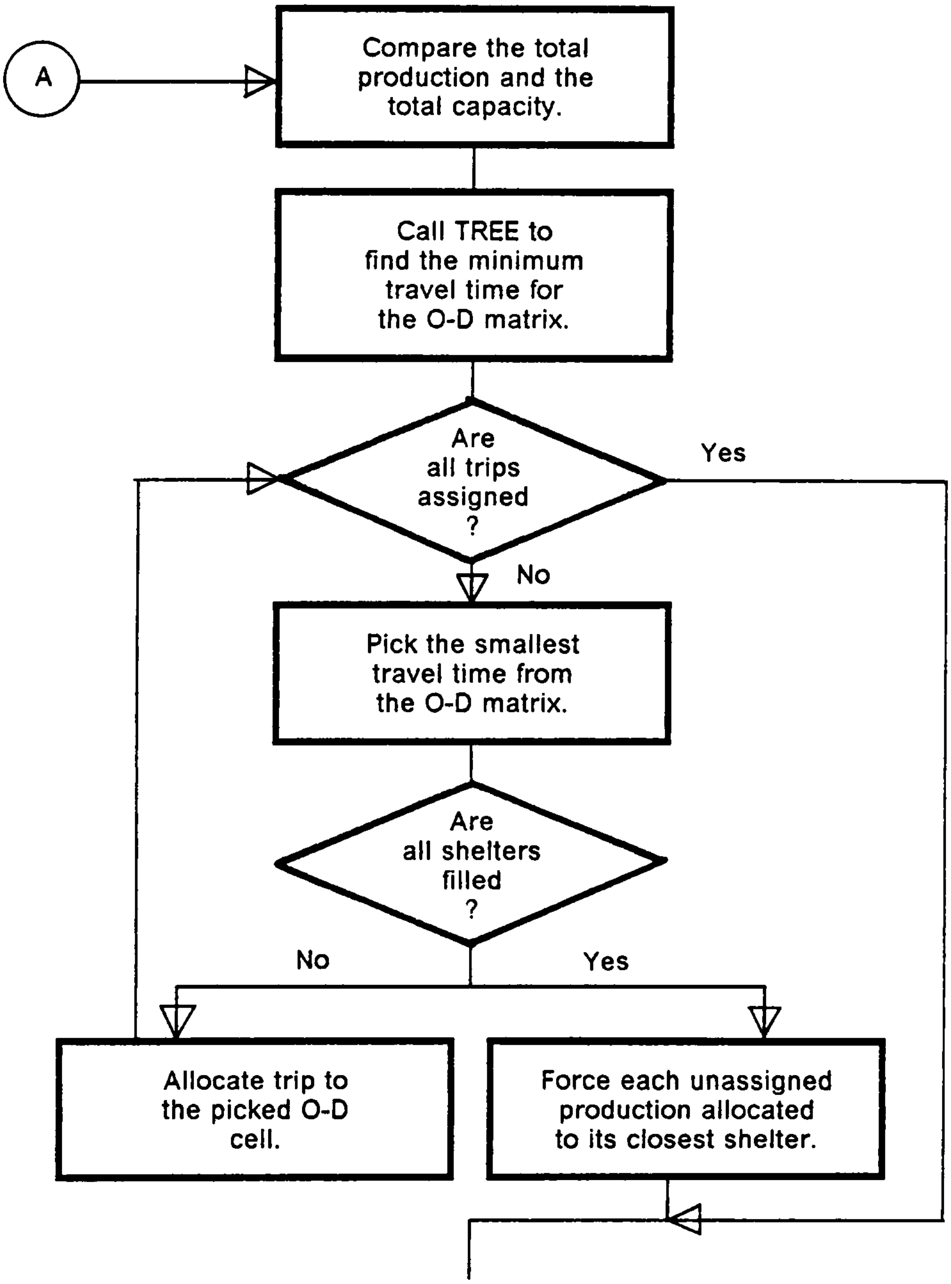


Subroutine EVAC



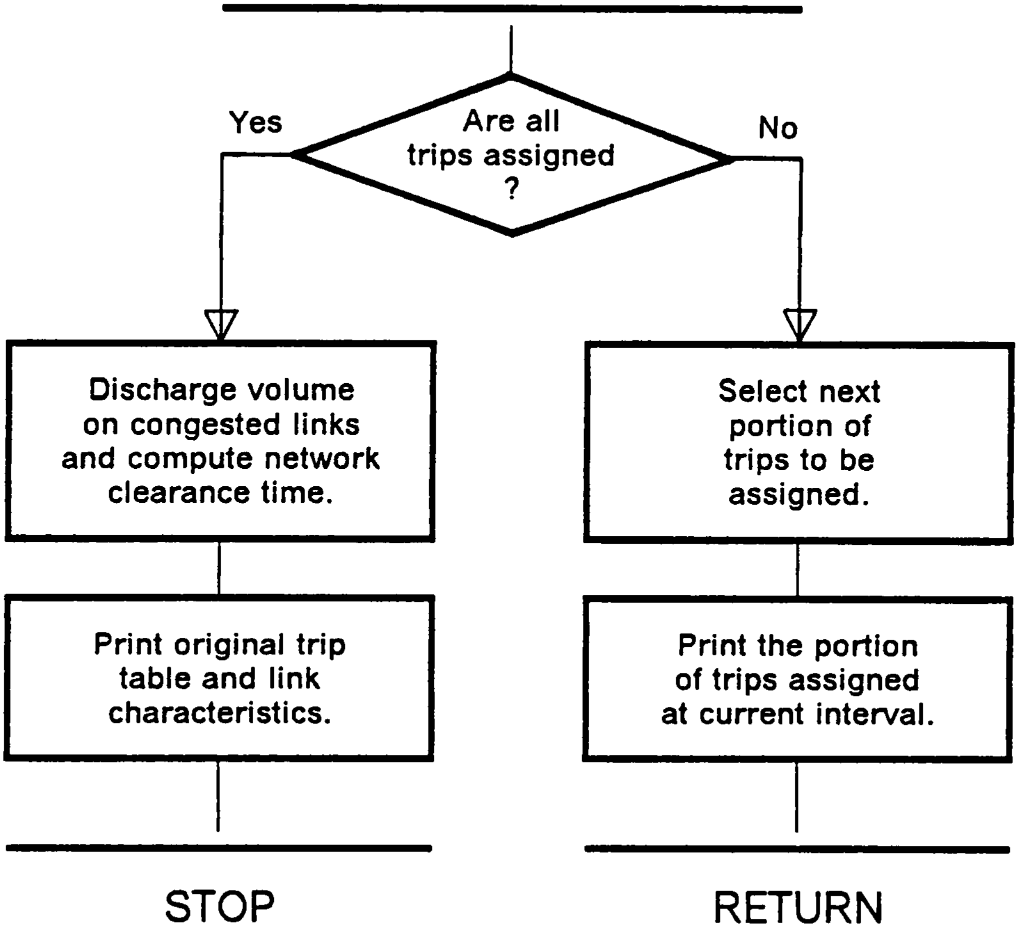

Subroutine ODTAB (1)
Is O-D table provided?
Yes
Read O-D table.
No
Read production.
Is shelter capacity provided?
Yes
A
No
Call TREE to find minimum travel time for the O-D matrix.
Use Gravity model to determine the O-D matrix.
RETURN

Subroutine ODTAB (2)

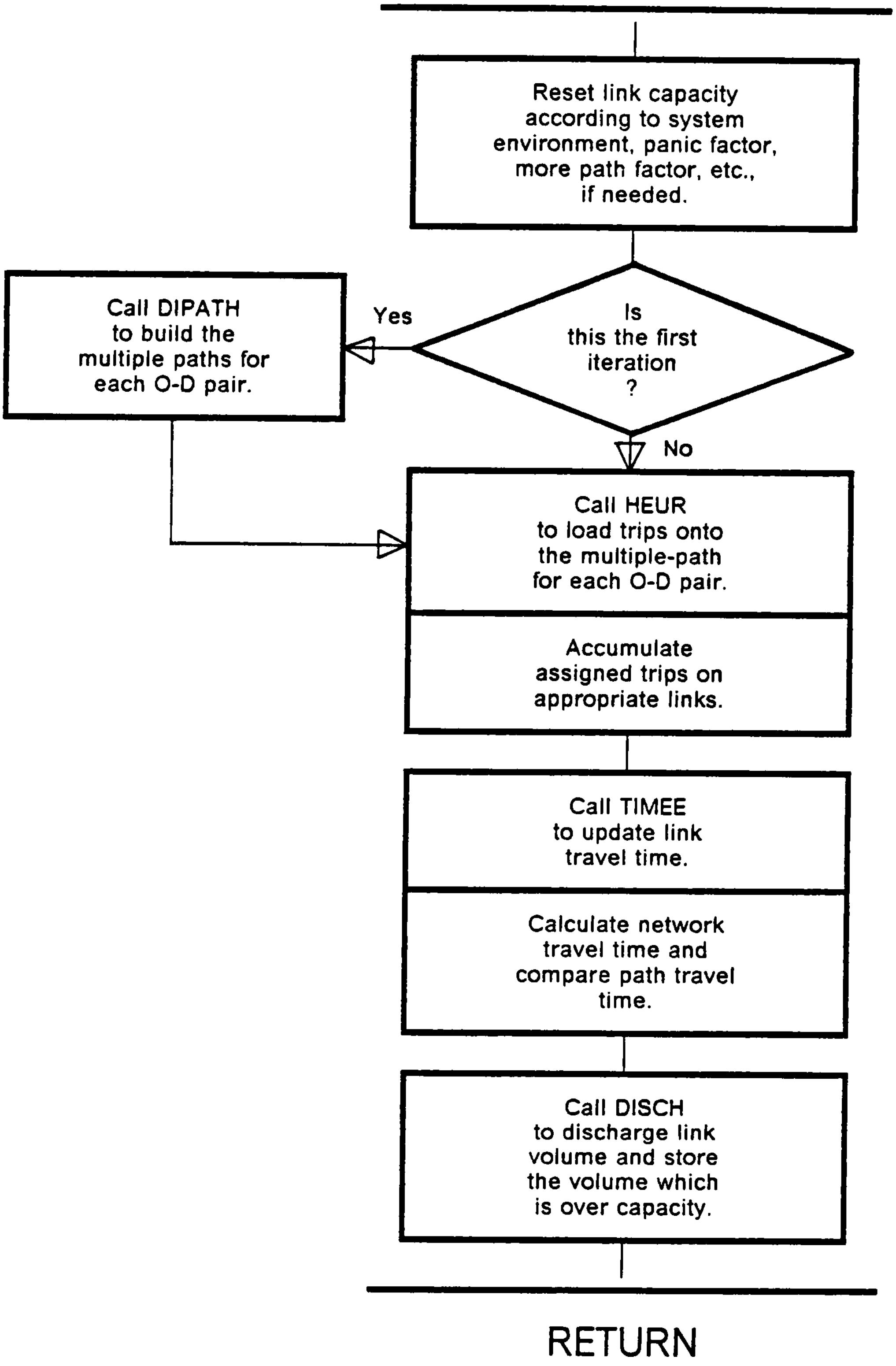


RETURN

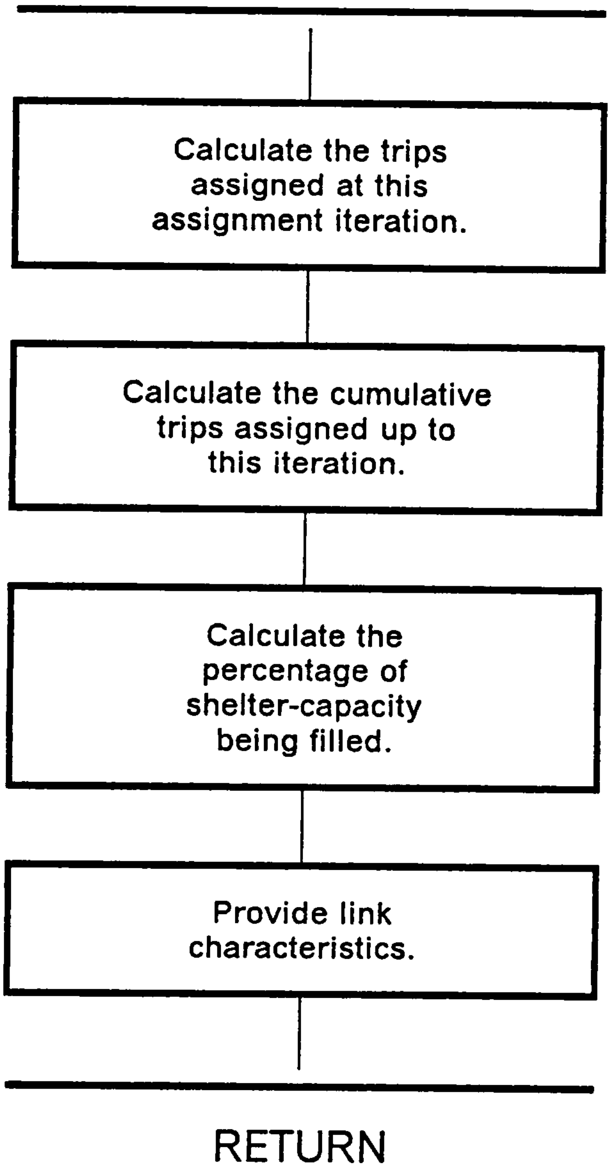
Subroutine SYSIN



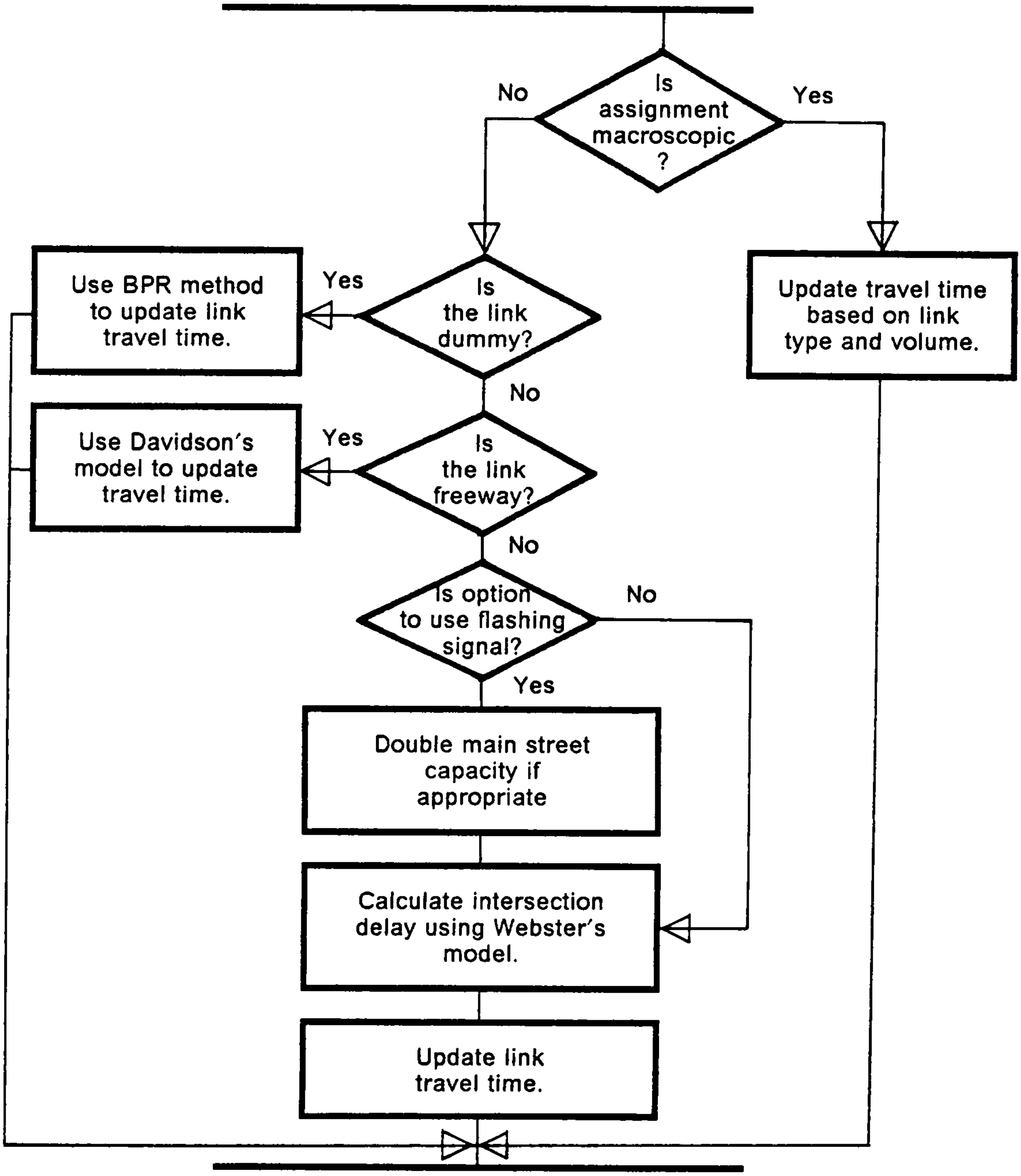
Subroutine STOCH



Subroutine SYSOUT

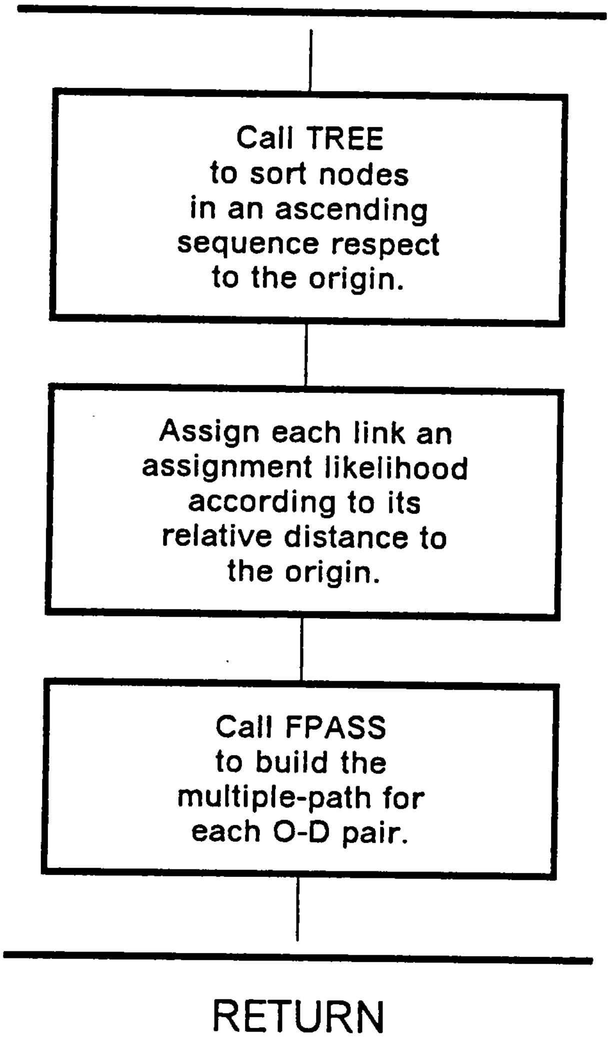


Subroutine TIMEE

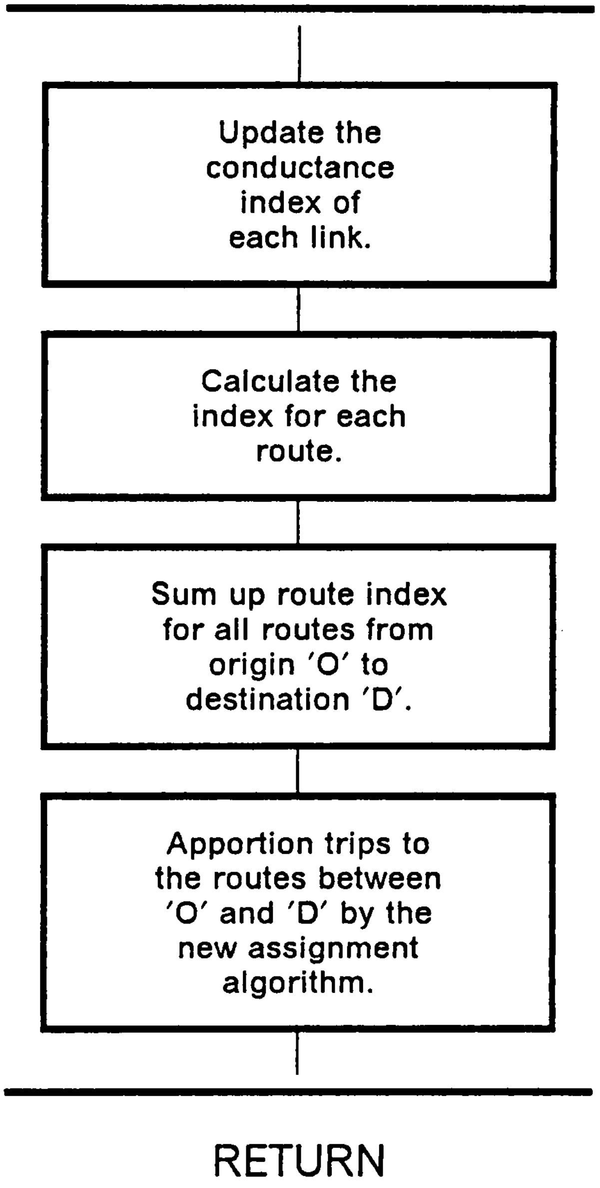


RETURN

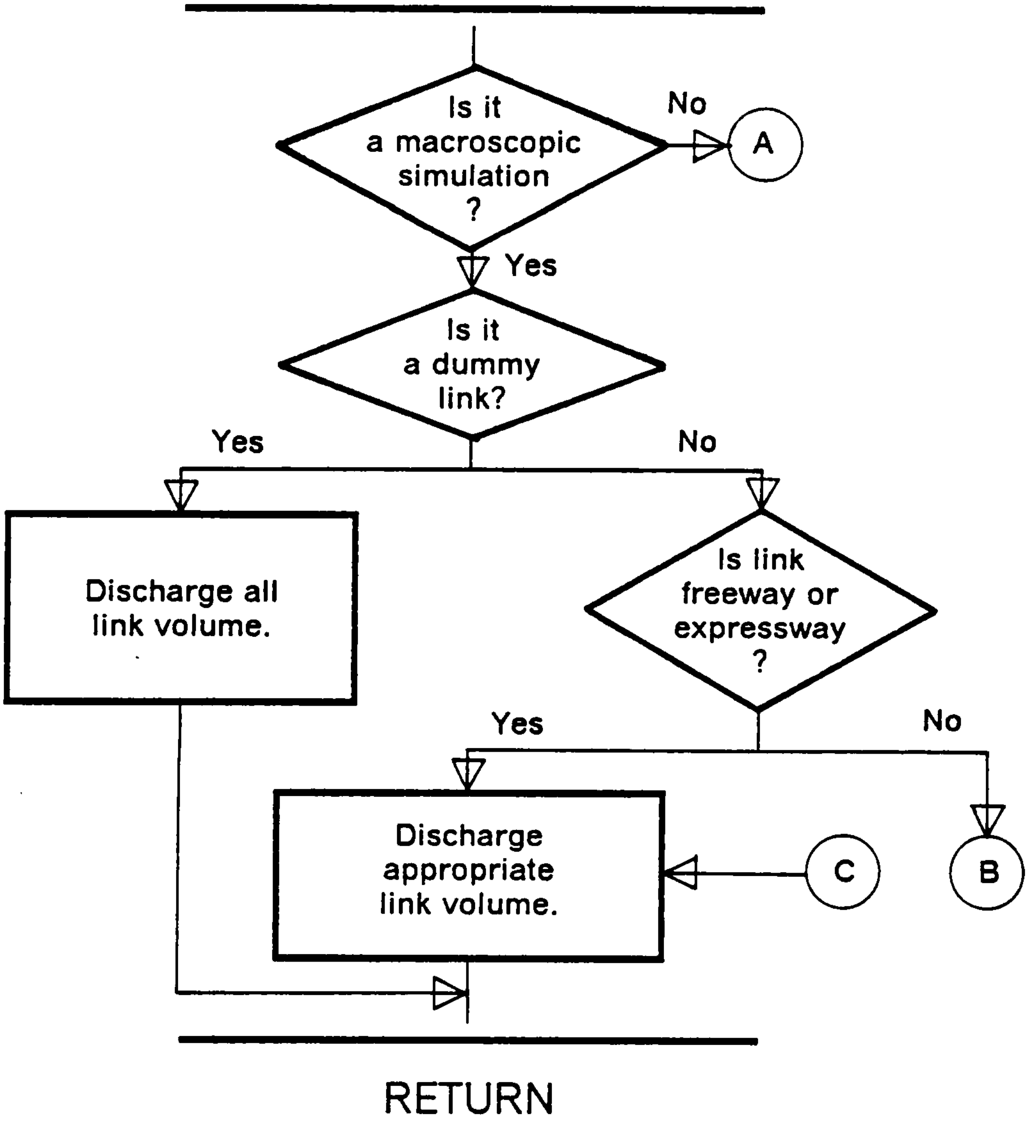
Subroutine DIPATH



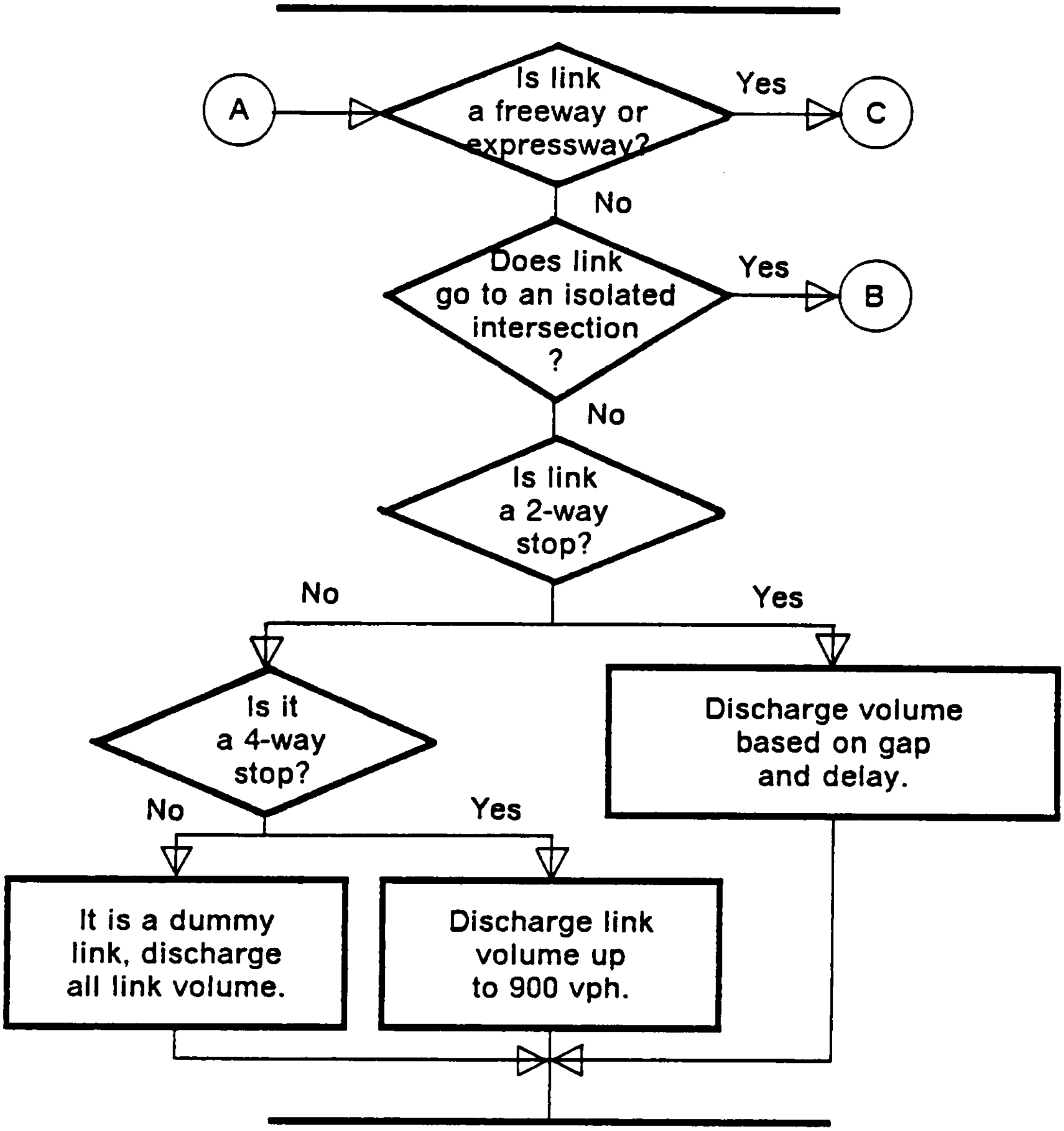
Subroutine HEUR



Subroutine DISCH (1)

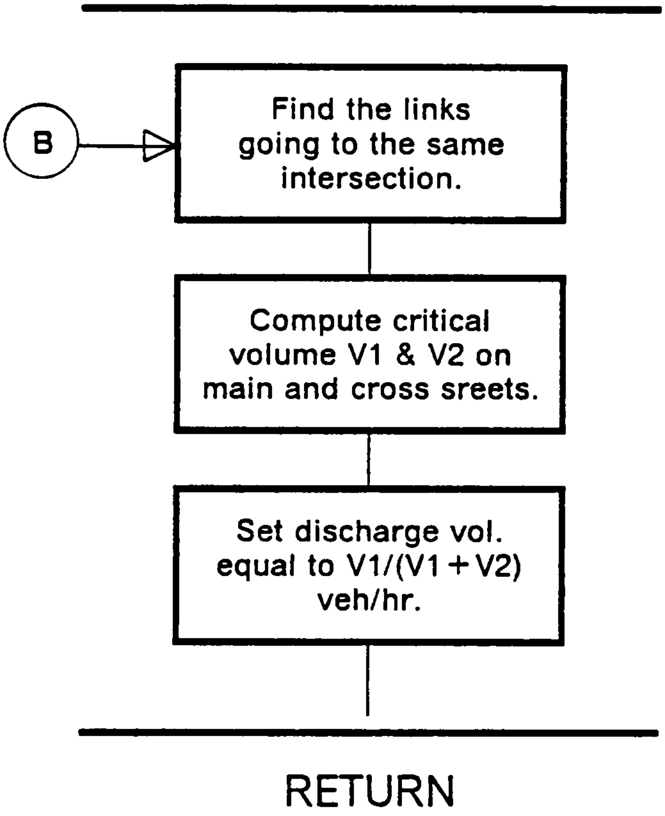


Subroutine DISCH (2)

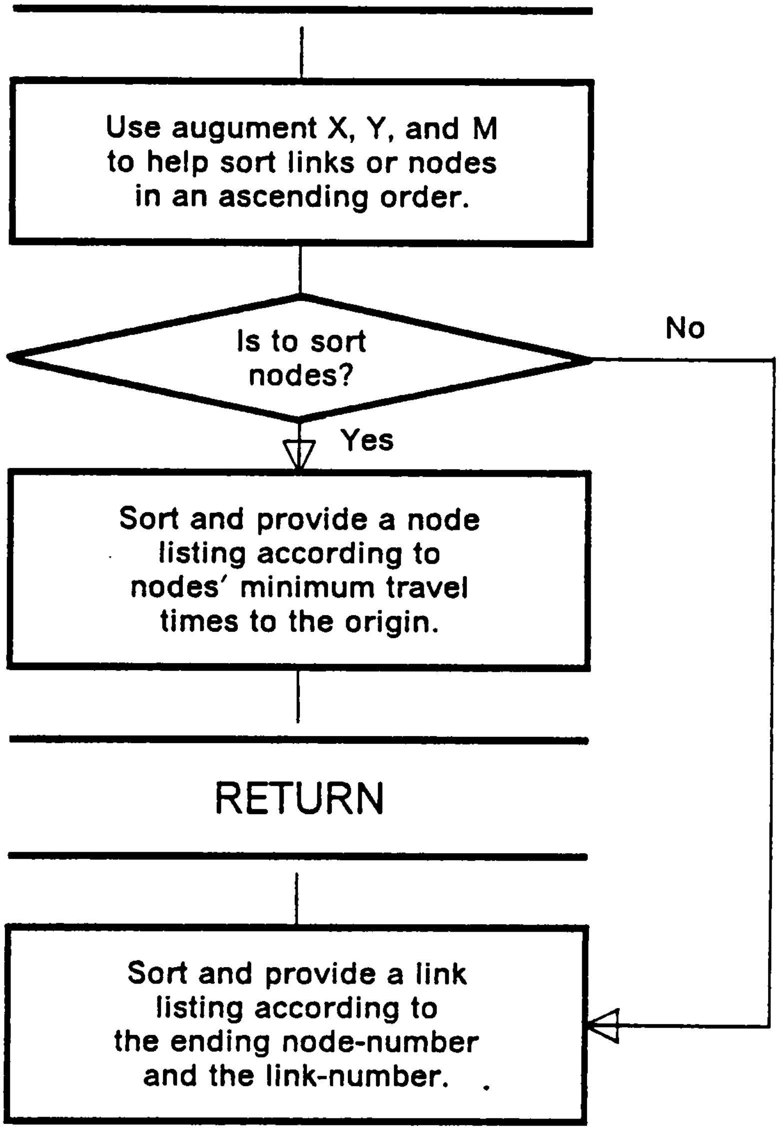


RETURN

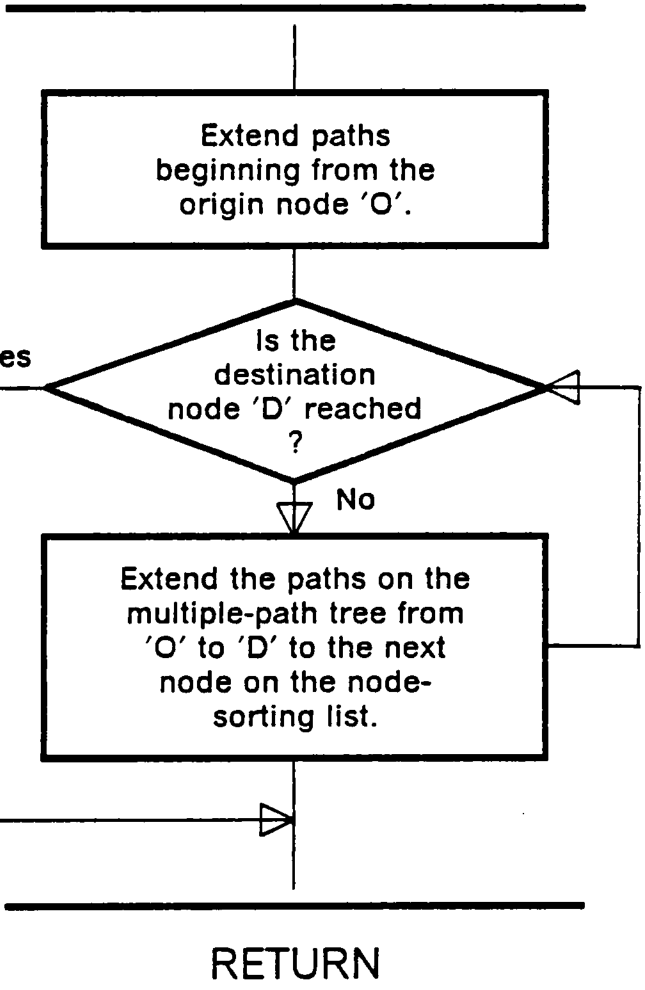
Subroutine DISCH (3)

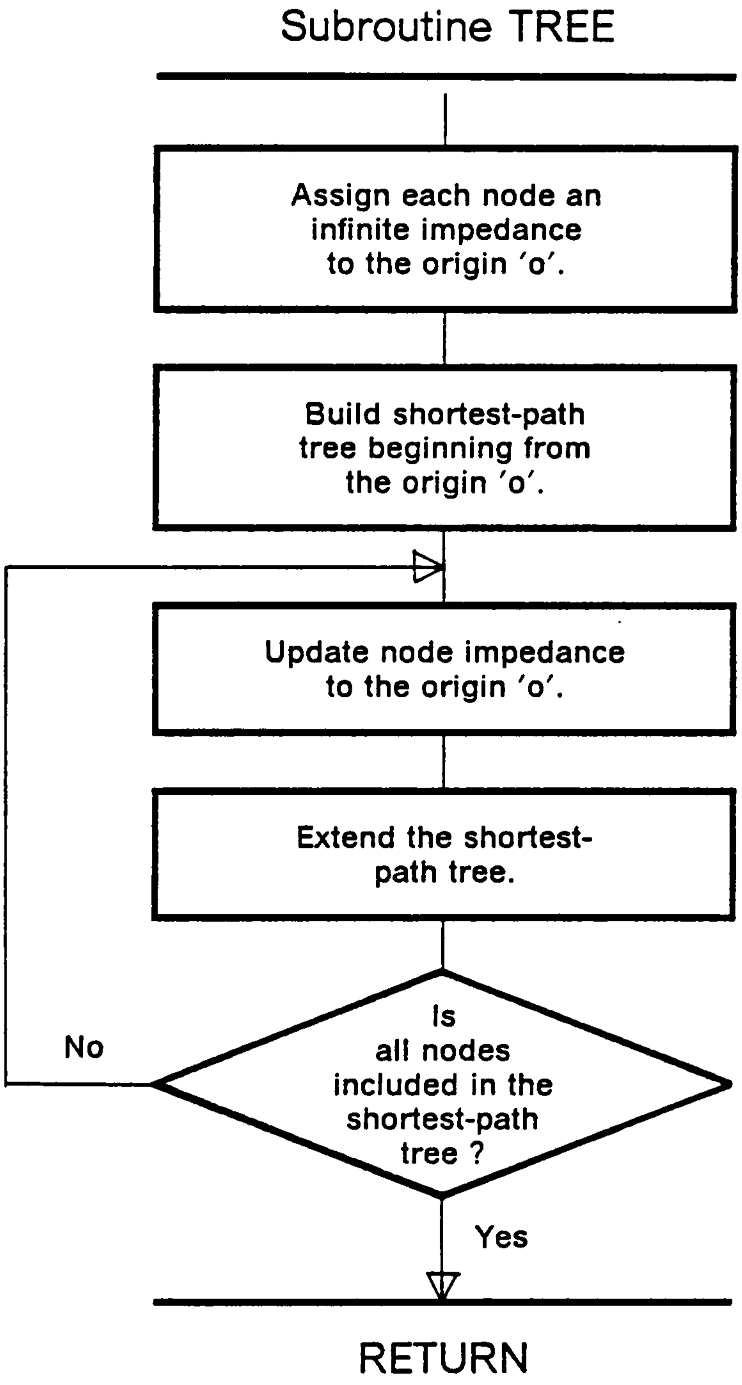


Subroutine SORTD



Subroutine FPASS




Subroutine TREE
Assign each node an infinite impedance to the origin 'o'.
Build shortest-path tree beginning from the origin 'o'.
Update node impedance to the origin 'o'.
Extend the shortest-path tree.
Is all nodes included in the shortest-path tree ?
No
Yes
RETURN

APPENDIX D. PROGRAM VARIABLE DEFINITION

A = LINK STARTING NODE (A-NODE)

AE = LINK ASSIGNMENT LIKELIHOOD

AINDX = THE LEADING LINK AMONG ALL LINKS USING NODE I AS STARTING NODE

ALI = LINK INDEX FOR HEURISTIC ASSIGNMENT

B = LINK ENDING NODE (B-NODE)

BB = GAP ACCEPTANCE DELAY (SECONDS)

BINDX = THE LEADING LINK AMONG ALL LINKS USING NODE I AS ENDING NODE

BNT = MINIMUM TRAVEL TIME FROM ROOT NODE TO NODE I

C = LINK CAPACITY (VPH)

CDEN = MAXIMUM NO. OF VEHICLE A LINK CAN HOLD (VEH)

CDISC = MAXIMUM LINK VOLUME (4/27 Kj * Vf)

CHOICE = O-D TABLE GENERATION OPTION

COI = CUMULATIVE ASSIGNED TRIPS UP TO CLOCK TIME ID

CYCT = CYCLE LENGTH

D = LINK DISTANCE (MILE)

DEN = VOLUME OF A CONGESTED LINK (VEH/TS)

DI = SHELTER OR DESTINATION CAPACITY (VEH)

DIG = AVAILABLE DI AFTER O-D TABLE HAS BEEN PARTIALLY DETERMINED

DP = % OF SHELTER CAPACITY BEING FILLED

EL = LINK PANIC FACTOR

EV = EXISTING TRAFFIC ON A LINK

EZ = LINK SUITABILITY FACTOR

G = CYCLE TIME ASSIGNED TO AN APPROACH (SECOND)

GAMMA = INVERSE OF GAP ACCEPTANCE DELAY

ID = SIMULATION CLOCK TIME (MIN)

IDEL = COMPUTED VEHICLE DELAY (SECOND/VEH)

IDST = THE BEGINNING NODE NUMBER OF ORIGINS

IND = NUMBER OF PROCESSED ITERATIONS OF AN O-D PAIR

IPR1 = PATHS PRINTING OPTION

IPR2 = LINK-CHARACTERISTICS PRINTING OPTION

ISCEN1 = USING SHOULDER OPTION

ISCEN2 = SEEDING NETWORK OPTION

ISCEN3 = USING FLASH-OPERATION OPTION

ISCEN4 = ONE-WAY TRAFFIC CONTROL OPTION

ITERD = DUMMY INDEX TO EXECUTE HEURISTIC ASSIGNMENT

JFAC = OPTION TO CAUSE A DESTINATION NODE HAVE MORE PATHES

LD = ARRAY TO RECORD THE O-D OF A PATH

LEND = ASSIGMENT-END INDEX FOR EACH O-D PAIR

LENGTH = PATH LENGTH

LV = ARRAY CONTAINING LINKS SEQUENCE OF PATHS

LV1 = ARRAY CONTAINING (DESTINATION, PATH NO., LINK SEQUENCE)

LN = ARRAY CONTAINING DESTINATION AND PATH'S LENGTH

LTYPE = LINK TYPE (ISOLATED SIGNALIZED, 2-WAY STOP, 4-WAY STOP, EXPRESSWAY, DUMMY)

MACRO = OPTION OF SIMULATION LEVEL, MACROSCOPIC=1.

MINT = TRAVEL TIME ARRAY FOR NODES IN A SHORTEST PATH TREE

MLINK = WARNING FOR PATH LENGTH CONSTRAINT

MPATH = WARNING FOR PATH NUMBER CONSTRAINT

NODEF = INDEX TO SHOW TRIPS BETWEEN AN O-D PAIR

NLINK = TOTAL LINKS NUMBER IN THE NETWORK

NODES = HIGHEST NODE NUMBER IN THE NETWORK

NTYPE = LINK TYPE ARRAY

O = TRIP PRODUCTION OF A ORIGIN

OG = PRODUCTION LEFT AFTER O-D TABLE IS PARTIALLY DECIDED

OUTPUT = NUMBER OF VEHICLES DISCHARGED FROM A CONGESTED LINK

P = NUMBER OF PATHS HAVE BEEN BUILT FOR AN O-D PAIR

P1 = NUMBER OF PATHS FOR AN O-D PAIR AFTER NEW PATHS ADDED

PARM1 = DIVERSION PROBABILITY (THETA) FOR DIAL'S ASSIGNMENT

PER = % OF TOTAL TRIPS TO BE LOADED IN AN INTERVAL

PET = EVACUATION TIME AFTER AN INCREMENTAL ASSIGNMENT

PN = NUMBER OF SHELTERS, WHICH AN ORIGIN PRODUCTION ARE ASSIGNED

PWT = PATH WEIGHT FOR HEURISTIC ASSIGNMENT

POI = TRIPS ASSIGNED IN AN INTERVAL

PROD = FRACTION OF AN PRODUCTION TO BE LOADED

RVL = REMAINING VOLUME ON A LINK

S = LINK SPEED (MPH)

SDI = SUMMATION OF TOTAL SHELTER CAPACITY (VEHICLES)

SO = LINK FREE-FLOW SPEED (MPH)

SOI = SUMMATION OF TOTAL PRODUCTION (VEHICLES)

STM = CUMULATED VEHICLE TRAVEL TIME UP TO TIME ID

STR = LINK NAME ARRAY

T = LINK TRAVEL TIME (MINUTES)

THL = THE TIME AT WHICH HALF OF TOTAL TRIPS ARE LOADED

TM = TOTAL VEHICLE TRAVEL TIME IN AN INTERVAL

TO = LINK FREE-FLOW TRAVEL TIME

TRIPS = O-D TRIPS TABLE

TRIP = O-D TRIPS TABLE FOR AN ASSIGNMENT ITERATION

TS = VEHICLE-LOADING TIME INTERVAL

V = LINK-VOLUME ASSIGNED FROM AN ORIGIN (VEH/TS)

VC = VOLUME/CAPACITY RATIO

VN = TRIP-VOLUME AT A NODE

VOLU = OVERALL VOLUME ON A LINK

VSEED = EXISTING TRAFFIC ON A LINK BEFORE VEHICLE-LOADING

V1 = ARRIVAL RATE ON A SUBJECT APPROACH (VEH/SECOND)

V2 = ARRIVAL RATE ON A PRIORITY APPROACH (VEH/SECOND)

WE = LINK ASSIGNMENT WEIGHT

X = CRITICAL ARRIVAL RATE (VEH/SECOND)

ZO = LOCATION OF A LINK IN THE EVACUATION ZONE

ZZ = ENDING NODE (B-NODE) OF A DOWNSTREAM LINK

APPENDIX E. PROGRAM LISTING

```
C     -------------------------
C     |  HEUPRAE-1 MAIN    |
C     -------------------------

       COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
      *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
      *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
      *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
      *POI,STM,LEND
       DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
      *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
      *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
      *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
      *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
       INTEGER A,B,AINDX,ZO,ELEV
       COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
       INTEGER SR,CR,RL,OPT,ZZ
       COMMON /CDIAL/ BD,BDL,MINT,BN,BINDX
       DIMENSION BD(336),BDL(336),MINT(105),BN(105),BINDX(105)
       INTEGER BINDX
       INTEGER MINT,BD,BDL,BN
       COMMON /CPATHA/ PN(41)
       INTEGER PN

       CALL INITIA
       CALL ODTAB
       CALL EVAC

       STOP
       END

C     -------------------------
C     |      INITIAL         |
C     -------------------------
C     PURPOSE: CHOOSE THE EVACUATION OPT, READ LINKS DATA, SORT LINKS
C     DATA, AND  ASSIGN THE INITIAL VALUE TO VARIABLES.
       SUBROUTINE INITIA
       COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
      *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
      *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
      *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
      *POI,STM,LEND
       DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
      *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
      *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
      *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
      *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
       INTEGER A,B,AINDX,ZO,ELEV
       COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
       INTEGER SR,CR,RL,OPT,ZZ
       COMMON /CDIAL/ BD,BDL,MINT,BN,BINDX
       DIMENSION BD(336),BDL(336),MINT(105),BN(105),BINDX(105)
```

```
      INTEGER BINDX,CN
      INTEGER MINT,BD,BDL,BN

C      -------------------------
C      |      OPTION           |
C      -------------------------
C      PURPOSE: INITIAL THE DECISION TO RUN MASSVAC2
      READ(5,10) IGN,IDS,ITE,NLINK
10     FORMAT(4I5)
      IDST=IGN+1
130    READ(5,140)MACRO,TS,THL,ISCEN1,ISCEN2,ISCEN3,ISCEN4,IPR1,IPR2
140    FORMAT(I5,2F5.0,6I5)
      READ(5,150) (JFAC(I), I=1,15)
150    FORMAT(15I2)
      READ(5,160) CHOICE
160    FORMAT(F6.0)
      READ(5,*) OPT,PARM1

C      -------------------------
C      |       RLINK           |
C      -------------------------
C      PURPOSE: READ LINK DATA
      DO 1010 I=1,NLINK
1010    READ(5,1015)ZO(I),A(I),B(I),ZZ(I),D(I),ELEV(I),C(I),SO(I),FL(I),
     *EL(I),EZ(I),NTYPE(I),VSEED(I),(STR(K,I),K=1,3)
1015   FORMAT(I3,1X,I3,1X,I3,1X,I3,F4.1,1X,I2,1X,F5.0,F3.0,3F5.0,I2,1X,
     *F5.0,1X,3A4)
      NODES=0
      DO 1065 LINK=1,NLINK
      IF(A(LINK).GT.NODES) NODES=A(LINK)
1065  IF(B(LINK).GT.NODES) NODES=B(LINK)
C      LOOP 1065 DETERMINES THE VALUE OF NODES.

      CN=A(1)
      AINDX(CN)=1
C      LOOP 190 FINDS LEADING LINK FOR EACH NODE
      DO 190 LINK=2,NLINK
       IF(A(LINK).EQ.CN) GO TO 190
        CN=A(LINK)
        AINDX(CN)=LINK
190    CONTINUE
      DO 1380 I=1,NLINK
       BD(I)=B(I)
1380   BDL(I)=I

      CALL SORTD(BD,BDL,NLINK)
C      THIS ST SORTS LINKS IN ASCENDING SEQUENCE BY NO. OF DESTINATION
C      NODE AND NO. OF LINK. LINKS WITH SAME DEST NODE WILL BE GROUPED.
C      BDL(I) RECORDS WHICH LINK IS AT ITH POS. ON THIS ASCENDING LIST.
      CN=BD(1)
C      CN IS THE B NODE OF LINK AT POS. 1
      BINDX(CN)=1
C      BINDX(NODE) IS THE POS. OF THE LEAD PRECEDING LINK TO ROOT NODE
C      ON THE ASCENDING DESTINATION LIST.
      DO 1390 LINK=2,NLINK
       IF(BD(LINK).EQ.CN)GO TO 1390
```

```
        CN=BD(LINK)
        BINDX(CN)=LINK
 1390  CONTINUE
 C     LOOP 1390 IS USED TO FIND THE LEAD PRECEDING LINK FOR EACH NODE

       DO 1020 I=1,NLINK
        TO(I)=D(I)/SO(I)*60.
        CDEN(I)=D(I)*5280./18.
        IF(NTYPE(I).EQ.4.OR.NTYPE(I).EQ.5) CDEN(I)=CDEN(I)*C(I)/2000.
        IF(NTYPE(I).NE.4.AND.NTYPE(I).NE.5)CDEN(I)=CDEN(I)*C(I)/900.
 1020  CDISC(I)=4./27.*(CDEN(I)/D(I))*SO(I)
 C     THE ST ABOVE IS FROM DREW'S BOOK. IT IS LINK'S MAX VOLUME. Q=K*V

 C     ----------------------------
 C     |       ASSIGN             |
 C     ----------------------------
 C     PURPOSE: ASSIGN INITIAL VALUE TO VARIABLES
       PET=0.
       DO 40 I=1,NLINK
        DEN(I)=0.
        VOLU(I)=0.
        RVL(I)=0.
        DO 40 J=1,IDS
 40     V(I,J)=0.
       DO 50 I=1,IDS
        DO 50 J=1,IGN
         IND(I,J)=0
         LEND(I,J)=0
        TRIP(I,J)=0.
 50     TRIPS(I,J)=0.
       DO 220 I=1,NLINK
        T(I)=TO(I)
        S(I)=SO(I)
        VC(I)=0.
 220   CONTINUE
       ID=0
       STM=0.

       RETURN
       END

 C     ----------------------------
 C     |    ODTABLE              |
 C     ----------------------------
 C     PURPOSE: READ THE OD TRIP TABLE OR INTERNALLY GENERATE THE
 C     OD TABLE AND PRINT THE TOTAL PRODUCTION AND SHELTER CAPACITY.
       SUBROUTINE ODTAB

       COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
      *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
      *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
      *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
      *POI,STM,LEND
       DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
      *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
      *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
```

```
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /CTREE/ FN,BNT,TSEQ
      DIMENSION FN(105),BNT(105),TSEQ(105)
      INTEGER FN
      DIMENSION TTT(41),TT(41,15),O(41),DI(15),IEM(41),OG(41),DIG(15)
      COMMON /CPATHA/ PN(41)
      INTEGER PN
      SOI=0.
      SDI=0.
C     SOI=SUM OF TOTAL PRODUCTION, SDI=SUM OF TOTAL SHELTER CAPACITY.
      IF(CHOICE.NE.2) GO TO 270
250   READ(5,260) ((TRIP(I,J),I=IDST,IDS),J=1,IGN)
260   FORMAT(13F5.0)
      DO 265 I=IDST,IDS
       DO 265 J=1,IGN
265   SOI=SOI+TRIP(I,J)
      WRITE(6,266) SOI
266   FORMAT(4X,'THE O-D TRIP TABLE IS GIVEN, TOTAL PROD IS ',F7.1)
      GO TO 600
270   READ(5,280) (O(I),I=IDST,IDS)
280   FORMAT(10F6.0)
      IF(CHOICE.EQ.0) GO TO 500
      READ(5,280) (DI(I),I=1,IGN)
C     THE STATEMENTS FOLLOWED ARE USED TO GENERATE TRIP TABLE BY MINIMUM
C     TRAVEL TIME ALLOCATION METHOD WITH ORIGIN TRIP PROD AND SHELTER
C     CAPACITY.
      DO 290 I=IDST,IDS
290   SOI=SOI+O(I)
      DO 300 I=1,IGN
300   SDI=SDI+DI(I)
      ICAP=0
      IF(SOI.GT.SDI) ICAP=1
C     ICAP IS THE FLAG WHICH STANDS FOR PROD > CAPACITY
      IF(ICAP.EQ.1) THEN
      SDF=SOI-SDI
      SS=SDF/SDI*100.
      WRITE(6,*) '**** WARNING ****'
      WRITE(6,305) SOI,SDI
      WRITE(6,310) SDF,SS
305   FORMAT(4X,'TOTAL PROD =',F8.1,' > TOTAL SHELTER CAP =',F8.1)
310   FORMAT(4X,'BY DIFF =',F8.1,' WHICH IS  ',F5.2,'% OF TOTAL CAP')
      ELSE
      SS=SOI/SDI
      WRITE(6,311) SOI,SDI,SS
311   FORMAT(4X,'TOTAL PROD = ',F8.1,' TOTAL CAP = ',F8.1,' THE PROD/CAP
     + = ',F5.3)
      END IF
315   DO 350 IOR=IDST,IDS
       CALL TREE(IOR)
       DO 350 J=1,IGN
        TT(IOR,J)=BNT(J)
350   CONTINUE
```

```
      DO 370 I=IDST,IDS
370    OG(I)=O(I)
      DO 380 J=1,IGN
380    DIG(J)=DI(J)
C      LOOP 400 FIGURES OUT WHICH SHELTER J IS CLOSEST TO PROD I
      DO 405 I=IDST,IDS
       IF(O(I).EQ.0.) GO TO 405
       AT=9999.
       DO 400 J=1,IGN
        IF(DIG(J).EQ.0.) GO TO 395
C        THIS AVOIDS ASSIGNING TRIP TO NO CAP SHELTER (RESERVED) WHEN
C        TOTAL PROD > TOTAL CAP.
         IF(TT(I,J).GE.AT) GO TO 400
          AT=TT(I,J)
          IEM(I)=J
          GO TO 400
395       TT(I,J)=9999.
400     CONTINUE
       WRITE(6,*) IEM(I),' IS THE CLOSEST SHELETER TO ORIGIN ',I
C       THE ABOVE ST. RECORDS THE CLOSEST SHELTER TO ORIG I IS J.
405    CONTINUE
C      GENERATE OD TAB BEGIN WITH OD PAIR OF THE SMALLEST TT(I,J)
410    BT=9999.
      IORI=0
      IDES=0
      DO 420 I=IDST,IDS
       DO 420 J=1,IGN
        IF(OG(I).EQ.0.) GO TO 420
         IF(TT(I,J).GE.BT) GO TO 420
          BT=TT(I,J)
          IORI=I
          IDES=J
420    CONTINUE
      IF((IORI.EQ.0).AND.(ICAP.EQ.0)) GO TO 600
C      END OF OD TABLE GENERATION
      IF(IORI.EQ.0) GO TO 480
C      ABOVE ST MEANS ALL SHELTER CAPACITY HAS BEEN FILLED OUT
      IF(OG(IORI).GT.DIG(IDES)) GO TO 440
      TRIP(IORI,IDES)=TRIP(IORI,IDES)+OG(IORI)
      DIG(IDES)=DIG(IDES)-OG(IORI)
      OG(IORI)=0.
      WRITE(6,430) TRIP(IORI,IDES),IORI,IDES
430    FORMAT(4X,'RENEWED O-D TRIP =',F8.0,' ORIG=',I2,' DEST=',I2)
      IF(DIG(IDES).GT.0.) GO TO 410
      GO TO 450
440    TRIP(IORI,IDES)=TRIP(IORI,IDES)+DIG(IDES)
      OG(IORI)=OG(IORI)-DIG(IDES)
      DIG(IDES)=0.
      WRITE(6,445) TRIP(IORI,IDES),IORI,IDES
445    FORMAT(4X,'RENEWED O-D TRIP =',F8.0,' ORIG=',I2,' DEST=',I2)
450    DO 460 I=IDST,IDS
460    TT(I,IDES)=9999.
      GO TO 410
C      ALL SHELTER CAPACITY HAS BEEN FILLED OUT. HYPOTHETICALLY
C      INCREASE THE CAPACITY TO ACCOMODATE THE TRIP
480    DO 490 I=IDST,IDS
```

```
      IF(OG(I).EQ.0.) GO TO 490
       TRIP(I,IEM(I))=TRIP(I,IEM(I))+OG(I)
       WRITE(6,485) OG(I),IEM(I),I
485   FORMAT(4X,'CAPACITY HAS BEEN HYPOTHETICALLY INCREASED BY ',F8.0,
     +'AT SHELTER=',I2,' AS THE TRIP FROM ORIG=',I2)
     WRITE(6,486) TRIP(I,IEM(I))
486   FORMAT(4X,'RENEWED O-D TRIP WITH CAPACITY INCREASED  =',F8.0)
     OG(I)=0.
490   CONTINUE
     GO TO 600
C     GENERATE OD TAB W/O SHELTER CAPACITY CONSTRAINT BY GRAVITY METHOD
500   FACT=1.
C     FACT IS THE EXPONT OF THE SPATIAL SEPARATION BETWEEN O-D. TRAVEL
C     TIME IS CURRENTLY USED AS SPATIAL SEPARATION IN THIS MODEL.
C     THE FRICTION TABLE IS ALSO AN ALTERNATIVE TO COUNT THE FRICTION.
     DO 510 IOR=IDST,IDS
      TTT(IOR)=0.
      CALL TREE(IOR)
      DO 510 J=1,IGN
       TT(IOR,J)=BNT(J)
       TTT(IOR)=TTT(IOR)+(1/TT(IOR,J)**FACT)
510   CONTINUE
     DO 530 I=IDST,IDS
      SOI=SOI+O(I)
      DO 530 J=1,IGN
       LL=IFIX(O(I)/TTT(I)/TT(I,J)**FACT+0.5)
       TRIP(I,J)=FLOAT(LL)
       IF(LL.GT.0) WRITE(6,525) TRIP(I,J),I,J
525   FORMAT(4X,'TRIP(O,D)=',F7.0,' FROM O=',I2,' TO D=',I2)
530   CONTINUE
     WRITE(6,531) SOI
531   FORMAT(4X,'NO SHELTER HAS CAP CONSTRAINT, TOTAL PROD = ',F8.1)
600   DO 605 I=1,IDS
605   PN(I)=0
     DO 610 I=IDST,IDS
      DO 610 J=1,IGN
       IF(TRIP(I,J).GT.0.) PN(I)=PN(I)+1
610   CONTINUE
     DO 620 I=IDST,IDS
      IF(PN(I).EQ.0) GO TO 620
       WRITE(6,630) I,PN(I)
620   CONTINUE
630   FORMAT(4X,'PROD FROM ORIGIN ',I3,' GOES TO ',I3,' DIFFERENT DEST')

     RETURN
     END

C     ---------------------------
C     |    TREE(ROOT)           |
C     ---------------------------
C     PURPOSE: TO BUILD THE SHORTEST PATH TREE FROM THE ROOT NODE TO
C     EVERY NODE IN THE NETWORK BY MOOR'S ALGORITHM

     SUBROUTINE TREE(ORIG)
     COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
    *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
```

```
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /CTREE/ FN,BNT,TSEQ
      DIMENSION FN(105),BNT(105),TSEQ(105)
      INTEGER FN,FROM,TOO,ORIG
      K=IDS+1
C     THE PURPOSE OF K IS EXPLAINED IN DO LOOP 1340
      DO 1310 I=1,NODES
       FN(I)=0
       BNT(I)=9999.
1310  TSEQ(I)=9999.
      BNT(ORIG)=0.
C     EXTEND THE TREE FROM ROOT NODE-ORIG
      LINK=AINDX(ORIG)
C     LINK IS THE FIRST POSITION WITHIN LINKS WITH NODE ORIG AS ITS
C     STARTING NODE
      FROM=ORIG
1320  CUMT=BNT(FROM)+T(LINK)
      TOO=B(LINK)
      IF(CUMT.GE.BNT(TOO)) GO TO 1330
      BNT(TOO)=CUMT
      FN(TOO)=FROM
C     USE FN TO RECORD THE PRECEDING NODE FROM FOR NODE TOO IN THE TREE
      TSEQ(TOO)=CUMT
1330  LINK=LINK+1
      IF(A(LINK).EQ.FROM) GO TO 1320
C     THE LINKS WITH SAME STARTING NODE HAVE BEEN SORTED/PUT IN SEQUENCE
C     FIND THE NEXT STARTING NODE WITH THE SMALLEST TRAVEL COST TO ROOT
C     NODE FROM THE END NODES OF EXISTING TREE
      FROM=0
      BIG=9999.
C     SINCE THE ORIGIN OR DESTINATION NODES ARE CONNECTED TO THE NETWORK
C     BY A SINGLE DUMMY ARC, THEY CAN BE CONSIDERED AS DEAD ENDS AND NO
C     LINK WILL EXTEND OUT FROM THEM. THUS LOOP 1340 BEGINS FROM IDS+1.
      DO 1340 I=K,NODES
       IF(TSEQ(I).GE.BIG) GO TO 1340
        FROM=I
        BIG=TSEQ(I)
1340  CONTINUE
      IF(FROM.EQ.0) RETURN
      TSEQ(FROM)=9999.
C     THE ABOVE ST MAKES NODE FROM WILL NOT BE A STARTING NODE AGAIN.
      LINK=AINDX(FROM)
      GO TO 1320
      END
```

```
C   ------------------------------------------
C   | EVACUATION  SIMULATION |
C   ------------------------------------------
C   PURPOSE: PROCESS THE SIMULATION OF EVACUATION UNTIL THE NETWORK
C   IS CLEARED.
     SUBROUTINE EVAC

      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CDIAL/ BD,BDL,MINT,BN,BINDX
      DIMENSION BD(336),BDL(336),MINT(105),BN(105),BINDX(105)
      INTEGER BINDX
      INTEGER MINT,BD,BDL,BN
      COMMON /CTREE/ FN,BNT,TSEQ
      DIMENSION FN(105),BNT(105),TSEQ(105)
      INTEGER FN,FROM,TOO
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /INDX/ I
      COMMON /CPATHA/ PN(41)
      INTEGER PN

70    CALL SYSIN
      CALL STOCH
      CALL SYSOUT
      GO TO 70

71    RETURN
      END

C   --------------------------------
C   |       SYSIN                  |
C   --------------------------------
C   PURPOSE: PREPARE INPUT DATA FOR EACH SIMULATION ITERATION DURING
C   NETWORK EVACUATION. THE INPUT DATA INCLUDE TRIPS, PROD, AND ID. IF
C   NO MORE TRIPS(I,J) WILL BE ASSIGNED TO THE NETWORK, SYSIN WILL
C   DISCHARGE NETWORK EXISTING VOL AND PRINT NETWORK EVACUATION
C   TIME. HEUPRAE THEN STOPS.
      SUBROUTINE SYSIN

      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
```

```
      *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
      *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
      *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
       INTEGER A,B,AINDX,ZO,ELEV
       COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
       INTEGER SR,CR,RL,OPT,ZZ
       Z=ALOG(49.)/THL
C       Z: LOADING PARAMETER
       H=THL
C       THL: HALF LOADING TIME, =1/2 ET
480    ID=ID+TS
       DO 510 I=IDST,IDS
        DO 510 J=1,IGN
         IF(LEND(I,J).EQ.1) PER=0.
         IF(LEND(I,J).EQ.1) GO TO 510
          IND(I,J)=IND(I,J)+1
          TDI=TS*IND(I,J)
          PAR=1/(1+EXP(-Z*(TDI-H)))
          IF(PAR.LT.0.95) GO TO 485
           PAR=1.0
           LEND(I,J)=1
485         IF(IND(I,J).EQ.1)GO TO 490
           TDIP=TS*(IND(I,J)-1)
           PARP=1/(1+EXP(-Z*(TDIP-H)))
           PER=PAR-PARP
           GO TO 510
490         PER=PAR
510     TRIPS(I,J)=TRIP(I,J)*PER
       DO 520 I=IDST,IDS
        DO 520 J=1,IGN
         IF(TRIPS(I,J).GE.1) GO TO 700
520     CONTINUE
       GO TO 540
C      LOOP 530 CHECKS FOR CONGESTED LINKS, DISCHARGES VOLUME, AND
C      COMPUTES NETWORK CLEARANCE TIME IF NO MORE TRIP WILL BE ASSIGNED
C      TO NETWORK FROM THE ORIGINS. THEN THE PROGRAM STOPS.
530     ID=ID+TS
       WRITE(6,535)
535     FORMAT('0','       ')
540     DO 610 I=1,NLINK
        IF(RVL(I).EQ.0)GO TO 610
         IF(RVL(I).LE.((C(I)/60.)*TS))GO TO 580
          OUTPUT=CDISC(I)*TS/60.
          IF(OUTPUT.GT.RVL(I))OUTPUT=RVL(I)
          RVL(I)=RVL(I)-OUTPUT
          CAP=C(I)/60.*TS
          WRITE(6,550)I,ZO(I),RVL(I),CAP,(STR(K,I),K=1,3),A(I),B(I)
550     FORMAT(' ','CONGESTED LINK',2X,I4,5X,'ZONE',2X,I5,5X,'VOLUME',2X,
      *F8.0,5X,'CAPACITY',2X,F5.0,5X,3A4,3X,'A-NODE',I3,3X,'B-NODE',I3)
       TE=ID
       IF(TE.GT.PET) GO TO 560
       GO TO 610
560     PET=TE
       WRITE(6,570)PET,ID
570     FORMAT(4X,'EVACUATION TIME1',F10.3,'ID=',I4)
```

```
      GO TO 610
580    V(I,ITE)=RVL(I)
      RVL(I)=0.
      LTYPE=NTYPE(I)

      CALL TIMEE

      TE=ID+T(I)
      IF(TE.GT.PET) GO TO 590
       GO TO 610
590    PET=TE
      WRITE(6,600)PET,ID
600    FORMAT(4X,'EVACUATION TIM2=',F10.3,'ID=',I4)
610    CONTINUE
      DO 620 I=1,NLINK
       IF(RVL(I).NE.0)GO TO 530
620    CONTINUE
      WRITE(6,625)
625    FORMAT('0',' SUMMATION O-D TRIP TABLE')
630    DO 650 K=IDST,IDS
       DO 650 L=1,IGN
        IF(TRIP(K,L).EQ.0)GO TO 650
         WRITE(6,640)K,L,TRIP(K,L)
640    FORMAT(' ','ORIGIN=',I2,5X,'DESTINATION=',I3,5X,'TRIP  =',F5.0)
650    CONTINUE
C     LOOP 650 PRINTS ORIGINAL TRIP TABLE.
      WRITE(6,655)
655    FORMAT('0',' LINK CHARACTERISTICS')
      DO 670 N=1,NLINK
       IF(VOLU(N).EQ.0)GO TO 670
        WRITE(6,660)N,ZO(N),VOLU(N),(STR(K,N),K=1,3),A(N),B(N)
660    FORMAT(' ','LINK',2X,I4,5X,'ZONE',2X,I3,5X,'OVERALL VOLUME',2X,
     *F8.0,5X,3A4,3X,'A-NODE',I3,3X,'B-NODE',I3)
670    CONTINUE
C     LOOP 670 PRINTS LINK'S CHARACTERISTICS IF LINK HAS BEEN USED.
      WRITE(6,672) STM
672    FORMAT('0','TOTAL VEH TRAVEL TIME = ',F9.1,' VEH-MIN')
      WRITE(6,675)PET
675    FORMAT('0',4X,'MAX NETWORK CLEARANCE TIME EQUALS TO',4X,F10.3)
      T20=THL-ALOG(4.)/Z
      TD=PET-T20
      WRITE(6,680) TD
680    FORMAT('0',4X,'THE EVACUATION TIME NEEDED IF 20% OF THE TRIP HAS',
     *' BEEN LOADED BEFORE ISSUING THE EVACUATION ORDER IS ',F10.3)
      RISKT=THL*2.*1.5
C     RISKT IS MAX ALLOWABLE NETWORK CLEARANCE TIME. 1.5 IS RISK FACTOR.
      IF(PET.LT.RISKT) WRITE(6,*) 'EVACUATION TIME IS ACCEPTABLE'
      IF(PET.LT.RISKT) STOP
      WRITE(6,690)
690    FORMAT(4X,'THE LOADING PERIOD IS NOT APPROPRIATE, EVACUATION TIME'
     *,' IS TOO LARGE')
      STOP

700    DO 710 I=IDST,IDS
       PROD(I)=0.0
```

```
       DO 710 J=1,IGN
710     PROD(I)=PROD(I)+TRIPS(I,J)
      WRITE(6,785)
785   FORMAT('0','       ')
      DO 800 K=IDST,IDS
       DO 800 L=1,IGN
        IF(TRIPS(K,L).EQ.0)GO TO 800
         WRITE(6,790)K,L,TRIPS(K,L)
790   FORMAT(' ','ORIGIN=',I2,5X,'DESTINATION=',I3,5X,'TRIP =',F5.0)
800   CONTINUE
C     LOOP 800 PRINTS TRIP TABLE FOR THE PRESENT TIME INTERVAL.
810   WRITE(6,820)
820   FORMAT(//5X,'PROBABILISTIC  TRAFFIC ASSIGNMENT')
830   WRITE(6,840)OPT,PARM1
840   FORMAT(/5X,'PRINT OPTION =',I3,5X,'PARAMETER  AS READ =',F11.5)

      RETURN
      END

C     ------------------------------
C     |        STOCH               |
C     ------------------------------
      SUBROUTINE STOCH

      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      DIMENSION EV(336)
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /CTREE/ FN,BNT,TSEQ
      DIMENSION FN(105),BNT(105),TSEQ(105)
      INTEGER FN,FROM,TOO,ORIG
      COMMON /CDIAL/ BD,BDL,MINT,BN,BINDX
      DIMENSION BD(336),BDL(336),MINT(105),BN(105),BINDX(105)
      INTEGER BINDX
      INTEGER MINT,BD,BDL,BN
      COMMON /INDX/ I
      COMMON /CPATHA/ PN(41)
      INTEGER PN
      COMMON /CPATHB/ LV,LD,NOPA,LENGTH,KPATH
      DIMENSION LV(150,105),LD(150,2),NOPA(41,15),LENGTH(150)
      IF(OPT.GT.1)OPT=1
      IF(ID.GE.(0.75*2*THL))GO TO 1610
C      ID: UPDATED CLOCK TIME, THL: HALF LOADING TIME
      GO TO 1670
1610  IF(ISCEN1.EQ.0)GO TO 1630
C     ISCEN1: SHOULDER USING OPTION
```

```
      DO 1620 II=1,NLINK
       IF(NTYPE(II).EQ.4) C(II)=C(II)+2000.
       CDEN(II)=D(II)*5280./18.
       IF(NTYPE(II).EQ.4.OR.NTYPE(II).EQ.5)CDEN(II)=CDEN(II)*C(II)/2000.
       IF(NTYPE(II).NE.4.AND.NTYPE(II).NE.5)CDEN(II)=CDEN(II)*C(II)/900.
       CDISC(II)=4./27.*(CDEN(II)/D(II))*SO(II)
C      THE UNIT OF CDISC IS VEH/HR
1620  CONTINUE
      ISCEN1=0.
1630  IF(ISCEN4.EQ.0)GO TO 1650
C     ISCEN4: OPT FOR ONE-WAY TRAFFIC
      DO 1640 JJ=1,NLINK
       IF(FL(JJ).EQ.1)C(JJ)=C(JJ)*2
       CDEN(JJ)=D(JJ)*5280./18.
       IF(NTYPE(JJ).EQ.4.OR.NTYPE(JJ).EQ.5)CDEN(JJ)=CDEN(JJ)*C(JJ)/2000.
       IF(NTYPE(JJ).NE.4.AND.NTYPE(JJ).NE.5)CDEN(JJ)=CDEN(JJ)*C(JJ)/900.
       CDISC(JJ)=4./27.*(CDEN(JJ)/D(JJ))*SO(JJ)
1640  CONTINUE
      ISCEN4=0.
1650  DO 1660 K=1,NLINK
       IF(EL(K).EQ.1)T(K)=9999.
1660  CONTINUE
C     EL: PANIC FACTOR
1670  DO 1680 J=1,NLINK
       IF(EZ(J).EQ.1)T(J)=9999.
1680  CONTINUE
C     EZ: SUITABILITY FACTOR

      POI=0.
      IF(ITERD.EQ.1) GO TO 1703

      KPATH=0
      DO 1685 K1=1,150
       LENGTH(K1)=0
       LD(K1,1)=0
       LD(K1,2)=0
       DO 1685 K2=1,105
1685  LV(K1,K2)=0
C************ INITIAL PATH INDEX
      DO 1700 ORIG=IDST,IDS
       IF(PROD(ORIG).LE.0.0)GO TO 1700

       CALL DIPATH(ORIG)

1700  CONTINUE
      ITERD=1
      DO 1702 K=1,KPATH
       K1=LD(K,1)
       K2=LD(K,2)
1702  NOPA(K1,K2)=NOPA(K1,K2)+1

1703  CALL HEUR1
      DO 1704 ORIG=IDST,IDS
       POI=POI+PROD(ORIG)
1704  COI=COI+PROD(ORIG)
```

```
1705  TM=0.0
C     TM RECORDS TOTAL TRAVEL TIME
      DO 1770 LINK=1,NLINK
       V(LINK,ITE)=RVL(LINK)
      DO 1710 ITER=IDST,IDS
1710  V(LINK,ITE)=V(LINK,ITE)+V(LINK,ITER)
C     LOOP 1710 SUMS THE TOTAL VOL ASSIGNED ON LINK AT CURRENT INTERVAL
      IF(ISCEN2.EQ.0)GO TO 1740
C     ISCEN2: OPT FOR SEEDING NETWORK
      IF(VSEED(LINK).EQ.0) GO TO 1740
      IF((VSEED(LINK)*TS/60.).LT.VSEED(LINK))GO TO 1720
      V(LINK,ITE)=V(LINK,ITE)+VSEED(LINK)
      VSEED(LINK)=0.
      GO TO 1740
1720  V(LINK,ITE)=V(LINK,ITE)+(VSEED(LINK)*TS/60.)
      VSEED(LINK)=VSEED(LINK)-(VSEED(LINK)*TS/60.)
C     VSEED IS UPDATED EVERY ITERATION.
1740  VOLU(LINK)=VOLU(LINK)+V(LINK,ITE)-RVL(LINK)
C     TO PREVENT RVL BEING DOUBLE COUNTED IN COMPUTING VOLU.
      RVL(LINK)=0.
      VC(LINK)=V(LINK,ITE)/(C(LINK)*TS/60.)
1750  I=LINK
      LTYPE=NTYPE(I)

      CALL TIMEE

1760  S(LINK)=D(LINK)*60.0/T(LINK)
1770  TM=TM+T(LINK)*V(LINK,ITE)
      WRITE(6,1775) TM
1775  FORMAT(4X,'TOTAL VEH TRAVEL TIME AT THIS INTERVAL = ',F9.1,' VEH-'
     +,'MIN')
      STM=STM+TM
C     STM IS THE SUMMATION OF TATAL VEH TRAVEL TIME UP TO THIS INTERVAL.
      IF(ID.GE.(0.75*2*THL))GO TO 1790
      GO TO 2100
C     LOOP 1840 INVOKES MORE PATHS OPT.
1790  DO 1840 I=1,NLINK
       DO 1840 IJ=1,IGN
        IF(B(I).EQ.IJ.AND.JFAC(IJ).EQ.1)GO TO 1800
C      JFAC IS MORE PATHS OPT.
        GO TO 1840
1800     JB=A(I)
        TMIN=9999.
        TMAX=0.
        DO 1820 J=1,NLINK
         IF(B(J).EQ.JB)GO TO 1810
          GO TO 1820
1810      IF(T(J).GT.TMAX)TMAX=T(J)
         IF(T(J).LT.TMIN)TMIN=T(J)
1820     CONTINUE
      DO 1830 K=1,NLINK
       IF(B(K).EQ.JB.AND.V(K,ITE).EQ.0)T(K)=T(K)*(TMAX-TMIN)/TMAX
1830  CONTINUE
C     LOOP 1830 REDUCES TRAVEL TIME ON LINK K TO MAKE IT MORE ATTRACTIVE
1840  CONTINUE
2100  DO 2235 M=1,KPATH
```

```
       ET=0.
       IF(IPR1.EQ.0)GO TO 2145
C       IPR1: OPTION FOR PATH PRINTING
       WRITE(6,2115)
2115    FORMAT('0','PATH TRACING BELOW')
       WRITE(6,2117) M,LD(M,1),LD(M,2)
2117    FORMAT('0',' PATH ',I3,' FROM ORIG ',I3,' TO DEST ',I3)
       DO 2130 KK=1,LENGTH(M)
        LN=LV(M,KK)
        WRITE(6,2120) KK,A(LN),B(LN),(STR(LJ,LN),LJ=1,3)
2120   FORMAT('0','THE ',I3,' LINK GOES FROM NODE ',I3,' TO NODE ',I3,
     *' NAME: ',3A4)
2130    CONTINUE
2145    DO 2150 KK=1,LENGTH(M)
2150    ET=ET+T(LV(M,KK))
       IF(ET.EQ.0.) GO TO 2235
        TE=ID+ET
C       TE AND PET ARE TRANSFERRED BY COMMON METHOD.
        IF(TE.GT.PET) GO TO 2200
         GO TO 2235
2200      PET=TE
2210      WRITE(6,2220) PET,ID
2220      FORMAT(4X,'EVACUATION TIME3',F10.3,'ID=',I4)
2235   CONTINUE

      IPR1=0
C     REASSIGN IPR1=0 THUS PATH TRACING CAN ONLY BE DONE ONCE.
      DO 2240 J=1,NLINK
2240   EV(J)=V(J,ITE)
C      V(J,ITE): TOTAL VOL ON  LINK J AT CURRENT TIME INTERVAL.
      WRITE(6,2265)
2265   FORMAT('0','       ')
      DO 2300 JD=1,NLINK
       IF(EV(JD).LE.((C(JD)/60.)*TS)) GO TO 2290
        DEN(JD)=EV(JD)
C        DEN: CONGESTED VOL OF A LINK. UNIT:VEH/TS
        CAP=C(JD)/60.*TS
      WRITE(6,2270)JD,ZO(JD),EV(JD),CAP,(STR(K,JD),K=1,3),A(JD),B(JD)
2270   FORMAT(' ','CONGESTED LINK',2X,I4,5X,'ZONE',2X,I3,5X,'VOLUME',2X,
     *F8.0,5X,'CAPACITY  ',F5.0,5X,3A4;3X,'A-NODE',I3,3X,'B-NODE',I3)
        I=JD
        LTYPE=NTYPE(I)

        CALL DISCH
C        CALL DISCH TO COMPUTE OUTPUT ONLY WHEN A LINK IS CONGESTED.
        IF(OUTPUT.LE.0)OUTPUT=CDISC(I)*TS/60.
C        CHECK THE LOGIC WHY OUTPUT IS POSSIBLE TO BE <= 0.
        IF(OUTPUT.GT.((C(I)/60.)*TS))OUTPUT=C(I)/60.*TS
        IF(NTYPE(I).EQ.5)OUTPUT=EV(I)
        RVL(I)=EV(I)-OUTPUT
C        FOR UNCONGESTED LINK RVL=0. IS SET IN LOOP 1770
        IF(RVL(I).EQ.0)DEN(I)=0.
C      ALL VOL ON A LINK WILL COMPLETELY BE DISCHARGED EXCEPT THE LINK IS
C      CONGESTED. RVL OF A LINK WILL BE 0 IF WHICH LINK IS NOT CONGESTED.
```

```
       GO TO 2300
2290    DEN(JD)=0.
2300  CONTINUE
      DO 2320 I=1,NLINK
       DO 2320 J=1,IDS
C      CLEAR THE LINK ASSIGNED VOL BEFORE LEAVING CURRENT TIME INTERVAL.
2320  V(I,J)=0.
      IF(ID/TS.GT.40)GO TO 2330
       GO TO 2350
2330  WRITE(6,2340)
2340  FORMAT(' ','INCREASE THE H VALUE OF THE S CURVE')

      STOP
2350  RETURN
      END

C     ---------------------------
C     |       SYSOUT            |
C     ---------------------------
C     PURPOSE: WRITES OUT THE CHARACTERISTICS OF THE LINKS WITH POSITIVE
C     VOLUME AFTER A TIME INTERVAL TS's TRIP ASSIGNMENT.
      SUBROUTINE SYSOUT

      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /CDIAL/ BD,BDL,MINT,BN,BINDX
      DIMENSION BD(336),BDL(336),MINT(105),BN(105),BINDX(105)
      INTEGER BINDX
      INTEGER MINT,BD,BDL,BN
      PP=POI/SOI*100.
      CP=COI/SOI*100.
C      PP AND CP ARE % OF PARTIAL AND CULMULATED TRIP ASSIGNED AT TS.
      WRITE(6,3100) ID,TS
3100  FORMAT(4X,'CURRENT CLOCK TIME = ',I4,' INTERVAL = ',F5.0)
      WRITE(6,3200) POI,PP
3200  FORMAT(4X,'TOTAL TRIP ASSIGNED AT THIS TS = ',F8.1,' IT IS ',F6.2,
     +'%')
      WRITE(6,3300) COI,CP
3300  FORMAT(4X,'CUMULATED TRIP ASSIGNED UP TO THIS TS = ',F8.1,
     +' IT IS ',F6.2,'%')
      IF(CHOICE.EQ.0) GO TO 3360
       DP=COI/SDI*100.
       WRITE(6,3350) DP
3350   FORMAT(4X,F6.2,'% OF SHELTER CAPACITY HAS BEEN FILLED')
```

```
3360  IF(IPR2.EQ.0) RETURN
C      IPR2: OPTION FOR LINK CHARACTERISTICS PRINTING.
      WRITE(6,3400)
3400  FORMAT(' ',' LINK  ANODE  BNODE  DIST  SPEEDO  SPEED  TIMEO   TIME
     *    STR    VOLUME  CAPACITY  V/C RATIO  RVOL   FL   EL
     *EZ ')
      DO 3420 I=1,NLINK
       IF(V(I,ITE).EQ.0)GO TO 3420
        WRITE(6,3410)  I,A(I),B(I),D(I),SO(I),S(I),TO(I),T(I),
     *(STR(K,I),K=1,3),V(I,ITE),C(I),VC(I),RVL(I),FL(I),EL(I),EZ(I)
3410  FORMAT(1X,I4,I5,I7,F9.2,1X,3F7.2,F8.2,1X,3A4,1X,F8.2,1X,F7.0,
     *1X,F9.2,1X,F8.2,1X,3F5.2)
3420  CONTINUE

      RETURN
      END

C      ----------------------------
C      |        TIMEE             |
C      ----------------------------
C      PURPOSE: UPDATE LINK I TRAVEL TIME BASED ON LINK TYPE AND TRAFFIC
C      SIMULATION OPTION (MACRO OR MICRO LEVEL).
C      COMMENT: THE REQUIREMENT FOR FLASHING CONTROL SEEMS INCOMPLETE, IT
C      ONLY CONSIDER THE LINK VOLUME, NOT CONSIDER THE VOL ON CROSS ST.
      SUBROUTINE TIMEE

      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /INDX/ I
      DIMENSION LLL(4)
      IF(V(I,ITE).GT.0)GO TO 2390
       T(I)=TO(I)
       RETURN

2390  IF(MACRO.EQ.0) GO TO 2530
C***   THIS IS THE MACROSCOPIC LOGIC
       IF(V(I,ITE).GT.C(I)*TS/60.)GO TO 2540
        IF(NTYPE(I).EQ.4.OR.NTYPE(I).EQ.5)V(I,ITE)=.5*V(I,ITE)
C      CURRENTLY, LANE OF TYPE 4 OR 5 IS TWO LANE EACH DIRECTION FREEWAY
      V(I,ITE)=V(I,ITE)*60./TS
      GO TO (2400,2430,2460,2490,2520),LTYPE
2400  IF(V(I,ITE)-750.)2410,2410,2420
2410  T(I)=(3.191+.000772*V(I,ITE))*D(I)
      GO TO 2960
2420  T(I)=(-30.58+.0458*V(I,ITE))*D(I)
```

```
      GO TO 2960
 2430 IF(V(I,ITE)-900.)2440,2440,2450
 2440 T(I)=(1.9036+.000561*V(I,ITE))*D(I)
      GO TO 2960
 2450 T(I)=(-23.9804+.029321*V(I,ITE))*D(I)
      GO TO 2960
 2460 IF(V(I,ITE)-1070.)2470,2470,2480
 2470 T(I)=(1.66355+.000605*V(I,ITE))*D(I)
      GO TO 2960
 2480 T(I)=(-22.8769+.02354*V(I,ITE))*D(I)
      GO TO 2960
 2490 IF(V(I,ITE)-1600.)2500,2500,2510
 2500 T(I)=(0.9644+.000333*V(I,ITE))*D(I)
      GO TO 2960
 2510 T(I)=(-18.1668+.012290*V(I,ITE))*D(I)
      GO TO 2960
 2520 T(I)=TO(I)
      GO TO 2960
C***  THIS IS THE MICROSCOPIC LOGIC, NTYPE=5 IS FOR DUMMY LINKS (BPR)
C***  NTYPE=2 IS FOR A TWO-WAY STOP SIGN, NTYPE=3 IS FOR FOUR-WAY STOP
C***  SIGN, NTYPE=1 IS FOR AN ISOLATED SIGNALIZED INTERSECTION,
C***  NTYPE=4 IS FOR EXPRESSWAYS.
 2530 GO TO (2560,2560,2560,2550,2540),LTYPE
 2540 CC=V(I,ITE)/(C(I)*TS/60.)
      T(I)=TIME(TO(I),CC)
      RETURN
 2550 IDEL=1200
      ARR=V(I,ITE)/(TS*60.*2.)
C      ARR IS THE VEHICLE ARRIVAL RATE: VEH/SECOND PER LANE
      SPD=SO(I)
      DST=D(I)*100.
      IF(ARR.LT.1)IDEL=IFIX(DST*ARR*36./(SPD*(1-ARR))+.5)
      IDEL=MIN0(IABS(IDEL),1200)
C      IDEL IS THE DELAY PERCEIVED BY EACH VEH ON LINK I, UNIT=SECCOND
      T(I)=TO(I)+IDEL/60.
      RETURN
 2560 IBIN=B(I)
      ICOUT=0
      DO 2570 KI=1,4
 2570 LLL(KI)=0
      DO 2590 KK=1,NLINK
      IF(B(KK).NE.IBIN)GO TO 2590
      DO 2580 KKK=1,4
      IF(LLL(KKK).NE.0)GO TO 2580
      LLL(KKK)=KK
      ICOUT=ICOUT+1
      GO TO 2590
 2580 CONTINUE
 2590 CONTINUE
      IF(ICOUT.EQ.1.OR.ICOUT.EQ.2)GO TO 2810
      IF(ICOUT.EQ.3)GO TO 2720
      IF(ZZ(LLL(1)).EQ.A(LLL(2))) GO TO 2600
      IF(ZZ(LLL(1)).EQ.A(LLL(3))) GO TO 2640
      IF(ZZ(LLL(1)).EQ.A(LLL(4))) GO TO 2680
```

```
      GO TO 2810
C     LINK LLL(1) AND LLL(2) ON THE SAME LINE
 2600 IF(LLL(1).EQ.I.OR.LLL(2).EQ.I) GO TO 2620
      V2=MAX1(V(LLL(1),ITE),V(LLL(2),ITE))
      V1=MAX1(V(LLL(3),ITE),V(LLL(4),ITE))
      IF(ISCEN3.EQ.0)GO TO 2610
C     ISCEN3=1 MEANS FLASHING OPT
      IF(V(LLL(3),ITE).GT.(0.3*C(LLL(3))*TS/60.)) GO TO 2610
      IF(V(LLL(4),ITE).GT.(0.3*C(LLL(4))*TS/60.)) GO TO 2610
C     CHANGE CROSS STREET TO FREEWAY OPERATION
      NTYPE(LLL(3))=2
      NTYPE(LLL(4))=2
      NTYPE(LLL(1))=4
      NTYPE(LLL(2))=4
      IF(C(LLL(1)).EQ.900.)C(LLL(1))=2000.
      IF(C(LLL(1)).EQ.1800.)C(LLL(1))=4000.
      IF(C(LLL(2)).EQ.900.)C(LLL(2))=2000.
      IF(C(LLL(2)).EQ.1800.)C(LLL(2))=4000.
 2610 VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
 2620 V1=MAX1(V(LLL(1),ITE),V(LLL(2),ITE))
      V2=MAX1(V(LLL(3),ITE),V(LLL(4),ITE))
      IF(ISCEN3.EQ.0)GO TO 2630
      IF(V(LLL(1),ITE).GT.(0.3*C(LLL(1))*TS/60.)) GO TO 2630
      IF(V(LLL(2),ITE).GT.(0.3*C(LLL(2))*TS/60.)) GO TO 2630
      NTYPE(LLL(1))=2
      NTYPE(LLL(2))=2
      NTYPE(LLL(3))=4
      NTYPE(LLL(4))=4
      IF(C(LLL(3)).EQ.900.)C(LLL(3))=2000.
      IF(C(LLL(3)).EQ.1800.)C(LLL(3))=4000.
      IF(C(LLL(4)).EQ.900.)C(LLL(4))=2000.
      IF(C(LLL(4)).EQ.1800.)C(LLL(4))=4000.
 2630 VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
 2640 IF(LLL(1).EQ.I.OR.LLL(3).EQ.I) GO TO 2660
      V2=MAX1(V(LLL(1),ITE),V(LLL(3),ITE))
      V1=MAX1(V(LLL(2),ITE),V(LLL(4),ITE))
      IF(ISCEN3.EQ.0)GO TO 2650
      IF(V(LLL(2),ITE).GT.(0.3*C(LLL(2))*TS/60.)) GO TO 2650
      IF(V(LLL(4),ITE).GT.(0.3*C(LLL(4))*TS/60.)) GO TO 2650
      NTYPE(LLL(2))=2
      NTYPE(LLL(4))=2
      NTYPE(LLL(1))=4
      NTYPE(LLL(3))=4
      IF(C(LLL(1)).EQ.900.)C(LLL(1))=2000.
      IF(C(LLL(1)).EQ.1800.)C(LLL(1))=4000.
      IF(C(LLL(3)).EQ.900.)C(LLL(3))=2000.
      IF(C(LLL(3)).EQ.1800.)C(LLL(3))=4000.
 2650 VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
 2660 V1=MAX1(V(LLL(1),ITE),V(LLL(3),ITE))
      V2=MAX1(V(LLL(2),ITE),V(LLL(4),ITE))
      IF(ISCEN3.EQ.0)GO TO 2670
      IF(V(LLL(1),ITE).GT.(0.3*C(LLL(1))*TS/60.)) GO TO 2670
      IF(V(LLL(3),ITE).GT.(0.3*C(LLL(3))*TS/60.)) GO TO 2670
```

```
      NTYPE(LLL(1))=2
      NTYPE(LLL(3))=2
      NTYPE(LLL(2))=4
      NTYPE(LLL(4))=4
      IF(C(LLL(2)).EQ.900.)C(LLL(2))=2000.
      IF(C(LLL(2)).EQ.1800.)C(LLL(2))=4000.
      IF(C(LLL(4)).EQ.900.)C(LLL(4))=2000.
      IF(C(LLL(4)).EQ.1800.)C(LLL(4))=4000.
 2670 VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
 2680 IF(LLL(1).EQ.I.OR.LLL(4).EQ.I) GO TO 2700
      V2=MAX1(V(LLL(1),ITE),V(LLL(4),ITE))
      V1=MAX1(V(LLL(3),ITE),V(LLL(2),ITE))
      IF(ISCEN3.EQ.0)GO TO 2690
      IF(V(LLL(2),ITE).GT.(0.3*C(LLL(2))*TS/60.)) GO TO 2690
      IF(V(LLL(3),ITE).GT.(0.3*C(LLL(3))*TS/60.)) GO TO 2690
      NTYPE(LLL(2))=2
      NTYPE(LLL(3))=2
      NTYPE(LLL(1))=4
      NTYPE(LLL(4))=4
      IF(C(LLL(1)).EQ.900.)C(LLL(1))=2000.
      IF(C(LLL(1)).EQ.1800.)C(LLL(1))=4000.
      IF(C(LLL(4)).EQ.900.)C(LLL(4))=2000.
      IF(C(LLL(4)).EQ.1800.)C(LLL(4))=4000.
 2690 VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
 2700 V1=MAX1(V(LLL(1),ITE),V(LLL(4),ITE))
      V2=MAX1(V(LLL(3),ITE),V(LLL(2),ITE))
      IF(ISCEN3.EQ.0)GO TO 2710
      IF(V(LLL(1),ITE).GT.(0.3*C(LLL(1))*TS/60.)) GO TO 2710
      IF(V(LLL(4),ITE).GT.(0.3*C(LLL(4))*TS/60.)) GO TO 2710
      NTYPE(LLL(1))=2
      NTYPE(LLL(4))=2
      NTYPE(LLL(2))=4
      NTYPE(LLL(3))=4
      IF(C(LLL(2)).EQ.900.)C(LLL(2))=2000.
      IF(C(LLL(2)).EQ.1800.)C(LLL(2))=4000.
      IF(C(LLL(3)).EQ.900.)C(LLL(3))=2000.
      IF(C(LLL(3)).EQ.1800.)C(LLL(3))=4000.
 2710 VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
C***  LOGIC OF TEE INTERSECTION
 2720 IF(ZZ(LLL(1)).EQ.0)GO TO 2800
      IF(ZZ(LLL(1)).EQ.A(LLL(2)))GO TO 2740
      IF(ZZ(LLL(1)).EQ.A(LLL(3)))GO TO 2770
      WRITE(6,2730)
 2730 FORMAT(2X,26HSOMETHING WRONG WITH 2OGIC)
      STOP
 2740 IF(LLL(1).EQ.I.OR.LLL(2).EQ.I)GO TO 2760
      V2=MAX1(V(LLL(1),ITE),V(LLL(2),ITE))
      V1=V(LLL(3),ITE)
      IF(ISCEN3.EQ.0)GO TO 2750
      IF(V(LLL(3),ITE).GT.(0.3*C(LLL(3))*TS/60.)) GO TO 2750
      NTYPE(LLL(3))=2
      NTYPE(LLL(1))=4
      NTYPE(LLL(2))=4
```

```
      IF(C(LLL(1)).EQ.900.)C(LLL(1))=2000.
      IF(C(LLL(1)).EQ.1800.)C(LLL(1))=4000.
      IF(C(LLL(2)).EQ.900.)C(LLL(2))=2000.
      IF(C(LLL(2)).EQ.1800.)C(LLL(2))=4000.
 2750 VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
 2760 V1=MAX1(V(LLL(1),ITE),V(LLL(2),ITE))
      V2=V(LLL(3),ITE)
      VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
 2770 IF(LLL(1).EQ.I.OR.LLL(3).EQ.I)GO TO 2790
      V2=MAX1(V(LLL(1),ITE),V(LLL(3),ITE))
      V1=V(LLL(2),ITE)
      IF(ISCEN3.EQ.0)GO TO 2780
      IF(V(LLL(2),ITE).GT.(0.3*C(LLL(2))*TS/60.)) GO TO 2780
      NTYPE(LLL(2))=2
      NTYPE(LLL(1))=4
      NTYPE(LLL(3))=4
      IF(C(LLL(1)).EQ.900.)C(LLL(1))=2000.
      IF(C(LLL(1)).EQ.1800.)C(LLL(1))=4000.
      IF(C(LLL(3)).EQ.900.)C(LLL(3))=2000.
      IF(C(LLL(3)).EQ.1800.)C(LLL(3))=4000.
 2780 VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
 2790 V1=MAX1(V(LLL(1),ITE),V(LLL(3),ITE))
      V2=V(LLL(2),ITE)
      VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)+V(LLL(4),ITE)
      GO TO 2820
C     LINK LLL(2) AND LLL(3) ON THE SAME LINE
 2800 IF(LLL(1).EQ.I) GO TO 2805
      V1=MAX1(V(LLL(2),ITE),V(LLL(3),ITE))
      V2=V(LLL(1),ITE)
      VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)
      GO TO 2820
 2805 V1=V(LLL(1),ITE)
      V2=MAX1(V(LLL(2),ITE),V(LLL(3),ITE))
      VTOTL=V(LLL(1),ITE)+V(LLL(2),ITE)+V(LLL(3),ITE)
      GO TO 2820
 2810 V1=V(I,ITE)
      V2=0.
      VTOTL=V1
 2820 IF(V1.EQ.0.AND.V2.EQ.0)V1=V(I,ITE)
      IF(V1.EQ.0)V1=V(I,ITE)
      IF(VTOTL.EQ.0)VTOTL=V(I,ITE)
      RATIO=V1/(V1+V2)
      V1=V1/(TS*60.)
      V2=V2/(TS*60.)
      VTOTL=VTOTL/(TS*60.)
      GO TO (2920,2830,2870,2960,2960),LTYPE
C     THEORY OF 2830 AND 2870 IS FROM UROAD OF UTPS
 2830 BB=(100.7*V2+6.88)*V2+0.194
      GAMMA=1./BB
      SQG=SQRT(GAMMA)
      SV=GAMMA-SQG*0.057735
      IF(V1.GT.SV) GO TO 2840
      DELA=1./(GAMMA-V1)
```

```
      GO TO 2860
 2840 IF(V1.GT.GAMMA*4.8833)GO TO 2850
      DELA=(BB*V1-1.)*300.+35.
      GO TO 2860
 2850 DELA=1200.
 2860 IDEL=MIN0(IABS(IFIX(DELA+.5)),1200)
      IF(IDEL.GT.1200)IDEL=1200
      T(I)=TO(I)+IDEL/60.
      RETURN
 2870 VS=VTOTL
      VZ=V(I,ITE)
      VZ=VZ/(TS*60.)
      VSS=2.*(V1**2+V2**2)
      SK=0.50
      IDEL=0
      IF(VSS.EQ.0) GO TO 2910
      SKR=1./SK
      SKS=SQRT(SK)
      SV=SK-SKS*0.057735
      IF(VS.GT.SV)GO TO 2880
      DELA=1./(SK-VS)-SKR
      GO TO 2900
 2880 TH=34.64/SKS-SKR
      IF(VS.GT.(5.-TH*.00333)*SK)GO TO 2890
      DELA=(VS-SK)*300.*SKR+TH
      GO TO 2900
 2890 DELA=1200.
 2900 IDEL=IFIX(DELA*VS*VZ/VSS+.5)
      IDEL=MIN0(IABS(IDEL),1200)
C     IF(IDEL.GT.1200)IDEL=1200
 2910 T(I)=TO(I)+IDEL/60.
      RETURN
 2920 VCRIT=(V1+V2)*TS*60.
      YY=VCRIT/1800.
      IF(YY.GE.1.) CYCT=120.
      IF(YY.GE.1.) GO TO 2925
      CYCT=20./(1.-YY)
      IF(CYCT.LT.40.)CYCT=40.
      IF(CYCT.GT.120.)CYCT=120.
 2925 V1=V1*TS*60.
      G=CYCT*V1/VCRIT
      VOLL=V(I,ITE)
      VOLL=VOLL/(TS*60.)
      IF(VOLL.LE.0)VOLL=0.01
      SS=G/CYCT
      X=VOLL/(0.5*SS)
C     THE SATURATION RATE IS 0.5 VEH/SEC (1800 VEH/HR)
      IF(X.LE.1)GO TO 2930
      GO TO 2940
 2930 DEL=0.9*(CYCT*((1-SS)**2)/(2*(1-SS*X))+(X**2)/(2*VOLL*(1-X)))
      GO TO 2950
 2940 DEL=TS**2/2*(VOLL-2*SS)
 2950 IF(DEL.LT.0)STOP
      IDEL=IFIX(DEL+.5)
      IF(IDEL.GT.1200)IDEL=1200
```

```
      T(I)=TO(I)+IDEL/60.
 2960 IF(MACRO.EQ.0)RETURN
      IF(NTYPE(I).EQ.4.OR.NTYPE(I).EQ.5)V(I,ITE)=2*V(I,ITE)
      V(I,ITE)=V(I,ITE)*TS/60.
      RETURN
      END
      FUNCTION TIME(TIMEO,VCR)
      TIME=TIMEO
      IF(VCR.LE.0.) RETURN
      TIME=TIMEO*(1.0+0.15*VCR**4)

      RETURN
      END

C     --------------------------
C     |  DISCHARGE              |
C     --------------------------
C      PURPOSE: DISCHARGE LINK I TRAFFIC THROUGH ITS DOWNSTREAM
C      INTERSECTION AT B NODE
      SUBROUTINE DISCH

      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /INDX/ I
      DIMENSION LLL(4)
C      NTYPE: 1=ISOLATED SIGNALIZED, 2=2 WAY STOP, 3=4-WAY STOP,
C      4=EXPRESSWAY, 5=DUMMY LINK.
      IF(MACRO.EQ.0)GO TO (3000,3280,3290,3300,3330),LTYPE
      GO TO (3000,3000,3000,3300,3330),LTYPE
C***   DISSIPATION OF VEHICULAR QUEUE AT SIGNALIZED INTERSECTIONS
 3000 IBIN=B(I)
      ICOUT=0
      DO 3010 KI=1,4
 3010 LLL(KI)=0
      DO 3030 KK=1,NLINK
       IF(B(KK).NE.IBIN.OR.V(KK,ITE).EQ.0) GO TO 3030
        ICOUT=ICOUT+1
        LLL(ICOUT)=KK
 3030 CONTINUE
C     LOOP 3030 TELLS HOW MANY AND WHICH LINKS USE IBIN AS B NODE
      IF(ICOUT.EQ.1.OR.ICOUT.EQ.2)GO TO 3320
      IF(ICOUT.EQ.3)GO TO 3100
C     THE FOLLOWING DEALS WITH 4 LEGS INTERSECTION
C     ZZ(I) TELLS THE DOWNSTREAM NODE OF LINK I: A---I---B-------ZZ(I)
```

```
      IF(ZZ(LLL(1)).EQ.0)GO TO 3100
C      ABOVE ST IS NOT APPROPRIATE.
C      TO FIND WHICH TWO LINKS ARE ON THE SAME LINE
      IF(ZZ(LLL(1)).EQ.A(LLL(2))) GO TO 3040
      IF(ZZ(LLL(1)).EQ.A(LLL(3))) GO TO 3060
      IF(ZZ(LLL(1)).EQ.A(LLL(4))) GO TO 3080
      GO TO 3320
C      3040: LINKS 1 AND 2 ARE ON THE SAME LINE
3040  IF(LLL(1).EQ.I.OR.LLL(2).EQ.I) GO TO 3050
      V2=MAX1(V(LLL(1),ITE),V(LLL(2),ITE))
      V1=MAX1(V(LLL(3),ITE),V(LLL(4),ITE))
      GO TO 3270
3050  V1=MAX1(V(LLL(1),ITE),V(LLL(2),ITE))
      V2=MAX1(V(LLL(3),ITE),V(LLL(4),ITE))
      GO TO 3270
C      3060: LINKS 1 AND 3 ARE ON THE SAME LINE
3060  IF(LLL(1).EQ.I.OR.LLL(3).EQ.I) GO TO 3070
      V2=MAX1(V(LLL(1),ITE),V(LLL(3),ITE))
      V1=MAX1(V(LLL(2),ITE),V(LLL(4),ITE))
      GO TO 3270
3070  V1=MAX1(V(LLL(1),ITE),V(LLL(3),ITE))
      V2=MAX1(V(LLL(2),ITE),V(LLL(4),ITE))
      GO TO 3270
C      3080: LINKS 1 AND 4 ARE ON THE SAME LINE
3080  IF(LLL(1).EQ.I.OR.LLL(4).EQ.I) GO TO 3090
      V2=MAX1(V(LLL(1),ITE),V(LLL(4),ITE))
      V1=MAX1(V(LLL(3),ITE),V(LLL(2),ITE))
      GO TO 3270
3090  V1=MAX1(V(LLL(1),ITE),V(LLL(4),ITE))
      V2=MAX1(V(LLL(3),ITE),V(LLL(2),ITE))
      GO TO 3270
C***  LOGIC OF TEE INTERSECTION
C      FIND THE LEG OF T INTERSECTION FIRST
3100  IF(ZZ(LLL(1)).EQ.0) GO TO 3260
      IF(ZZ(LLL(1)).EQ.A(LLL(2)))GO TO 3210
      IF(ZZ(LLL(1)).EQ.A(LLL(3)))GO TO 3230
      WRITE(6,3200)
3200  FORMAT(2X,'SOMETHING WRONG WITH THE LOGIC OF T INTERSECTION')
      STOP
C      3210: LINKS 1 AND 2 ARE ON THE SAME LINE
3210  IF(LLL(1).EQ.I.OR.LLL(2).EQ.I)GO TO 3220
      V2=MAX1(V(LLL(1),ITE),V(LLL(2),ITE))
      V1=V(LLL(3),ITE)
      GO TO 3270
3220  V1=MAX1(V(LLL(1),ITE),V(LLL(2),ITE))
      V2=V(LLL(3),ITE)
      GO TO 3270
C      3230: LINKS 1 AND 3 ARE ON THE SAME LINE
3230  IF(LLL(1).EQ.I.OR.LLL(3).EQ.I)GO TO 3240
      V2=MAX1(V(LLL(1),ITE),V(LLL(3),ITE))
      V1=V(LLL(2),ITE)
      GO TO 3270
3240  V1=MAX1(V(LLL(1),ITE),V(LLL(3),ITE))
      V2=V(LLL(2),ITE)
      GO TO 3270
3260  IF(LLL(1).EQ.I) GO TO 3265
```

```
      V1=MAX1(V(LLL(2),ITE),V(LLL(3),ITE))
      V2=V(LLL(1),ITE)
      GO TO 3270
3265  V1=V(LLL(1),ITE)
      V2=MAX1(V(LLL(2),ITE),V(LLL(3),ITE))
C     V2 IS THE CRITICAL VOL ON CROSS ST
3270  IF(V1.EQ.0.AND.V2.EQ.0) V1=1.
      RATIO=V1/(V1+V2)
      OUTPUT=(1800./60.)*TS*RATIO*C(I)/900.
C     THE CAPACITY 900 FOR TWO LANE TWO WAY HIGHWAY IS TOO CONSERVATIVE
      IF(OUTPUT.GT.(CDISC(I)*TS/60.))OUTPUT=CDISC(I)*TS/60.
      IF(OUTPUT.GT.((C(I)/60.)*TS))OUTPUT=C(I)*TS/60.
      RETURN
3280  IBIN=B(I)
      V2=0.
      DO 3285 KK=1,NLINK
      IF(KK.EQ.I) GO TO 3285
      IF(B(KK).NE.IBIN.OR.A(KK).NE.ZZ(I)) GO TO 3285
      V2=V2+V(KK,ITE)
3285  CONTINUE
      V2=V2/(TS*60.)
C     V2 IS THE ARRIVAL RATE PER SECOND.
      BB=(100.7*V2+6.88)*V2+0.194
      OUTPUT=(1100./BB)*TS/60.
      IF(OUTPUT.GT.(CDISC(I)*TS/60.))OUTPUT=CDISC(I)*TS/60.
      IF(OUTPUT.GT.((C(I)/60.)*TS))OUTPUT=C(I)*TS/60.
      RETURN
3290  OUTPUT=900.*TS/60.
      IF(OUTPUT.GT.(CDISC(I)*TS/60.))OUTPUT=CDISC(I)*TS/60.
      IF(OUTPUT.GT.((C(I)/60.)*TS))OUTPUT=C(I)/60.*TS
      RETURN
C***  LOGIC OF DISSIPATING QUEUES FROM FREEWAYS AND EXPRESSWAYS
3300  DENS=(V(I,ITE)*60./TS)/(S(I)*C(I)/2000.)
      IF(DENS.GT.99)GO TO 3310
      OUTPUT=(74.3*DENS-0.75*DENS*DENS)*C(I)/2000.*TS/60.
      IF(OUTPUT.GT.(CDISC(I)*TS/60.))OUTPUT=CDISC(I)*TS/60.
      IF(OUTPUT.GT.((C(I)/60.)*TS))OUTPUT=C(I)/60.*TS
      RETURN
3310  OUTPUT=CDISC(I)*TS/60.
      IF(OUTPUT.GT.((C(I)/60.)*TS))OUTPUT=C(I)/60.*TS
      RETURN
3320  OUTPUT=V(I,ITE)
      IF(OUTPUT.GT.(CDISC(I)*TS/60.))OUTPUT=CDISC(I)*TS/60.
      IF(OUTPUT.GT.((C(I)/60.)*TS))OUTPUT=C(I)/60.*TS
      RETURN
3330  OUTPUT=V(I,ITE)

      RETURN
      END

C     ---------------------------
C     |      DIAL & PATH        |
C     ---------------------------
C     PURPOSE: ASSIGN EACH LINK AN ASSIGNMENT-LIKELIHOOD AND
C     BUILD THE MULTIPLE-PATH FOR EACH O-D PAIR.
      SUBROUTINE DIPATH(ORIG)
```

```
      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /CTREE/ FN,BNT,TSEQ
      DIMENSION FN(105),BNT(105),TSEQ(105)
      INTEGER FN,FROM,TOO,ORIG
      COMMON /CDIAL/ BD,BDL,MINT,BN,BINDX
      DIMENSION BD(336),BDL(336),MINT(105),BN(105),BINDX(105)
      INTEGER BINDX
      INTEGER MINT,BD,BDL,BN
      COMMON /CPATHA/ PN(41)
      INTEGER PN
      COMMON /CPATHB/ LV,LD,NOPA,LENGTH,KPATH
      DIMENSION LV(150,105),LD(150,2),NOPA(41,15),LENGTH(150)
      COMMON /CDIPH/ NODEP,NODIX,NODEF,LV1,LD1,LENG,MLINK,LPATH,MPATH,
     *AE,WE,VN
      DIMENSION NODEP(105,2),NODIX(105),NODEF(105),LV1(300,105),LD1(300)
     *,LENG(300),MLINK(300),AE(336),WE(336),VN(105)
      REAL MI,MJ
C-----------------------------------------------------
C      PN: NO OF DEST WHICH ORIG TREE NEEDS TO REACH.
C      LV: PATH'S LINK SEQUENCE. LD KEEPS TRACK OF PATH'S O-D.
C      LENGTH: PATH LENGTH. KPATH: TOTAL PATHS BUILT IN THIS ITERATION.
C      NODEP(I,1): NO OF DIF PATH GOES FROM NODE I TO ORIG, THEY BEGIN
C      WITH PATH NODEP(I,2). THESE PATHS MAY NOT BE CONSECUTIVE. NODIX:
C      DUMMY VAR TELLS IF NODE I HAS EVER BEEN USED AS PRECEEDING LINK
C      NODE. IF IT IS USED, NODEP(I,1) NEW PATHS WILL BE BUILT. NODEF
C       =1 MEANS DEST I SHOULD HAVE PATH BUILT TO FROM ORIG.
C      LD1 KEEPS PATH'S TEMPORARY ENDING NODE.
C      MPATH=1 MEANS THAT THIS ORIG HAS COME TO ITS MAX PATH CONSTRAINT
C      MLINK(P1)=1 MEANS THAT PATH P1 HAS COME TO ITS LENGTH CONSTRAINT
C      MAX PATH LENGTH IN THIS MODEL IS 105 LINKS
C-----------------------------------------------------
      MPATH=0
      DO 1350 I=1,NLINK
       AE(I)=0.0
C       AE(I): ASSIGNMENT-LIKELIHOOD
1350   WE(I)=0.0
C      WE(I) IS THE ASSIGNMENT WEIGHT OF LINK I
      DO 1360 I=1,105
       NODEF(I)=0
1360   VN(I)=0.0
      DO 1365 I=1,IGN
       IF(TRIPS(ORIG,I).GE.(.5)) NODEF(I)=1
1365   CONTINUE
```

```
      THETA=PARM1

      CALL TREE(ORIG)

      DO 1370 I=1,NODES
       MINT(I)=IFIX(BNT(I)*120.+.5)
1370  BN(I)=I
C     AMPLIFY BNT TO SORT NODE SEQUENCE
C     BNT(I) IS THE MINIMUM TRAVEL TIME TO NODE I FROM THE ORIGIN NODE
C     USE MINT AND BN AS SORTING VARIABLE THUS THE NETWORK CODING WILL
C     NOT BE AFFECTED.
      CALL SORTD(MINT,BN,NODES)
C     THIS ST SORTS NODES IN ASCENDING SEQUENCE BY TRAVEL TIME TO  THE
C     ORIGIN NODE. NODE BN WITH SMALL MINT WILL LEAD THE SEQUENCE
C     BN(I) RECORDS THE ITH NEAREST NODE TO THE ORIGIN.
C     LOOP 1420 COMPUTES LINK'S ASSIGNMENT PROBABILITY LIKELIHOOD WITH
C     RESPECT TO THE LINK ON THE SHORTEST PATH HAS LIKELIHOOD=1.
1407  FORMAT(3X,' NODE= ',I3,' TIME= ',F8.3)
1408  DO 1420 LINK=1,NLINK
       IF(DEN(LINK).GT.(2.*C(LINK)*TS/60.))GO TO 1420
        I=A(LINK)
        J=B(LINK)
        MI=BNT(I)
        MJ=BNT(J)
        IF(MI.GE.MJ)GO TO 1420
         DELTAT=MJ-MI-T(LINK)
C     T(LINK) CAN BE REPLACED BY OTHER IMPEDANCE VARIABLE IF CONSIDERED.
C     DELTAT SHOULD LESS THAN OR EQUAL TO 0.
C     IF(DELTAT.GT.(-3.).AND.DELTAT.LE.1)GO TO 1410
C     GO TO 1420
1410  AE(LINK)=EXP(THETA*DELTAT)
      IF(AE(LINK).LT.(0.100)) AE(LINK)=0.
1420  CONTINUE

      CALL FPASS(ORIG)

C****************************************
C     REORDER "NPATH" USEFUL PATHS BEFORE RETURN
      NPATH=0
      DO 1590 I=1,IGN
       IF(NODEF(I).EQ.0) GO TO 1590
        NPATH=NPATH+NODEP(I,1)
1590  CONTINUE
      WRITE(6,1591) ORIG,NPATH
1591  FORMAT(3X,'ORIG = ',I3,' USEFUL PATH =',I3)
C     ONLY "NPATH" PATHS ARE USEFUL TO KEEP
      IF(KPATH+NPATH.LE.150) GO TO 15902
       WRITE(6,15901) ORIG
15901 FORMAT(3X,' KPATH WILL EXTEND PATH NO CONSTRAINT AT ORIG= ',I3)
      STOP

15902 CON=1.
      M=0
      DO 1610 I=1,LPATH
       KEND=LD1(I)
```

```
      IF(NODEF(KEND).EQ.0) GO TO 1610
C       WRITE(6,1592) ORIG,KEND
1592  FORMAT(3X,'ORIG = ',I3,' KEND = ',I3)
       M=M+1
      IF(LENG(I).GT.105) WRITE(6,*) '*PATH LENGTH IS GREATER THAN 105*'
      DO 1600 J=1,LENG(I)
1600  LV(KPATH+M,J)=LV1(I,J)
      LD(KPATH+M,1)=ORIG
      LD(KPATH+M,2)=KEND
      LENGTH(KPATH+M)=LENG(I)
1610  CONTINUE
      KPATH=KPATH+NPATH

      RETURN
      END

C     -----------------------
C     |   FORWARD  PASS     |
C     -----------------------
C      PURPOSE: CALCULATE LINK'S WEIGHT AND BUILDS PATHS FROM ORIGIN TO
C      DEST WITH + TRIP BY FORWARD PASS METHOD
      SUBROUTINE FPASS(ORIG)

      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /CTREE/ FN,BNT,TSEQ
      DIMENSION FN(105),BNT(105),TSEQ(105)
      INTEGER FN,FROM,TOO,ORIG
      COMMON /CDIAL/ BD,BDL,MINT,BN,BINDX
      DIMENSION BD(336),BDL(336),MINT(105),BN(105),BINDX(105)
      INTEGER BINDX
      INTEGER MINT,BD,BDL,BN
      COMMON /CPATHA/ PN(41)
      INTEGER PN
      COMMON /CDIPH/ NODEP,NODIX,NODEF,LV1,LD1,LENG,MLINK,LPATH,MPATH,
     *AE,WE,VN
      DIMENSION NODEP(105,2),NODIX(105),NODEF(105),LV1(300,105),LD1(300)
     *,LENG(300),MLINK(300),AE(336),WE(336),VN(105)
      DO 1420 I=1,105
       NODEP(I,1)=0
       NODEP(I,2)=0
1420  NODIX(I)=0
      DO 1425 K1=1,300
       LD1(K1)=0
       LENG(K1)=0
```

```
      MLINK(K1)=0
      DO 1425 K2=1,105
1425   LV1(K1,K2)=0
     SW=0.0
C     SW IS THE WEIGHT SUMMATION FOR LINKS WHICH PRECEED NODE IB.
C***************************************************************
C     LOOP 1480 CALCULATES LINK'S WEIGHT AND BUILDS PATHS FROM ORIG TO
C     EACH DEST BY FORWARD PASS METHOD.
     LPATH=1
C     LPATH: NO OF PATHS HAVE BEEN BUILT FROM NODE ORIG.
     KPN=0
C     KPN: NO OF DEST WITH  +VOL HAS BEEN REACHED. STOP PATH BUILDING
C     WHEN KPN=PN(ORIG)
     DO 1480 I=2,NODES
      IB=BN(I)
      IF(BNT(IB).EQ.9999.)GO TO 1475
C       IB IS THE NODE CURRENTLY DEALT WITH.
       LOC1=BINDX(IB)
1430  LINK=BDL(LOC1)
C     LOC1 & LOC2 TELL THE POS. ON THE ASCENDING LIST.
       JB=A(LINK)
       IF(JB.EQ.ORIG)GO TO 1470
        IF(AE(LINK).EQ.0.0)GO TO 1460
         LOC2=BINDX(JB)
1440  LFEED=BDL(LOC2)
C     LFEED IS A LINK USES JB AS B NODE.
     IF(B(LFEED).NE.JB)GO TO 1450
      SW=SW+WE(LFEED)
      LOC2=LOC2+1
      IF(LOC2.GT.NLINK)GO TO 1450
       GO TO 1440
1450  WE(LINK)=AE(LINK)*SW
     SW=0.0
     IF(NODIX(JB).EQ.1) GO TO 1457

C******** UPDATE EXISTING PATHS ********
     M1=NODEP(JB,1)
     M2=NODEP(JB,2)
     N2=0
     DO 1455 LL=1,M1
1451   IF(LD1(M2+N2).EQ.JB) GO TO 1453
C     LOOP 1451 LOOKS FOR PATHS END AT NODE JB.
       N2=N2+1
       GO TO 1451
1453  IF(MLINK(M2+N2).EQ.1) GO TO 14532
      NODIX(JB)=1
      NODEP(IB,1)=NODEP(IB,1)+1
     IF((NODEP(IB,2).EQ.0).OR.(NODEP(IB,2).GT.M2+N2)) NODEP(IB,2)=M2+N2
      LENG(M2+N2)=LENG(M2+N2)+1
      IF(LENG(M2+N2).EQ.105) MLINK(M2+N2)=1
      LV1(M2+N2,LENG(M2+N2))=LINK
      LD1(M2+N2)=IB
14532 N2=N2+1
1455  CONTINUE
C     LOOP 1455 EXTENDS LENGTH OF PATHS WHICH END AT NODE JB.
     GO TO 1460
```

```
C******** ADD NEW PATHS ********
C    AS JB IS NOT ENDING NODE FOR ANY PATH, THIS PART IS MORE COMPLICAT
1457  IF(MPATH.NE.1) GO TO 14572
C    WRITE(6,14570) ORIG
14570 FORMAT(3X,'LOCAL PATH > 300 AT ORIG1= ',I3)
      GO TO 1460
14572 M1=NODEP(JB,1)
      M2=NODEP(JB,2)
      N2=0
C    LOOP 1459 CHECKS THESE M PATHS AND ADDS NEW PATH IF NEEDED.
      DO 1459 LL=1,M1
14573  DO 14574 K3=1,LENG(M2+N2)
14574   IF(B(LV1(M2+N2,K3)).EQ.JB) GO TO 14575
C      LOOP 14574 CHECK IF PATH PASSES THROUGH NODE JB
       N2=N2+1
       GO TO 14573
14575  IF(K3.EQ.105) GO TO 14585
        IF(MPATH.NE.1) GO TO 14577
         GO TO 1460
14577   LPATH=LPATH+1
        IF(LPATH.EQ.300) MPATH=1
        DO 1458 K4=1,K3
1458      LV1(LPATH,K4)=LV1(M2+N2,K4)
        NODEP(IB,1)=NODEP(IB,1)+1
        LENG(LPATH)=K3+1
        IF((K3+1).EQ.105) MLINK(LPATH)=1
        LV1(LPATH,K3+1)=LINK
        LD1(LPATH)=IB
14585  N2=N2+1
1459  CONTINUE
C    IF NODIX=1, ADD NEW PATHS

C******* GO TO NEXT LINK *******
1460  LOC1=LOC1+1
      IF(LOC1.GT.NLINK)GO TO 1475
       IF(BD(LOC1).NE.IB)GO TO 1475
        GO TO 1430
1470  WE(LINK)=1.
C    LINK HERE IS THE DUMMY LINK CONNECTED TO NODE ORIG.
      NODIX(ORIG)=1
      LV1(1,1)=LINK
      LD1(1)=IB
      LENG(1)=1
      NODEP(IB,1)=1
      NODEP(IB,2)=1
      GO TO 1460
1475  IF(NODEF(IB).EQ.0) GO TO 1480
      KPN=KPN+1
      IF(KPN.NE.PN(ORIG)) GO TO 1480
       GO TO 1485
1480  CONTINUE

1485  RETURN
      END
```

```
C     -----------------------------------
C     |  HEURISTIC ASSIGNMENT 3   |
C     -----------------------------------
C     PURPOSE: HEURISTIC ASSIGNMENT FROM PROPOSAL 1
      SUBROUTINE HEUR1

      COMMON /ALL/ A,B,AINDX,D,SO,S,TO,T,C,TRIPS,V,VC,NODES,PROD,
     *TRIP,TE,PET,ID,RVL,FL,EL,CHOICE,EZ,ITERD,
     *TS,THL,ZO,STR,ELEV,ISCEN1,ISCEN2,ISCEN3,VOLU,VSEED,IND,
     *CDEN,DEN,CDISC,ISCEN4,IPR1,IPR2,JFAC,IDS,IGN,ITE,IDST,SOI,SDI,COI,
     *POI,STM,LEND
      DIMENSION A(337),B(336),AINDX(105),D(336),SO(336),ZO(336),
     *T(336),C(336),V(336,41),TRIPS(41,15),VC(336),PROD(41),
     *RVL(336),FL(336),EL(336),EZ(336),VOLU(336),VSEED(336),LEND(41,15),
     *S(336),TO(336),TRIP(41,15),IND(41,15),CDISC(336),JFAC(15),
     *ELEV(336),NTYPE(336),ZZ(336),STR(3,336),CDEN(336),DEN(336)
      INTEGER A,B,AINDX,ZO,ELEV
      COMMON /CTRL/ SR,CR,RL,NLINK,PARM1,OPT,MACRO,OUTPUT,LTYPE,NTYPE,ZZ
      INTEGER SR,CR,RL,OPT,ZZ
      COMMON /CPATHA/ PN(41)
      INTEGER PN,ORIG
      COMMON /CPATHB/ LV,LD,NOPA,LENGTH,KPATH
      DIMENSION LV(150,105),LD(150,2),NOPA(41,15),LENGTH(150)
      DIMENSION ALI(336),PWT(150)
      DO 3900 I=1,150
3900  PWT(I)=0.
      DO 3910 I=1,NLINK
       ALI(I)=(C(I)*TS/60.-RVL(I))/T(I)
       IF(ALI(I).GT.0.) GO TO 3910
        WRITE(6,3920) A(I),B(I),ALI(I)
3920    FORMAT(3X,'CONGESTED LINK WITH A= ',I3,' B= ',I3,' HAS
     * ALI(I)=',F8.2)
        ALI(I)=0.
3910  CONTINUE

      DO 3950 K=1,KPATH
       SALI=0.
       SD=0.
       DO 3940 K1=1,LENGTH(K)
        IF(ALI(K1).LE.0.) GO TO 3950
         SALI=SALI+ALI(K1)*D(K1)
3940    SD=SD+D(K1)
       PWT(K)=SALI/SD
3950  CONTINUE

      DO 4300 ORIG=IDST,IDS
       DO 4200 J=1,IGN
        K=NOPA(ORIG,J)
        IF(K.EQ.0) GO TO 4200
         SPWT=0.
         DO 4100 K1=1,KPATH
4100      IF(LD(K1,1).EQ.ORIG.AND.LD(K1,2).EQ.J)SPWT=SPWT+PWT(K1)
         IF(SPWT.NE.0.) GO TO 4150
          WRITE(6,4120) ORIG,J
4120        FORMAT(3X,'NO PATH IS AVAILABLE FROM ',I3,' TO ',I3)
          IND(ORIG,J)=IND(ORIG,J)-1
```

```
        LEND(ORIG,J)=0
        PROD(ORIG)=PROD(ORIG)-TRIPS(ORIG,J)
        GO TO 4200
4150     DO 4160 K1=1,KPATH
        IF(LD(K1,1).NE.ORIG.OR.LD(K1,2).NE.J) GO TO 4160
         DO 4155 L=1,LENGTH(K1)
          N=LV(K1,L)
          V(N,ORIG)=V(N,ORIG)+(PWT(K1)/SPWT)*TRIPS(ORIG,J)
4155       CONTINUE
4160     CONTINUE
4200    CONTINUE
4300   CONTINUE

      RETURN
      END

C     ------------------------------
C     |         SORTD              |
C     ------------------------------
C     PURPOSE: SORT X-Y PAIR IN ASCENDING ORDER BY X(I)*1000+Y(I)
      SUBROUTINE SORTD(X,Y,M)
      DIMENSION X(336),Y(336)
      INTEGER X,Y,XY1,XY2,ISAVX,ISAVY
      K=M-1
1200  J=1
1210  L=0
      DO 1230 I=J,K
       XY1=X(I)*1000+Y(I)
       XY2=X(I+1)*1000+Y(I+1)
       IF(XY1.LE.XY2)GO TO 1230
        IF(L.NE.0)GO TO 1220
         J1=I-1
1220  ISAVX=X(I)
      ISAVY=Y(I)
      X(I)=X(I+1)
      Y(I)=Y(I+1)
      X(I+1)=ISAVX
      Y(I+1)=ISAVY
      L=I
1230  CONTINUE
      IF(L.EQ.0)RETURN
      K=L-1
      IF(J1.LE.0)GO TO 1200
      J=J1
      GO TO 1210
      END
```

APPENDIX F. LISTING OF IEDSS CONTROL MODULE

```
REM
REM      This is the SYSTEM CONTROL MODULE of the IEDSS
REM      Enter the option to execute the IEDSS
REM        0: QUIT
REM        1: GRAPHIC DISPLAY
REM        2: DATA-BASE MANAGEMENT
REM        3: EVACUATION SIMULATION
REM
PAUSE
ECHO OFF
if %1==0 goto STOP
if %1==1 goto GRAPH
if %1==2 goto DBASE
if %1==3 goto ESIMU
:GRAPH
GRAPH.BAT
:DBASE
DBASE.BAT
:ESIMU.BAT
ESIMU.BAT
:STOP
```

APPENDIX G. DEMOGRAPHIC DATA OF VIRGINIA BEACH

```
VA BEACH DEMOGRAPHIC DATA-BASE FOR EVACUATION
 78   ZONES
 12  11667    6568    4454     0    50    50     0  3
 13   2899    2190    1073     0     0   100     0  3
 14   6101    4605    1788   100     0     0     0  3
 15   5814    4505    1523     0     0     0   100  3
 16   1640    1318     587     0     0    30    70  3
 17   5611    3970    1464     0     0    30    70  3
 18   3989    2769     758     0     0     0   100  3
 19   5099    4096    1741     0     0     0   100  3
 20  19995   13688    3563     0     0     0   100  3
 21   4138    2825    1196     0     0     0   100  3
 22   7952    5073    2297     0     0     0   100  3
 23  10918    7130    3192     0     0    50    50  3
 24   1705    1119     557     0     0   100     0  3
 25   6013    4304    1261     0     0    50    50  3
 26   1189     782     280     0     0     0   100  3
 27   6199    4315    2076     0    50    50     0  4
 28   9909    6846    1415     0     0   100     0  4
 29   3958    2820    1255     0     0    10    90  4
 30  10698    6317    2456     0     0   100     0  4
 31   5490    3910    1665     0    70    30     0  4
 32   4762    3771    1282     0     0   100     0  3
 33  11415    6969    2624     0     0   100     0  4
 34   1432    1141     528     0     0   100     0  3
 35   1368     913     391     0     0     0   100  4
 36   1342     893     198     0     0     0   100  4
 37   7198    4789    1382     0   100     0     0  4
 38   8282    5487    2711     0   100     0     0  4
 39  18224   12241    3420     0    20    80     0  4
 40  15436   10747    3448     0    50    50     0  4
 41   3240    2294     708     0    10     0    90  3
 42   6717    4892    1925     0    25    75     0  3
 43   4525    3243    1392     0     0    50    50  3
 44   8304    5668    2494     0     5    95     0  3
 45    738     600     124     0   100     0     0  1
 46    937     761     186     0     0     0   100  1
 47   7233    5412    1443     0    10     0    90  1
 48   3742    2904    1113     0    10     0    90  1
 49   1876    1216     366     0    10     0    90  1
 50   9964    7215    3130     0     0     0   100  1
 51   5013    3640    1291     0    10    90     0  1
 52   3613    3199    1250   100     0     0     0  1
 53    843     747     346   100     0     0     0  1
 54   3264    2891     967   100     0     0     0  1
 55     25      10       0     0   100     0     0  1
```

56	1418	701	345	0	0	100	0	1
57	2465	2369	1304	0	100	0	0	1
58	3032	2619	1476	0	100	0	0	1
59	2033	1666	987	0	0	50	50	1
60	853	695	370	0	30	70	0	1
61	888	727	504	0	50	50	0	1
62	4476	3586	1636	0	50	50	0	1
63	1008	502	576	0	0	100	0	1
64	4876	2685	1818	0	50	50	0	1
65	773	619	442	0	50	50	0	1
66	5860	3432	1618	0	0	100	0	1
67	1445	1158	214	0	100	0	0	1
68	1401	935	685	0	0	0	100	1
69	5099	3496	1335	0	20	0	80	1
70	2693	2158	1811	0	40	40	20	1
71	7631	3452	4300	0	0	0	100	2
72	13548	9075	3739	0	100	0	0	2
73	11302	8113	10971	0	0	0	100	2
74	3722	2233	255	0	0	100	0	2
75	5649	3565	561	0	50	0	50	2
76	963	611	276	0	0	100	0	2
77	2758	2022	216	0	100	0	0	2
78	2715	1992	702	100	0	0	0	2
79	6574	3598	303	0	50	50	0	2
80	6334	4619	1374	0	0	0	100	4
81	7559	5115	1516	0	0	100	0	4
82	2681	1897	812	0	0	100	0	4
83	15823	11052	3034	0	50	50	0	4
84	10096	7052	372	0	0	100	0	4
85	14863	7135	3560	0	0	100	0	4
86	12694	8865	2355	0	0	100	0	4
87	8542	6230	1066	0	0	100	0	4
88	8837	6253	3092	0	0	0	100	4
89	2037	1441	657	0	0	100	0	4

VIRGINIA BEACH EVACUATION ROUTE DATA BASE

```
   13  3  2  5 HERMITAGE ELEMENTARY SCHOOL
S-PLEASURE HOUSE R
S-PLEASURE HOUSE R, W-NORTHAMPTON B, E-I64
   14  3  1  5 HERMITAGE ELEMENTARY SCHOOL
W-SHORE D, N-NORTHAMPTON B, S-PLEASURE HOUSE R
W-SHORE D, W-NORTHAMPTON B, E-I64
   27  4  2  2 WINDSOR OAKS ELEMENTARY SCHOOL
N-S. PLAZA TRAIL, E-OLD FORGE R, S-PRESIDENTIAL B
W-SOUTH B, N-INDEPENDENCE B, W-US44
   31  4  1  5 KEMPSVILLE HIGH SCHOOL
E-SUSQUEHANNA D, E-PRINCESS ANNE R, S-KEMPSVILLE R
W-SUSQUEHANNA D, N-SOUTH NEWTOWN R, W-US44
   37  4  2  3 GREEN RUN HIGH SCHOOL
S-LYNNHAVEN P, N-INDEPENDENCE B
```

NW-HOLLAND R, N-ROSEMONT R, W-US44
38 4 2 1 PLAZA JR HIGH SCHOOL
E-BOW CREEK B, N-SOUTH LYNNHAVEN R
W-BOW CREEK B, N-ROSEMONT R, W-US44
39 4 4 3 GREEN RUN HIGH SCHOOL
W-DAHLIA D
N-ROSEMONT R, W-HOLLAND R, W-US44
40 4 2 2 WINDSOR OAKS ELEMENTARY SCHOOL
N-S. PLAZA TRAIL, N-PRESIDENTIAL B
NW-HOLLAND R, NW-INDEPENDENCE B, W-US44
41 3 2 3 KINGSTON ELEMENTARY SCHOOL
S-LITTLE NECK R, W-KINGS GRANT R
S-LITTLE NECK R, W-VA BEACH B, S-ROSEMONT R, W-US44
42 3 2 4 KINGS GRANT ELEMENTARY SCHOOL
N-NORTH LYNNHAVEN R OR KINGS GRANT R, W-KINGS GRANT R
S-NORTH LYNNHAVEN R, W-US44
44 3 2 2 PRINCESS ANNE HIGH SCHOOL
S-LYNN SHORES D, W-VA BEACH B
W-NORTH LYNNHAVEN R, S-INDEPENDENCE B, W-US44
45 1 2 7 DEY ELEMENTARY SCHOOL
E-ADAM KEELING, S-NORTH GREAT NECK
E-ADAM KEELING, S-NORTH GREAT NECK, E-VA BEACH B, S-LYNN PARKWAY, W-US44
47 1 3 6 COX HIGH SCHOOL
N-NORTH GREAT NECK
S-NORTH GREAT NECK, E-VA BEACH B, S-LYNNHAVEN PARKWAY, W-US44
48 1 4 6 COX HIGH SCHOOL
N-NORTH GREAT NECK
S-NORTH GREAT NECK, E-VA BEACH B, S-LYNNHAVEN PWAY, W-US44
49 1 2 6 COX HIGH SCHOOL
N-FIRST COLONIAL, N-NORTH GREAT
S-FIRST COLONIAL, W-LASKIN, W-US44
51 1 3 6 COX HIGH SCHOOL
S-REAGAN A, N-NORTH GREAT NECK
S-REAGAN A, S-NORTH GREAT NECK, W-VA BEACH B, N-LYNNHAVEN P, W-US44
52 1 1 7 DEY ELEMENTARY SCHOOL
E-SHORE D, S-NORTH GREAT NECK
E-SHORE D, S-NORTH GREAT NECK, E-VA BEACH B, S-LYNNHAVEN P, W-US44
53 1 1 8 GREAT NECK JR HIGH SCHOOL
W-BROAD BAY, S-NORTH GREAT NECK
W-BROAD BAY R, S-NORTH GREAT NECK, E-VA BEACH B, S-LYNNHAVEN, W-US44
54 1 1 8 GREAT NECK JR HIGH SCHOOL
W-SHORE D, S-NORTH GREAT NECK
E-SHORE D, S-NORTH GREAT NECK, E-VA BEACH B, S-LYNNHAVEN P, W-US44
55 1 2 8 GREAT NECK JR HIGH SCHOOL
W-SHORE D, S-NORTH GREAT NECK

```
W-SHORE D, S-NORTH GREAT NECK,E-VA BEACH B, S-LYNNHAVEN
P, W-US44
   56  1  4  8 GREAT NECK JR HIGH SCHOOL
W-ATLANTIC A, W-SHORE D, S-NORTH GREAT NECK
S-ATLANTIC A, S-PACIFIC A, W-LASKIN R, W-US44
   57  1  2  6 COX HIGH SCHOOL
S-ATLANTIC A, S-PACIFIC A, W-LASKIN, N-FIRST COLONIAL,
N-NORTH GREAT NECK
S-ATLANTIC A, S-PACIFIC A, W-LASKIN, W-US44
   58  1  2  4 LYNNHAVEN JR HIGH SCHOOL
S-ATLANTIC A, S-PACIFIC A, W-LASKIN, N-FIRST COLONIAL
S-ATLANTIC A, S-PACIFIC A, W-LASKIN, W-US44
   59  1  2  5 FIRST COLONIAL HIGH SCHOOL
S-PACIFIC A, W-LASKIN, N-FIRST COLONIAL, N-MILL DAM
S-ATLANTIC A, S-PACIFIC A, W-LASKIN, W-US44
   60  1  2  5 FIRST COLONIAL HIGH SCHOOL
S-CARDINAL, W-LASKIN, N-FIRST COLONIAL, N-MILL DAM
S-CARDINAL, W-LASKIN, W-US44
   61  1  2  5 FIRST COLONIAL HIGH SCHOOL
W-LASKIN, N-FIRST COLONIAL, N-MILL DAM
W-LASKIN, W-US44
   62  1  2  1 VA BEACH JR HIGH SCHOOL
24TH STREET
W-US44
   63  1  3  1 VA BEACH JR HIGH SCHOOL
N-PARKS A
W-US44
   64  1  2  2 PAVILION CENTER
N-PARKS A
N-NORTH BIRDNECK, W-US44
   65  1  2  2 PAVILION CENTER
N-PACIFIC A, W-21ST STREET
N-PACIFIC A, W-22ND ST, W-US44
   66  1  2  3 SEATACK ELEMENTARY SCHOOL
N-BIRDNECK
N-SOUTH BIRDNECK, W-US44
   67  1  2  3 SEATACK ELEMENTARY SCHOOL
N-GENERAL BOOTH B, N-PACIFIC A, W-VA BEACH B, N-BIRDNECK
N-GENERAL BOOTH B, N-PACIFIC A, W-22ND ST, W-US44
   70  1  2  3 SEATACK ELEMENTARY SCHOOL
E-LASKIN, S-BIRDNECK
W-LASKIN, W-US44
   71  2  4  1 FIRE TRAINING CENTER
S-OCEANA B, N-GENERAL BOOTH B, N-SOUTH BIRDNECK R
N-OCEANA B, W-VA BEACH B, W-US44
   72  2  2  1 FIRE TRAINING CENTER
W-DAM NECK R, N-GENERAL BOOTH B, N-SOUTH BIRDNECK R
N-OCEANA B, W-VA BEACH B, W-US44
   73  2  4  1 FIRE TRAINING CENTER
E-EAGLEWOOD D, N-GENERAL BOOTH B, N-SOUTH BIRDNECK R
N-OCEANA B, W-VA BEACH B, W-US44
```

74 2 1 4 KELLAM HIGH SCHOOL
SE-PRINCESS ANNE R, HOLLAND R
W-PRINCESS ANNE R, N-BAXTER R, N-INDEPENDENCE, W-US44
75 2 1 5 NORTH LANDING ELEMENTARY SCHOOL
SE-LONDON BRIDGE R, S-OCEANA B, W-PRINCESS ANNE R, W-NORTH LANDING R
N-LONDON BRIDGE R, W-POTTERS R, N-LYNNHAVEN P, W-US44
76 2 1 3 PRINCESS ANNE JR HIGH SCHOOL
N-SEABOARD R
N-SEABOARD R, W-PRINCESS ANNE R, N-BAXTER R, N-INDEPENDENCE, W-US44
77 2 2 3 PRINCESS ANNE JR HIGH SCHOOL
W-SANDBRIDGE R, W-PRINCESS ANNE R, S-SEABOARD R
N-OCEANA B, N-GENERAL BOOTH B, N-SOUTH BIRDNECK R, W-US44
78 2 1 2 PRINCESS ANNE ELEMENTARY SCHOOL
W-SANDBRIDGE R, W-PRINCESS ANNE R, S-SEABOARD R
N-OCEANA B, N-GENERAL BOOTH B, N-SOUTH BIRDNECK R, W-US44
79 2 4 2 NORTH LANDING ELEMENTARY SCHOOL
SE-SALEM R, E-NORTH LANDING R
W-PRINCESS ANNE, W-PROVIDENCE, S-KEMPSVILLE, W-INDIAN RIVER, E-I64