A PLANNING METHOD FOR REGIONAL ROAD INVESTMENTS

By

LEON DARREL CHAPMAN

Bachelor of Science
Oklahoma State University
Stillwater, Oklahoma
1964

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1968

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
DOCTOR OF PHILOSOPHY
July, 1971

ACKNOWLEDGEMENTS

I wish to express my appreciation to the chairman of my committee and thesis adviser, Professor Bennett Basore, for his assistance during this research. I am truly grateful for the many hours he has devoted while assisting my preparation of this thesis.

I also wish to thank Professor Palmer Terrell for his many efforts concerning the dynamic programming formulation contained in this work. I am indebted to the remaining members of my committee for their cooperation and assistance during my entire Ph.D. graduate program.

The financial support from the Center for Systems Science which was formed under Grant GU-3160 from the National Science Foundation is gratefully acknowledged. These investigations were made possible by my participation in the Multidisciplinary Studies Group which was formed under this NSF grant.

I further wish to express my appreciation to my wife for her patience, sacrifices, and encouragement during my graduate studies.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Statement of the Problem. Highway researchers have been plagued for a number of years with the problem of how to decide when and where to build new roads on a regional, intercity, or statewide basis. Most planning procedures to date have been limited in scope by considering only a small network and a small number of alternative network investments. Just recently, with the use of dynamic programming, there have been some formulations of the problem of investing in a road network over an extended time period where the interdependency of road investments were considered. (For example, see Tillman (30); Neufville and Mori (21); Funk and Tillman (12); Bergendahl (4); and Morlok (20)).

The primary aim of this research effort has been to develop a planning process for regional road investments that will optimize over several known investment alternatives. The optimal investment alternatives which occur over several specified time periods will minimize operators' costs, construction costs, and maintenance costs. Operators' costs may include such things as vehicle operation, road accidents, and drivers' time. Construction costs may be formulated as a cost function being composed of right-of-way, road construction, bridges, and signs. Maintenance costs are the monies necessary to maintain or for minor repairs on existing facilities. Most statewide highway planning agencies have been using the Bureau of Public Roads Urban Transportation Program

System 360 (BPR) (31,32) for detailed urban studies. Included in the BPR package is a set of programs dealing with spiderweb networks that may be utilized for intercity network studies. Therefore, if the BPR programs can be utilized, the planning process could be implemented with minimum effort by the highway planner.

1.2 Components of Regional Highway Planning. Every regional highway planning process must address itself to the solution of the traffic generation, the traffic distribution, the traffic assignment, and the road investment problems. A structure for a regional highway planning process is depicted in Figure 1.

The first component of this regional highway planning process is the traffic generation problem. This problem may be defined as the determination of factors which contribute to the desire for travel between two nodes (i,j). Numerous socio-economic factors that could contribute to trip generation have been considered by many researchers. In mentioning a few of the more common factors, one would include population, automobile tag registrations, drivers license, personal income, manufacturing, employment, and wholesale sales. Regression analysis has normally been employed for the determination of the most important trip generation factors.

The second component of regional planning deals with traffic distribution. Basically, the problem is to find the number of trips that exist between each node-pair for a given set of nodes. Trip counts must be distributed for the present and projected into the future for different time period considerations. The traffic distribution problem can be treated with several different approaches. For example, traffic distribution has been studied through use of the Fratar Model, the

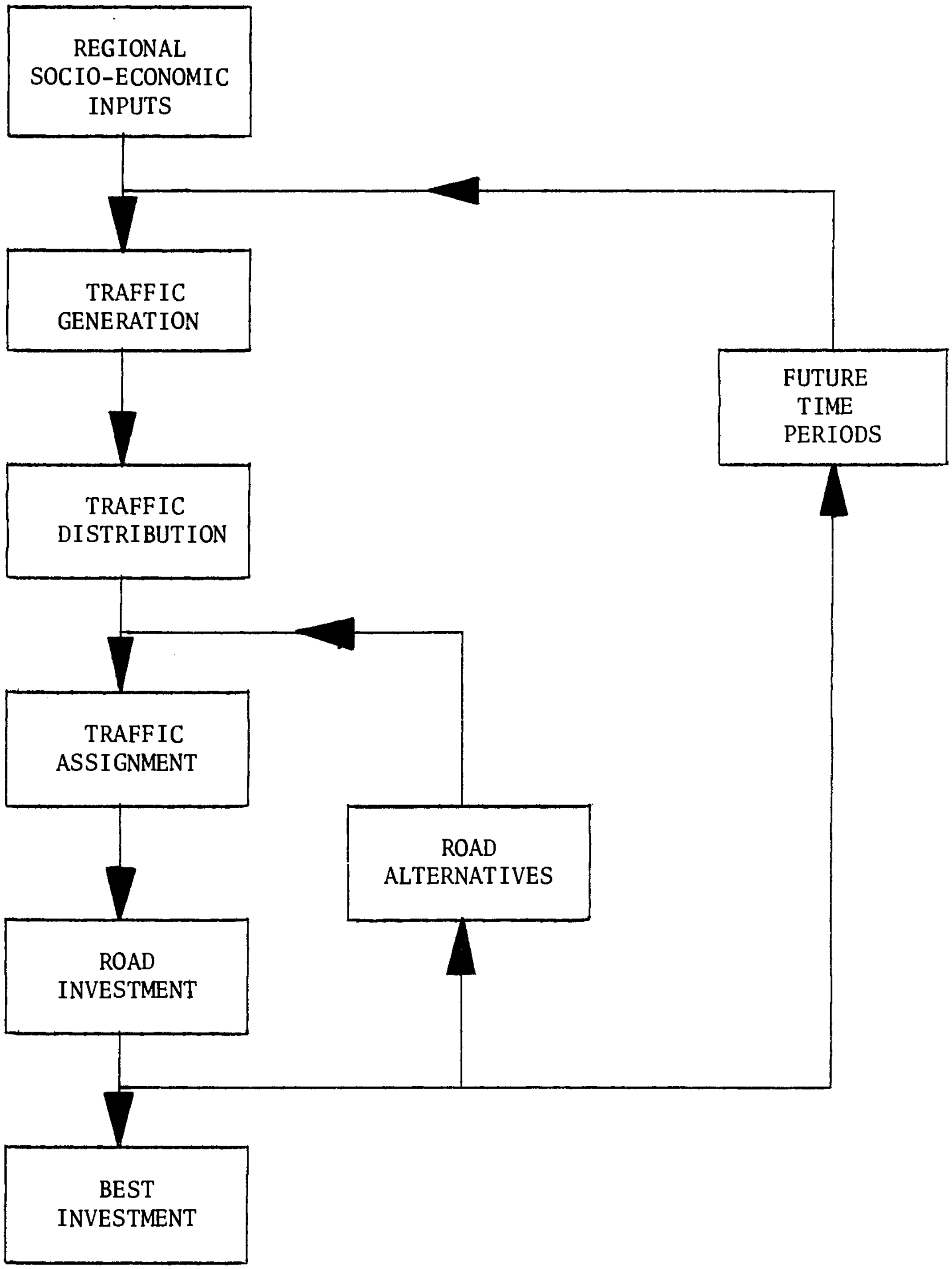

Figure 1. Structure of a Regional Highway Planning Process

Gravity Model and the Intervening Opportunities Model (31). Although others exist, these are probably the most widely used models for traffic distribution.

The third component of regional highway planning is the traffic assignment problem. To solve the traffic assignment problem one must find the traffic allocation that minimizes the sum of operators' cost over all roads for a given set of traffic demands (trips) between every node-pair. The normal assumption made is that the operators' cost has a one-to-one correspondence with the operators' time. With the assumption of only travel time cost, the problem reduces to the minimization of the total travel time on the network. The usual solution of this problem has been formulated by Garrison and Marble (14) through the use of Linear Programming. The BPR programs (32) provide a means of assignment through use of the minimum trees which may utilize minimum impedance or minimum time routes. A minimum time tree with reference to a particular node called N_1 would consist of all the minimum time routes to every node in the network from node N_1. The impedance of a road is a function of the roads' resistance to travel. The impedance might include such things as travel time, distance, ease of travel, and road composition.

The final problem in the regional highway planning process is the road investment problem. This problem begins by assuming the traffic generation, traffic distribution, and traffic assignment problems have been solved. A network operators' cost must be calculated for each feasible network. Construction costs for new roads and maintenance costs for every network are assumed to be given. Knowing these inputs, the problem of road investment is to find the best investment policy for constructing new roads in such a manner that the sum of operators' costs plus construction costs plus maintenance costs is minimized. Road investment analysis should consider several time periods over which investment can be made; thus, coupling the analysis procedure with the budget limitations for investments and the construction process during each period. Since the service functions of new and old roads within a given network extends over a time frame of many years, this multiple-time period characteristic will better serve to portray the real world situation. Several alternative road investments should be considered during each time period. The decision of time and cost of an investment may be dependent upon previous decisions of investment. The ability to cope with these interdependencies should be a characteristic of all regional highway planning models.

1.3 Constraints in Regional Planning. In listing some of the constraints involved in the regional planning process, the number of trips between nodes must be forecasted for the present and each future time period. This in turn implies the utilization of a form of forecasting of growth rates or population increases to project trips into the future.

Another limitation on the solution is the fixed budget during each planning period. This restricts the possible investment plans that may be considered during each planning period. An example of a typical restriction would be to specify that at least 85% of the budgetary monies allocated during each time period be invested.

For realistic highway planning, one might consider a planning period of say 20 years with possible investment alternatives every five years. This would dictate a fixed budget during each five year time period which cannot be exceeded. Since a limited budget exists for each time period (five years) and a variation in trips also exists, it becomes quite evident that any network planning must be done in stages. This restriction tends to point to the utilization of some form of dynamic programming in which an optimal investment is found by optimizing over the total of the time periods (20 years); the optimal decision of investments will be dependent upon the investments made during each time period.

If every possible alternative investment could be considered at each stage, then a global optimum could be obtained; however, due to combinatorial reasons, this number of alternative investment plans becomes so excessive that no current computers can handle such a large problem when dealing with 40 to 100 nodes. Thus a constrained extremum must be found. An appropriate way to constrain the problem is to specify the alternative investment programs to be considered. This specification of alternatives can be a major problem if one is trying to choose <u>the</u> best alternative investment plans from the complete set of alternative plans; however, it seems likely that most highway planners are able to construct a reasonable set of restricted alternative investment programs for consideration for future development.

In summary, the aim of this dissertation has been the development of a regional planning process for optimizing investments in a highway network. The determination of an optimal investment scheme in planning the construction of new highways and improvements has been illustrated. The optimal investment will provide a network that will satisfy the demands placed on it by the users and concomitantly minimize construction costs, network operators' costs, and maintenance costs. The proposed regional highway planning method has the advantage of being application oriented, requires only small implementation procedures, and demands only slight computational times by the computer. In Chapter II a review of the current techniques for solving the road investment problem is presented. Chapter III describes the proposed two-part model which has been developed for solving the traffic generation, the traffic distribution, the traffic assignment, and the regional road investment problems. A brief narrative description of the computer programs utilized within this planning model is presented in Chapter IV. Two example problems are presented in Chapter V to illustrate the ability of this planning model to cope with a large number of nodes in a network, a large set of decisions, and a large number of feasible output networks. Chapter VI provides a summary of the model described in this dissertation and describes the more important traits for employing this planning method.

CHAPTER II

REVIEW OF LITERATURE

2.1 Introduction. The principal objective of a transportation study is to produce a plan for a future transportation network. Drake and Hoel (10) have stated, "Because of previous trends, it would appear that the practice of statewide transportation planning will become as significant as has urban transportation planning." Traffic generation, traffic distribution, traffic assignment, and road investment techniques are an important part of this planning process. In recent years considerable effort has been devoted to traffic generation, traffic distribution, and traffic assignment studies. Several methods of solutions have become available for these problems, and research personnel are now in a position to resolve the road investment problem. Most of the work that has been done to date has been primarily oriented toward urban traffic problems.

2.2 Traffic Generation. Traffic generation may be thought of as the resultant factor of the separation of interurban activities. Many factors have been shown to correlate with trip generation potential. Regression analysis has normally been applied to the many factors that have been considered for trip generation in determining the prime constituents. To list a few of the more common factors, one would include population, personal income, automobile tag registrations, drivers license, manufacturing, employment, wholesale sales, and retail sales.

The implication here is that many factors have been considered and regression analysis has been employed for testing the correlation of these factors.

2.3 Traffic Distribution. In looking into the area of traffic distribution, one soon discovers that a multitude of research and progress has been accomplished in this area. Traffic distribution is a phenomenon, such that trips between nodes are brought into being. Determining the number of such trips is accomplished by actual count or estimated by traffic generation models. The most frequently used models in this area are the Fratar Model (11), the Gravity Model, and the Intervening Opportunities Model. A large part of the following information concerning these three models has been obtained from Urban Transportation Planning System 360 -- General Information, U. S. Bureau of Public Roads (31).

In discussing the first of these models, the Fratar Model, one would classify this method as a growth factor technique. The Fratar Model was developed by Thomas J. Fratar in 1954. A basic assumption in this model is the change in trips in an interchange is directly proportional to the change in trips in the origin and destination zones contributing to the interchange. In predicting future travel, the model uses present travel flows and the predicted growth factors of the origin and destination zones as inputs.

The Fratar Model is applied in the following manner. Each nodepair trip in the base year trip table is multiplied by its destination growth factor. The total from each origin is compared to the desired total and each trip originating from that point is multiplied by the ratio of the totals. This process is repeated by comparing the destination totals to the desired totals and then multiplying each of the elements by the ratio of the actual to the desired total. Then the origins are again compared and multiplied by the corresponding ratios. This method is continued for several iterations until the actual totals become sufficiently close to the desired totals.

A shortcoming of this method is that a base year trip table must be known before starting any iterations. No origin or destination zones are permitted to be removed or added to the system. Therefore, if there were no trips existing between two zones in the base year, an unlimited growth rate would not produce any trips between those two zones. Growth factor techniques tend to retain the instability of prediction between zones where a very small number of trips exist during the base year. Normally, the growth pattern by nature tends to expand in the previously underdeveloped areas. This being the case, the inability of the growth factor techniques to expand existing trips which are near zero becomes a very detrimental drawback for this method. The model is further described in Appendix A.1.

The Intervening Opportunity Model assumes the individual traveler tends to make his trips as short as possible. It is assumed that since a traveler considers the possiblity of stopping at any destination, then there exists a probability of actually stopping at each destination. It is further assumed that the traveler rank-orders the destination zones according to distance or travel time from the origin zone. The model is described in its mathematical form in Appendix A.2.

Probably the most widely used trip distribution model is the Gravity Model. The term Gravity Model comes from the similarity of the trip potential-distance relationship with Newton's law of mutual attraction of physical bodies which has the mass-distance relationship. Basically, the model states that the number of trips from one zone to another is directly proportional to the total number of trips produced by the first zone multiplied by the trips attracted by the second zone and inversely proportional to the distance factor or travel time factor between the two zones. The application of gravity models is widespread and many variations of the basic model have been employed. This model does not require a base year trip table. It can also produce trips between zones which previously had no existing trips. One limitation of this model is the necessity for calibration. The calibration procedure adjusts the base year travel pattern to fit the most current origin-destination trip counts so that the model can be used as a predictive tool. Mathematical formulas for the Gravity Model are presented in Appendix A.3. Apparently, there is little difference in the performance of the Intervening Opportunities and Gravity Model (17).

Other models concerning trip distribution have been proposed such as the Abstract Mode Model (24). Additional information may be obtained in <u>Guidelines for Trip Generation Analysis</u> (16). Similar information and further descriptions of these models have been produced by Overgaard (23) and Deutschman (9). A survey of travel demands has been accomplished by Miller (19), which provides a detailed bibliography in this area.

<u>2.4 Traffic Assignment.</u> Traffic assignment may be defined as a method of allocating a given set of trips between all nodes to a specific transportation network. Trips are loaded onto networks for several reasons. For example, assume a trip table exists for five years in the future. If the trips from this table can be loaded or assigned

to the existing network, then an estimate of the volume levels on each road will have been determined. The future volume levels on the network links will assist in the planning of highway investments of the future in preventing overcrowded highways or inadequate facilities. Traffic assignment also assists in evaluation of different alternative network proposals. A network operators' cost may be determined from the volume of trips on each road and the cost of travel over each road. By minimizing the network operators' costs for several alternative networks, an optimal or minimum cost network may be useful information when planning new constructions.

Beckman (3) and Bergendahl (5) consider some of the various forms of traffic assignment. Jewell (18) describes several models for traffic assignment in considerable detail and several algorithms for multicommodity flow. Almond (1) considers the problem of fixed journey time versus variable journey time as a function of flow. In many cases, the most simple and most efficient method of traffic assignment has been done on a minimum-time route basis. The BPR computer programs (32) assign traffic on the basis of a minimum-time path or minimum-impedance path. These programs are readily available for use on the IBM 360 computer. To obtain a minimum-time or minimum-impedance assignment, it is desirable to use the BPR programs because of their ability to model a large network while using only small amounts of computer time.

2.5 Road Investment. Road investment modeling may be defined as the determination of an investment plan for adding new roads or links to a network which will reduce the total travel time or network operators' costs plus construction costs plus maintenance costs.

Several researchers have attempted to solve the urban road investment problem. Garrison and Marble (14) used a linear programming formulation for the analysis of network improvements. This included the idea of adding capacity and improving the level of service on the link. They assumed the travel cost on each link to be constant. With additional investment in each link, the capacity of the link increased in a linear fashion. The objective was to minimize the travel cost and the investment. Such constraints as flow balance, capacity limits, and budget were included.

Roberts and Funk (13) in 1964 used a linear programming model to add links to a transportation network. The first step was to choose all possible link additions to the network. If the link was added to the network, then it was included in the objective function. Zero-one integer programming was used as a solution technique.

Moving away from the more widely researched urban transportation network problem which is mainly restricted to a grid-type network, Ridley (25) developed an investment policy to reduce the total travel time in a transportation network. The flow on each link was assumed far below the link capacity. He also assumed the travel time could be represented as a linear decreasing function of the investment. The objective was to minimize the total travel time in the network. The objective function was nonlinear since the travel time was a function of both traffic volume and investment. The transportation network was represented by an abstract graph of nodes and arcs. Real-valued variables and functions were defined on the graph to represent travel times, traffic flows, and investment. The travel time on each arc was assumed known as a function of investment in each arc. The assignment of traffic flow to the network was based on the minimum travel time between the nodes under consideration. Ridley's technique finds the optimal set of arcs to invest in to produce the minimum travel time in the network. Upper and lower bounds were found on the minimum travel time for an investment. The method of bounded subsets, which is a form of the branch and bound methods, was adopted to find the optimal investment for a given budget. The technique starts with the maximum investment in the network and progresses toward the budget constraint. A major drawback in Ridley's method is the large number of possible link additions which must be considered. The efficiency of the method is restricted to eliminating a few projects from a set.

In 1969, Scott (27) reported on three different approaches to the solution of the optimal network problem. These solution techniques included: 1) integer programming; 2) backtrack programming; and 3) certain approximative solution algorithms. Integer programming has been used by many researchers and has been discussed in this report. Backtrack programming attempts to reduce the total number of combinatorial possibilities which must be examined in searching for the optimal solution. The method partitions the total solution space into sectors of feasibility and non-feasibility. A systematic search over the combinatorial tree for the network is utilized in searching for the optimal solution. Scott has worked a set of small problems with this method, but noted that the algorithm requires lengthy computations even for small systems of vertices. The algorithm has the advantage of permitting network optimization over any linear or monotonic nonlinear objective function.

The third method that Scott expands on is the approximative solution algorithms. The first algorithm establishes the minimum spanning tree of the network and proceeds toward the optimal solution by adding new arcs in a progressive manner. The algorithm terminates when the cost of the entire network exceeds the set budget constraint. The second approximate algorithm begins with the maximum set of links for the network and deletes links in a systematic order. The performance of the approximative algorithms is compared to the solutions obtained by the backtrack programming method. The approximative algorithms yield excellent results while requiring only small amounts of computer time to obtain the final solution. One restriction of this scheme is that of limited testing on very small networks of ten nodes or less. This method does not solve the road investment problem; it only determines an optimal network given an upper bound on the total length of the network. No staging or multiple-time period considerations are involved.

In 1970, Richard de Neufville and Yasuo Mori (21) extended Funk and Tillman's (12) work. They have developed a computer program using dynamic programming for staging the construction of highway links. The program permits: 1) examination of several stages for each project proposed; 2) multiple-time periods; 3) budget limitations; 4) system costs; and 5) benefits accrued by changes occurring over several time periods. The variable increment dynamic programming procedure was utilized for increasing computational effectiveness. This method incorporates several desirable traits for the solution of the road investment problem.

Tillman (30) proposed a dynamic programming model to determine when the links of highways, railways, and airlines should be added to provide maximum economic benefits to the country. This model differs from similar models in that the resource to be allocated over time is in terms of links of the transportation system rather than the budget. For the model to be fruitful, accurate forecasts of the effect on the system when various links are added and the economic return from each link are necessary. A simple deterministic example problem was illustrated by Tillman. A probabilistic version of dynamic programming could also be used to solve the problem.

Bergendahl (4) formulated a solution method to the road investment problem by combining linear programming and dynamic programming. Linear programming was used to find the optimal traffic operation which minimized travelers' time for each alternative investment. The time for investment was found by using dynamic programming to find the minimum social costs for road investment and traffic operation up to a time horizon, where the future effects were estimated in a scrap value. The scrap value accounted for the future worth of the finalized network. Budget constraints were introduced when the government weighs the investment costs greater than the operating costs; another formulation placed an upper bound on the annual investment budget. An example problem was worked which consisted of 13 nodes and five possible link additions. However, several alternative investment plans were available for each link addition (i.e., no link addition, two-lane addition, four-lane addition). This method considered multiple-time periods and is considered by this author to be the most sagacious way to formulate a solution to the road investment problem.

Bergendahl's formulation has assumed road decisions, construction costs, and social costs occur at the beginning of each planning period. This implies that roads must be constructed instantaneously which in no way ties to the real world situation. Considering the fact that today's existing network is the only feasible input for the beginning stage, it is normally easier to solve the problem by starting at the free end point of the problem (last future planning period) and working toward the fixed end point (the present). Bergendahl's solution by dynamic programming is completely opposite to the above staging scheme. In this research, Bergendahl's basic idea has been expanded upon and an easier application implementation technique has been developed which may consider networks containing a maximum of 8,170 nodes and in the order of 1,000 alternative investment plans at each planning period.

In the article by Sonia Stairs (28), several of the various methods proposed to solve the road investment problem are presented. He discusses the limitation of Ridley's (25) branch and bound method and the integer programming solution proposed by Roberts and Funk (13). Stairs describes a heuristic method for solving the road investment problem that requires a planner to have direct input-output relations with a computer. The planner manually inserts various alternative investment plans into a computer which in turn supplies the consequences of the investment plan almost instantaneously. This "Interactive Program" would be feasible on a "time-sharing" computer system. The crux of the scheme is to try changing the network and then calculate the effect on the network costs. By perturbing the network with several alternatives, it may become apparent that some alternatives are not worth considering. Therefore, a fast reduction in the possible alternatives may be arrived at by the planner. Local effects of changes could be estimated by trials on small sections of the network. This type of method would be useful in providing information about the region of local optimization and sensitivity of the various alternatives.

Stairs points out there are two distinct objectives of network selection. One is to find the best network. The other is to select a network which is better than any network found by current methods. For small problems, one can investigate every alternative but for large problems one shall probably be limited to the second objective. Stairs ignored the time dimension for planning investments to simplify his procedure.

2.6 Summary of Literature Review. In considering the progress that has been made in regional highway planning, the surface has just been dented with techniques that provide a plausible method for solution. Although advancing efforts have been accomplished in developing models for traffic generation, traffic distribution, and traffic assignment, little progress has been made in coupling these methods with a suitable solution to the road investment problem. The main obstacle has previously been the lack of a suitable technique for solving the road investment problem. However, with the advent of dynamic programming, the broader consideration of regional road investment can now be handled, within certain constraints (8).

CHAPTER III

MODEL THEORY AND DESCRIPTION

<u>3.1 Introduction</u>. The proposed planning method for intercity road investments minimizes the sum of operators' costs, construction costs, and maintenance costs incurred over several time periods. Network operators' costs are formulated as the sum of the costs for vehicle operation, the costs of road accidents, and the costs for travel time. The construction costs have been assumed to be a function of right-of-way, road construction, bridges, and signs. Maintenance costs are the costs for maintaining an existing facility.

A beginning assumption for this method considers a network consisting of nodes and links to represent an intercity road system. The nodes represent the location of the various cities within the planning region and the links correspond to the existing and proposed road patterns that interconnect the cities. A set of demands (trips) are developed for each node-pair during each time period. The demands for travel are satisfied by a set of minimum-time routes through the network of linkages. Investments are in the form of additions of new links to an existing network.

The model for the road investment problem consists of two parts. First, the network operators' cost for each possible network is calculated. The network operators' cost is determined from knowledge of the total trip volumes on each link of the network N_i. The network, N_i, will be considered as fixed during each time period, k, which is assumed to be a duration of five years. Therefore, a network operators' cost table will be developed listing the cost for each specified time period. The second part of the solution uses dynamic programming to minimize the sum of construction costs, network operators' costs, and maintenance costs over multiple-time periods.

Since investments are to be made at various points in time, a technique to account for the change in the value of a dollar with time must be employed. A dollar spent now is not the same as a dollar spent a year from now. If one were given the alternative of receiving a dollar either today or a year from today, most people would choose the former (assuming he neglects the effects of interest, inflation, and income tax advantage); however, if the same person were required to give up a dollar either today or a year from now, he would in all likelihood choose the latter. Clearly, the value of the dollar is not independent of time.

The intent of the above example is to illustrate that the reality of an interest (discount) rate as a reflector of the time preference should be divorced from the erroneous conception that interest charges are only relevant when the borrowing of funds is involved. In analyzing the benefits accrued and costs involved with road network investment alternatives, it becomes very advantageous to view an interest rate or discounting rate as a tool for transposing benefits and costs which result at various points in time to comparable values at some common point in time; therefore, all monies within this dissertation will be discounted to their present worth value with an assumed interest rate of seven percent.

3.2 Network Operators' Cost Formulation. The first assumption in determining a network operators' cost is that the regional road network can be isolated from another region. (Fixed boundary conditions on any network will always be a point to question.) The existing roads will be considered as links of a graph and with nodes representing the cities within the planning region. Dummy nodes may be used to represent the intersection of roads between cities. These dummy nodes would be viewed as zero trip distribution nodes. Each investment alternative will be viewed as the opening of a link at the end of a time period while the decision was made at the beginning of the time period. The network will be considered as fixed during each time period, i.e., traffic must be free to operate on all links within this period. Five years has been chosen as the duration of a period. This seems plausible when consideration is given to the time lag for construction of a road after the decision has been made that the road must be built.

The demand for travel can be defined as a relation D_{ij}^{k} between two nodes (i,j) in the network during period k. The demands may be expressed in terms of trips through use of origin-destination data. Trips (demands) can be distributed during each time period by employing an Intuitive Gravity Model (2). By an Intuitive Gravity Model, it is inferred that trip distribution is dependent on a single factor similar to mass in the Gravitational Model. This assumes that the desire for intercity travel is uniform over the region being considered. Personal income for each node can be considered as the sole factor that generates trips between nodes. This follows from the assumption that the demand (trips) is highly correlated with economic activity, and personal income would also be highly correlated with such activity. The desire for travel is counteracted by the distance separating the cities or nodes. The distribution model is

$$D_{ij}^{k} = \beta \frac{(PI_i^k)(PI_j^k)}{r_{ij}^{\alpha}} \qquad (3.2.1)$$

where:

D_{ij}^{k} = demand between nodes (i,j) during period k;

PI_i^k = personal income for node i during period k;

PI_j^k = personal income for node j during period k;

r_{ij} = distance between nodes (i,j);

α = 2.78 - impeding factor (6);

β = 440 = calibration constant to express demands in terms of trips (2).

A trip table will be calculated for each time period k. A growth factor for personal income will be used in order to simulate the increase in trips during future time periods. (The Fratar Model could be used in this case after obtaining a base year trip table to project future trips; however, the author feels it would be more accurate to use some forecasting technique for increasing personal income for each time period and then reuse the Gravity Model to build a new trip table.) Trip tables will be constructed for each time period (k = 1,2,...,n).

After the trip tables have been constructed, each trip table must be assigned to every feasible network that can exist during each period k. These trips will be assigned to each network by employing a minimum-time route. The BPR programs develop a minimum-time route tree for each node to every other node in the network. Therefore, if you have N nodes in a network, there will be N trees for that network. The tree building starts at node 1 and finds the shortest time route to node 2. It then determines the shortest time route between nodes 1 and 3. The process continues until a route is determined for nodes 1 and N. This would compose the tree for node 1 (TR_1). After all trees ($TR_1, TR_2, \ldots, TR_N$) have been built for a network, the trips will be loaded through the use of the minimum trees onto the network. This will form a total load on the network which has minimized travel time costs on a given network.

For mathematical formulation, the following definition will be made. The links will be numbered $\ell = 1,2,\ldots,m$ where there are m links in a network. The link volume or total trips on a link will be denoted as $LKVOL_k(\ell)$ to represent the average total trips per period on link ℓ during time period k. The operating cost on each link during period k will be a function of this link volume. In this analysis, all links will be assumed to operate well below their capacity. This assumption rules out congestion effects on roads between cities. The effects of congestion and travelers impedance can be handled by the BPR programs if the road impedance or resistance to travel is known. Although this impedance concept is not being used in this report, this method of using impedance of a link to find minimum-impedance trees would provide a very meaningful solution to this problem. Further research in the determination of impedance is being conducted by Osborne (22).

The operators' cost on a given link ℓ is assumed to be a function of the time period k. Therefore, a link operators' cost for each link ℓ, $LKCOST_k(\ell)$, must be known valid, for time period k. The total network operators' cost during period k may then be formulated as:

$$NOC_k = \sum_{\ell} LKCOST_k(\ell)[LKVOL_k(\ell)] \qquad (3.2.2)$$

where $LKCOST_k(\ell)$ is the average operating cost for link ℓ for one trip during period k. From Equation (3.2.2), the network operators' cost for a given network is determined for period k. One must then calculate this network operators' cost (NOC_k) for every feasible network that may exist during period k.

With the computation of the above cost tables, one has completed the first step in the solution of the regional road investment problem. These derived network operators' costs will be used as inputs to the road investment problem.

3.3 Regional Road Investment Formulation. The second part of the solution is to determine the optimal investment scheme. At this point, the problem of determining an optimal road investment sequence may be formulated as a multistage decision process. A multistage decision process is a method in which a sequence of decisions are made such that a decision in one period affects the decisions made in succeeding periods. A method of solving this type of problem is dynamic programming (DP).

At this point, a short digression to explain dynamic programming seems to be in order. Basically, dynamic programming is a decomposition technique for solving multistage decision problems. The DP approach is to decompose the n-decision variable problem into n one-decision variable problems. In many cases, the n subproblems are much easier to solve than the original problem. The decomposition is accomplished in such a manner that the optimal solution to the n-variable problem is obtained from the optimal solutions to the n one-dimensional problems.

Dynamic programming is based on Bellman's (7) "principle of optimality" which states: "An optimal policy (set of decisions) has the property that whatever the initial state and decisions are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision."

Consider the n-stage decision process shown in Figure 2. The beginning stage is considered to be stage n. The numbering of stages is assumed to represent the number of stages remaining in the total planning period of n stages. Each stage in the process has an input S_k, which in turn was the output of a previous stage, $\tilde{S}_{k+1}$. The input to stage k is acted upon by a decision D_k which transforms the input to some output function $S_k = \tilde{S}_k<D_k, S_k>$[1], producing a cost C_k and an output $\tilde{S}_k$. The output $\tilde{S}_k$ now becomes the input S_{k-1} to the next stage k-1. The decisions, D_k, at each stage are restricted to a known set of permissible alternatives A_k. The cost at stage k is a function of the input S_k and decision D_k. Therefore, $C_k = C_k<D_k,S_k>$ measures the cost of making decision D_k when the input to the stage is S_k.

The objective of this multistage decision process is to optimize some function of the individual stage costs, $F<C_1,C_2,C_3,\ldots,C_k>$. The composition of the n-stage cost function determines whether a given problem can be solved by dynamic programming.

---

[1] $T_k<D_k, S_k>$ implies $\tilde{S}_k$ is a function of D_k and S_k.

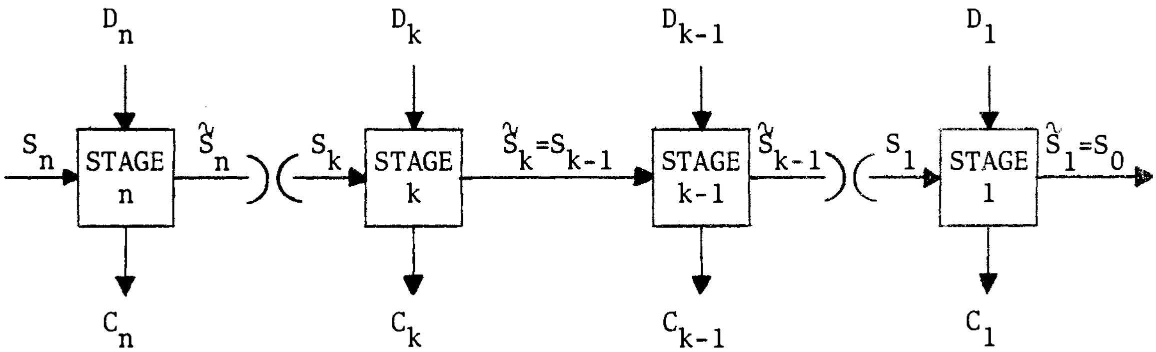

Figure 2. n-Stage Decision Process

Now consider the implication of the "principle of optimality" to the above multistage decision process. Suppose the objective is to minimize the n-stage objective function $F_n\langle S_n D_n\rangle$ which is the sum of the individual stage costs,

$$F_n^*\langle S_n\rangle^2 = \min_{D_n,\ldots,D_1} [C_1\langle D_1,S_1\rangle + C_2\langle D_2,S_2\rangle + \ldots + C_n\langle D_n,S_n\rangle]$$

$$= \min_{D_n,\ldots,D_1} [\sum_{k=1}^{n} C_k\langle D_k,S_k\rangle] \tag{3.3.1}$$

subject to

$$\tilde{S}_k = S_{k-1} = T_k\langle S_k,D_k\rangle \qquad k = 1,2,\ldots,n .$$

Equation (3.3.1) may be written as

$$F_n^*\langle S_n\rangle = \{\min_{D_n} (\min_{D_{n-1},\ldots,D_1} [C_1\langle D_1,S_1\rangle + \ldots + C_n\langle D_n,S_n\rangle])\} \tag{3.3.2}$$

subject to

---

[2] $F_n^*\langle S_n\rangle = \min_{D_n}[F\langle S_n,D_n\rangle]$.

$$\tilde{S}_k = S_{k-1} = T_k<S_k,D_k> \qquad k = 1,2,\ldots,n \ .$$

Note that $C_n<D_n,S_n>$ does not depend on $D_{n-1},\ldots,D_1$. Therefore, the minimization with respect to these variables may be accomplished as shown below

$$F_n^*<S_n> = \min_{D_n}\{C_n<D_n,S_n> + \min_{D_{n-1},\ldots,D_1}[C_{n-1}<D_{n-1},S_{n-1}> + \ldots + C_1<D_1,S_1>]\} \tag{3.3.3}$$

subject to

$$\tilde{S}_k = S_{k-1} = T_k<S_k,D_k> \qquad k = 1,2,\ldots,n \ .$$

Using Equation (3.3.1), one may rewrite Equation (3.3.3) in the following manner:

$$F_k^*<S_k> = \min_{D_k}[C_k<D_k,S_k> + F_{k-1}^*<S_{k-1}>], \qquad k = 1,2,\ldots,n \tag{3.3.4}$$

where

$$\tilde{S}_k = S_{k-1} = T_k<D_k,S_k> \qquad k = 1,2,\ldots,n \ .$$

$$F_0^*<S_0> = 0.$$

Equation (3.3.4) is the mathematical statement of Bellman's "principle of optimality" for a serial multistage decision process.

Now one can verbally bond Equation (3.3.4) with the n-stage decision process shown in Figure 2. The solution procedure takes place in the following manner: Start with the last stage which is stage number 1, and determine the optimal D_1 (denoted by D_1^*) by minimizing the cost C_1 for each value of input S_1. Then, the optimal D_1^* has been determined independent of the decisions made at prior stages.

Next, one can consider the last two stages, stages 1 and 2. Remembering that regardless of the input S_1 to stage 1, the optimal decision D_1^* to minimize the cost C_1 from stage 1 will be chosen. Consider all possible inputs to stage 2. From these inputs determine the optimal decision D_2^* by minimizing the total cost from the last two stages. It should be mentioned that the input to stage 1 is a function of the decision and states at stage 2, but one has previously determined the optimal decision D_1^* at stage 1 for every possible input state. One has now completed the optimization of a two-stage process. Extending the procedure to stage 3, one determines D_3^* by considering the cost of the 3rd, 2nd, and 1st stages. Since the optimal cost C_2^* at stage 2 is for all succeeding stages, one does not have to reconsider stage 1. Therefore, for each input S_3, one determines D_3^* considering the cost of stage 3 and the total cost of succeeding stages C_2^*. The process is continued until one has reached the final stage where one determines D_n^*. Through this decomposition procedure one has determined an optimal decision at each stage for the entire problem. The optimal decisions must be traced back through the solution starting at the final stage n to arrive at the individual decisions at each stage.

With the above description of DP, one is in a position to formulate a solution for the road investment problem. One can assume that each stage in the DP technique is a period of time t, say five years. This would be the approximate time required to construct new road alternatives for a known set of projects. The input to the stages will be defined as the existing road network which includes any links that have been constructed in the previous stages. It will be convenient to introduce the concept of "state" of the road network. A state S_k will be inputs to the stages which in essence is the set of road network links that are open for traveling during period k. The decisions D_k at each stage k will be which of the alternative roads or links to open for travel at the end of period k. The function $T_k<D_k,S_k>$ which transforms one state S_k into the next state S_{k-1} is assumed to take place at the end of time period k and before the beginning of time period k-1. The transformation may be written as follows:

$$\tilde{S}_k = T_k<D_k,S_k> = S_{k-1} \qquad (3.3.5)$$

The initial state of the road network prior to any road investments will be defined as S_n. An investment alternative for four time periods will be a sequence of state pairs $[(S_1,\tilde{S}_1), (S_2,\tilde{S}_2), (S_3,\tilde{S}_3), (S_4,\tilde{S}_4)]$. Corresponding with these state pairs will be a set of decisions (D_1,D_2,D_3,D_4).

The cost $C_k<S_k,D_k>$ is the sum of the network operators' cost plus construction cost plus maintenance cost. It is assumed that the construction costs $CC_k<S_k,D_k>$ are known for every possible transition $(S_k,\tilde{S}_k)$. These costs may be formulated as a function of the costs for right-of-way, road construction, bridges, and signs. The maintenance costs $MC_k<S_k>$ will be considered to be the required cost to maintain the network which exists at the beginning of each period. The decisions and construction costs are assumed to occur at the beginning of each time period. The network operators' costs are charged at the end of the period. All costs are assumed to be fixed for a duration of five years. Each cost will be discounted to its present worth value at the beginning of the last stage n. This procedure permits comparison of cost alternatives based on their present worth value over multiple-time periods.

Mathematically, the total cost of each stage $C_k<S_k,D_k>$ may be expressed as

$$C_k<S_k,D_k> = PS(NOC_k<S_k,D_k>) + CC_k<S_k,D_k> + MC_k<S_k>$$
$$= PS(NOC_k<\tilde{S}_k>) + CC_k<S_k,D_k> + MC_k<S_k> \quad (3.3.6)$$

where

$$PS = \frac{1}{(1+i)^t}, \quad \begin{array}{l} i = \text{interest rate (.07 assumed);} \\ t = \text{time period of one stage (5 years).} \end{array} \quad (3.3.7)$$

The PS factor properly discounts the network operators' cost which is charged at the end of each stage; therefore, the operators' cost of the newly formed roads is reflected back to the beginning of the stage.

This charging of the network operators' cost at the end of the time period serves to delay the benefits resulting from the new road additions. In Equation (3.3.6), the NOC is actually the cost for operating on the new facilities, $\tilde{S}_k$, after the construction decision has been made. This technique provides a method of accounting for the benefits accrued in the final planning stage of the DP process; otherwise, the last stage would always represent a decisionless stage because any investment in new road constructions would tend to increase the objective function without any offsetting benefits. The only problem encountered with this method occurs at stage n (input stage) of the DP process. The network operators' cost which should be charged for operating on today's existing highway network is neglected in stage n. Since this cost is fixed and independent of the decisions made at stage n, its consequence will only increase the total objective function by a fixed amount; therefore neglecting the input stage network operators' cost will have no affect on the optimal solution calculated by the DP program. The above assumptions are consistent when a reapplication of the DP process for the next n-stage planning period is encountered.

A mathematical formulation of the road investment problem may be formulated as

$$F_k^*<S_k> = \min_{D_k}[C_k<D_k,S_k> + PS(F_{k-1}^*<S_{k-1}>)], \qquad k = 1,2,\ldots,n \qquad (3.3.8)$$

where

$$\tilde{S}_k = S_{k-1} = T_k<D_k,S_k> \qquad k = 1,2,\ldots,n$$

$$F_0^*<S_0> = 0.$$

The PS factor in Equation (3.3.8) discounts all accumulated costs from the beginning of one stage k to the beginning of the previous stage k+1.

So far no mention of budget constraints during each stage has been considered. A limit on the monies to allocate for construction costs in each period is logical. Since one is normally working under a budget for each period, Equation (3.3.8) should incorporate these constraints. Therefore, Equation (3.3.8) may be rewritten to include the budget constraint as

$$F_k^*<S_k> = \min_{D_k}[C_k<D_k,S_k> + PS(F_{k-1}^*<S_{k-1}>)], \qquad k = 1,2,\ldots,n \qquad (3.3.9)$$

where

$$\tilde{S}_k = S_{k-1} = T_k<D_k,S_k> \qquad k = 1,2,...,n$$

$$F_0^*<S_0> = 0$$

subject to

$$CC_k \leq B_k \qquad k = 1,2,...,n$$

where:

CC_k = construction cost for period k;

B_k = budget limitation for period k.

In summary, the solution to Equation (3.3.9) will provide the optimal solution to the road investment problem for the feasible investments considered (i.e., this solution minimizes the sum of network operators' costs plus maintenance costs plus construction costs). In this planning method for regional road investments one must know the network operators' costs for each time period as found in Equation (3.2.2). In addition, one needs the cost of maintenance for each network during each period and the construction cost for each feasible investment. Coupled with these costs, the budget limitation for each period must be known. An assumed interest rate to discount monies and the length of a time period provide all the necessary inputs to find a solution to the regional road investment problem. Keep in mind, however, that the network operators' costs must be formulated through the use of a gravity model to distribute trips for each time period considered. The solution procedure uses the BPR programs to determine network operators' costs and then DP to solve the investment problem.

CHAPTER IV

COMPUTER PROGRAM DESCRIPTIONS

4.1 Introduction. All of the following programs were written for the IBM System 360 and were executed on a Model 65 computer. Only those programs concerning spiderweb networks were employed from the BPR package of programs (32). From this set of programs, a technique was developed to compute the network operators' cost for every feasible network state. The calculation of the network operators' costs required the following BPR programs: 1) Build Spiderweb; 2) Format Spiderweb; 3) Build Spider Trees; 4) Format Spider Trees; and 5) Load Spiderweb. In conjunction with these BPR programs, three supplementary programs were developed. These programs were entitled: 1) Build Trip Table; 2) Format Link Volume; and 3) Network Operators' Cost.

After the computation of all network operators' costs, a final program using dynamic programming was employed for the optimization process. A brief description of each of the programs will now be presented. The major portion of the BPR program descriptions has been procured from the General Information documentation by the Bureau of Public Roads (31).

4.2 Build Spiderweb (BLDSPWB). A spiderweb network is a simplified network of direct connections between centroids. The spiderweb programs were originally written for use in nationwide traffic assignments where counties are used as zones and the network connects centroids representing the zonal system. Assignment of trips to such a network represents major corridor movements within the area described by the network. The input for this program requires the coordinates for each centroid in the area plus additional description as desired by the user. In addition, an input of connector cards is required stating which centroids should be connected to each centroid. To minimize the coding requirements, only centroids connected to higher numbered centroids need be recorded. With the addition of the last two programs which couple the BPR programs, it is mandatory that each link connection card be a sequence of higher ordering of nodes. The program permits either a maximum of four connectors or eight connectors at each centroid. For an eight connector network, a maximum of 8,170 nodes may comprise a spiderweb network. A maximum of 16,362 nodes may compose a network with only four connectors per node. Eight connector networks were utilized in the research presented in this dissertation. From the coordinates and connectors, this program prepares a network and writes it out together with such additional information as included in the coordinate file as a network record which is very similar to a historical record. A list of the input data cards and execution cards is illustrated in Appendix B.1.

4.3 Format Spiderweb Network (FMTSPWB). This program provides a standard format or computer print out of a spiderweb network description as initially built by the program BLDSPWB. The output format supplies the x-y coordinates of node A, the node number, the impedance and distance of the link to every node B connected to node A. This information is formated for every node in the spiderweb network. A listing of the execution set-up for FMTSPWB is shown in Appendix B.1. A representative output from this program is shown in Appendix B.2.

4.4 Build Spider Trees (BLDSPTR). This program uses the network record described above as input and produces a tree file and/or a tree time file under user option. This permits the user to either select minimum-time routes or minimum-impedance routes through the network. Since this is a spiderweb network with up to eight connectors, turn penalities and capacity limitations are not permitted. A tree may be defined as a record showing the shortest route from a given node to all other nodes in the highway network. These spider trees provide a minimum-time route when loading a trip table onto the network. Spider trees may be constructed to provide minimum-impedance routes through the network. This would be a more realistic situation that would account for the variable road conditions that affect the travelers' choice of roads. The execution of BLDSPTR is shown in Appendix B.1.

4.5 Format Spider Trees (FMTSPTR). This program provides a standard format of spiderweb trees. The program FMTSPTR permits a maximum of thirteen trees to be printed in a single computer run. Normally, it is not necessary to format all or even thirteen trees from a given network. A cross-sectional sample of trees taken throughout the network can be printed for the user to check minimum-time routes for accuracy. This program may be deleted after sample runs have insured the correct operation of the BLDSPTR program. A listing of the execution of the FMTSPTR program and its output is presented in Appendix B.3 and B.4, respectively.

4.6 Build Trip Table (BLDTT). This program was originally written for the MSG group at Oklahoma State University during their study of the "Travel Demand in Oklahoma" (2). The program has been written in Fortran IV computer language for use on the IBM System 360 computer. BLDTT permitted the coupling of the Intuitive Gravity Model in Equation (3.2.1) with the BPR programs. The BLDTT program requires the latitude and longitude in degrees and minutes and personal income for each node in the network as inputs. The trip tables generated by this program were assumed to be symmetric. The trip table output has been written in the proper form to input with the next BPR program, Load Spiderweb. Utilization of this program provides the solution for the traffic distribution problem in Figure 1 on page 3. The BLDTT program is executed only one time for each five year time period employed by the DP technique. Each period trip table was stored on a disk data set which provided an input for the execution of Load Spiderweb program. A program listing of BLDTT is shown in Appendix B.5. Representative outputs from this program are illustrated in Chapter V, Tables VII, VIII, IX, and X.

<u>4.7 Load Spiderweb (LDSPWB)</u>. This program requires as input the network record, the tree file, and a trip table. The output is an updated network record which includes the link volumes found by this program. The trips are loaded onto the network through the minimum trees found by the BLDSPTR program. The program is flexible as to zones that may be loaded. Options are provided for loading all zone-to-zone movements, selected origins to selected destinations, all origins to selected destinations, or selected origins to all destinations. The BPR program to format the spiderweb loads was not sufficient in the calculation of a network operators' cost. Therefore, a converter program was added to provide the required output. The LDSPWB program is the last description of the BPR programs utilized in this planning method.

The execution of this BPR program is contained in Appendix B.6.

4.8 Format Link Volume (FMTLKVOL). This program written in the PL1 computer language for the IBM System 360 Model 65 computer is strictly a read-write type program. It reads the output from the LDSPWB program saving only the link volumes and sequentially assigns numbers to the links within the network. The numbering is accomplished by starting at node 1 and assigning each link connected to node 1 a sequencing number that begins at the lowest external node connection. After all links connected to node 1 have been assigned a link number, the program then proceeds to node 2 and continues the assignment. This process is continued until every link has been assigned an index number. This ordering of the links is very useful when calculating the network operators' cost. The printed output lists the A node, the A to B node connection, and the link A-B volume. The program listing of FMTLKVOL is presented in Appendix B.7.

4.9 Network Operators' Cost (NTOPCOST). This program written in the Fortran IV computer language performs the multiplications and summations required in Equation (3.2.2) in calculating the network operators' cost for each network. Several feasible network operators' costs may be calculated by using the same spiderweb network. The only change required is the link travelers' costs for traveling on a two-lane road as compared to traveling on a four-lane road. Therefore, the network operators' costs for all feasible networks having the same link connections, but permitting two to four-lane changes, are computed in a single run of the above programs. The NTOPCOST program will loop back and recalculate the network operators' costs for each two to four-lane change that creates a feasible network state.

The required inputs for this program are the link numbers that may change, the network number, the state of each network, the period number, and a cost table for all possible link operators' costs. The program uses the network state and selects from the matrix of link costs the proper two or four-lane costs for a given network. On the average, a total of eight different network operators' costs were calculated in a single run of this program. The program listing of this program is shown in Appendix B.8. The output from the execution of Program 2 in Appendix B.6 which includes a sequence of four period outputs from the programs FMTLKVOL and NTOPCOST is shown in Appendix B.9.

4.10 Dynamic Programming (DP). Written in the Fortran IV computer language, this DP program minimizes the sum of network operators' costs, maintenance costs, and construction costs over a specified number of time periods. Budget limitations on new construction are imposed during each period. The solution of Equations (3.3.6), (3.3.7), and (3.3.9) are found by starting at the free end of the problem and working toward the fixed input (the road network that exists today).

Required inputs are the following: 1) the network operators' costs for each period and each specified network; 2) maintenance costs for each network and each period; 3) construction costs for each feasible decision; 4) budget limitations for each period; 5) the interest rate for discounting monies; 6) the decision numbers and decision states; and 7) the network numbers and network states.

The output of this program provides the optimal cost accumulations for each period for the various network states. The optimal decisions and up to two ties at each stage are printed and the optimal decisions are traced back through the stages after a solution has been found. In addition, the construction costs for each feasible decision, the maintenance costs, and the operators' costs are printed as part of the output.

A listing of this program is presented in Appendix B.10. Representative output from this program is shown in Tables XIV, XV, XVI, and XVII, in Chapter V.

4.11 Program Utilization. The execution of programs required in determining the various network operators' costs after the trip tables were constructed by the BLDTT program was combined into just two programs. Referring to Figure 3, the first program executed the BLDSPWB, the FMTSPWB, the BLDSPTR, and the FMTSPTR programs. This program constructed a highway network and all of the minimum-time routes in preparation of the loading of trips onto this network. The second program was repeated the same number of times as the number of trip tables constructed to handle each of the specified time periods. The NTOPCOST program calculated each network operators' cost created by two to four-lane changes within a fixed network configuration. After the complete set of network operators' costs have been computed, the dynamic programming program can be run one time for the determination of the optimal costs and optimal decisions at each stage.

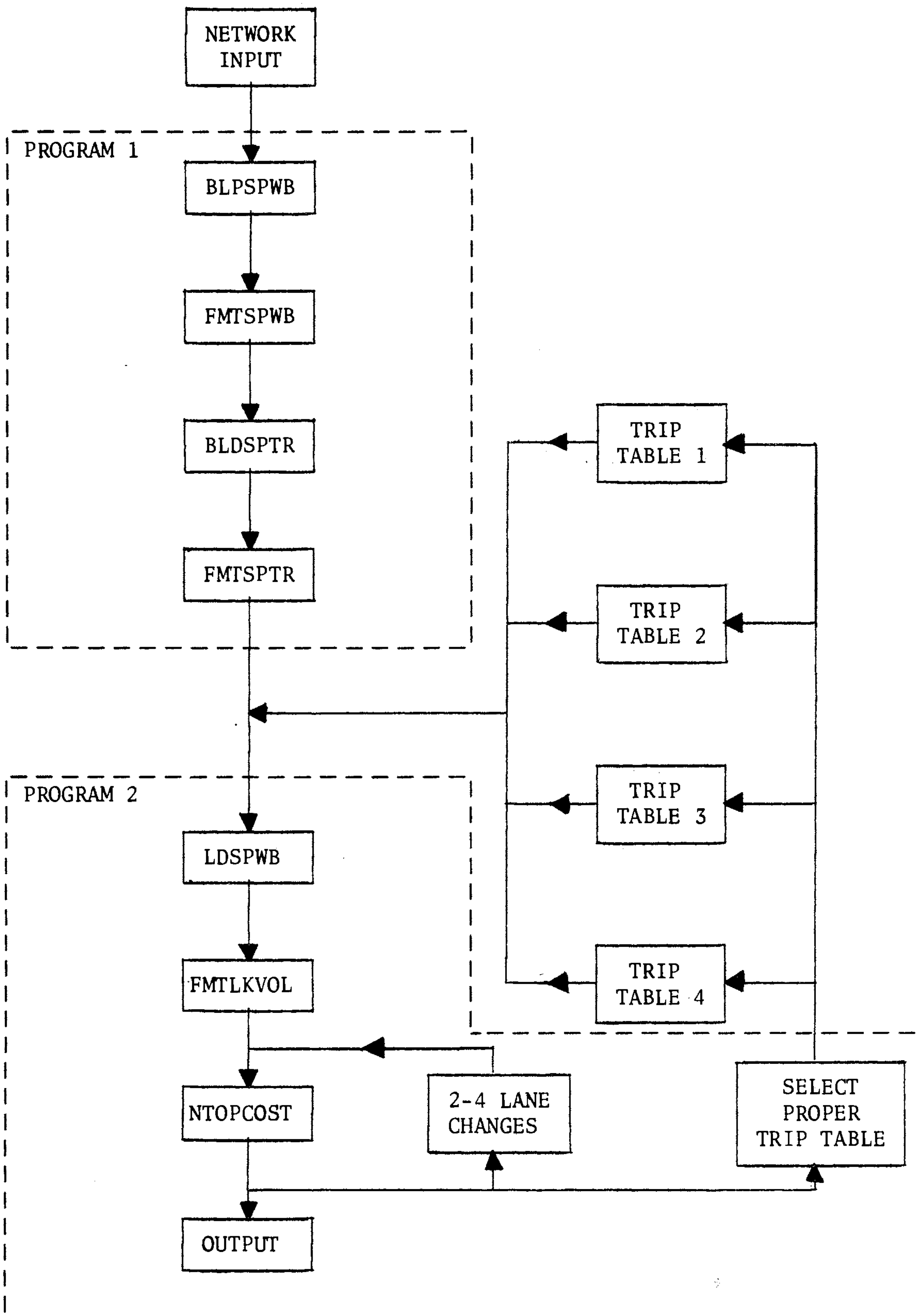

Figure 3. Computer Block Diagram for Calculation of Network Operators' Cost

CHAPTER V

APPLICATION OF THE MODEL

5.1 Introduction. Two example problems are presented in this chapter to illustrate the regional planning model developed in Chapter III. The first example problem, which is rather trivial assumes a network of six nodes and two possible link additions. A pictorial sample of the various network link interconnections and minimum trees is presented. Three stages for planning the construction of new roads has been assumed. There are nine feasible network states at the input of the last two stages and 16 feasible decisions to consider for altering each of these nine states. The required input data has been assumed and the optimal investment policy has been computed. This problem is trivial in the sense that it can be worked by hand in a short amount of time; however, it should provide a meaningful description of the modeling effort being employed.

The second problem that is presented is far from trivial and serves to demonstrate the power, the ease of application, the small cost, and the versatility of this planning method. The problem begins with a network that is composed of the largest 53 personal income towns (regions) in Oklahoma. Eight possible links may be changed within this network. From these, 200 feasible network states and 326 alternate investment decisions were assumed. The optimal decision policies are presented for each stage of the total planning period of 20 years.

5.2 A Simple Six-Node Problem. A very small problem was considered first to insure that all computer programs were operating correctly. Although the numbers used throughout this example are not realistic, the example problem serves to exhibit the technique of applying the planning method for regional road investments. The network in Figure 4 is assumed to represent six cities (nodes) with six existing roads (solid lines) connecting the cities. The dashed lines represent roads (links) that have been proposed for addition to the existing network. The nodes are numbered 1 through 6. The numbers between the nodes represent distance in miles. Three time periods each consisting of five years were assumed. The following trip tables in Table I were given inputs for each period. These tables represent three outputs from the traffic generation and traffic distribution problems. The traffic generation problem would require 1970 personal income data for each node in the network. This personal income data would then be projected into each of the future periods (1975 and 1980). To generate the trip tables or solve the traffic distribution problem would then be a simple matter of solving Equation (3.2.1) for the trip table entries.

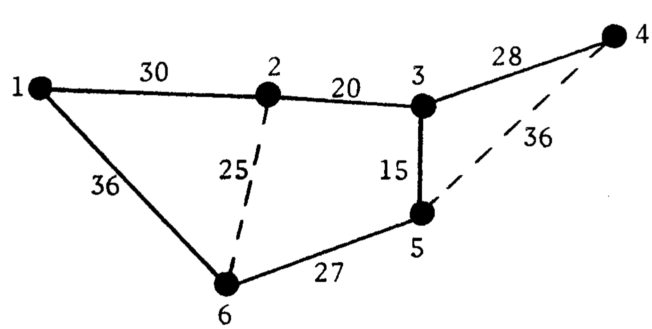

Figure 4. Assumed Six-Node Network
(Approximate Scale)

TABLE I

TRIP TABLES FOR PERIODS 1970, 1975, AND 1980

	1	2	3	4	5	6
			TT3 - 1970			
1	0	15	10	5	4	12
2	15	0	8	6	6	9
3	10	8	0	10	15	8
4	5	6	10	0	10	8
5	4	6	15	10	0	10
6	12	9	8	8	10	0
			TT2 - 1975			
1	0	18	12	6	5	14
2	18	0	10	7	7	11
3	12	10	0	12	18	10
4	6	7	12	0	12	10
5	5	7	18	12	0	12
6	14	11	10	10	12	0
			TT3 - 1980			
1	0	30	15	8	15	16
2	30	0	15	20	20	25
3	15	15	0	25	30	35
4	8	20	25	0	30	35
5	15	20	30	30	0	15
6	16	25	35	35	15	0

Four different networks were used as spiderweb networks in the Bureau of Public Roads Programs. These networks are shown in Figure 5.

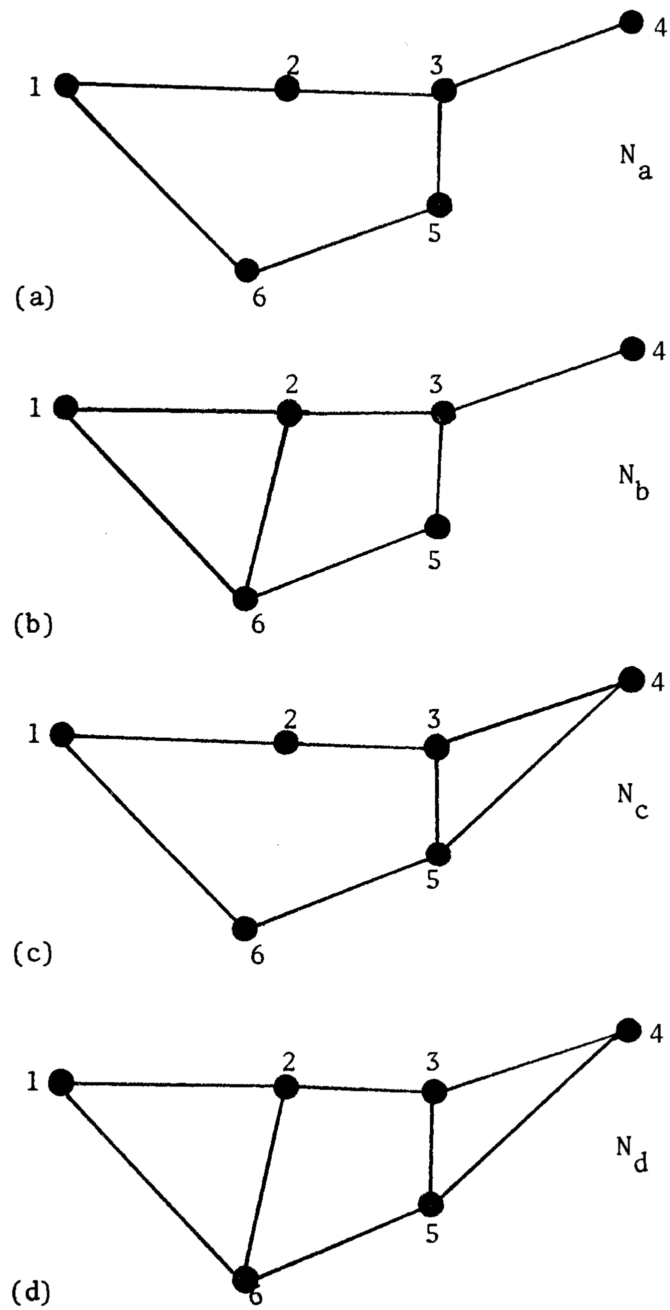

Figure 5. Spiderweb Network Link Connections (Approximate Scale)

The following programs from the BPR package were used: 1) BLDSPWB-to build the spiderweb networks in Figure 5; 2) FMTSPWB-to format the spiderweb network descriptions; 3) BLDSPTR-to build spider trees (minimum-time routes); 4) FMTSPTR-to format the spider trees; and 5) LDSPWB-to load the three trip tables on each network. In addition to these, the two coupling programs, FMTLKVOL and NTOPCOST, were employed to calculate the network operators' costs for each network. The outputs of the network operators' cost program served as inputs for the dynamic programming program.

A set of minimum-time trees constructed by the BLDSPTR program is shown in Figure 6 for network N_a of Figure 5a. The LDSPWB program provided the following link volumes (LKVOL) for network N_a in Figure 5. A link cost (LKCOST) matrix was assumed to represent operators' costs per trip on each link during each time period. Using the link costs in Table III and the link volumes in Table II and Equation (3.2.2), the network operators' costs (NOC) for network N_a are calculated to be 1068, 1535, and 4329 for the respective periods of 1970, 1975, and 1980. The proposed links to interconnect nodes 2-6 and 4-5 were assumed to have three possible states:

0 --- represents no road;

2 --- represents a two-lane road;

4 --- represents a four-lane road.

A number consisting of two digits was used to represent the state of the network. Each digit of this number will use the three digits above to represent the state of each link. This limits the maximum number of feasible states or networks at nine. In this problem, all nine networks were considered feasible. Maintenance costs were assumed to be the

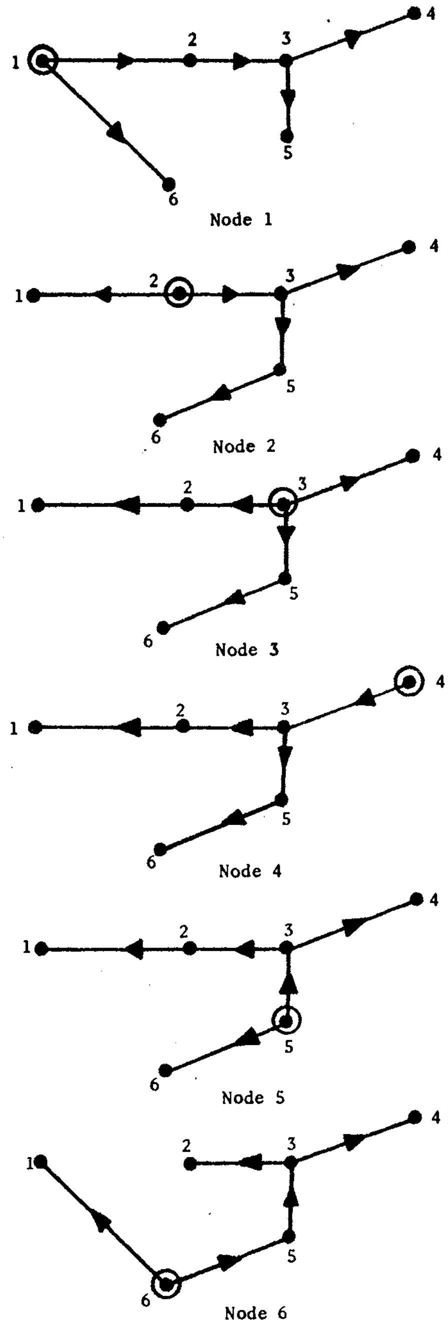

Figure 6. Minimum-Time Route Trees. (Approximate Scale)

TABLE II

LINK VOLUME TABLE FOR NETWORK N_a

Link (Node No.)	LKVOL (Total Trips/Period) 1970	1975	1980
1-2	60	72	106
1-6	32	38	62
2-3	88	106	206
3-4	78	94	236
3-5	112	136	350
5-6	78	96	250

TABLE III

LINK COST TABLE

Link	Two-Lane Costs			Four-Lane Costs		
(Node No.)	1970	1975	1980	1970	1975	1980
1-2	3.0	3.6	4.3			
1-6	3.6	4.3	5.0			
2-3	2.0	2.4	3.5			
2-6	2.5	3.0	3.2	2.0	2.4	2.5
3-4	2.8	3.4	4.1			
3-5	1.5	1.7	2.5			
4-5	3.6	4.3	4.0	3.0	3.6	3.2
5-6	2.7	3.2	4.0			

required monies to maintain the existing network during each period. Maintenance costs and network operators' costs are shown in Table IV. A one digit code has been used to represent the decision for each proposed link addition. The number of digits in the decision will be the same as the number of proposed link additions (two in this problem). The decision code is listed below:

0 --- represents no construction;

2 --- construct a two-lane road;

3 --- convert a two-lane to a four-lane;

4 --- construct a four-lane road.

The construction cost for each decision is shown in Table V. No budget limitations were imposed upon the decisions at each stage for this sample problem.

TABLE IV

NETWORK OPERATORS' COSTS AND MAINTENANCE COSTS

Network Number	State 2-6	State 4-5	Network Operators' Costs 1970	1975	1980	Maintenance Costs 1970	1975	1980
1	0	0	1068	1535	4329	200	240	300
2	2	0	1001	1440	3989	210	253	310
3	4	0	992	1427	3954	215	260	318
4	0	2	1043	1499	3991	215	263	330
5	0	4	1021	1469	3887	220	270	335
6	2	2	976	1405	3651	225	270	340
7	4	2	967	1392	3616	230	280	345
8	2	4	954	1374	3547	225	275	330
9	4	4	945	1361	3512	235	280	350

TABLE V

DECISIONS AND CONSTRUCTION COSTS

Decision Number	Decision 2-6/4-5		Construction Costs
1	0	0	0.0
2	2	0	100.0
3	4	0	187.5
4	0	2	144.0
5	0	4	300.0
6	2	2	244.0
7	4	2	331.5
8	2	4	400.0
9	4	4	487.5
10	3	0	125.0
11	3	2	275.0
12	3	3	350.0
13	3	4	440.0
14	0	3	225.0
15	2	3	320.0
16	4	3	420.0

The dynamic programming technique uses only the last three stages shown in Figure 2. At each stage in the process of DP, there are nine possible input network states (except stage three which has only the existing network as its input) and 16 possible decisions for each stage that must be considered. The problem is solved by starting at the free end, stage 1 (1980), and working toward the fixed end, stage 3 (1970). This type of system is classified as a serial system with one fixed end point and one free end point. These types of problems can be solved by starting at either the free or fixed end, but normally they are easier to solve by starting at the free end point. The dependency of a decision made in one time period affecting the decision in another time period is properly accounted for in the DP technique.

The use of the DP computer program which used Equations (3.3.6), (3.3.7), and (3.3.9) and the costs in Tables IV and V yielded the optimal decisions of $D_3^* = 2\ 0$, $D_2^* = 0\ 0$, and $D_1^* = 0\ 2$. The corresponding optimal cost values are 4596, 4621, and 4105. This implies we should build a two-lane road to interconnect nodes 2 and 6 in 1970. We do no constructing in 1975 and in 1980 another two-lane road to interconnect nodes 4 and 5 should be constructed.

5.3 A Complex 53-Node Problem. To better illustrate the power of this planning method, a 53-node problem shown in Figure 7 has been solved. The nodes represent the largest 53 cities based on personal income within the State of Oklahoma. A spiderweb network shown in Figure 7 was constructed to represent major road connections between these cities. The dashed linkages which are numbered from 1 through 8 represent the eight possible road connections that may be added to the existing network (solid lines). The basic problem is the determination of the optimal construction scheme during the next 20 years that minimizes the sum of operators' costs, construction costs, and maintenance costs.

The problem was formulated exactly as described in Chapter III. Four time periods each consisting of five years were considered. This required four trip tables to be constructed to represent the solution of the traffic generation and the traffic distribution during each period. An initial set of personal incomes by county for the year 1967 were obtained from the U.S. Department of Commerce, Office of Business

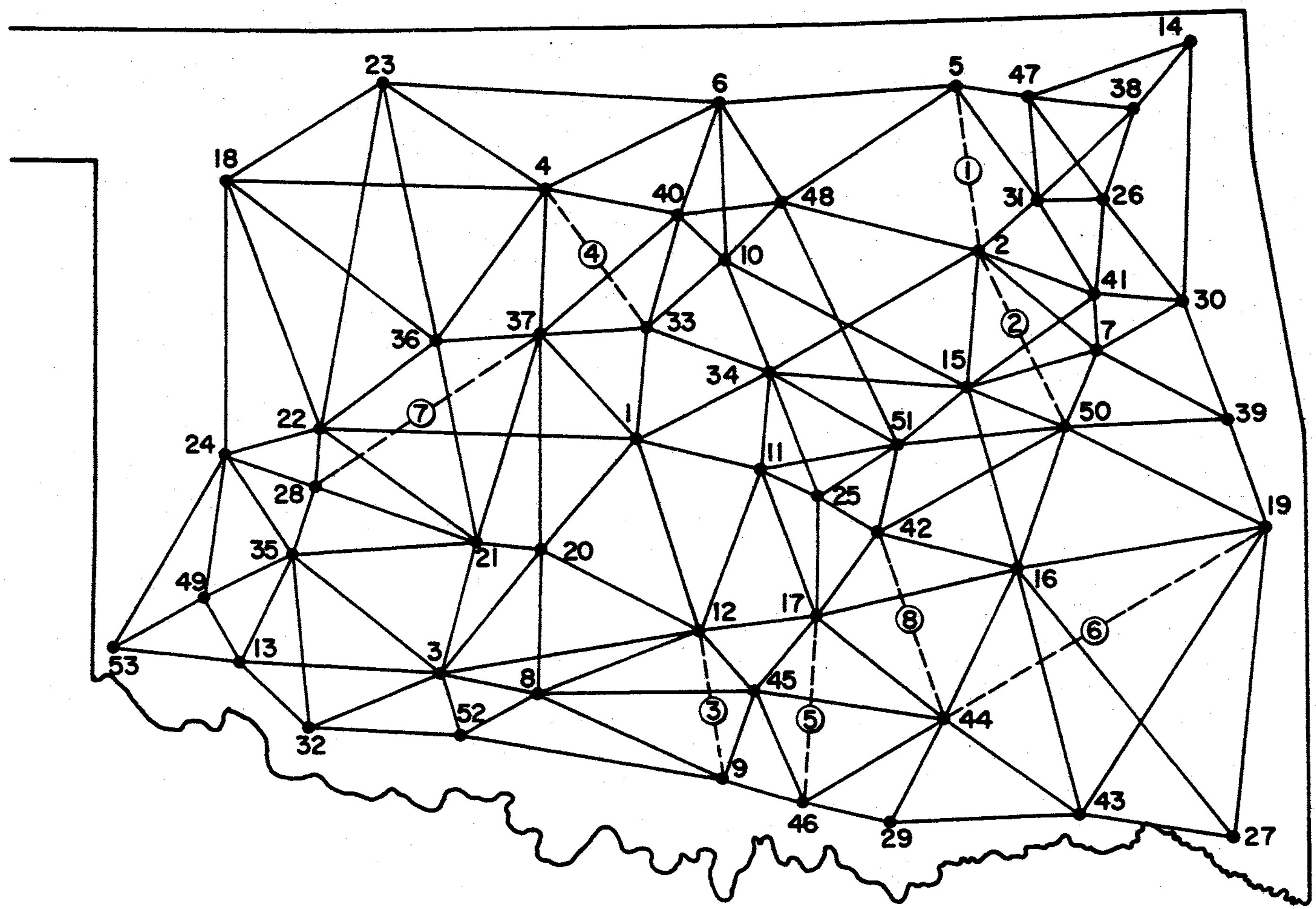

Figure 7. 53-Node Network

Economics, in Washington, D. C. In each of the future periods, different personal incomes were assumed to represent the growth or decay of each of the particular regions about the 53 nodes. These personal incomes for each node are given in Table VI along with the node number, the x-y coordinate location in degrees and minutes of latitude and longitude, and the name of the city. The Intuitive Gravity Model in Equation (3.2.1) was used with a calibration constant of $\beta = 440$ (16). The BLDTT program generated four trip tables as shown in Tables VII, VIII, IX, and X to represent the respective periods of 1970, 1975, 1980, and 1985.

This completed the solution of the traffic generation and traffic distribution problems depicted in Figure 1. The BLDTT program for constructing these trip tables was executed only one time for each five year planning period. The output trip tables were stored on disk data sets for repeated use by the LDSPWB program.

Each of the link additions were represented by a one digit code as shown on page 45. This dictated an eight digit number to represent the state of the network. Links connecting nodes 2-5, 2-50, and 9-12 were assumed to have two-lane roads in existence prior to any planning periods. Therefore, the only decisions that would be feasible for these particular links would be the conversion from a two-lane to a four-lane road. Each decision consisted of an eight digit number being formed from the code as shown on page 48.

The state and decision numbers were coded to represent the links of the network as shown in Table XI. Since links 2-5, 2-50, and 9-12 were assumed to be existing two-lane roads, the initial state to be used as input for stage four in the DP program would be 22200000. If a

TABLE VI

NODE DESCRIPTION AND PERSONAL INCOME FOR GRAVITY MODEL INPUT

			1970	1975	1980	1985	
NODE	LATITUDE	LONGITUDE	PERSONAL INCOME M$	PERSONAL INCOME M$	PERSONAL INCOME M$	PERSONAL INCOME M$	
1	35. 28.	97. 33.	1874.9	2011.3	2416.3	2621.3	OK OKLAC
2	36. 7.	95. 58.	1569.7	1832.1	2365.5	2534.6	OK TULSA
3	34. 36.	98. 25.	343.0	421.2	660.3	725.6	OK LAWTO
4	36. 24.	97. 54.	198.7	210.4	291.5	352.7	OK ENID
5	36. 44.	95. 59.	167.8	185.4	210.3	274.3	OK BARTL
6	36. 41.	97. 4.	167.4	189.8	220.6	254.6	OK PONCC
7	35. 45.	95. 21.	147.3	173.5	192.4	231.0	OK MUSKO
8	34. 30.	97. 57.	110.2	154.2	161.8	194.5	OK DUNCA
9	34. 11.	97. 8.	124.1	164.1	172.4	196.7	OK ARDMO
10	36. 7.	97. 3.	102.0	121.1	131.7	161.3	OK STILL
11	35. 20.	96. 55.	86.4	101.4	115.8	145.3	OK SHAWN
12	34. 44.	97. 14.	83.0	96.5	112.3	132.0	OK PAULV
13	34. 39.	99. 21.	79.9	82.4	94.6	99.8	OK ALTUS
14	36. 53.	94. 54.	80.2	91.8	99.7	120.6	OK MIAMI
15	35. 38.	95. 59.	75.1	58.6	112.8	136.9	OK OKMUL
16	34. 56.	95. 46.	75.5	76.0	81.6	97.4	OK MCALE
17	34. 47.	96. 41.	70.7	72.8	77.9	81.3	OK ADA
18	36. 26.	99. 25.	85.0	99.8	112.4	132.8	OK WOODW
19	35. 3.	94. 36.	79.4	85.9	95.6	105.3	OK POTEA
20	35. 3.	97. 57.	62.6	75.2	83.4	96.8	OK CHICK
21	35. 4.	98. 16.	65.5	72.6	84.6	97.4	OK ANADA
22	35. 32.	98. 59.	62.6	86.7	91.5	107.6	OK CLINT
23	36. 48.	98. 43.	64.5	69.8	73.4	82.0	OK ALVA
24	35. 25.	99. 26.	51.9	66.2	72.4	81.5	OK ELKC
25	35. 15.	96. 40.	50.7	65.8	66.8	72.8	OK SEMIN
26	36. 19.	95. 19.	50.1	59.2	62.3	68.5	OK PRYOR
27	33. 54.	94. 50.	70.1	85.9	89.6	97.8	OK IDABE
28	35. 17.	98. 59.	43.6	55.8	62.2	70.5	OK CORDE
29	33. 59.	96. 24.	42.6	52.1	57.5	63.8	OK DURAN
30	35. 57.	94. 58.	44.4	62.8	73.6	89.7	OK TAHLE
31	36. 20.	95. 37.	37.1	51.1	56.8	65.7	OK CLARE
32	34. 24.	99. 3.	36.1	41.5	47.6	54.1	OK FREDE
33	35. 53.	97. 26.	35.9	45.8	51.7	63.9	OK GUTHR
34	35. 43.	96. 54.	33.3	40.5	49.7	54.2	OK CHAND
35	35. 3.	99. 4.	31.3	34.9	41.6	43.8	OK HOBAR
36	35. 52.	98. 26.	31.1	37.8	41.3	47.6	OK WATON
37	35. 53.	97. 56.	29.4	33.8	39.5	39.5	OK KINGF
38	36. 39.	95. 11.	28.6	31.2	34.3	41.8	OK VINIT
39	35. 27.	94. 49.	43.0	50.5	55.6	63.7	OK SALLI
40	36. 17.	97. 18.	27.8	33.9	36.2	44.8	OK PERRY
41	35. 58.	95. 23.	25.8	29.7	33.6	46.8	OK WAGON
42	35. 5.	96. 25.	24.1	28.7	30.9	37.4	OK HOLDE
43	34. 1.	95. 31.	21.4	19.6	21.9	23.2	OK HUGO
44	34. 22.	96. 8.	21.4	21.4	26.7	31.6	OK ATOKA
45	34. 31.	96. 58.	20.1	25.7	27.6	35.8	OK SULPH
46	34. 6.	96. 46.	20.6	18.7	21.7	22.4	OK MADIL
47	36. 42.	95. 36.	18.8	20.6	24.2	29.4	OK NOWAT
48	36. 21.	96. 46.	19.2	21.6	26.8	29.6	OK PAWNE
49	34. 54.	99. 31.	36.8	40.6	42.6	51.8	OK MANGU
50	35. 29.	95. 32.	17.1	21.4	26.8	31.6	OK CHECO
51	35. 25.	96. 20.	16.0	17.6	21.0	23.5	OK OKEMA
52	34. 22.	98. 18.	15.6	16.7	18.4	20.3	OK WALTE
53	34. 42.	99. 56.	13.6	15.8	17.5	18.6	OK HOLLI

TABLE VII

1970 TRIP TABLE

Destinations (node numbers)

Origins (node numbers)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0.	1808.	793.	675.	105.	268.	90.	330.	179.	677.	1564.	592.	59.	19.	118.	71.	234.	53.	21.	1157.
2	1808.	0.	71.	147.	1706.	387.	1476.	29.	39.	392.	176.	50.	9.	144.	1498.	122.	66.	13.	63.	27.
3	793.	71.	0.	21.	5.	9.	5.	825.	50.	10.	18.	50.	96.	1.	5.	5.	15.	7.	2.	156.
4	675.	147.	21.	0.	16.	136.	6.	6.	4.	79.	13.	6.	3.	2.	6.	3.	4.	16.	1.	9.
5	105.	1706.	5.	16.	0.	70.	31.	2.	2.	25.	7.	3.	1.	32.	16.	4.	3.	1.	3.	2.
6	268.	387.	9.	136.	70.	0.	10.	3.	3.	140.	10.	4.	1.	5.	9.	3.	3.	4.	2.	4.
7	90.	1476.	5.	6.	31.	1C.	C.	2.	4.	10.	10.	4.	1.	12.	110.	26.	6.	1.	24.	2.
8	330.	29.	825.	6.	2.	3.	2.	0.	52.	4.	10.	52.	10.	0.	2.	3.	11.	2.	1.	61.
9	179.	39.	50.	4.	2.	3.	4.	52.	0.	3.	12.	75.	3.	1.	3.	7.	39.	1.	2.	10.
10	677.	392.	10.	79.	25.	140.	1C.	4.	3.	0.	29.	6.	1.	2.	13.	4.	5.	2.	1.	5.
11	1564.	176.	18.	13.	7.	10.	10.	10.	12.	29.	0.	45.	1.	1.	19.	10.	46.	1.	2.	13.
12	592.	50.	50.	6.	3.	4.	4.	52.	75.	6.	45.	0.	2.	1.	5.	6.	90.	1.	1.	29.
13	59.	9.	96.	3.	1.	1.	1.	10.	3.	1.	1.	2.	0.	0.	1.	1.	1.	2.	0.	5.
14	19.	144.	1.	2.	32.	5.	12.	0.	1.	2.	1.	1.	0.	0.	3.	1.	1.	0.	2.	0.
15	118.	1498.	5.	6.	16.	9.	11C.	2.	3.	13.	19.	5.	1.	3.	0.	24.	8.	1.	5.	2.
16	71.	122.	5.	3.	4.	3.	26.	3.	7.	4.	10.	6.	1.	1.	24.	0.	19.	0.	11.	2.
17	234.	66.	15.	4.	3.	3.	6.	11.	39.	5.	46.	90.	1.	1.	8.	19.	0.	1.	2.	6.
18	53.	13.	7.	16.	1.	4.	1.	2.	1.	2.	1.	1.	2.	0.	1.	0.	1.	0.	0.	2.
19	21.	63.	2.	1.	3.	2.	24.	1.	2.	1.	2.	1.	0.	2.	5.	11.	2.	0.	0.	1.
20	1157.	27.	156.	9.	2.	4.	2.	61.	10.	5.	13.	29.	5.	0.	2.	2.	6.	2.	1.	0.
21	540.	21.	286.	9.	1.	3.	1.	45.	7.	4.	7.	12.	9.	0.	1.	1.	4.	2.	0.	291.
22	129.	13.	32.	12.	1.	3.	1.	5.	2.	3.	2.	2.	10.	0.	1.	1.	1.	10.	0.	7.
23	52.	17.	4.	45.	2.	8.	1.	1.	1.	4.	1.	1.	1.	0.	1.	0.	1.	28.	0.	1.
24	50.	7.	20.	5.	1.	1.	C.	3.	1.	1.	1.	1.	14.	0.	0.	0.	1.	7.	0.	3.
25	354.	122.	8.	5.	4.	5.	8.	5.	8.	11.	486.	22.	1.	1.	19.	12.	50.	1.	2.	4.
26	23.	662.	1.	2.	42.	5.	6C.	1.	1.	3.	2.	1.	0.	22.	9.	2.	1.	0.	3.	0.
27	14.	16.	2.	1.	1.	1.	3.	1.	2.	1.	1.	1.	0.	0.	1.	4.	2.	0.	6.	0.
28	87.	8.	43.	6.	1.	2.	1.	5.	2.	2.	1.	2.	16.	0.	0.	0.	1.	4.	0.	7.
29	28.	13.	5.	1.	1.	1.	2.	3.	31.	1.	2.	5.	0.	0.	1.	4.	8.	0.	1.	1.
30	17.	201.	1.	1.	9.	2.	175.	0.	1.	2.	1.	1.	0.	7.	8.	3.	1.	0.	7.	0.
31	23.	1732.	1.	2.	73.	6.	34.	0.	1.	4.	2.	1.	0.	9.	10.	2.	1.	0.	2.	0.
32	30.	4.	105.	1.	0.	1.	C.	9.	2.	1.	1.	2.	88.	0.	0.	0.	1.	1.	0.	3.
33	1202.	56.	7.	42.	4.	16.	2.	2.	2.	86.	15.	4.	1.	0.	3.	1.	2.	1.	0.	5.
34	468.	136.	5.	10.	5.	10.	4.	2.	2.	65.	69.	5.	0.	1.	9.	2.	5.	1.	1.	3.
35	47.	5.	49.	3.	0.	1.	0.	5.	1.	1.	1.	1.	36.	0.	0.	0.	1.	2.	0.	4.
36	17C.	12.	9.	30.	1.	4.	1.	2.	1.	4.	2.	1.	2.	0.	1.	0.	1.	5.	0.	4.
37	571.	21.	8.	61.	2.	7.	1.	2.	1.	11.	4.	2.	1.	0.	1.	1.	1.	2.	0.	5.
38	10.	129.	1.	1.	27.	3.	5.	0.	0.	1.	1.	0.	0.	87.	2.	1.	0.	0.	1.	0.
39	15.	78.	1.	1.	3.	1.	63.	0.	1.	1.	1.	1.	0.	2.	6.	7.	1.	0.	57.	0.
40	142.	58.	3.	65.	5.	76.	2.	1.	1.	196.	4.	1.	0.	1.	2.	1.	1.	1.	0.	1.
41	15.	483.	1.	1.	10.	2.	431.	0.	1.	2.	1.	1.	0.	4.	14.	3.	1.	0.	3.	0.
42	76.	50.	3.	2.	2.	2.	5.	2.	4.	3.	27.	8.	0.	0.	10.	16.	45.	0.	1.	1.
43	7.	7.	1.	0.	0.	0.	1.	1.	2.	0.	1.	1.	0.	0.	1.	3.	1.	0.	1.	0.
44	18.	12.	2.	1.	1.	1.	2.	1.	7.	1.	2.	3.	0.	0.	1.	9.	10.	0.	1.	1.
45	54.	10.	7.	1.	1.	1.	1.	7.	71.	1.	5.	59.	0.	0.	1.	2.	43.	0.	0.	2.
46	21.	7.	4.	1.	0.	0.	1.	3.	106.	0.	2.	6.	0.	0.	1.	2.	7.	0.	0.	1.
47	9.	162.	0.	1.	138.	3.	5.	0.	0.	2.	1.	0.	0.	11.	2.	0.	0.	0.	0.	0.
48	48.	145.	1.	8.	13.	64.	2.	0.	0.	73.	3.	1.	0.	1.	3.	1.	1.	0.	0.	1.
49	27.	4.	24.	2.	0.	1.	C.	3.	1.	1.	1.	1.	160.	0.	0.	0.	0.	2.	0.	2.
50	14.	111.	1.	1.	2.	1.	113.	0.	1.	1.	2.	1.	0.	1.	29.	10.	1.	0.	3.	0.
51	52.	91.	2.	1.	2.	2.	6.	1.	1.	4.	18.	2.	0.	0.	35.	6.	5.	0.	1.	1.
52	26.	3.	405.	1.	0.	0.	C.	70.	3.	0.	1.	2.	3.	0.	0.	0.	1.	0.	0.	4.
53	6.	1.	4.	0.	0.	0.	0.	1.	0.	0.	0.	0.	14.	0.	0.	0.	0.	0.	0.	0.

TABLE VII (Continued)

Destinations (node numbers)

Origins (node numbers)	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1	540.	129.	52.	50.	354.	23.	14.	87.	28.	17.	23.	30.	1202.	468.	47.	170.	571.	10.	15.	142.
2	21.	13.	17.	7.	122.	662.	16.	8.	13.	201.	1732.	4.	56.	136.	5.	12.	21.	129.	78.	58.
3	216.	32.	4.	20.	8.	1.	2.	43.	5.	1.	1.	105.	7.	5.	49.	9.	8.	1.	1.	3.
4	9.	12.	45.	5.	5.	2.	1.	6.	1.	1.	2.	1.	42.	10.	3.	30.	61.	1.	1.	65.
5	1.	1.	2.	1.	4.	42.	1.	1.	1.	9.	73.	0.	4.	5.	0.	1.	2.	27.	3.	5.
6	3.	3.	8.	1.	5.	5.	1.	2.	1.	2.	6.	1.	16.	10.	1.	4.	7.	3.	1.	76.
7	1.	1.	1.	0.	8.	60.	3.	1.	2.	175.	34.	0.	2.	4.	0.	1.	1.	9.	63.	2.
8	45.	5.	1.	3.	5.	1.	1.	5.	3.	0.	0.	9.	2.	2.	5.	2.	2.	0.	0.	1.
9	7.	2.	1.	1.	8.	1.	2.	2.	31.	1.	1.	2.	2.	2.	1.	1.	1.	0.	1.	1.
10	4.	3.	4.	1.	11.	3.	1.	2.	1.	2.	4.	1.	86.	65.	1.	4.	11.	1.	1.	196.
11	7.	2.	1.	1.	486.	2.	1.	1.	2.	1.	2.	1.	15.	69.	1.	2.	4.	1.	1.	4.
12	12.	2.	1.	1.	22.	1.	1.	2.	5.	1.	1.	2.	4.	5.	1.	1.	2.	0.	1.	1.
13	9.	10.	1.	14.	1.	0.	0.	16.	0.	0.	0.	88.	1.	0.	36.	2.	1.	0.	0.	0.
14	0.	0.	0.	0.	1.	22.	0.	0.	0.	7.	9.	0.	0.	1.	0.	0.	0.	87.	2.	1.
15	1.	1.	1.	0.	19.	5.	1.	0.	1.	8.	10.	0.	3.	9.	0.	1.	1.	2.	6.	2.
16	1.	1.	0.	0.	12.	2.	4.	0.	4.	3.	2.	0.	1.	2.	0.	0.	1.	1.	7.	1.
17	4.	1.	1.	1.	50.	1.	2.	1.	8.	1.	1.	1.	2.	5.	1.	1.	1.	0.	1.	1.
18	2.	10.	28.	7.	1.	0.	0.	4.	0.	0.	0.	1.	1.	1.	2.	5.	2.	0.	0.	1.
19	0.	0.	0.	0.	2.	3.	6.	0.	1.	7.	2.	0.	0.	1.	0.	0.	0.	1.	57.	0.
20	291.	7.	1.	3.	4.	0.	0.	7.	1.	0.	0.	3.	5.	3.	4.	4.	5.	0.	0.	1.
21	0.	16.	1.	6.	3.	0.	0.	18.	1.	0.	0.	5.	3.	2.	11.	6.	5.	0.	0.	1.
22	16.	0.	3.	78.	1.	0.	0.	214.	0.	0.	0.	3.	2.	1.	24.	17.	4.	0.	0.	1.
23	1.	3.	0.	2.	1.	0.	0.	1.	0.	0.	0.	0.	2.	1.	1.	4.	2.	0.	0.	2.
24	6.	78.	2.	0.	0.	0.	0.	52.	0.	0.	0.	3.	1.	0.	22.	3.	1.	0.	0.	0.
25	3.	1.	1.	0.	0.	1.	1.	1.	2.	1.	1.	0.	4.	19.	0.	1.	1.	0.	1.	2.
26	0.	0.	0.	0.	1.	0.	0.	0.	0.	32.	160.	0.	1.	1.	0.	0.	0.	44.	4.	1.
27	0.	0.	0.	0.	1.	0.	0.	0.	2.	1.	0.	0.	0.	0.	0.	0.	0.	0.	2.	0.
28	18.	214.	1.	52.	1.	0.	0.	0.	0.	0.	0.	4.	1.	1.	115.	5.	2.	0.	0.	0.
29	1.	0.	0.	0.	2.	0.	2.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
30	0.	0.	0.	0.	1.	32.	1.	0.	0.	0.	9.	0.	0.	1.	0.	0.	0.	5.	20.	0.
31	0.	0.	0.	0.	1.	160.	0.	0.	0.	9.	0.	0.	1.	1.	0.	0.	0.	14.	2.	1.
32	5.	3.	0.	3.	0.	0.	0.	4.	0.	0.	0.	0.	0.	0.	6.	1.	0.	0.	0.	0.
33	3.	2.	2.	1.	4.	1.	0.	1.	0.	0.	1.	0.	0.	17.	1.	3.	22.	0.	0.	19.
34	2.	1.	1.	0.	19.	1.	0.	1.	0.	1.	1.	0.	17.	0.	0.	1.	3.	0.	1.	5.
35	11.	24.	1.	22.	0.	0.	0.	115.	0.	0.	0.	6.	1.	0.	0.	2.	1.	0.	0.	0.
36	6.	17.	4.	3.	1.	0.	0.	5.	0.	0.	0.	1.	3.	1.	2.	0.	19.	0.	0.	1.
37	5.	4.	2.	1.	1.	0.	0.	2.	0.	0.	0.	0.	22.	3.	1.	19.	0.	0.	0.	5.
38	0.	0.	0.	0.	0.	44.	0.	0.	0.	5.	14.	0.	0.	0.	0.	0.	0.	0.	1.	0.
39	0.	0.	0.	0.	1.	4.	2.	0.	0.	20.	2.	0.	0.	1.	0.	0.	0.	1.	0.	0.
40	1.	1.	2.	0.	2.	1.	0.	0.	0.	0.	1.	0.	19.	5.	0.	1.	5.	0.	0.	0.
41	0.	0.	0.	0.	1.	39.	0.	0.	0.	39.	19.	0.	0.	1.	0.	0.	0.	3.	5.	0.
42	1.	0.	0.	0.	83.	1.	1.	0.	1.	1.	1.	0.	1.	3.	0.	0.	0.	0.	1.	0.
43	0.	0.	0.	0.	1.	0.	12.	0.	4.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
44	0.	0.	0.	0.	2.	0.	2.	0.	15.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
45	1.	0.	0.	0.	4.	0.	0.	0.	4.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.
46	1.	0.	0.	0.	1.	0.	1.	0.	33.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
47	0.	0.	0.	0.	0.	15.	0.	0.	0.	2.	19.	0.	0.	0.	0.	0.	0.	18.	1.	0.
48	0.	0.	1.	0.	1.	1.	0.	0.	0.	0.	1.	0.	3.	4.	0.	0.	1.	0.	0.	9.
49	4.	8.	1.	20.	0.	0.	0.	12.	0.	0.	0.	8.	0.	0.	25.	1.	0.	0.	0.	0.
50	0.	0.	0.	0.	2.	2.	0.	0.	0.	4.	2.	0.	0.	1.	0.	0.	0.	0.	6.	0.
51	0.	0.	0.	0.	33.	1.	0.	0.	0.	1.	1.	0.	1.	5.	0.	0.	0.	0.	1.	0.
52	5.	1.	0.	1.	0.	0.	0.	1.	0.	0.	0.	4.	0.	0.	1.	0.	0.	0.	0.	0.
53	1.	1.	0.	2.	0.	0.	0.	1.	0.	0.	0.	2.	0.	0.	1.	0.	0.	0.	0.	0.

TABLE VII (Continued)

Origins (node numbers) / Destinations (node numbers)

	41	42	43	44	45	46	47	48	49	50	51	52	53
1	15.	76.	7.	18.	54.	21.	9.	48.	27.	14.	52.	26.	6.
2	483.	50.	7.	12.	10.	7.	162.	145.	4.	111.	91.	3.	1.
3	1.	3.	1.	2.	7.	4.	0.	1.	24.	1.	2.	405.	4.
4	1.	2.	0.	1.	1.	1.	1.	8.	2.	1.	1.	1.	0.
5	10.	2.	0.	1.	1.	0.	138.	13.	0.	2.	2.	0.	0.
6	2.	2.	C.	1.	1.	0.	3.	64.	1.	1.	2.	0.	0.
7	431.	5.	1.	2.	1.	1.	5.	2.	0.	113.	6.	0.	0.
8	0.	2.	1.	1.	7.	3.	C.	0.	3.	0.	1.	70.	1.
9	1.	4.	2.	7.	71.	106.	C.	0.	1.	1.	1.	3.	0.
10	2.	3.	0.	1.	1.	0.	2.	73.	1.	1.	4.	0.	0.
11	1.	27.	1.	2.	5.	2.	1.	3.	1.	2.	18.	1.	0.
12	1.	8.	1.	3.	59.	6.	C.	1.	1.	1.	2.	2.	0.
13	0.	0.	0.	0.	0.	0.	C.	0.	160.	0.	0.	3.	14.
14	4.	0.	0.	0.	0.	0.	11.	1.	0.	1.	0.	0.	0.
15	14.	10.	1.	1.	1.	1.	2.	3.	0.	29.	35.	0.	0.
16	3.	16.	3.	9.	2.	2.	0.	1.	0.	10.	6.	0.	0.
17	1.	45.	1.	10.	43.	7.	C.	1.	0.	1.	5.	1.	0.
18	0.	0.	0.	0.	0.	0.	C.	0.	2.	0.	0.	0.	0.
19	3.	1.	1.	1.	0.	0.	0.	0.	0.	3.	1.	0.	0.
20	0.	1.	0.	1.	2.	1.	C.	1.	2.	0.	1.	4.	0.
21	0.	1.	0.	0.	1.	1.	0.	0.	4.	0.	0.	5.	1.
22	0.	0.	0.	0.	0.	0.	0.	0.	8.	0.	0.	1.	1.
23	0.	0.	0.	0.	0.	0.	C.	1.	1.	0.	0.	0.	0.
24	0.	0.	0.	0.	0.	0.	C.	0.	20.	0.	0.	1.	2.
25	1.	83.	1.	2.	4.	1.	C.	1.	0.	2.	33.	0.	0.
26	39.	1.	0.	0.	0.	0.	15.	1.	0.	2.	1.	0.	0.
27	0.	1.	12.	2.	0.	1.	0.	0.	0.	0.	0.	0.	0.
28	0.	0.	0.	0.	0.	0.	C.	0.	12.	0.	0.	1.	1.
29	0.	1.	4.	15.	4.	33.	C.	0.	0.	0.	0.	0.	0.
30	39.	1.	0.	0.	0.	0.	2.	0.	0.	4.	1.	0.	0.
31	19.	1.	0.	0.	0.	0.	19.	1.	0.	2.	1.	0.	0.
32	0.	0.	0.	0.	0.	0.	0.	0.	8.	0.	0.	4.	2.
33	C.	1.	0.	0.	0.	0.	0.	3.	0.	0.	1.	0.	0.
34	1.	3.	0.	0.	1.	0.	0.	4.	0.	1.	5.	0.	0.
35	0.	0.	0.	0.	0.	0.	C.	0.	25.	0.	0.	1.	1.
36	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.	0.	0.	0.
37	0.	0.	0.	0.	0.	0.	C.	1.	0.	0.	0.	0.	0.
38	3.	0.	0.	0.	0.	0.	18.	0.	0.	0.	0.	0.	0.
39	5.	1.	0.	0.	0.	0.	1.	0.	0.	6.	1.	0.	0.
40	0.	0.	0.	0.	0.	0.	C.	9.	0.	0.	0.	0.	0.
41	0.	1.	0.	0.	0.	0.	2.	1.	0.	5.	1.	0.	0.
42	1.	0.	0.	2.	2.	1.	0.	0.	0.	1.	13.	0.	0.
43	0.	0.	0.	3.	0.	1.	C.	0.	0.	0.	0.	0.	0.
44	0.	2.	3.	0.	2.	3.	C.	0.	0.	0.	0.	0.	0.
45	0.	2.	0.	2.	0.	6.	0.	0.	0.	0.	0.	0.	0.
46	0.	1.	1.	3.	6.	0.	0.	0.	0.	0.	0.	0.	0.
47	2.	0.	0.	0.	0.	0.	C.	1.	0.	0.	0.	0.	0.
48	1.	0.	0.	0.	0.	0.	1.	0.	0.	0.	1.	0.	0.
49	0.	0.	0.	0.	0.	0.	C.	0.	0.	0.	0.	1.	11.
50	5.	1.	0.	0.	0.	0.	C.	0.	0.	0.	1.	0.	0.
51	1.	13.	0.	0.	0.	0.	C.	1.	0.	1.	0.	0.	0.
52	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.	0.	0.	0.
53	0.	0.	0.	0.	0.	C.	C.	0.	11.	0.	0.	0.	0.

TABLE VIII

1975 TRIP TABLE

Destinations (node numbers)

Origins (node numbers)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0.	2263.	1044.	766.	125.	326.	113.	495.	254.	863.	1969.	739.	66.	23.	167.	76.	258.	67.	24.	1491.
2	2263.	0.	101.	181.	2199.	512.	2029.	48.	60.	543.	241.	68.	11.	193.	2296.	144.	79.	18.	80.	37.
3	1044.	101.	0.	27.	7.	13.	7.	1417.	82.	15.	27.	71.	121.	2.	7.	6.	19.	10.	2.	230.
4	766.	181.	27.	0.	19.	164.	7.	9.	6.	99.	16.	8.	4.	3.	8.	3.	4.	20.	1.	12.
5	125.	2199.	7.	19.	0.	88.	41.	3.	3.	33.	9.	3.	1.	41.	24.	5.	3.	2.	4.	2.
6	326.	512.	13.	164.	88.	0.	13.	5.	4.	188.	14.	5.	2.	6.	13.	3.	4.	5.	2.	5.
7	113.	2029.	7.	7.	41.	13.	C.	4.	6.	13.	13.	5.	1.	17.	170.	31.	8.	1.	31.	2.
8	495.	48.	1417.	9.	3.	5.	4.	0.	97.	6.	16.	84.	14.	1.	4.	4.	15.	3.	1.	103.
9	254.	60.	82.	6.	3.	4.	6.	97.	0.	5.	18.	116.	4.	1.	6.	9.	53.	1.	2.	16.
10	863.	543.	15.	99.	53.	188.	13.	6.	5.	0.	40.	8.	2.	3.	21.	4.	6.	3.	2.	8.
11	1969.	241.	27.	16.	9.	14.	13.	16.	18.	40.	0.	61.	2.	2.	30.	12.	56.	2.	2.	18.
12	739.	68.	71.	8.	3.	5.	5.	84.	116.	8.	61.	0.	3.	1.	7.	7.	108.	1.	2.	40.
13	66.	11.	121.	4.	1.	2.	1.	14.	4.	2.	2.	3.	0.	0.	1.	1.	1.	3.	0.	6.
14	23.	193.	2.	3.	41.	6.	17.	1.	1.	3.	2.	1.	0.	0.	5.	2.	1.	0.	2.	1.
15	167.	2296.	7.	8.	24.	13.	17C.	4.	6.	21.	30.	7.	1.	5.	0.	31.	11.	1.	7.	3.
16	76.	144.	6.	3.	5.	3.	31.	4.	9.	4.	12.	7.	1.	2.	31.	0.	20.	0.	12.	2.
17	258.	79.	19.	4.	3.	4.	8.	15.	53.	6.	56.	108.	1.	1.	11.	20.	0.	1.	2.	8.
18	67.	18.	10.	20.	2.	5.	1.	3.	1.	3.	2.	1.	3.	0.	1.	0.	1.	0.	0.	2.
19	24.	80.	2.	1.	4.	2.	31.	1.	2.	2.	2.	2.	0.	2.	7.	12.	2.	0.	0.	1.
20	1491.	37.	230.	12.	2.	5.	2.	103.	16.	8.	18.	40.	6.	1.	3.	2.	8.	2.	1.	0.
21	642.	27.	390.	11.	2.	4.	2.	70.	10.	5.	9.	16.	11.	0.	2.	1.	4.	3.	0.	387.
22	191.	21.	55.	17.	2.	4.	1.	10.	3.	4.	3.	4.	15.	0.	1.	1.	2.	16.	0.	12.
23	61.	21.	5.	52.	2.	10.	1.	2.	1.	5.	2.	1.	1.	0.	1.	0.	1.	36.	0.	2.
24	68.	11.	31.	7.	1.	2.	1.	5.	2.	2.	2.	2.	19.	0.	1.	0.	1.	11.	0.	4.
25	493.	184.	13.	7.	6.	7.	12.	8.	13.	17.	740.	33.	1.	1.	33.	16.	67.	1.	2.	7.
26	25.	912.	2.	3.	54.	7.	E4.	1.	1.	5.	3.	1.	0.	29.	15.	3.	1.	0.	3.	1.
27	18.	23.	3.	1.	1.	1.	4.	1.	4.	1.	2.	2.	0.	1.	2.	5.	2.	0.	8.	1.
28	119.	12.	67.	7.	1.	2.	1.	10.	3.	2.	2.	3.	21.	0.	1.	0.	1.	6.	0.	10.
29	37.	19.	8.	1.	1.	1.	2.	6.	50.	1.	3.	8.	1.	0.	2.	5.	11.	0.	1.	2.
30	25.	331.	2.	2.	14.	4.	251.	1.	1.	3.	2.	1.	0.	12.	15.	5.	1.	0.	11.	1.
31	34.	2784.	2.	3.	112.	10.	56.	1.	1.	7.	3.	1.	0.	15.	18.	3.	1.	0.	2.	1.
32	36.	6.	148.	2.	0.	1.	0.	14.	3.	1.	1.	2.	105.	0.	0.	0.	1.	1.	0.	4.
33	1645.	84.	10.	56.	5.	23.	3.	4.	3.	130.	22.	5.	1.	1.	5.	1.	3.	2.	1.	7.
34	610.	194.	7.	12.	7.	14.	6.	3.	3.	93.	99.	7.	1.	1.	15.	3.	6.	1.	1.	4.
35	56.	7.	67.	3.	1.	1.	0.	8.	2.	1.	1.	2.	41.	0.	0.	0.	1.	2.	0.	6.
36	222.	17.	14.	38.	1.	5.	1.	4.	2.	5.	3.	2.	2.	0.	1.	0.	1.	7.	0.	6.
37	704.	28.	11.	75.	2.	9.	1.	4.	2.	15.	6.	3.	1.	0.	2.	1.	1.	3.	0.	7.
38	11.	164.	1.	1.	33.	3.	12.	0.	0.	2.	1.	0.	0.	109.	3.	1.	0.	0.	1.	0.
39	19.	107.	2.	1.	4.	2.	88.	1.	1.	2.	2.	1.	0.	3.	9.	8.	2.	0.	72.	0.
40	186.	83.	4.	84.	7.	105.	2.	2.	1.	284.	6.	2.	0.	1.	3.	1.	1.	1.	0.	2.
41	19.	649.	1.	1.	12.	3.	585.	1.	1.	3.	2.	1.	0.	5.	21.	3.	1.	0.	3.	0.
42	97.	69.	5.	2.	2.	2.	7.	3.	6.	4.	38.	11.	0.	1.	16.	19.	55.	0.	1.	2.
43	7.	7.	1.	0.	0.	0.	1.	1.	2.	0.	1.	1.	0.	0.	1.	3.	1.	0.	1.	0.
44	20.	14.	3.	1.	1.	1.	2.	2.	9.	1.	2.	4.	0.	0.	2.	9.	10.	0.	1.	1.
65	74.	16.	11.	1.	1.	1.	1.	12.	121.	1.	8.	88.	1.	0.	2.	3.	56.	0.	1.	4.
46	20.	7.	5.	1.	0.	0.	1.	4.	128.	1.	2.	6.	0.	0.	1.	2.	7.	0.	0.	1.
47	10.	208.	1.	1.	167.	4.	7.	0.	0.	2.	1.	0.	0.	14.	3.	1.	0.	0.	1.	0.
48	58.	190.	2.	10.	16.	82.	3.	1.	1.	98.	3.	1.	0.	1.	4.	1.	1.	0.	1.	0.
49	31.	5.	33.	2.	0.	1.	0.	5.	1.	1.	1.	1.	183.	0.	0.	0.	0.	2.	0.	3.
50	18.	162.	1.	1.	3.	2.	167.	1.	1.	2.	3.	1.	0.	1.	47.	12.	2.	0.	4.	0.
51	61.	117.	2.	2.	2.	2.	8.	1.	2.	5.	23.	3.	0.	0.	51.	7.	6.	0.	1.	1.
52	30.	4.	532.	1.	0.	0.	C.	104.	5.	0.	1.	3.	3.	0.	0.	0.	1.	0.	0.	5.
53	7.	1.	6.	1.	0.	0.	C.	1.	0.	0.	0.	0.	17.	0.	0.	0.	0.	1.	0.	0.

TABLE VIII (Continued)

Destinations (node numbers)

Origins (node numbers)	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1	642.	191.	61.	68.	493.	29.	18.	119.	37.	25.	34.	36.	1645.	610.	56.	222.	704.	11.	19.	186.
2	27.	21.	21.	11.	184.	912.	23.	12.	19.	331.	2784.	6.	84.	194.	7.	17.	28.	164.	107.	83.
3	390.	55.	5.	31.	13.	2.	3.	67.	8.	2.	2.	148.	10.	7.	67.	14.	11.	1.	2.	4.
4	11.	17.	52.	7.	7.	3.	1.	7.	1.	2.	3.	2.	56.	12.	3.	38.	75.	1.	1.	84.
5	2.	2.	2.	1.	6.	54.	1.	1.	1.	14.	112.	0.	5.	7.	1.	1.	2.	33.	4.	7.
6	4.	4.	10.	2.	7.	7.	1.	2.	1.	4.	10.	1.	23.	14.	1.	5.	9.	3.	2.	105.
7	2.	1.	1.	1.	12.	84.	4.	1.	2.	291.	56.	0.	3.	6.	0.	1.	1.	12.	88.	2.
8	70.	10.	2.	5.	8.	1.	1.	10.	6.	1.	1.	14.	4.	3.	8.	4.	4.	0.	1.	2.
9	10.	3.	1.	2.	13.	1.	4.	3.	50.	1.	1.	3.	3.	3.	2.	2.	2.	0.	1.	1.
10	5.	4.	5.	2.	17.	5.	1.	2.	1.	3.	7.	1.	130.	93.	1.	5.	15.	2.	2.	284.
11	9.	3.	2.	2.	740.	3.	2.	2.	3.	2.	3.	1.	22.	99.	1.	3.	6.	1.	2.	6.
12	16.	4.	1.	2.	33.	1.	2.	3.	8.	1.	1.	2.	5.	7.	2.	2.	3.	0.	1.	2.
13	11.	15.	1.	19.	1.	0.	0.	21.	1.	0.	0.	105.	1.	1.	41.	2.	1.	0.	0.	0.
14	0.	0.	0.	0.	1.	29.	1.	0.	0.	12.	15.	0.	1.	1.	0.	0.	0.	109.	3.	1.
15	2.	1.	1.	1.	33.	15.	2.	1.	2.	15.	18.	0.	5.	15.	0.	1.	2.	3.	9.	3.
16	1.	1.	0.	0.	16.	3.	5.	0.	5.	5.	3.	0.	1.	3.	0.	0.	1.	1.	8.	1.
17	4.	2.	1.	1.	67.	1.	2.	1.	11.	1.	1.	1.	3.	6.	1.	1.	1.	0.	2.	1.
18	3.	16.	36.	11.	1.	0.	0.	6.	0.	0.	0.	1.	2.	1.	2.	7.	3.	0.	0.	1.
19	0.	0.	0.	0.	2.	3.	8.	0.	1.	11.	2.	0.	1.	1.	0.	0.	0.	1.	72.	0.
20	387.	12.	2.	4.	7.	1.	1.	10.	2.	1.	1.	4.	7.	4.	6.	6.	7.	0.	0.	2.
21	0.	24.	2.	8.	4.	0.	0.	25.	1.	0.	0.	6.	5.	2.	14.	8.	6.	0.	0.	1.
22	24.	0.	5.	137.	2.	0.	0.	379.	1.	0.	0.	4.	3.	1.	37.	28.	6.	0.	0.	1.
23	2.	5.	0.	3.	1.	0.	0.	2.	0.	0.	0.	0.	2.	1.	1.	5.	3.	0.	0.	2.
24	8.	137.	3.	0.	1.	0.	0.	85.	0.	0.	0.	4.	1.	1.	31.	5.	2.	0.	0.	1.
25	4.	2.	1.	1.	0.	2.	1.	1.	3.	2.	2.	1.	7.	30.	1.	1.	2.	1.	2.	3.
26	0.	0.	0.	0.	2.	0.	1.	0.	0.	53.	261.	0.	1.	2.	0.	0.	0.	57.	6.	1.
27	0.	0.	0.	0.	1.	1.	0.	0.	4.	1.	1.	0.	0.	0.	0.	0.	0.	0.	2.	0.
28	25.	379.	2.	85.	1.	0.	0.	0.	0.	0.	0.	5.	2.	1.	164.	8.	3.	0.	0.	1.
29	1.	1.	0.	0.	3.	0.	4.	0.	0.	1.	0.	0.	1.	1.	0.	0.	0.	0.	1.	0.
30	0.	0.	0.	0.	2.	53.	1.	0.	1.	0.	18.	0.	1.	1.	0.	0.	0.	8.	34.	1.
31	0.	0.	0.	0.	2.	261.	1.	0.	0.	18.	0.	0.	1.	2.	0.	0.	0.	22.	3.	1.
32	6.	4.	0.	4.	1.	0.	0.	5.	0.	0.	0.	0.	0.	0.	8.	1.	1.	0.	0.	0.
33	5.	3.	2.	1.	7.	1.	0.	2.	1.	1.	1.	0.	0.	26.	1.	5.	32.	0.	0.	30.
34	2.	1.	1.	1.	30.	2.	0.	1.	1.	1.	2.	0.	26.	0.	0.	1.	4.	1.	1.	8.
35	14.	37.	1.	31.	1.	0.	0.	164.	0.	0.	0.	8.	1.	0.	0.	2.	1.	0.	0.	0.
36	8.	28.	5.	5.	1.	0.	0.	8.	0.	0.	0.	1.	5.	1.	2.	0.	26.	0.	0.	2.
37	6.	6.	3.	2.	2.	0.	0.	3.	0.	0.	0.	1.	32.	4.	1.	26.	0.	0.	0.	6.
38	0.	0.	0.	0.	1.	57.	0.	0.	0.	8.	22.	0.	0.	1.	0.	0.	0.	0.	1.	0.
39	0.	0.	0.	0.	2.	6.	2.	0.	1.	34.	3.	0.	0.	1.	0.	0.	0.	1.	0.	0.
40	1.	1.	2.	1.	3.	1.	0.	1.	0.	1.	1.	0.	30.	8.	0.	2.	6.	0.	0.	0.
41	0.	0.	0.	0.	2.	53.	1.	0.	0.	64.	30.	0.	1.	1.	0.	0.	0.	4.	7.	0.
42	1.	1.	0.	0.	129.	1.	1.	0.	2.	1.	1.	0.	2.	4.	0.	0.	1.	0.	1.	1.
43	0.	0.	0.	0.	1.	0.	13.	0.	4.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.
44	0.	0.	0.	0.	2.	0.	2.	0.	18.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.
45	2.	1.	0.	0.	6.	0.	1.	0.	6.	0.	0.	0.	1.	1.	0.	0.	0.	0.	0.	0.
46	1.	0.	0.	0.	1.	0.	1.	0.	37.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
47	0.	0.	0.	0.	1.	19.	0.	0.	0.	3.	29.	0.	0.	1.	0.	0.	0.	22.	1.	0.
48	1.	0.	1.	0.	2.	1.	0.	0.	0.	1.	2.	0.	4.	5.	0.	1.	1.	1.	0.	12.
49	4.	12.	1.	28.	0.	0.	0.	17.	0.	0.	0.	10.	0.	0.	31.	1.	1.	0.	0.	0.
50	0.	0.	0.	0.	3.	3.	1.	0.	0.	7.	3.	0.	0.	1.	0.	0.	0.	1.	8.	0.
51	1.	0.	0.	0.	46.	1.	0.	0.	1.	1.	1.	0.	1.	6.	0.	0.	0.	0.	1.	1.
52	6.	1.	0.	1.	0.	0.	0.	1.	0.	0.	0.	4.	0.	0.	1.	0.	0.	0.	0.	0.
53	1.	2.	0.	3.	0.	0.	0.	2.	0.	0.	0.	2.	0.	0.	2.	0.	0.	0.	0.	0.

TABLE VIII (Continued)

Origins (node numbers) / Destinations (node numbers)

	41	42	43	44	45	46	47	48	49	50	51	52	53
1	19.	97.	7.	20.	74.	20.	10.	58.	31.	18.	61.	30.	7.
2	649.	69.	7.	14.	16.	7.	208.	190.	5.	162.	117.	4.	1.
3	1.	5.	1.	3.	11.	5.	1.	2.	33.	1.	2.	532.	6.
4	1.	2.	0.	1.	1.	1.	1.	10.	2.	1.	2.	1.	1.
5	12.	2.	0.	1.	1.	0.	167.	16.	0.	3.	2.	0.	0.
6	3.	2.	0.	1.	1.	0.	4.	82.	1.	2.	2.	0.	0.
7	585.	7.	1.	2.	1.	1.	7.	3.	0.	167.	8.	0.	0.
8	1.	3.	1.	2.	12.	4.	0.	1.	5.	1.	1.	104.	1.
9	1.	6.	2.	9.	121.	128.	0.	1.	1.	1.	2.	5.	0.
10	3.	4.	0.	1.	1.	1.	2.	58.	1.	2.	5.	0.	0.
11	2.	38.	1.	2.	8.	2.	1.	3.	1.	3.	23.	1.	0.
12	1.	11.	1.	4.	88.	6.	0.	1.	1.	1.	3.	3.	0.
13	0.	0.	0.	0.	1.	0.	0.	0.	183.	0.	0.	3.	17.
14	5.	1.	0.	0.	0.	0.	14.	1.	0.	1.	0.	0.	0.
15	21.	16.	1.	2.	2.	1.	3.	4.	0.	47.	51.	0.	0.
16	3.	19.	3.	9.	3.	2.	1.	1.	0.	12.	7.	0.	0.
17	1.	55.	1.	10.	56.	7.	0.	1.	0.	2.	6.	1.	0.
18	0.	0.	0.	0.	0.	0.	0.	0.	2.	0.	0.	0.	1.
19	3.	1.	1.	1.	1.	0.	1.	0.	0.	4.	1.	0.	0.
20	0.	2.	0.	1.	4.	1.	0.	1.	3.	0.	1.	5.	0.
21	0.	1.	0.	0.	2.	1.	0.	1.	4.	0.	1.	6.	1.
22	0.	1.	0.	0.	1.	0.	0.	0.	12.	0.	0.	1.	2.
23	0.	0.	0.	0.	0.	0.	0.	1.	1.	0.	0.	0.	0.
24	0.	0.	0.	0.	0.	0.	0.	0.	28.	0.	0.	1.	3.
25	2.	129.	1.	2.	6.	1.	1.	2.	0.	3.	46.	0.	0.
26	53.	1.	0.	0.	0.	0.	19.	1.	0.	3.	1.	0.	0.
27	1.	1.	13.	2.	1.	1.	0.	0.	0.	1.	0.	0.	0.
28	0.	0.	0.	0.	0.	0.	0.	0.	17.	0.	0.	1.	2.
29	0.	2.	4.	18.	6.	37.	0.	0.	0.	0.	1.	0.	0.
30	64.	1.	0.	0.	0.	0.	3.	1.	0.	7.	1.	0.	0.
31	30.	1.	0.	0.	0.	0.	29.	2.	0.	3.	1.	0.	0.
32	0.	0.	0.	0.	0.	0.	0.	0.	10.	0.	0.	4.	2.
33	1.	2.	0.	0.	1.	0.	0.	4.	0.	0.	1.	0.	0.
34	1.	4.	0.	0.	1.	0.	1.	5.	0.	1.	6.	0.	0.
35	0.	0.	0.	0.	0.	0.	0.	0.	31.	0.	0.	1.	2.
36	0.	0.	0.	0.	0.	0.	0.	1.	1.	0.	0.	0.	0.
37	0.	1.	0.	0.	0.	0.	0.	1.	1.	0.	0.	0.	0.
38	4.	0.	0.	0.	0.	0.	22.	1.	0.	1.	0.	0.	0.
39	7.	1.	1.	1.	0.	0.	1.	0.	0.	8.	1.	0.	0.
40	0.	1.	0.	0.	0.	0.	0.	12.	0.	0.	1.	0.	0.
41	0.	1.	0.	0.	0.	0.	2.	1.	0.	7.	1.	0.	0.
42	1.	0.	0.	2.	3.	1.	0.	1.	0.	2.	17.	0.	0.
43	0.	0.	0.	3.	0.	1.	0.	0.	0.	0.	0.	0.	0.
44	0.	2.	3.	0.	2.	3.	0.	0.	0.	0.	1.	0.	0.
45	0.	3.	0.	2.	0.	8.	0.	0.	0.	0.	1.	1.	0.
46	0.	1.	1.	3.	8.	0.	0.	0.	0.	0.	0.	0.	0.
47	2.	0.	0.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.
48	1.	1.	0.	0.	0.	0.	1.	0.	0.	0.	1.	0.	0.
49	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	14.
50	7.	2.	0.	0.	0.	0.	0.	0.	0.	0.	2.	0.	0.
51	1.	17.	0.	1.	1.	0.	0.	1.	0.	2.	0.	0.	0.
52	0.	0.	0.	0.	1.	0.	0.	0.	1.	0.	0.	0.	0.
53	0.	0.	0.	0.	0.	0.	0.	0.	14.	0.	0.	0.	0.

TABLE IX

1980 TRIP TABLE

Destinations (node numbers)

Origins (node numbers)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0.	3510.	1967.	1276.	170.	456.	151.	624.	320.	1127.	2701.	1033.	91.	30.	229.	98.	332.	91.	33.	1987.
2	3510.	0.	205.	324.	3221.	768.	2908.	65.	82.	763.	356.	103.	16.	270.	3391.	199.	110.	26.	115.	53.
3	1967.	205.	0.	59.	12.	23.	13.	2331.	134.	25.	48.	130.	218.	3.	13.	10.	32.	18.	4.	401.
4	1276.	324.	59.	0.	29.	263.	11.	13.	8.	149.	26.	12.	6.	4.	12.	4.	6.	32.	2.	18.
5	170.	3221.	12.	29.	0.	116.	51.	3.	4.	40.	11.	4.	1.	50.	31.	6.	4.	2.	5.	3.
6	456.	768.	23.	263.	116.	0.	17.	6.	5.	238.	19.	7.	2.	8.	18.	4.	5.	7.	3.	6.
7	151.	2908.	13.	11.	51.	17.	0.	4.	7.	16.	17.	7.	1.	20.	216.	37.	9.	1.	38.	3.
8	624.	65.	2331.	13.	3.	6.	4.	0.	107.	7.	20.	103.	17.	1.	5.	4.	17.	3.	1.	120.
9	320.	82.	134.	8.	4.	5.	7.	107.	0.	6.	22.	142.	5.	1.	7.	10.	60.	2.	3.	19.
10	1127.	763.	25.	149.	40.	238.	16.	7.	6.	0.	49.	11.	2.	4.	26.	5.	7.	4.	2.	9.
11	2701.	356.	48.	26.	11.	19.	17.	20.	22.	49.	0.	82.	2.	2.	39.	15.	68.	2.	3.	22.
12	1033.	103.	130.	12.	4.	7.	7.	103.	142.	11.	82.	0.	4.	1.	10.	9.	134.	2.	2.	52.
13	91.	16.	218.	6.	1.	2.	1.	17.	5.	2.	2.	4.	0.	0.	1.	1.	1.	4.	0.	8.
14	30.	270.	3.	4.	50.	8.	20.	1.	1.	4.	2.	1.	0.	0.	6.	2.	1.	1.	3.	1.
15	229.	3391.	13.	12.	31.	18.	216.	5.	7.	26.	39.	10.	1.	6.	0.	38.	14.	1.	9.	4.
16	98.	199.	10.	4.	6.	4.	37.	4.	10.	5.	15.	9.	1.	2.	38.	0.	22.	1.	15.	2.
17	332.	110.	32.	6.	4.	5.	9.	17.	60.	7.	68.	134.	1.	1.	14.	22.	0.	1.	3.	9.
18	91.	26.	18.	32.	2.	7.	1.	3.	2.	4.	2.	2.	4.	1.	1.	1.	1.	0.	0.	3.
19	33.	115.	4.	2.	5.	3.	38.	1.	3.	2.	3.	2.	0.	3.	9.	15.	3.	0.	0.	1.
20	1987.	53.	401.	18.	3.	6.	3.	120.	19.	9.	22.	52.	8.	1.	4.	2.	9.	3.	1.	0.
21	898.	41.	712.	18.	2.	5.	2.	86.	12.	7.	12.	22.	14.	1.	3.	2.	5.	4.	1.	500.
22	242.	28.	91.	25.	2.	5.	1.	11.	4.	5.	4.	5.	18.	0.	1.	1.	2.	19.	0.	14.
23	77.	28.	9.	75.	3.	12.	1.	2.	1.	5.	2.	1.	1.	1.	1.	0.	1.	42.	0.	2.
24	90.	15.	53.	10.	1.	3.	1.	6.	2.	2.	2.	2.	24.	0.	1.	0.	1.	13.	0.	5.
25	601.	242.	21.	10.	7.	9.	14.	9.	14.	19.	858.	39.	1.	1.	38.	17.	73.	1.	2.	8.
26	37.	1240.	3.	4.	65.	8.	58.	1.	1.	5.	3.	1.	0.	34.	17.	3.	1.	0.	4.	1.
27	22.	31.	5.	1.	2.	1.	5.	2.	4.	1.	2.	2.	0.	1.	3.	6.	2.	0.	9.	1.
28	159.	18.	118.	12.	1.	3.	1.	11.	3.	3.	3.	4.	27.	0.	1.	1.	1.	7.	0.	13.
29	49.	27.	13.	2.	1.	1.	3.	6.	58.	1.	4.	10.	1.	0.	3.	6.	13.	0.	2.	2.
30	36.	501.	3.	3.	18.	5.	379.	1.	2.	4.	3.	2.	0.	15.	20.	6.	2.	0.	14.	1.
31	46.	3995.	3.	5.	141.	12.	69.	1.	1.	8.	4.	1.	0.	18.	23.	3.	2.	0.	3.	1.
32	50.	9.	266.	3.	1.	1.	1.	17.	4.	1.	1.	3.	138.	0.	1.	0.	1.	1.	0.	5.
33	2231.	122.	18.	88.	7.	30.	4.	5.	3.	159.	28.	7.	1.	1.	6.	2.	4.	2.	1.	9.
34	900.	307.	13.	21.	9.	20.	9.	4.	4.	125.	139.	10.	1.	1.	21.	4.	8.	1.	1.	6.
35	81.	10.	126.	5.	1.	2.	1.	9.	2.	1.	2.	2.	56.	0.	1.	0.	1.	3.	0.	8.
36	291.	24.	24.	58.	2.	6.	1.	4.	2.	6.	4.	3.	2.	0.	1.	1.	1.	8.	0.	8.
37	589.	42.	19.	121.	3.	12.	2.	4.	2.	19.	8.	4.	1.	0.	2.	1.	2.	3.	0.	9.
38	15.	233.	1.	2.	41.	4.	14.	0.	1.	2.	1.	1.	0.	130.	4.	1.	0.	0.	1.	0.
39	25.	152.	3.	2.	5.	2.	107.	1.	2.	2.	2.	1.	0.	3.	12.	9.	2.	0.	89.	1.
40	239.	114.	7.	124.	9.	130.	3.	2.	1.	330.	7.	2.	1.	1.	4.	1.	1.	2.	0.	2.
41	26.	948.	2.	2.	16.	4.	734.	1.	1.	3.	3.	1.	0.	6.	28.	4.	1.	0.	4.	0.
42	126.	96.	8.	3.	3.	3.	8.	3.	7.	5.	47.	13.	0.	1.	19.	22.	64.	0.	2.	2.
43	10.	11.	2.	0.	0.	0.	2.	1.	3.	0.	1.	1.	0.	0.	1.	4.	2.	0.	2.	0.
44	30.	22.	5.	1.	1.	1.	3.	2.	12.	1.	3.	5.	0.	0.	3.	13.	14.	0.	2.	1.
45	95.	22.	19.	2.	1.	1.	2.	14.	136.	2.	9.	110.	1.	0.	2.	3.	64.	0.	1.	4.
46	28.	11.	9.	1.	0.	1.	1.	5.	156.	1.	2.	9.	0.	0.	1.	2.	8.	0.	0.	1.
47	14.	315.	1.	2.	223.	6.	9.	0.	0.	3.	1.	0.	0.	18.	3.	1.	0.	0.	1.	0.
48	87.	305.	3.	17.	22.	118.	4.	1.	1.	132.	5.	1.	0.	1.	6.	1.	1.	1.	0.	1.
49	40.	7.	54.	3.	1.	1.	0.	5.	2.	1.	1.	1.	220.	0.	0.	0.	1.	2.	0.	3.
50	28.	263.	2.	2.	5.	2.	233.	1.	1.	2.	4.	1.	0.	1.	67.	17.	2.	0.	6.	1.
51	88.	180.	4.	3.	3.	3.	10.	2.	2.	6.	31.	4.	0.	1.	70.	9.	8.	0.	1.	1.
52	40.	5.	919.	1.	0.	1.	0.	121.	6.	1.	1.	4.	4.	0.	0.	0.	1.	0.	0.	6.
53	9.	2.	10.	1.	0.	0.	0.	1.	0.	0.	0.	0.	21.	0.	0.	0.	0.	1.	0.	1.

TABLE IX (Continued)

Destinations (node numbers)

Origins (node numbers)	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1	898.	242.	77.	90.	601.	37.	22.	159.	49.	36.	46.	50.	2231.	900.	81.	291.	989.	15.	25.	239.
2	41.	28.	28.	15.	242.	1240.	31.	18.	27.	501.	3995.	9.	122.	307.	10.	24.	42.	233.	152.	114.
3	712.	91.	9.	53.	21.	3.	5.	118.	13.	3.	3.	266.	18.	13.	126.	24.	19.	1.	3.	7.
4	18.	25.	75.	10.	10.	4.	1.	12.	2.	3.	5.	3.	88.	21.	5.	58.	121.	2.	2.	124.
5	2.	2.	3.	1.	7.	65.	2.	1.	1.	18.	141.	1.	7.	9.	1.	2.	3.	41.	5.	9.
6	5.	5.	12.	3.	9.	8.	1.	3.	1.	5.	12.	1.	30.	20.	2.	6.	12.	4.	2.	130.
7	2.	1.	1.	1.	14.	98.	5.	1.	3.	379.	69.	1.	4.	9.	1.	1.	2.	14.	107.	3.
8	86.	11.	2.	6.	9.	1.	2.	11.	6.	1.	1.	17.	5.	4.	9.	4.	4.	0.	1.	2.
9	12.	4.	1.	2.	14.	1.	4.	3.	58.	2.	1.	4.	3.	4.	2.	2.	2.	1.	2.	1.
10	7.	5.	5.	2.	19.	5.	1.	3.	1.	4.	8.	1.	159.	125.	1.	6.	19.	2.	2.	330.
11	12.	4.	2.	2.	858.	3.	2.	3.	4.	3.	4.	1.	28.	139.	2.	4.	8.	1.	2.	7.
12	22.	5.	1.	2.	39.	1.	2.	4.	10.	2.	1.	3.	7.	10.	2.	3.	4.	1.	1.	2.
13	14.	18.	1.	24.	1.	0.	0.	27.	1.	0.	0.	138.	1.	1.	56.	2.	1.	0.	0.	1.
14	1.	0.	1.	0.	1.	34.	1.	0.	0.	15.	18.	0.	1.	1.	0.	0.	0.	130.	3.	1.
15	3.	1.	1.	1.	38.	17.	3.	1.	3.	20.	23.	1.	6.	21.	1.	1.	2.	4.	12.	4.
16	2.	1.	0.	0.	17.	3.	6.	1.	6.	6.	3.	0.	2.	4.	0.	1.	1.	1.	9.	1.
17	5.	2.	1.	1.	73.	1.	2.	1.	13.	2.	2.	1.	4.	8.	1.	1.	2.	0.	2.	1.
18	4.	19.	42.	13.	1.	0.	0.	7.	0.	0.	0.	1.	2.	1.	3.	8.	3.	0.	0.	2.
19	1.	0.	0.	0.	2.	4.	5.	0.	2.	14.	3.	0.	1.	1.	0.	0.	0.	1.	89.	0.
20	500.	14.	2.	5.	8.	1.	1.	13.	2.	1.	1.	5.	9.	6.	8.	8.	9.	0.	1.	2.
21	0.	29.	2.	10.	4.	1.	1.	33.	1.	1.	1.	8.	6.	4.	19.	11.	9.	0.	0.	2.
22	29.	0.	6.	158.	2.	0.	C.	445.	1.	0.	0.	5.	4.	2.	47.	32.	8.	0.	0.	2.
23	2.	6.	0.	3.	1.	0.	0.	2.	0.	0.	1.	1.	3.	1.	1.	6.	4.	0.	0.	2.
24	10.	158.	3.	0.	1.	0.	C.	103.	0.	0.	0.	5.	1.	1.	41.	6.	2.	0.	0.	1.
25	4.	2.	1.	1.	0.	2.	1.	1.	3.	2.	3.	1.	8.	38.	1.	1.	3.	1.	2.	3.
26	1.	0.	0.	0.	2.	0.	1.	0.	0.	66.	305.	0.	1.	2.	0.	0.	0.	66.	7.	1.
27	1.	0.	0.	0.	1.	1.	0.	0.	4.	2.	1.	0.	0.	1.	0.	0.	0.	0.	2.	0.
28	33.	445.	2.	103.	1.	0.	0.	0.	0.	0.	0.	7.	2.	1.	217.	10.	4.	0.	0.	1.
29	1.	1.	0.	0.	3.	0.	4.	0.	0.	1.	0.	0.	1.	1.	0.	0.	0.	0.	1.	0.
30	1.	0.	0.	0.	2.	66.	2.	0.	1.	0.	23.	0.	1.	2.	0.	0.	0.	11.	44.	1.
31	1.	0.	1.	0.	3.	305.	1.	0.	0.	23.	0.	0.	2.	3.	0.	0.	1.	27.	4.	1.
32	8.	5.	1.	5.	1.	0.	0.	7.	0.	0.	0.	0.	1.	0.	11.	1.	1.	0.	0.	0.
33	6.	4.	3.	1.	8.	1.	C.	2.	1.	1.	2.	1.	0.	37.	1.	6.	42.	0.	1.	36.
34	4.	2.	1.	1.	38.	2.	1.	1.	1.	2.	3.	0.	37.	0.	1.	2.	5.	1.	1.	10.
35	19.	47.	1.	41.	1.	0.	0.	217.	0.	0.	0.	11.	1.	1.	0.	3.	2.	0.	0.	0.
36	11.	32.	6.	6.	1.	0.	0.	10.	0.	0.	0.	1.	6.	2.	3.	0.	34.	0.	0.	2.
37	9.	8.	4.	2.	3.	0.	0.	4.	0.	0.	1.	1.	42.	5.	2.	34.	0.	0.	0.	8.
38	0.	0.	0.	0.	1.	66.	C.	0.	0.	11.	27.	0.	0.	1.	0.	0.	0.	0.	2.	0.
39	0.	0.	0.	0.	2.	7.	2.	0.	1.	44.	4.	0.	1.	1.	0.	0.	0.	2.	0.	0.
40	2.	2.	2.	1.	3.	1.	0.	1.	0.	1.	1.	0.	36.	10.	0.	2.	8.	0.	0.	0.
41	0.	0.	0.	0.	2.	63.	1.	0.	0.	85.	38.	0.	1.	2.	0.	0.	0.	5.	9.	1.
42	1.	1.	0.	0.	141.	1.	1.	0.	2.	1.	1.	0.	2.	6.	0.	0.	1.	0.	1.	1.
43	0.	0.	0.	0.	1.	0.	15.	0.	5.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.
44	1.	0.	0.	0.	3.	0.	3.	0.	25.	1.	0.	0.	0.	1.	0.	0.	0.	0.	1.	0.
45	2.	1.	0.	0.	6.	0.	1.	1.	7.	0.	0.	0.	1.	1.	0.	0.	1.	0.	0.	0.
46	1.	0.	0.	0.	2.	0.	1.	0.	47.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
47	0.	0.	0.	0.	1.	24.	0.	0.	0.	4.	37.	0.	0.	1.	0.	0.	0.	28.	1.	1.
48	1.	1.	1.	0.	2.	2.	0.	0.	0.	1.	3.	0.	6.	8.	0.	1.	2.	1.	0.	16.
49	6.	14.	1.	32.	0.	0.	C.	20.	0.	0.	0.	12.	1.	0.	38.	1.	1.	0.	0.	0.
50	0.	0.	0.	0.	3.	4.	1.	0.	1.	11.	4.	0.	1.	2.	0.	0.	0.	1.	11.	0.
51	1.	0.	0.	0.	56.	1.	C.	0.	1.	1.	2.	0.	2.	9.	0.	0.	1.	0.	1.	1.
52	7.	1.	0.	1.	1.	0.	C.	2.	0.	0.	0.	6.	0.	0.	2.	0.	0.	0.	0.	0.
53	1.	2.	0.	4.	0.	C.	0.	2.	0.	0.	0.	3.	0.	0.	2.	0.	0.	0.	0.	0.

TABLE IX (Continued)

Destinations (node numbers)

Origins (node numbers)

	41	42	43	44	45	46	47	48	49	50	51	52	53
1	26.	126.	10.	30.	95.	28.	14.	87.	40.	28.	88.	40.	9.
2	948.	96.	11.	22.	22.	11.	315.	305.	7.	263.	180.	5.	2.
3	2.	8.	2.	5.	19.	9.	1.	3.	54.	2.	4.	919.	10.
4	2.	3.	0.	1.	2.	1.	2.	17.	3.	2.	3.	1.	1.
5	16.	3.	0.	1.	1.	0.	223.	22.	1.	5.	3.	0.	0.
6	4.	3.	0.	1.	1.	1.	6.	118.	1.	2.	3.	1.	0.
7	734.	8.	2.	3.	2.	1.	9.	4.	0.	233.	10.	0.	0.
8	1.	3.	1.	2.	14.	5.	0.	1.	5.	1.	2.	121.	1.
9	1.	7.	3.	12.	136.	156.	0.	1.	2.	1.	2.	6.	0.
10	3.	5.	0.	1.	2.	1.	3.	132.	1.	2.	6.	1.	0.
11	3.	47.	1.	3.	9.	2.	1.	5.	1.	4.	31.	1.	0.
12	1.	13.	1.	5.	110.	9.	0.	1.	1.	1.	4.	4.	0.
13	0.	0.	0.	0.	1.	0.	0.	0.	220.	0.	0.	4.	21.
14	6.	1.	0.	0.	0.	0.	18.	1.	0.	1.	1.	0.	0.
15	28.	19.	1.	3.	2.	1.	3.	6.	0.	67.	70.	0.	0.
16	4.	22.	4.	13.	3.	2.	1.	1.	0.	17.	9.	0.	0.
17	1.	64.	2.	14.	64.	8.	0.	1.	1.	2.	8.	1.	0.
18	0.	0.	0.	0.	0.	0.	0.	1.	2.	0.	0.	0.	1.
19	4.	2.	2.	2.	1.	0.	1.	0.	0.	6.	1.	0.	0.
20	0.	2.	0.	1.	4.	1.	0.	1.	3.	1.	1.	6.	1.
21	0.	1.	0.	1.	2.	1.	0.	1.	4.	0.	1.	7.	1.
22	0.	1.	0.	0.	1.	0.	0.	1.	14.	0.	0.	1.	2.
23	0.	0.	0.	0.	0.	0.	0.	1.	1.	0.	0.	0.	0.
24	0.	0.	0.	0.	0.	0.	0.	0.	32.	0.	0.	1.	4.
25	2.	141.	1.	3.	6.	2.	1.	2.	0.	3.	56.	1.	0.
26	63.	1.	0.	0.	0.	0.	24.	2.	0.	4.	1.	0.	0.
27	1.	1.	15.	3.	1.	1.	0.	0.	0.	1.	0.	0.	0.
28	0.	0.	0.	0.	1.	0.	0.	0.	20.	0.	0.	2.	2.
29	0.	2.	5.	25.	7.	47.	0.	0.	0.	1.	1.	0.	0.
30	85.	1.	0.	1.	0.	0.	4.	1.	0.	11.	1.	0.	0.
31	38.	1.	0.	0.	0.	0.	37.	3.	0.	4.	2.	0.	0.
32	0.	0.	0.	0.	0.	0.	0.	0.	12.	0.	0.	6.	3.
33	1.	2.	0.	0.	1.	0.	0.	6.	1.	1.	2.	0.	0.
34	2.	6.	0.	1.	1.	0.	1.	8.	0.	2.	9.	0.	0.
35	0.	0.	0.	0.	0.	0.	0.	0.	38.	0.	0.	2.	2.
36	0.	0.	0.	0.	0.	0.	0.	1.	1.	0.	0.	0.	0.
37	0.	1.	0.	0.	1.	0.	0.	2.	1.	0.	1.	0.	0.
38	5.	0.	0.	0.	0.	0.	26.	1.	0.	1.	0.	0.	0.
39	9.	1.	1.	1.	0.	0.	1.	0.	0.	11.	1.	0.	0.
40	1.	1.	0.	0.	0.	0.	1.	16.	0.	0.	1.	0.	0.
41	0.	1.	0.	0.	0.	0.	3.	1.	0.	11.	1.	0.	0.
42	1.	0.	1.	3.	4.	1.	0.	1.	0.	2.	22.	0.	0.
43	0.	1.	0.	4.	0.	1.	0.	0.	0.	0.	0.	0.	0.
44	0.	3.	4.	0.	3.	4.	0.	0.	0.	1.	1.	0.	0.
45	0.	4.	0.	3.	0.	9.	0.	0.	0.	0.	1.	1.	0.
46	0.	1.	1.	4.	9.	0.	0.	0.	0.	0.	0.	0.	0.
47	3.	0.	0.	0.	0.	0.	0.	1.	0.	1.	0.	0.	0.
48	1.	1.	0.	0.	0.	0.	1.	0.	0.	1.	1.	0.	0.
49	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	16.
50	11.	2.	0.	1.	0.	0.	1.	1.	0.	0.	3.	0.	0.
51	1.	22.	0.	1.	1.	0.	0.	1.	0.	3.	0.	0.	0.
52	0.	0.	0.	0.	1.	0.	0.	0.	1.	0.	0.	0.	0.
53	0.	0.	0.	0.	0.	0.	0.	0.	16.	0.	0.	0.	0.

TABLE X

1985 TRIP TABLE

Destinations (node numbers)

Origins (node numbers)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0.	4081.	2344.	1675.	241.	571.	196.	814.	396.	1497.	3677.	1317.	104.	39.	302.	127.	376.	116.	39.	2502.
2	4081.	0.	241.	421.	4502.	950.	3738.	84.	100.	1001.	478.	130.	18.	350.	4410.	255.	123.	33.	136.	66.
3	2344.	241.	0.	78.	17.	29.	17.	3080.	168.	34.	66.	168.	253.	4.	18.	13.	36.	24.	5.	511.
4	1675.	421.	78.	0.	46.	368.	16.	19.	12.	220.	39.	17.	7.	6.	18.	6.	8.	45.	3.	25.
5	241.	4502.	17.	46.	0.	174.	80.	5.	6.	64.	19.	7.	1.	80.	49.	9.	5.	4.	7.	4.
6	571.	950.	29.	368.	174.	0.	24.	8.	7.	336.	27.	9.	3.	11.	25.	6.	6.	10.	3.	8.
7	196.	3738.	17.	16.	80.	24.	C.	6.	9.	24.	25.	10.	1.	29.	315.	53.	11.	2.	50.	4.
8	814.	84.	3080.	19.	5.	8.	6.	0.	146.	11.	30.	145.	22.	1.	7.	6.	22.	4.	2.	167.
9	396.	100.	168.	12.	6.	7.	9.	146.	0.	9.	32.	190.	6.	2.	10.	14.	71.	2.	4.	25.
10	1497.	1001.	34.	220.	64.	336.	24.	11.	9.	0.	76.	15.	2.	6.	38.	7.	9.	6.	3.	13.
11	3677.	478.	66.	39.	19.	27.	25.	30.	32.	76.	0.	120.	3.	3.	59.	23.	90.	3.	4.	33.
12	1317.	130.	168.	17.	7.	9.	10.	145.	190.	15.	120.	0.	5.	1.	14.	13.	164.	2.	3.	71.
13	104.	18.	253.	7.	1.	3.	1.	22.	6.	2.	3.	5.	0.	0.	1.	1.	2.	4.	0.	9.
14	39.	350.	4.	6.	80.	11.	29.	1.	2.	6.	3.	1.	0.	0.	9.	3.	1.	1.	4.	1.
15	302.	4410.	18.	18.	49.	25.	315.	7.	10.	38.	59.	14.	1.	9.	0.	56.	18.	2.	13.	5.
16	127.	255.	13.	6.	9.	6.	53.	6.	14.	7.	23.	13.	1.	3.	56.	0.	28.	1.	19.	3.
17	376.	123.	36.	8.	5.	6.	11.	22.	71.	9.	90.	164.	2.	1.	18.	28.	0.	1.	3.	11.
18	116.	33.	24.	45.	4.	10.	2.	4.	2.	6.	3.	2.	4.	1.	2.	1.	1.	0.	0.	4.
19	39.	136.	5.	3.	7.	3.	5C.	2.	4.	3.	4.	3.	0.	4.	13.	19.	3.	0.	0.	1.
20	2502.	66.	511.	25.	4.	8.	4.	167.	25.	13.	33.	71.	9.	1.	5.	3.	11.	4.	1.	0.
21	1122.	5C.	900.	24.	4.	7.	3.	119.	16.	10.	17.	29.	17.	1.	4.	2.	6.	5.	1.	668.
22	309.	36.	118.	36.	3.	7.	2.	16.	5.	7.	6.	6.	22.	1.	2.	1.	2.	27.	1.	19.
23	93.	34.	11.	102.	4.	16.	2.	2.	1.	7.	3.	2.	2.	1.	1.	1.	1.	56.	0.	2.
24	110.	18.	65.	14.	2.	3.	1.	8.	3.	3.	3.	3.	28.	0.	1.	1.	1.	17.	0.	7.
25	710.	282.	25.	13.	9.	11.	18.	12.	17.	25.	1174.	49.	1.	2.	50.	22.	83.	1.	3.	10.
26	44.	1461.	4.	5.	93.	10.	125.	1.	2.	7.	4.	2.	0.	45.	23.	4.	2.	1.	5.	1.
27	27.	37.	5.	2.	2.	1.	6.	2.	5.	1.	3.	3.	0.	1.	3.	8.	3.	0.	11.	1.
28	196.	22.	147.	16.	2.	4.	1.	16.	4.	4.	4.	5.	32.	0.	1.	1.	2.	9.	0.	17.
29	59.	32.	16.	2.	2.	2.	4.	9.	73.	2.	6.	13.	1.	1.	3.	8.	15.	0.	2.	3.
30	47.	655.	5.	5.	29.	7.	554.	2.	2.	6.	5.	2.	0.	22.	29.	9.	2.	1.	19.	1.
31	58.	4951.	4.	7.	212.	16.	58.	1.	2.	11.	6.	2.	0.	25.	33.	4.	2.	1.	4.	1.
32	62.	10.	333.	4.	1.	1.	1.	23.	5.	1.	2.	4.	165.	0.	1.	1.	1.	2.	0.	7.
33	2991.	161.	25.	132.	11.	43.	6.	8.	5.	241.	44.	10.	2.	1.	9.	3.	5.	3.	1.	13.
34	1065.	359.	16.	28.	13.	25.	11.	6.	5.	166.	190.	12.	1.	2.	28.	5.	9.	1.	1.	7.
35	92.	11.	146.	7.	1.	2.	1.	12.	3.	2.	2.	3.	63.	0.	1.	0.	1.	4.	0.	9.
36	364.	29.	30.	81.	3.	9.	1.	6.	2.	9.	5.	4.	3.	0.	2.	1.	1.	11.	0.	10.
37	1072.	45.	21.	147.	4.	14.	2.	5.	2.	24.	10.	5.	2.	1.	2.	1.	2.	4.	0.	11.
38	19.	305.	2.	3.	65.	6.	21.	1.	1.	3.	2.	1.	0.	191.	6.	1.	1.	0.	2.	0.
39	31.	187.	3.	2.	8.	3.	147.	1.	2.	3.	3.	2.	0.	5.	16.	13.	2.	0.	112.	1.
40	320.	152.	9.	186.	14.	186.	4.	3.	2.	500.	11.	3.	1.	1.	6.	1.	2.	2.	1.	3.
41	39.	1415.	3.	4.	29.	6.	1227.	1.	2.	5.	5.	2.	0.	10.	47.	6.	2.	0.	6.	1.
42	165.	124.	10.	4.	4.	4.	11.	5.	10.	7.	71.	19.	1.	1.	28.	32.	81.	0.	2.	3.
43	11.	12.	2.	1.	1.	0.	2.	1.	3.	1.	1.	1.	0.	0.	1.	5.	2.	0.	2.	0.
44	38.	28.	7.	1.	1.	1.	4.	3.	17.	1.	5.	7.	0.	0.	4.	18.	17.	0.	2.	1.
45	134.	30.	27.	3.	2.	2.	3.	21.	202.	3.	15.	168.	1.	0.	3.	5.	87.	0.	1.	6.
46	32.	12.	10.	1.	1.	1.	1.	6.	184.	1.	3.	10.	0.	0.	1.	2.	9.	0.	1.	2.
47	19.	410.	2.	3.	353.	8.	12.	0.	1.	4.	2.	1.	0.	26.	5.	1.	1.	0.	1.	0.
48	104.	360.	4.	23.	32.	150.	6.	1.	1.	178.	7.	2.	0.	2.	8.	1.	1.	1.	1.	1.
49	52.	9.	73.	5.	1.	1.	1.	7.	2.	1.	1.	2.	282.	0.	1.	0.	1.	4.	0.	4.
50	35.	332.	3.	2.	7.	3.	325.	1.	2.	3.	5.	2.	0.	2.	96.	23.	3.	0.	8.	1.
51	106.	215.	5.	4.	5.	4.	14.	2.	3.	8.	43.	6.	0.	1.	94.	12.	9.	0.	1.	2.
52	47.	6.	1114.	2.	0.	1.	0.	160.	7.	1.	2.	5.	4.	0.	0.	0.	1.	0.	0.	8.
53	11.	2.	12.	1.	0.	0.	0.	2.	1.	0.	0.	0.	24.	0.	0.	0.	0.	1.	0.	1.

TABLE X (Continued)

Destinations (node numbers)

Origins (node numbers)	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1	1122.	309.	93.	110.	710.	44.	27.	196.	59.	47.	58.	62.	2991.	1065.	92.	364.	1072.	19.	31.	320.
2	50.	36.	34.	18.	282.	1461.	37.	22.	32.	655.	4951.	10.	161.	359.	11.	29.	45.	305.	187.	152.
3	900.	118.	11.	65.	25.	4.	5.	147.	16.	5.	4.	333.	25.	16.	146.	30.	21.	2.	3.	9.
4	24.	36.	102.	14.	13.	5.	2.	16.	2.	5.	7.	4.	132.	28.	7.	81.	147.	3.	2.	186.
5	4.	3.	4.	2.	9.	93.	2.	2.	2.	29.	212.	1.	11.	13.	1.	3.	4.	65.	8.	14.
6	7.	7.	16.	3.	11.	10.	1.	4.	2.	7.	16.	1.	43.	25.	2.	9.	14.	6.	3.	186.
7	3.	2.	2.	1.	18.	129.	6.	1.	4.	554.	96.	1.	6.	11.	1.	1.	2.	21.	147.	4.
8	119.	16.	2.	8.	12.	1.	2.	16.	9.	2.	1.	23.	8.	6.	12.	6.	5.	1.	1.	3.
9	16.	5.	1.	3.	17.	2.	5.	4.	73.	2.	2.	5.	5.	5.	3.	2.	2.	1.	2.	2.
10	10.	7.	7.	3.	25.	7.	1.	4.	2.	6.	11.	1.	241.	166.	2.	9.	24.	3.	3.	500.
11	17.	6.	3.	3.	1174.	4.	3.	4.	6.	5.	6.	2.	44.	190.	2.	5.	10.	2.	3.	11.
12	29.	6.	2.	3.	49.	2.	3.	5.	13.	2.	2.	4.	10.	12.	3.	4.	5.	1.	2.	3.
13	17.	22.	2.	28.	1.	0.	0.	32.	1.	0.	0.	165.	2.	1.	63.	3.	2.	0.	0.	1.
14	1.	1.	1.	0.	2.	45.	1.	0.	1.	22.	25.	0.	1.	2.	0.	0.	1.	191.	5.	1.
15	4.	2.	1.	1.	50.	23.	3.	1.	3.	29.	33.	1.	9.	28.	1.	2.	2.	6.	16.	6.
16	2.	1.	1.	1.	22.	4.	8.	1.	8.	9.	4.	1.	3.	5.	0.	1.	1.	1.	13.	1.
17	6.	2.	1.	1.	83.	2.	3.	2.	15.	2.	2.	1.	5.	9.	1.	1.	2.	1.	2.	2.
18	5.	27.	56.	17.	1.	1.	0.	9.	0.	1.	1.	2.	3.	1.	4.	11.	4.	0.	0.	2.
19	1.	1.	0.	0.	3.	5.	11.	0.	2.	19.	4.	0.	1.	1.	0.	0.	0.	2.	112.	1.
20	668.	19.	2.	7.	10.	1.	1.	17.	3.	1.	1.	7.	13.	7.	9.	10.	11.	0.	1.	3.
21	0.	40.	3.	13.	6.	1.	1.	43.	2.	1.	1.	11.	9.	4.	23.	14.	10.	0.	1.	3.
22	40.	0.	7.	210.	2.	1.	0.	594.	1.	1.	1.	7.	6.	2.	58.	44.	9.	0.	0.	2.
23	3.	7.	0.	4.	1.	1.	0.	3.	0.	1.	1.	1.	4.	1.	1.	7.	4.	0.	0.	3.
24	13.	210.	4.	0.	1.	0.	0.	132.	0.	0.	0.	6.	2.	1.	48.	8.	3.	0.	0.	1.
25	6.	2.	1.	1.	0.	3.	2.	1.	4.	3.	3.	1.	11.	45.	1.	2.	3.	1.	2.	4.
26	1.	1.	1.	0.	3.	0.	1.	0.	1.	88.	388.	0.	2.	2.	0.	0.	1.	89.	8.	1.
27	1.	0.	0.	0.	2.	1.	0.	0.	5.	2.	1.	0.	1.	1.	0.	0.	0.	0.	3.	0.
28	43.	594.	3.	132.	1.	0.	0.	0.	1.	0.	0.	9.	3.	1.	259.	13.	4.	0.	0.	1.
29	2.	1.	0.	0.	4.	1.	5.	1.	0.	1.	1.	1.	1.	1.	0.	0.	0.	0.	1.	0.
30	1.	1.	1.	0.	3.	88.	2.	0.	1.	0.	33.	0.	1.	2.	0.	0.	1.	16.	61.	1.
31	1.	1.	1.	0.	3.	388.	1.	0.	1.	33.	0.	0.	2.	4.	0.	1.	1.	37.	5.	2.
32	11.	7.	1.	6.	1.	0.	0.	9.	1.	0.	0.	0.	1.	1.	13.	1.	1.	0.	0.	0.
33	9.	6.	4.	2.	11.	2.	1.	3.	1.	1.	2.	1.	0.	49.	1.	9.	52.	1.	1.	56.
34	4.	2.	1.	1.	45.	2.	1.	1.	1.	2.	4.	1.	49.	0.	1.	2.	6.	1.	1.	13.
35	23.	58.	1.	48.	1.	0.	0.	259.	0.	0.	0.	13.	1.	1.	0.	4.	2.	0.	0.	1.
36	14.	44.	7.	8.	2.	0.	0.	13.	0.	0.	1.	1.	9.	2.	4.	0.	39.	0.	0.	4.
37	10.	9.	4.	3.	3.	1.	0.	4.	0.	1.	1.	1.	52.	6.	2.	39.	0.	0.	0.	10.
38	0.	0.	0.	0.	1.	89.	0.	0.	0.	16.	37.	0.	1.	1.	0.	0.	0.	0.	2.	1.
39	1.	0.	0.	0.	2.	8.	3.	0.	1.	61.	5.	0.	1.	1.	0.	0.	0.	2.	0.	1.
40	3.	2.	3.	1.	4.	1.	0.	1.	0.	1.	2.	0.	56.	13.	1.	4.	10.	1.	1.	0.
41	1.	0.	0.	0.	3.	97.	1.	0.	1.	144.	61.	0.	1.	2.	0.	0.	0.	9.	14.	1.
42	2.	1.	0.	0.	185.	1.	1.	1.	3.	2.	2.	0.	3.	8.	0.	1.	1.	0.	2.	1.
43	0.	0.	0.	0.	1.	0.	17.	0.	6.	1.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.
44	1.	0.	0.	0.	4.	0.	3.	0.	33.	1.	1.	0.	1.	1.	0.	0.	0.	0.	1.	0.
45	4.	1.	0.	1.	9.	0.	1.	1.	10.	1.	0.	1.	1.	2.	1.	1.	1.	0.	1.	1.
46	1.	0.	0.	0.	2.	0.	1.	0.	54.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.
47	0.	0.	0.	0.	1.	32.	0.	0.	0.	6.	53.	0.	1.	1.	0.	0.	0.	42.	1.	1.
48	1.	1.	1.	0.	3.	2.	0.	0.	0.	1.	4.	0.	8.	9.	0.	1.	2.	1.	1.	23.
49	8.	20.	1.	44.	1.	0.	0.	28.	0.	0.	0.	17.	1.	0.	49.	2.	1.	0.	0.	0.
50	1.	0.	0.	0.	4.	6.	1.	0.	1.	16.	5.	0.	1.	2.	0.	0.	0.	1.	15.	1.
51	1.	1.	0.	0.	69.	2.	1.	0.	1.	2.	2.	0.	2.	11.	0.	0.	1.	0.	1.	1.
52	9.	2.	0.	1.	1.	0.	0.	2.	1.	0.	0.	7.	1.	0.	2.	1.	0.	0.	0.	0.
53	1.	2.	0.	4.	0.	0.	0.	2.	0.	0.	0.	3.	0.	0.	3.	0.	0.	0.	0.	0.

TABLE X (Continued)

Destinations (node numbers)

Origins (node numbers)

	41	42	43	44	45	46	47	48	49	50	51	52	53
1	39.	165.	11.	38.	134.	32.	19.	104.	52.	35.	106.	47.	11.
2	1415.	124.	12.	28.	30.	12.	410.	360.	9.	332.	215.	6.	2.
3	3.	10.	2.	7.	27.	10.	2.	4.	73.	3.	5.	1114.	12.
4	4.	4.	1.	1.	3.	1.	3.	23.	5.	2.	4.	2.	1.
5	29.	4.	1.	1.	2.	1.	353.	32.	1.	7.	5.	0.	0.
6	6.	4.	0.	1.	2.	1.	8.	150.	1.	3.	4.	1.	0.
7	1227.	11.	2.	4.	3.	1.	12.	6.	1.	329.	14.	0.	0.
8	1.	5.	1.	3.	21.	6.	0.	1.	7.	1.	2.	160.	2.
9	2.	10.	3.	17.	202.	184.	1.	1.	2.	2.	3.	7.	1.
10	5.	7.	1.	1.	3.	1.	4.	178.	1.	3.	8.	1.	0.
11	5.	71.	1.	5.	15.	3.	2.	7.	1.	5.	43.	2.	0.
12	2.	19.	1.	7.	168.	10.	1.	2.	2.	2.	6.	5.	0.
13	0.	1.	0.	0.	1.	0.	0.	0.	282.	0.	0.	4.	24.
14	10.	1.	0.	0.	0.	0.	26.	2.	0.	2.	1.	0.	0.
15	47.	28.	1.	4.	3.	1.	5.	8.	1.	96.	94.	0.	0.
16	6.	32.	5.	18.	5.	2.	1.	1.	0.	23.	12.	0.	0.
17	2.	81.	2.	17.	87.	9.	1.	1.	1.	3.	9.	1.	0.
18	0.	0.	0.	0.	0.	0.	0.	1.	4.	0.	0.	0.	1.
19	6.	2.	2.	2.	1.	1.	1.	1.	0.	8.	1.	0.	0.
20	1.	3.	0.	1.	6.	2.	0.	1.	4.	1.	2.	8.	1.
21	1.	2.	0.	1.	4.	1.	0.	1.	8.	1.	1.	9.	1.
22	0.	1.	0.	0.	1.	0.	0.	1.	20.	0.	1.	2.	2.
23	0.	0.	0.	0.	0.	0.	0.	1.	1.	0.	0.	0.	0.
24	0.	0.	0.	0.	1.	0.	0.	0.	44.	0.	0.	1.	4.
25	3.	185.	1.	4.	9.	2.	1.	3.	1.	4.	69.	1.	0.
26	97.	1.	0.	0.	0.	0.	32.	2.	0.	6.	2.	0.	0.
27	1.	1.	17.	3.	1.	1.	0.	0.	0.	1.	1.	0.	0.
28	0.	1.	0.	0.	1.	0.	0.	0.	28.	0.	0.	2.	2.
29	1.	3.	6.	33.	10.	54.	0.	0.	0.	1.	1.	1.	0.
30	144.	2.	1.	1.	1.	0.	6.	1.	0.	16.	2.	0.	0.
31	61.	2.	0.	1.	0.	0.	53.	4.	0.	5.	2.	0.	0.
32	0.	0.	0.	0.	1.	0.	0.	0.	17.	0.	0.	7.	3.
33	1.	3.	0.	1.	1.	0.	1.	8.	1.	1.	2.	1.	0.
34	2.	8.	0.	1.	2.	1.	1.	9.	0.	2.	11.	0.	0.
35	0.	0.	0.	0.	1.	0.	0.	0.	49.	0.	0.	2.	3.
36	0.	1.	0.	0.	1.	0.	0.	1.	2.	0.	0.	1.	0.
37	0.	1.	0.	0.	1.	0.	0.	2.	1.	0.	1.	0.	0.
38	9.	0.	0.	0.	0.	0.	42.	1.	0.	1.	0.	0.	0.
39	14.	2.	1.	1.	1.	0.	1.	1.	0.	15.	1.	0.	0.
40	1.	1.	0.	0.	1.	0.	1.	23.	0.	1.	1.	0.	0.
41	0.	2.	0.	1.	0.	0.	5.	1.	0.	17.	2.	0.	0.
42	2.	0.	1.	4.	6.	1.	0.	1.	0.	3.	30.	0.	0.
43	0.	1.	0.	5.	1.	1.	0.	0.	0.	0.	0.	0.	0.
44	1.	4.	5.	0.	5.	5.	0.	0.	0.	1.	1.	0.	0.
45	0.	6.	1.	5.	0.	13.	0.	0.	0.	1.	1.	1.	0.
46	0.	1.	1.	5.	13.	0.	0.	0.	0.	0.	0.	0.	0.
47	5.	0.	0.	0.	0.	0.	0.	1.	0.	1.	0.	0.	0.
48	1.	1.	0.	0.	0.	0.	1.	0.	0.	1.	1.	0.	0.
49	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	21.
50	17.	3.	0.	1.	1.	0.	1.	1.	0.	0.	4.	0.	0.
51	2.	30.	0.	1.	1.	0.	0.	1.	0.	4.	0.	0.	0.
52	0.	0.	0.	0.	1.	0.	0.	0.	1.	0.	0.	0.	0.
53	0.	0.	0.	0.	0.	0.	0.	0.	21.	0.	0.	0.	0.

feasible decision policy of 00322000 is assumed to transform the existing state, then the new state would be 22422000 as shown in Table XI. A two-lane road was considered to be the only feasible additions for interconnecting nodes 17-46, 19-44, 28-37, and 42-44. The link which interconnects nodes 4-33 could represent either a two-lane or a four-lane road.

TABLE XI

CODE FOR STATES AND DECISIONS

Digit Number	1	2	3	4	5	6	7	8
Link Connection	2-5	2-50	9-12	4-33	17-46	19-44	28-37	42-44
Existing State	2	2	2	0	0	0	0	0
Feasible Decision	0	0	3	2	2	0	0	0
New State	2	2	4	2	2	0	0	0

With the above limitations imposed upon the state of each link, 200 feasible network states were constructed. A network operators' cost was determined for each of the 200 networks by employing program 1 and program 2 in Figure 3, on page 40. Operators' costs for every link were assumed. The assumed two-lane travelers' cost are given in Table XII. Table XIII contains the two and four-lane costs for the eight

TABLE XII

TWO-LANE LINK COSTS

LINK NUMBER	LINK CONNECTIONS	Two - Lane				LINK NUMBER	LINK CONNECTIONS	Two - Lane			
		1970	1975	1980	1985			1970	1975	1980	1985
1	1-11	3.7	3.9	4.0	4.2	51	9-52	6.8	7.3	7.4	7.5
2	1-12	5.2	5.4	5.7	6.0	52	10-15	6.8	7.0	7.4	7.5
3	1-20	3.5	3.6	3.8	3.9	53	10-34	2.9	3.1	3.4	3.5
4	1-22	8.1	8.4	8.5	8.7	54	10-34	2.9	3.1	3.4	3.5
5	1-33	3.0	3.2	3.6	3.7	55	10-40	1.8	1.9	2.0	2.3
6	1-34	3.7	3.9	4.0	4.1	56	10.48	2.3	2.4	2.7	2.9
7	1-37	3.6	3.9	4.2	4.5	57	11-12	4.3	4.6	4.8	4.9
8	2-5	4.3	4.4	4.7	5.0	58	11-17	4.0	4.4	4.5	4.6
9	2-7	4.3	4.5	4.8	5.0	59	11-25	1.5	1.7	1.8	2.0
10	2-15	3.3	3.5	3.8	4.1	60	11-34	2.7	3.0	3.2	3.3
11	2-31	2.5	2.7	3.0	3.2	61	11-51	3.3	3.5	3.7	3.9
12	2-34	5.6	5.8	5.9	6.0	62	12-17	3.1	3.3	3.7	3.8
13	2-41	3.4	3.6	3.9	4.0	63	12-20	4.5	4.9	5.2	5.4
14	2-48	4.7	4.8	5.0	5.2	64	12-45	2.3	2.5	2.8	3.0
15	2-50	5.0	5.3	5.6	5.7	65	13-32	2.4	2.7	3.0	3.1
16	3-8	2.8	3.0	3.4	3.5	66	13-35	3.2	3.4	3.6	3.8
17	3-12	6.8	7.0	7.3	7.4	67	13-49	2.0	2.1	2.3	2.5
18	3-13	5.3	5.6	5.7	5.9	68	13-53	3.3	3.5	3.8	4.0
19	3-20	3.9	4.1	4.2	4.4	69	14-30	6.5	6.9	7.3	7.4
20	3-21	3.3	3.5	3.8	3.9	70	14-38	2.3	2.5	2.7	2.8
21	3-32	3.9	4.3	4.4	4.6	71	14-47	4.1	4.5	4.6	4.8
22	3-35	4.8	5.0	5.3	5.4	72	15-16	5.0	5.2	5.4	5.6
23	3-52	1.8	2.0	2.3	2.4	73	15-34	5.2	5.5	5.6	5.7
24	4-6	5.0	5.2	5.4	5.5	74	15-41	4.1	4.2	4.3	4.4
25	4-18	8.4	8.6	8.8	8.9	75	15-50	2.7	2.8	3.1	3.4
26	4-23	5.3	5.4	5.6	5.8	76	15-51	2.5	2.6	2.8	3.1
27	4-36	4.7	5.0	5.2	5.3	77	16-17	5.3	5.5	5.6	5.7
28	4-37	3.6	3.7	3.8	4.2	78	16-19	6.7	7.1	7.3	7.5
29	4-40	3.4	3.8	3.9	4.1	79	16-27	8.9	9.1	9.5	9.6
30	5-6	6.0	6.3	6.4	6.5	80	16-42	3.8	4.2	4.3	4.4
31	5-31	3.4	3.5	3.6	3.7	81	16-43	6.5	7.0	7.1	7.2
32	5-47	2.1	2.5	2.6	2.8	82	16-44	4.4	4.8	5.0	5.3
33	5-48	5.1	5.5	5.6	5.7	83	16-50	4.0	4.0	4.2	4.7
34	6-10	3.9	4.1	4.4	4.5	84	17-25	3.2	3.6	3.6	3.8
35	6-23	9.2	9.3	9.8	9.9	85	17-42	2.6	3.0	3.0	3.2
36	6-40	3.1	3.2	3.6	3.8	86	17-44	4.3	4.6	4.8	5.0
37	6-48	2.8	3.0	3.4	3.5	87	17-45	2.5	2.7	2.8	2.9
38	7-15	3.7	4.0	4.2	4.3	88	18-22	6.7	7.2	7.4	7.6
39	7-30	2.6	2.7	3.2	3.3	89	18-23	4.6	5.0	5.1	5.2
40	7-39	3.6	3.7	3.8	4.3	90	18-24	7.0	7.3	7.6	7.7
41	7-41	1.5	1.7	1.9	2.3	91	18-36	6.7	7.0	7.2	7.4
42	7-50	2.1	2.4	2.6	2.7	92	19-27	8.1	8.3	8.4	8.5
43	8-9	5.2	5.5	5.7	5.8	93	19-39	3.0	3.2	3.4	3.6
44	8-12	4.5	4.6	5.0	5.2	94	19-43	8.8	9.3	9.7	9.8
45	8-20	3.8	4.2	4.3	4.4	95	19-50	6.1	6.3	6.4	6.5
46	8-45	5.6	5.8	6.0	6.2	96	20-21	1.8	2.0	2.1	2.2
47	8-52	2.2	2.4	2.6	2.8	97	20-37	5.8	5.9	6.0	6.3
48	9-12	4.1	4.2	4.6	4.9	98	21-22	5.2	5.3	5.4	5.8
49	9-45	2.5	2.7	2.8	2.9	99	21-28	4.3	4.7	4.8	4.9
50	9-46	2.2	2.4	2.6	2.7	100	21-35	4.5	4.8	4.9	5.0

TABLE XII (Continued)

LINK NUMBER	LINK CONNECTIONS	Two - Lane 1970	1975	1980	1985	LINK NUMBER	LINK CONNECTIONS	Two - Lane 1970	1975	1980	1985
101	21-36	5.6	5.7	5.8	6.3	126	30-41	2.3	2.4	2.6	2.7
102	21-37	6.0	6.3	6.5	6.6	127	31-38	3.3	3.5	3.7	4.0
103	22-33	8.9	9.2	9.7	9.9	128	31-41	2.9	3.3	3.4	3.6
104	22-24	2.7	2.9	3.2	3.2	129	31-47	2.5	2.6	2.8	3.0
105	22-28	1.7	2.0	2.2	2.2	130	32-35	4.5	5.1	5.1	5.2
106	22-36	3.9	4.2	4.5	4.6	131	32-52	4.3	4.4	4.9	5.0
107	23-36	6.6	7.2	7.3	7.4	132	33-34	3.2	3.5	3.7	4.0
108	24-28	2.7	3.0	3.3	3.4	133	33-37	3.8	3.0	3.3	3.6
109	24-35	3.3	3.7	3.8	3.9	134	33-40	2.9	3.1	3.5	3.8
110	24-49	3.6	4.0	4.2	4.3	135	34-51	3.8	4.2	4.5	4.7
111	24-53	5.7	6.0	6.2	6.4	136	35-49	2.8	3.0	3.3	3.5
112	25-34	3.5	3.8	3.9	4.0	137	36-37	2.8	3.0	3.6	3.6
113	25-42	1.8	2.3	2.5	2.6	138	37-40	4.5	5.0	5.2	5.2
114	25-51	2.2	2.5	2.7	2.9	139	38-47	2.3	2.3	2.5	2.5
115	26-30	3.2	3.4	3.6	3.8	140	39-50	4.0	4.0	4.3	4.5
116	26-31	1.7	2.0	2.3	2.3	141	40-48	3.0	3.2	3.4	3.8
117	26-38	2.4	2.8	2.9	3.0	142	42-50	5.7	6.2	6.5	6.6
118	26-41	2.5	2.5	2.6	2.7	143	42-51	2.4	2.5	2.7	3.0
119	26-47	3.1	3.4	3.5	3.8	144	43-44	4.3	4.4	4.6	4.8
120	27-43	4.0	4.1	4.2	4.4	145	44-45	4.9	5.2	5.2	5.4
121	28-35	1.7	2.0	2.3	2.4	146	44-46	4.1	4.3	4.6	4.6
122	29-43	5.1	5.4	5.6	5.9	147	45-46	3.1	3.2	3.4	3.5
123	29-44	3.1	3.1	3.4	3.6	148	48-51	6.9	7.2	7.6	7.7
124	29-46	2.3	2.5	2.7	3.0	149	49-53	2.7	3.0	3.1	3.2
125	30-39	3.6	4.0	4.2	4.5	150	50-51	4.5	4.8	5.2	5.3

TABLE XIII

TWO AND FOUR-LANE LINK COSTS FOR EACH FEASIBLE LINK ADDITION

LINK NUMBER	LINK CONNECTIONS	Two - Lane					Four - Lane			
		1970	1975	1980	1985		1970	1975	1980	1985
1	2-5	4.3	4.4	4.7	5.0		4.1	4.2	4.3	4.4
2	2-50	5.0	5.3	5.6	5.7		4.6	4.7	4.9	5.1
3	9-12	4.1	4.2	4.6	4.9		3.7	3.9	4.0	4.3
4	4-33	4.4	4.6	4.8	5.0		3.8	4.2	4.2	4.3
5	17-46	4.8	5.0	5.4	5.5		4.5	4.5	4.7	4.8
6	19-44	9.1	9.3	9.5	9.9		8.0	8.2	8.7	9.0
7	28-37	7.2	7.7	7.8	7.9		6.5	6.7	7.0	7.1
8	42-44	5.2	5.4	5.6	5.8		4.6	4.7	4.8	5.0

links that may change in the network. The 200 network operators' costs were actually determined with 23 runs of the BPR programs as shown in Figure 3. This was possible since the network connections did not change when considering only two to four-lane changes. The only change required was to impose a new link cost to represent the travelers' cost on a four-lane road. These changes were accomplished by the NTOPCOST program. Operators' cost on a given link was assumed to vary between periods. This would account for increased travelers' costs due to vehicle expenses, road deterioration, and increased road volumes.

The 200 network operators' costs and maintenance costs for each period are shown in Table XIV. The maintenance costs were assumed to increase with each time period and for each link addition to the existing network. This is based on increased volumes and increased number of links requiring higher maintenance costs. These numbers were arbitrarily assumed and may or may not represent actual maintenance costs. In Table XV the 326 feasible decisions with their corresponding construction costs are shown. The network operators' costs, maintenance costs, and construction costs have been reduced by a factor of 1825. This was imposed on each cost since the link volume calculated in Equation (3.2.2) was assumed to represent average daily traffic volume; therefore, a conversion factor of 1825 (5 years * 365 days) would be required to represent the total traffic volumes during each five year period. Since this same number should be applied to maintenance costs and construction costs, these costs have been reduced by this same factor. This reduction in costs in no way affects the optimal decisions arrived at by the DP optimization process.

The optimal stage costs and corresponding optimal decisions

TABLE XIV

NETWORK PERIOD OPERATORS' COSTS AND MAINTENANCE COSTS

NETNU	STATE	OPCOST,PERIOD=1,2,3,4				MAINTENANCE COST,PERIOD=1,2,3,4			
1	2 2 2 0 0 0 0 0	1049888.	779092.	509304.	355385.	230975.	155818.	81489.	53308.
2	2 2 4 0 0 0 0 0	1049043.	778436.	509041.	355149.	231552.	156208.	81693.	53441.
3	2 4 2 0 0 0 0 0	1049480.	778717.	509105.	355293.	231552.	156208.	81693.	53441.
4	4 2 2 0 0 0 0 0	1044329.	776450.	508399.	354685.	231552.	156208.	81693.	53441.
5	4 4 2 0 0 0 0 0	1043921.	776075.	508200.	354594.	231899.	156441.	81815.	53521.
6	4 2 4 0 0 0 0 0	1043484.	775794.	508136.	354450.	231899.	156441.	81815.	53521.
7	2 4 4 0 0 0 0 0	1048635.	778061.	508841.	355058.	231899.	156441.	81815.	53521.
8	4 4 4 0 0 0 0 0	1043076.	775418.	507937.	354359.	232130.	156597.	81896.	53575.
9	2 2 2 2 2 0 0 0	1048936.	778546.	508973.	355152.	231899.	156441.	81815.	53521.
10	2 2 4 2 2 0 0 0	1048092.	777890.	508709.	354917.	232130.	156597.	81896.	53575.
11	2 4 2 2 2 0 0 0	1048529.	778171.	508774.	355061.	232130.	156597.	81896.	53575.
12	4 2 2 2 2 0 0 0	1043377.	775903.	508068.	354453.	232130.	156597.	81896.	53575.
13	4 4 2 2 2 0 0 0	1042969.	775528.	507869.	354361.	232295.	156708.	81955.	53613.
14	4 2 4 2 2 0 0 0	1042532.	775247.	507805.	354217.	232295.	156708.	81955.	53613.
15	2 4 4 2 2 0 0 0	1047684.	777514.	508510.	354826.	232295.	156708.	81955.	53613.
16	4 4 4 2 2 0 0 0	1042124.	774872.	507605.	354126.	232419.	156792.	81998.	53641.
17	2 2 2 4 2 0 0 0	1048610.	778355.	508892.	355061.	232130.	156597.	81896.	53575.
18	2 2 4 4 2 0 0 0	1047766.	777699.	508629.	354826.	232295.	156708.	81955.	53613.
19	2 4 2 4 2 0 0 0	1048203.	777980.	508693.	354970.	232295.	156708.	81955.	53613.
20	4 2 2 4 2 0 0 0	1043051.	775713.	507987.	354361.	232295.	156708.	81955.	53613.
21	4 4 2 4 2 0 0 0	1042643.	775337.	507788.	354270.	232419.	156792.	81998.	53641.
22	4 2 4 4 2 0 0 0	1042206.	775056.	507724.	354126.	232419.	156792.	81998.	53641.
23	2 4 4 4 2 0 0 0	1047358.	777324.	508429.	354735.	232419.	156792.	81998.	53641.
24	4 4 4 4 2 0 0 0	1041798.	774681.	507525.	354035.	232515.	156857.	82032.	53663.
25	2 2 2 2 0 2 0 0	1048955.	778570.	508989.	355171.	231899.	156441.	81815.	53521.
26	2 2 4 2 0 2 0 0	1048111.	777913.	508726.	354936.	232130.	156597.	81896.	53575.
27	2 4 2 2 0 2 0 0	1048548.	778194.	508790.	355080.	232130.	156597.	81896.	53575.
28	4 2 2 2 0 2 0 0	1043397.	775927.	508084.	354472.	232130.	156597.	81896.	53575.
29	4 4 2 2 0 2 0 0	1042989.	775552.	507885.	354380.	232295.	156708.	81955.	53613.
30	4 2 4 2 0 2 0 0	1042552.	775271.	507821.	354236.	232295.	156708.	81955.	53613.
31	2 4 4 2 0 2 0 0	1047703.	777538.	508527.	354845.	232295.	156708.	81955.	53613.
32	4 4 4 2 0 2 0 0	1042144.	774896.	507622.	354145.	232419.	156792.	81998.	53641.
33	2 2 2 4 0 2 0 0	1048629.	778379.	508908.	355080.	232130.	156597.	81896.	53575.
34	2 2 4 4 0 2 0 0	1047785.	777722.	508645.	354845.	232295.	156708.	81955.	53613.
35	2 4 2 4 0 2 0 0	1048222.	778004.	508709.	354989.	232295.	156708.	81955.	53613.
36	4 2 2 4 0 2 0 0	1043070.	775736.	508004.	354380.	232295.	156708.	81955.	53613.
37	4 4 2 4 0 2 0 0	1042662.	775361.	507804.	354289.	232419.	156792.	81998.	53641.
38	4 2 4 4 0 2 0 0	1042226.	775080.	507740.	354145.	232419.	156792.	81998.	53641.
39	2 4 4 4 0 2 0 0	1047377.	777347.	508446.	354754.	232419.	156792.	81998.	53641.
40	4 4 4 4 0 2 0 0	1041818.	774705.	507541.	354054.	232515.	156857.	82032.	53663.
41	2 2 2 2 0 0 2 0	1048917.	778523.	508978.	355168.	231899.	156441.	81815.	53521.
42	2 2 4 2 0 0 2 0	1048073.	777867.	508715.	354932.	232130.	156597.	81896.	53575.
43	2 4 2 2 0 0 2 0	1048510.	778148.	508779.	355076.	232130.	156597.	81896.	53575.
44	4 2 2 2 0 0 2 0	1043358.	775881.	508073.	354468.	232130.	156597.	81896.	53575.
45	4 4 2 2 0 0 2 0	1042950.	775506.	507874.	354377.	232295.	156708.	81955.	53613.
46	4 2 4 2 0 0 2 0	1042513.	775224.	507810.	354233.	232295.	156708.	81955.	53613.
47	2 4 4 2 0 0 2 0	1047665.	777492.	508516.	354841.	232295.	156708.	81955.	53613.
48	4 4 4 2 0 0 2 0	1042105.	774849.	507611.	354142.	232419.	156792.	81998.	53641.
49	2 2 2 4 0 0 2 0	1048591.	778332.	508897.	355076.	232130.	156597.	81896.	53575.
50	2 2 4 4 0 0 2 0	1047747.	777676.	508634.	354841.	232295.	156708.	81955.	53613.
51	2 4 2 4 0 0 2 0	1048184.	777957.	508698.	354985.	232295.	156708.	81955.	53613.
52	4 2 2 4 0 0 2 0	1043032.	775690.	507993.	354377.	232295.	156708.	81955.	53613.
53	4 4 2 4 0 0 2 0	1042624.	775315.	507793.	354286.	232419.	156792.	81998.	53641.
54	4 2 4 4 0 0 2 0	1042187.	775034.	507729.	354142.	232419.	156792.	81998.	53641.
55	2 4 4 4 0 0 2 0	1047339.	777301.	508435.	354750.	232419.	156792.	81998.	53641.
56	4 4 4 4 0 0 2 0	1041779.	774658.	507530.	354050.	232515.	156857.	82032.	53663.
57	2 2 2 2 0 0 0 2	1048939.	778560.	508988.	355173.	231899.	156441.	81815.	53521.
58	2 2 4 2 0 0 0 2	1048095.	777903.	508724.	354938.	232130.	156597.	81896.	53575.
59	2 4 2 2 0 0 0 2	1048532.	778185.	508788.	355082.	232130.	156597.	81896.	53575.
60	4 2 2 2 0 0 0 2	1043380.	775917.	508083.	354473.	232130.	156597.	81896.	53575.
61	4 4 2 2 0 0 0 2	1042972.	775542.	507884.	354382.	232295.	156708.	81955.	53613.
62	4 2 4 2 0 0 0 2	1042535.	775261.	507819.	354238.	232295.	156708.	81955.	53613.
63	2 4 4 2 0 0 0 2	1047687.	777528.	508525.	354846.	232295.	156708.	81955.	53613.
64	4 4 4 2 0 0 0 2	1042127.	774886.	507620.	354147.	232419.	156792.	81998.	53641.
65	2 2 2 4 0 0 0 2	1048613.	778369.	508907.	355082.	232130.	156597.	81896.	53575.
66	2 2 4 4 0 0 0 2	1047769.	777713.	508643.	354847.	232295.	156708.	81955.	53613.
67	2 4 2 4 0 0 0 2	1048206.	777994.	508708.	354990.	232295.	156708.	81955.	53613.
68	4 2 2 4 0 0 0 2	1043054.	775727.	508002.	354382.	232295.	156708.	81955.	53613.
69	4 4 2 4 0 0 0 2	1042646.	775351.	507803.	354291.	232419.	156792.	81998.	53641.
70	4 2 4 4 0 0 0 2	1042209.	775070.	507739.	354147.	232419.	156792.	81998.	53641.
71	2 4 4 4 0 0 0 2	1047361.	777337.	508444.	354755.	232419.	156792.	81998.	53641.
72	4 4 4 4 0 0 0 2	1041801.	774695.	507539.	354056.	232515.	156857.	82032.	53663.
73	2 2 2 0 2 2 0 0	1050587.	779835.	509902.	355719.	231899.	156441.	81815.	53521.
74	2 2 4 0 2 2 0 0	1050523.	779780.	509882.	355699.	232130.	156597.	81896.	53575.
75	2 4 2 0 2 2 0 0	1050179.	779460.	509702.	355628.	232130.	156597.	81896.	53575.
76	4 2 2 0 2 2 0 0	1045028.	777192.	508997.	355020.	232130.	156597.	81896.	53575.
77	4 4 2 0 2 2 0 0	1044620.	776817.	508798.	354929.	232295.	156708.	81955.	53613.
78	4 2 4 0 2 2 0 0	1044965.	777137.	508977.	354999.	232295.	156708.	81955.	53613.
79	2 4 4 0 2 2 0 0	1050115.	779404.	509683.	355607.	232295.	156708.	81955.	53613.
80	4 4 4 0 2 2 0 0	1044557.	776762.	508778.	354908.	232419.	156792.	81998.	53641.
81	2 2 2 0 2 0 2 0	1050548.	779788.	509891.	355716.	231899.	156441.	81815.	53521.
82	2 2 4 0 2 0 2 0	1050485.	779733.	509871.	355695.	232130.	156597.	81896.	53575.
83	2 4 2 0 2 0 2 0	1050140.	779413.	509691.	355625.	232130.	156597.	81896.	53575.
84	4 2 2 0 2 0 2 0	1044990.	777146.	508986.	355016.	232130.	156597.	81896.	53575.
85	4 4 2 0 2 0 2 0	1044582.	776771.	508787.	354925.	232295.	156708.	81955.	53613.
86	4 2 4 0 2 0 2 0	1044926.	777091.	508966.	354996.	232295.	156708.	81955.	53613.
87	2 4 4 0 2 0 2 0	1050077.	779358.	509672.	355604.	232295.	156708.	81955.	53613.
88	4 4 4 0 2 0 2 0	1044518.	776716.	508767.	354904.	232419.	156792.	81998.	53641.
89	2 2 2 0 2 0 0 2	1050581.	779836.	509908.	355725.	231899.	156441.	81815.	53521.
90	2 2 4 0 2 0 0 2	1050524.	779787.	509890.	355706.	232130.	156597.	81896.	53575.
91	2 4 2 0 2 0 0 2	1050173.	779461.	505709.	355634.	232130.	156597.	81896.	53575.
92	4 2 2 0 2 0 0 2	1045022.	777193.	509003.	355026.	232130.	156597.	81896.	53575.
93	4 4 2 0 2 0 0 2	1044614.	776818.	508804.	354934.	232295.	156708.	81955.	53613.
94	4 2 4 0 2 0 0 2	1044966.	777144.	508985.	355006.	232295.	156708.	81955.	53613.
95	2 4 4 0 2 0 0 2	1050116.	779412.	509691.	355615.	232295.	156708.	81955.	53613.
96	4 4 4 0 2 0 0 2	1044558.	776769.	508786.	354915.	232419.	156792.	81998.	53641.
97	2 2 2 0 0 2 2 0	1056756.	784332.	513694.	358270.	231899.	156441.	81815.	53521.
98	2 2 4 0 0 2 2 0	1056748.	784327.	513692.	358267.	232130.	156597.	81896.	53575.
99	2 4 2 0 0 2 2 0	1056348.	783957.	513495.	358179.	232130.	156597.	81896.	53575.
100	4 2 2 0 0 2 2 0	1051197.	781689.	512790.	357570.	232130.	156597.	81896.	53575.

TABLE XIV (Continued)

101	4 4 2 0 0 2 2 0	1050789.	781314.	512590.	357479.	232295.	156708.	81955.	53613.
102	4 2 4 0 0 2 2 0	1051188.	781685.	512787.	357568.	232295.	156708.	81955.	53613.
103	2 4 4 C 0 2 2 0	1056340.	783952.	513493.	358176.	232295.	156708.	81955.	53613.
104	4 4 4 0 0 2 2 0	1050780.	781309.	512588.	357477.	232419.	156792.	81998.	53641.
105	2 2 2 0 0 2 0 2	1056779.	784368.	513704.	358275.	231899.	156441.	81815.	53521.
106	2 2 4 0 0 2 0 2	1056770.	784364.	513701.	358273.	232130.	156597.	81896.	53575.
107	2 4 2 0 0 2 0 2	1056371.	783993.	513505.	358184.	232130.	156597.	81896.	53575.
108	4 2 2 0 0 2 0 2	1051219.	781726.	512799.	357575.	232130.	156597.	81896.	53575.
109	4 4 2 C 0 2 0 2	1050811.	781351.	512600.	357484.	232295.	156708.	81955.	53613.
110	4 2 4 0 0 2 0 2	1051211.	781721.	512797.	357573.	232295.	156708.	81955.	53613.
111	2 4 4 0 0 2 0 2	1056362.	783988.	513502.	358181.	232295.	156708.	81955.	53613.
112	4 4 4 0 0 2 0 2	1050803.	781346.	512597.	357482.	232419.	156792.	81998.	53641.
113	2 2 2 0 0 0 2 2	1053682.	782252.	512154.	357097.	231899.	156441.	81815.	53521.
114	2 2 4 0 0 0 2 2	1053620.	782205.	512139.	357086.	232130.	156597.	81896.	53575.
115	2 4 2 C 0 0 2 2	1053274.	781876.	511955.	357006.	232130.	156597.	81896.	53575.
116	4 2 2 0 0 0 2 2	1048123.	7796C9.	511249.	356398.	232130.	156597.	81896.	53575.
117	4 4 2 0 0 0 2 2	1047715.	779234.	511050.	356307.	232295.	156708.	81955.	53613.
118	4 2 4 0 0 0 2 2	1048061.	779562.	511234.	356387.	232295.	156708.	81955.	53613.
119	2 4 4 0 0 0 2 2	1053212.	781830.	511940.	356995.	232295.	156708.	81955.	53613.
120	4 4 4 0 0 0 2 2	1047653.	779187.	511035.	356295.	232419.	156792.	81998.	53641.
121	2 2 2 2 2 2 0 0	1048905.	778528.	508955.	355141.	232130.	156597.	81896.	53575.
122	2 2 4 2 2 2 0 0	1048061.	777872.	508692.	354906.	232295.	156708.	81955.	53613.
123	2 4 2 2 2 2 0 0	1048498.	778153.	508756.	355050.	232295.	156708.	81955.	53613.
124	4 2 2 2 2 2 0 0	1043346.	775886.	508050.	354442.	232295.	156708.	81955.	53613.
125	2 2 2 4 2 2 0 0	1048579.	778337.	508874.	355050.	232295.	156708.	81955.	53613.
126	2 2 4 4 2 2 0 0	1047735.	777681.	508611.	354815.	232419.	156792.	81998.	53641.
127	2 4 2 4 2 2 0 0	1048171.	777962.	508675.	354959.	232419.	156792.	81998.	53641.
128	4 2 2 4 2 2 0 0	1043020.	775695.	507969.	354351.	232419.	156792.	81998.	53641.
129	2 2 2 2 2 0 2 0	1053012.	781679.	511192.	356638.	232130.	156597.	81896.	53575.
130	2 2 4 2 2 0 2 C	1052167.	781023.	510928.	356403.	232295.	156708.	81955.	53613.
131	2 4 2 2 2 0 2 0	1052604.	781304.	510993.	356547.	232295.	156708.	81955.	53613.
132	4 2 2 2 2 0 2 0	1047454.	779037.	510287.	355939.	232295.	156708.	81955.	53613.
133	2 2 2 4 2 0 2 0	1052685.	781488.	511111.	356547.	232295.	156708.	81955.	53613.
134	2 2 4 4 2 0 2 0	1051840.	780832.	510848.	356312.	232419.	156792.	81998.	53641.
135	2 4 2 4 2 0 2 0	1052277.	781113.	510912.	356456.	232419.	156792.	81998.	53641.
136	4 2 2 4 2 0 2 0	1047128.	778846.	510206.	355847.	232419.	156792.	81998.	53641.
137	2 2 2 2 2 0 0 2	1048891.	776521.	508957.	355144.	232130.	156597.	81896.	53575.
138	2 2 4 2 2 0 0 2	1048047.	777865.	508693.	354909.	232295.	156708.	81955.	53613.
139	2 4 2 2 2 0 0 2	1048484.	778146.	508758.	355053.	232295.	156708.	81955.	53613.
140	4 2 2 2 2 0 0 2	1043333.	775879.	508052.	354445.	232295.	156708.	81955.	53613.
141	2 2 2 4 2 0 0 2	1048566.	778330.	508876.	355053.	232295.	156708.	81955.	53613.
142	2 2 4 4 2 0 0 2	1047721.	777674.	508613.	354818.	232419.	156792.	81998.	53641.
143	2 4 2 4 2 0 0 2	1048158.	777955.	508677.	354962.	232419.	156792.	81998.	53641.
144	4 2 2 4 2 0 0 2	1043006.	775688.	507971.	354354.	232419.	156792.	81998.	53641.
145	2 2 2 2 0 2 2 0	1048886.	778505.	508960.	355157.	232130.	156597.	81896.	53575.
146	2 2 4 2 0 2 2 0	1048042.	777849.	508697.	354922.	232295.	156708.	81955.	53613.
147	2 4 2 2 0 2 2 0	1048479.	778130.	508761.	355066.	232295.	156708.	81955.	53613.
148	4 2 2 2 0 2 2 0	1043327.	775863.	508056.	354457.	232295.	156708.	81955.	53613.
149	2 2 2 4 0 2 2 0	1048560.	778315.	508880.	355066.	232295.	156708.	81955.	53613.
150	2 2 4 4 0 2 2 0	1047715.	777658.	508616.	354831.	232419.	156792.	81998.	53641.
151	2 4 2 4 0 2 2 0	1048152.	777939.	508680.	354975.	232419.	156792.	81998.	53641.
152	4 2 2 4 0 2 2 0	1043001.	775672.	507975.	354366.	232419.	156792.	81998.	53641.
153	2 2 2 2 0 2 0 2	1048908.	778542.	508970.	355162.	232130.	156597.	81896.	53575.
154	2 2 4 2 0 2 0 2	1048064.	777886.	508706.	354927.	232295.	156708.	81955.	53613.
155	2 4 2 2 0 2 0 2	1048501.	778167.	508771.	355071.	232295.	156708.	81955.	53613.
156	4 2 2 2 0 2 0 2	1043349.	775900.	508065.	354463.	232295.	156708.	81955.	53613.
157	2 2 2 4 0 2 C 2	1048582.	778351.	508889.	355071.	232295.	156708.	81955.	53613.
158	2 2 4 4 0 2 0 2	1047738.	777695.	508626.	354836.	232419.	156792.	81998.	53641.
159	2 4 2 4 C 2 0 2	1048174.	777976.	508690.	354580.	232419.	156792.	81998.	53641.
160	4 2 2 4 0 2 C 2	1043023.	7757C9.	507984.	354371.	232419.	156792.	81998.	53641.
161	2 2 2 2 0 0 2 2	1048869.	778496.	508959.	355159.	232130.	156597.	81896.	53575.
162	2 2 4 2 0 0 2 2	1048025.	777839.	508695.	354923.	232295.	156708.	81955.	53613.
163	2 4 2 2 0 0 2 2	1048462.	778120.	508760.	355067.	232295.	156708.	81955.	53613.
164	4 2 2 2 0 0 2 2	1043311.	775853.	508054.	354459.	232295.	156708.	81955.	53613.
165	2 2 2 4 0 0 2 2	1048544.	778305.	508878.	355067.	232295.	156708.	81955.	53613.
166	2 2 4 4 0 0 2 2	1047699.	777648.	508615.	354832.	232419.	156792.	81998.	53641.
167	2 4 2 4 0 0 2 2	1048136.	777930.	508679.	354976.	232419.	156792.	81998.	53641.
168	4 2 2 4 0 0 2 2	1042984.	775662.	507973.	354368.	232419.	156792.	81998.	53641.
169	2 2 2 0 2 2 2 0	1050516.	779771.	509873.	355705.	232130.	156597.	81896.	53575.
170	2 2 4 0 2 2 2 0	1050453.	779715.	509853.	355684.	232295.	156708.	81955.	53613.
171	2 4 2 0 2 2 2 0	1050108.	779395.	509674.	355614.	232295.	156708.	81955.	53613.
172	4 2 2 0 2 2 2 0	1044958.	777128.	508968.	355006.	232295.	156708.	81955.	53613.
173	2 2 2 0 2 2 C 2	1050549.	779818.	509890.	355715.	232130.	156597.	81896.	53575.
174	2 2 4 0 2 2 0 2	1050492.	779769.	5C9872.	355695.	232295.	156708.	81955.	53613.
175	2 4 2 0 2 2 0 2	1050141.	779443.	509691.	355623.	232295.	156708.	81955.	53613.
176	4 2 2 0 2 2 C 2	1044991.	777176.	508985.	355015.	232295.	156708.	81955.	53613.
177	2 2 2 0 0 2 2 2	1056709.	784304.	513675.	358261.	232130.	156597.	81896.	53575.
178	2 2 4 0 0 2 2 2	1056700.	784299.	513673.	358258.	232295.	156708.	81955.	53613.
179	2 4 2 0 0 2 2 2	1056301.	783929.	513476.	358170.	232295.	156708.	81955.	53613.
180	4 2 2 0 0 2 2 2	1051149.	781662.	512770.	357561.	232295.	156708.	81955.	53613.
181	2 2 2 2 2 2 2 0	1048835.	778464.	508926.	355127.	232295.	156708.	81955.	53613.
182	2 2 4 2 2 2 2 0	1047991.	7778C8.	508663.	354892.	232419.	156792.	81998.	53641.
183	2 4 2 2 2 2 2 0	1048428.	778089.	508727.	355036.	232419.	156792.	81998.	53641.
184	4 2 2 2 2 2 2 0	1043276.	775822.	508021.	354428.	232419.	156792.	81998.	53641.
185	2 2 2 4 2 2 2 0	1048510.	778273.	508845.	355036.	232419.	156792.	81998.	53641.
186	2 2 4 4 2 2 2 0	1047665.	777617.	508582.	354801.	232515.	156857.	82032.	53663.
187	2 4 2 4 2 2 2 0	1048102.	777898.	508646.	354945.	232515.	156857.	82032.	53663.
188	4 2 2 4 2 2 2 0	1042950.	775631.	507941.	354336.	232515.	156857.	82032.	53663.
189	2 2 2 0 2 2 2 2	1050478.	779754.	509861.	355700.	232295.	156708.	81955.	53613.
190	2 2 4 0 2 2 2 2	1050422.	779705.	509843.	355681.	232419.	156792.	81998.	53641.
191	2 4 2 C 2 2 2 2	1050070.	779379.	509662.	355609.	232419.	156792.	81998.	53641.
192	4 2 2 0 2 2 2 2	1044921.	777112.	508957.	355001.	232419.	156792.	81998.	53641.
193	2 2 2 2 2 2 2 2	1048790.	778439.	5C8910.	355120.	232419.	156792.	81998.	53641.
194	2 2 4 2 2 2 2 2	1047946.	777783.	508647.	354884.	232515.	156857.	82032.	53663.
195	2 4 2 2 2 2 2 2	1048383.	778064.	508711.	355028.	232515.	156857.	82032.	53663.
196	4 2 2 2 2 2 2 2	1043232.	775797.	508005.	354420.	232515.	156857.	82032.	53663.
197	2 2 2 4 2 2 2 2	1048465.	778249.	508829.	355028.	232515.	156857.	82032.	53663.
198	2 2 4 4 2 2 2 2	1047620.	777592.	508566.	354793.	232592.	156909.	82059.	53681.
199	2 4 2 4 2 2 2 2	1048057.	777873.	508630.	354937.	232592.	156909.	82059.	53681.
200	4 2 2 4 2 2 2 2	1042905.	775606.	507924.	354329.	232592.	156909.	82059.	53681.

TABLE XV

FEASIBLE DECISIONS AND CONSTRUCTION COSTS

DNO	DEC	CC
1	0 0 0 3 0 0 0 0	273.
2	0 C 3 0 0 0 0 0	255.
3	0 3 0 0 0 0 0 0	310.
4	3 0 0 0 0 0 0 0	267.
5	C 0 3 3 0 0 0 0	528.
6	0 3 0 3 C 0 0 0	583.
7	3 0 0 3 C 0 0 0	540.
8	0 3 3 0 C C 0 0	565.
9	3 0 3 0 C 0 0 0	522.
10	0 3 3 3 0 0 0 0	838.
11	3 0 3 3 C 0 0 0	795.
12	0 0 0 0 0 0 0 0	0.
13	0 0 0 3 0 0 0 2	632.
14	0 0 3 0 0 0 0 2	614.
15	0 3 0 0 0 0 0 2	669.
16	3 0 0 0 C 0 0 2	626.
17	0 0 3 3 0 0 0 2	887.
18	0 3 0 3 0 0 0 2	942.
19	3 0 0 3 0 0 0 2	899.
20	0 3 3 0 0 0 0 2	924.
21	3 0 3 0 0 0 0 2	881.
22	0 3 3 3 C 0 0 2	1197.
23	3 0 3 3 0 0 0 2	1154.
24	0 C 0 0 0 0 0 2	359.
25	C 0 0 3 0 0 2 0	770.
26	0 0 3 0 0 0 2 0	752.
27	C 3 0 0 0 0 2 0	807.
28	3 0 0 C C 0 2 0	764.
29	0 0 3 3 0 0 2 0	1025.
3C	C 3 0 3 0 0 2 0	1080.
31	3 0 0 3 0 0 2 0	1037.
32	0 3 3 0 0 0 2 0	1062.
33	3 0 3 C 0 C 2 0	1019.
34	0 3 3 3 0 0 2 0	1335.
35	3 0 3 3 0 0 2 C	1292.
36	0 0 0 0 0 0 2 0	497.
37	0 0 0 3 0 2 0 0	956.
38	0 C 3 0 0 2 0 0	938.
39	C 3 0 0 C 2 0 0	993.
40	3 0 C 0 0 2 0 0	950.
41	0 0 3 3 0 2 0 0	1211.
42	0 3 0 3 C 2 0 0	1266.
43	3 0 0 3 C 2 0 0	1223.
44	0 3 3 0 C 2 0 0	1248.
45	3 0 3 0 C 2 0 0	1205.
46	0 3 3 3 C 2 0 0	1521.
47	3 0 3 3 C 2 0 C	1478.
48	C 0 0 0 0 2 0 0	683.
49	C 0 0 3 C 0 2 2	1129.
50	0 0 3 0 0 0 2 2	1111.
51	C 3 0 0 0 0 2 2	1166.
52	3 0 0 C 0 0 2 2	1123.
53	0 0 3 3 0 0 2 2	1384.
54	C 3 0 3 0 0 2 2	1439.
55	3 0 0 3 C 0 2 2	1396.
56	0 3 3 0 0 0 2 2	1421.
57	3 C 3 0 0 0 2 2	1378.
58	0 3 3 3 0 0 2 2	1694.
59	3 0 3 3 0 0 2 2	1651.
60	C 0 0 0 0 0 2 2	856.
61	0 0 0 3 C 2 0 2	1315.
62	0 0 3 0 0 2 0 2	1297.
63	0 3 0 0 C 2 0 2	1352.
64	3 0 0 0 0 2 0 2	1309.
65	0 0 3 3 0 2 0 2	1570.
66	0 3 0 3 C 2 0 2	1625.
67	3 0 0 3 C 2 0 2	1582.
68	0 3 3 0 0 2 0 2	1607.
69	3 0 3 0 C 2 0 2	1564.
70	0 3 3 3 C 2 0 2	1880.
71	3 0 3 3 0 2 0 2	1837.
72	0 0 0 0 0 2 0 2	1042.
73	0 0 0 3 2 0 0 0	604.
74	0 C 3 0 2 0 0 0	586.
75	0 3 0 0 2 0 0 0	641.
76	3 C 0 0 2 0 0 0	598.
77	0 0 3 3 2 0 0 0	859.
78	C 3 0 3 2 0 0 0	914.
79	3 0 0 3 2 0 0 0	871.
80	0 3 3 0 2 0 0 0	896.
81	3 0 3 0 2 0 0 0	853.
82	0 3 3 3 2 0 0 0	1169.
83	3 0 3 3 2 C 0 0	1126.
84	C 0 C C 2 0 0 0	331.
85	C C C 3 2 0 0 2	963.
86	0 0 3 0 2 0 0 2	945.
87	C 3 C 0 2 0 0 2	1000.
88	3 0 0 C 2 0 0 2	957.
89	0 0 3 3 2 0 0 2	1218.
90	0 3 C 3 2 0 0 2	1273.
91	3 0 C 3 2 0 0 2	1230.
92	0 3 3 0 2 0 0 2	1255.
93	3 C 3 0 2 C 0 2	1212.
94	0 3 3 3 2 0 0 2	1528.
95	3 0 3 3 2 0 0 2	1485.
96	0 0 C 0 2 0 0 2	690.
97	0 0 0 3 0 2 2 0	1453.
98	0 0 3 0 0 2 2 0	1435.
99	C 3 0 0 0 2 2 0	1490.
100	3 0 0 0 0 2 2 0	1447.
101	0 0 3 3 0 2 2 0	17C8.
102	0 3 C 3 0 2 2 0	1763.
103	3 0 C 3 0 2 2 0	1720.
104	0 3 3 0 0 2 2 0	1745.
105	3 0 3 0 0 2 2 0	1702.
106	0 3 3 3 0 2 2 0	2018.
107	3 0 3 3 0 2 2 0	1975.
108	0 0 0 0 0 2 2 0	1180.
1C9	0 0 0 3 2 0 2 0	1101.
110	0 0 3 0 2 C 2 0	1083.
111	0 3 0 C 2 0 2 0	1138.
112	3 0 C C 2 C 2 0	1095.
113	0 0 3 3 2 0 2 0	1356.
114	0 3 0 3 2 0 2 0	1411.
115	3 0 C 3 2 0 2 0	1368.
116	0 3 3 0 2 0 2 0	1393.
117	3 0 3 0 2 0 2 0	1350.
118	0 3 3 3 2 0 2 0	1666.
119	3 0 3 3 2 0 2 0	1623.
120	0 0 0 C 2 0 2 0	828.
121	0 0 C 3 2 2 0 0	1287.
122	0 0 3 C 2 2 0 0	1269.
123	0 3 C 0 2 2 0 0	1324.
124	3 0 0 0 2 2 0 0	1281.
125	0 0 3 3 2 2 0 0	1542.
126	0 3 C 3 2 2 0 0	1597.
127	3 0 0 3 2 2 0 0	1554.
128	0 3 3 0 2 2 0 0	1579.
129	3 0 3 0 2 2 0 0	1536.
130	0 3 3 3 2 2 0 0	1852.
131	3 0 3 3 2 2 0 0	1809.
132	0 0 0 0 2 2 0 0	1014.
133	0 0 C 3 0 2 2 2	1812.
134	0 0 3 0 0 2 2 2	1794.
135	0 3 C 0 0 2 2 2	1849.
136	3 0 C 0 0 2 2 2	18C6.
137	0 0 3 3 0 2 2 2	2067.
138	0 3 C 3 0 2 2 2	2122.
139	3 0 C 3 0 2 2 2	2079.
140	0 3 3 0 0 2 2 2	2104.
141	3 0 3 0 0 2 2 2	2061.
142	0 3 3 3 0 2 2 2	2377.
143	3 0 3 3 0 2 2 2	2334.
144	0 0 C 0 0 2 2 2	1539.
145	0 0 C 3 2 0 2 2	1460.
146	0 C 3 0 2 0 2 2	1442.
147	C 3 C 0 2 0 2 2	1497.
148	3 0 0 C 2 0 2 2	1454.
149	0 0 3 3 2 0 2 2	1715.
150	0 3 0 3 2 0 2 2	1770.
151	3 0 0 3 2 0 2 2	1727.
152	0 3 3 0 2 0 2 2	1752.
153	3 0 3 C 2 0 2 2	1709.
154	0 3 3 3 2 0 2 2	2025.
155	3 0 3 3 2 0 2 2	1982.
156	0 0 0 0 2 0 2 2	1187.
157	0 0 0 3 2 2 2 0	1784.
158	0 0 3 0 2 2 2 0	1766.
159	0 3 0 0 2 2 2 0	1821.
160	3 0 C 0 2 2 2 0	1778.
161	0 0 3 3 2 2 2 0	2039.
162	0 3 C 3 2 2 2 0	2094.
163	3 0 C 3 2 2 2 0	2051.
164	0 3 3 0 2 2 2 0	2076.
165	3 0 3 C 2 2 2 0	2C33.
166	0 3 3 3 2 2 2 0	2349.
167	3 0 3 3 2 2 2 0	2306.
168	0 0 C 0 2 2 2 0	1511.
169	0 0 0 3 2 2 2 2	2143.
17C	0 0 3 0 2 2 2 2	2125.
171	0 3 0 0 2 2 2 2	2180.
172	3 0 C C 2 2 2 2	2137.
173	0 0 3 3 2 2 2 2	2398.
174	0 3 0 3 2 2 2 2	2453.
175	3 0 0 3 2 2 2 2	2410.
176	0 3 3 0 2 2 2 2	2435.
177	3 0 3 0 2 2 2 2	2392.
178	0 3 3 3 2 2 2 2	2708.
179	3 0 3 3 2 2 2 2	2665.
180	0 0 0 0 2 2 2 2	1870.
181	0 0 0 2 2 0 0 0	634.
182	0 0 0 2 0 2 0 0	986.
183	0 0 0 2 0 0 2 0	800.
184	0 0 0 2 0 0 0 2	662.
185	0 0 0 0 2 2 0 0	1014.
186	0 0 0 0 2 0 2 0	828.
187	0 0 0 0 2 0 0 2	690.
188	0 0 0 0 0 2 2 0	1180.
189	0 0 0 0 0 2 0 2	1042.
190	0 0 0 0 0 0 2 2	856.
191	0 0 0 2 2 2 0 0	1317.
192	0 0 0 2 2 0 2 0	1131.
193	0 0 0 2 2 0 0 2	993.
194	0 0 0 2 0 2 2 0	1483.
195	0 0 0 2 0 2 0 2	1345.
196	0 0 0 2 0 0 2 2	1159.
197	C 0 0 0 2 2 2 0	1511.
198	0 0 0 0 2 2 0 2	1373.
199	0 0 0 0 2 0 2 2	1187.
200	0 0 0 0 0 2 2 2	1539.
201	0 0 0 4 2 0 0 0	876.
202	0 0 C 4 0 2 0 0	1228.
203	0 0 0 4 0 0 2 0	1042.
204	0 0 0 4 0 0 0 2	904.
205	C 0 0 4 2 2 0 0	1559.
206	0 0 0 4 2 0 2 0	1373.
2C7	0 0 0 4 2 0 0 2	1235.
208	0 0 0 4 0 2 2 0	1725.
209	0 0 0 4 0 2 0 2	1587.
210	0 0 0 4 0 0 2 2	14Cl.
211	0 0 3 2 2 0 0 0	889.
212	0 0 3 2 0 2 0 0	1241.
213	0 0 3 2 0 0 2 0	1055.
214	0 0 3 2 0 0 0 2	917.
215	0 0 3 2 2 2 0 0	1572.
216	0 0 3 2 2 0 2 0	1386.
217	0 0 3 2 0 2 2 0	1738.
218	0 0 3 2 0 2 0 2	1600.
219	0 0 3 2 0 0 2 2	1414.
220	0 0 3 4 2 0 0 0	1131.
221	0 0 3 4 0 2 0 0	1483.
222	0 0 3 4 0 0 2 0	1297.
223	0 0 3 4 0 0 0 2	1159.
224	0 0 3 4 2 2 0 0	1814.
225	0 0 3 4 2 0 2 0	1628.
226	0 0 3 4 0 2 2 0	1980.
227	0 0 3 4 0 2 0 2	1842.
228	0 0 3 4 0 0 2 2	1656.
229	0 3 0 2 2 0 0 0	944.
230	0 3 0 2 0 2 0 0	1296.
231	0 3 0 2 0 0 2 0	1110.
232	0 3 C 2 0 0 0 2	972.
233	0 3 0 0 2 2 0 0	1324.
234	0 3 0 0 2 0 2 0	1138.
235	0 3 0 0 2 0 0 2	1000.
236	0 3 0 0 0 2 2 0	1490.
237	0 3 0 0 0 2 0 2	1352.
238	0 3 0 0 0 0 2 2	1166.
239	0 3 0 2 2 2 0 0	1627.
24C	0 3 0 2 2 0 2 0	1441.
241	0 3 0 2 2 0 0 2	1303.
242	0 3 C 2 0 2 2 0	1793.
243	0 3 0 2 0 2 0 2	1655.
244	0 3 0 2 0 0 2 2	1469.
245	0 3 0 C 2 2 2 0	1821.
246	0 3 0 0 2 2 0 2	1683.
247	0 3 0 0 C 2 2 2	1949.
248	C 3 0 4 2 0 0 0	1186.
249	0 3 0 4 0 2 0 0	1538.
250	0 3 0 4 0 0 2 C	1352.
251	C 3 0 4 0 0 C 2	1214.
252	C 3 0 4 2 2 0 0	1869.
253	0 3 0 4 2 0 2 0	1683.
254	0 3 0 4 2 0 0 2	1545.
255	0 3 0 4 0 2 2 0	2035.
256	0 3 0 4 0 2 0 2	1897.
257	C 3 0 4 0 0 2 2	1711.
258	3 0 0 2 2 0 0 0	901.
259	3 0 0 2 0 2 0 C	1253.
260	3 0 0 2 0 0 2 0	1067.
261	3 0 0 2 0 0 0 2	929.
262	3 0 0 0 2 2 0 0	1281.
263	3 0 0 0 2 0 2 0	1095.
264	3 0 0 0 2 0 0 2	957.
265	3 C 0 0 0 2 2 0	1447.
266	3 0 0 0 0 2 0 2	1309.
267	3 0 0 0 0 0 2 2	1123.
268	3 0 0 2 2 2 0 0	1584.
269	3 0 0 2 2 0 2 0	1398.
270	3 0 0 2 2 0 0 2	1260.
271	3 0 0 2 0 2 2 0	1750.
272	3 0 0 2 0 2 0 2	1612.
273	3 0 0 2 0 0 2 2	1426.
274	3 0 0 0 2 2 2 0	1778.
275	3 0 0 0 2 2 0 2	1640.
276	3 0 0 0 0 2 2 2	1806.
277	3 0 0 4 2 0 0 0	1143.
278	3 C 0 4 C 2 0 0	1495.
279	3 0 0 4 0 0 2 0	1309.
280	3 0 0 4 0 0 0 2	1171.
281	3 0 0 4 2 2 0 0	1826.
282	3 0 0 4 2 0 2 0	1640.
283	3 0 0 4 2 0 0 2	1502.
284	3 0 0 4 0 2 2 0	1992.
285	3 0 0 4 0 2 0 2	1854.
286	3 0 C 4 0 0 2 2	1668.
287	0 3 3 2 2 0 0 0	1199.
288	0 3 3 2 0 2 0 0	1551.
289	0 3 3 2 0 0 2 0	1365.
290	0 3 3 2 0 0 0 2	1227.
291	0 3 3 2 2 2 0 0	1882.
292	0 3 3 2 2 0 2 0	1696.
293	0 3 3 2 2 C 0 2	1558.
294	0 3 3 2 0 2 2 C	2048.
295	0 3 3 2 0 2 C 2	1910.
296	0 3 3 2 C 0 2 2	1724.
297	0 3 3 4 2 0 0 0	1441.
298	0 3 3 4 0 2 0 0	1793.
299	0 3 3 4 0 0 2 0	1607.
300	0 3 3 4 0 0 C 2	1469.
301	0 3 3 4 2 2 0 0	2124.
302	0 3 3 4 2 0 2 0	1938.
303	0 3 3 4 2 0 0 2	1800.
304	0 3 3 4 0 2 2 0	2290.
305	0 3 3 4 0 2 0 2	2152.
306	0 3 3 4 0 0 2 2	1966.
307	3 0 3 2 2 0 C 0	1156.
308	3 0 3 2 0 2 C C	1508.
309	3 0 3 2 0 C 2 0	1322.
310	3 0 3 2 0 0 0 2	1184.
311	3 0 3 2 2 2 0 0	1839.
312	3 0 3 2 2 C 2 0	1653.
313	3 0 3 2 2 0 0 2	1515.
314	3 C 3 2 0 2 2 C	2005.
315	3 0 3 2 0 2 0 2	1867.
316	3 0 3 2 0 0 2 2	1681.
317	3 0 3 4 2 0 0 0	1398.
318	3 0 3 4 0 2 0 0	1750.
319	3 C 3 4 0 0 2 0	1564.
320	3 0 3 4 0 0 0 2	1426.
321	3 0 3 4 2 2 0 0	2081.
322	3 0 3 4 2 0 2 0	1895.
323	3 0 3 4 2 0 0 2	1757.
324	3 0 3 4 0 2 2 0	2247.
325	3 0 3 4 0 2 0 2	2109.
326	3 0 3 4 0 0 2 2	1923.

computed by the DP program are shown in Table XVI. Table XVII then illustrates the output of tracing the optimal decisions through the four stages and the resulting new network state at each stage. Stage number two has an optimal decision of 03322000 which corresponds to decision number 287. As indicated in Table XVIII, there are more optimal decisions in this stage. This implies there exists alternative optimal decisions for this stage. To find this alternative decision, one considers the input state to stage two which is the new state, 42200000, in stage three as shown in Table XVII. Tracing this input state in Table XVI, one finds this corresponds to network number four. Tracing network number four to the second period decisions, one finds optimal decision numbers of 287 and 8. Therefore, decision number 8, which is 03300000 as found in Table XV, provides the same cost as decision number 287, which is 03322000. This implies the network operators' cost savings from the construction of the two-lane roads built by decision number 287 equals the cost for constructing the two two-lane roads. Since one has the same total costs for either decision, one would surely choose decision 287 since it would provide two additional two-lane roads at no extra cost.

The optimal decisions in Table XVII dictate the following construction policies: 1) in 1970, the optimal decision D_4^* = 30000000 corresponds to the conversion of the two-lane road existing between Tulsa and Bartlesville (nodes 2-5) to a four-lane road. After a five year delay for construction lag time, one now has a new network state of 42200000; 2) the optimal decision in stage three is D_3^* = 00000000 which implies one has no constructions during this 1975-1980 period; 3) but in 1980, D_2^* = 03322000 which implies one converts the two-lane roads between nodes

TABLE XVI

OPTIMAL STAGE COSTS AND CORRESPONDING OPTIMAL DECISIONS

NN	STATE	TOTAL PERIOD COSTS(1,2,3,4)				PERIOD (1,2,3,4) DECISIONS---COL 1 = DNO, COL 2&3=ALT DNO											
1	2 2 2 0 0 0 0 0	975444.	1405239.	1446243.	1337571.	307	0	0	9	0	0	4	0	0	4	0	0
2	2 2 4 0 0 0 0 0	975767.	1405373.	1446244.		258	0	0	4	0	0	4	0	0			
3	2 6 2 0 0 0 0 0	975731.	1405316.	1446247.		307	0	0	307	0	0	9	0	0			
4	4 2 2 0 0 0 0 0	975755.	1405359.	1446180.		211	0	0	287	8	0	12	0	0			
5	4 4 2 0 0 0 0 0	975810.	1405283.	1446103.		211	0	0	211	0	0	2	0	0			
6	4 2 4 0 0 0 0 0	975846.	1405338.	1446099.		181	0	0	229	12	0	12	0	0			
7	2 4 4 0 0 0 0 0	975822.	1405295.	1446115.		258	0	0	258	0	0	4	0	0			
8	4 4 4 0 0 0 0 0	975786.	1405184.	1445929.		181	0	0	181	0	0	12	0	0			
9	2 2 2 2 2 0 0 0	975734.	1405303.	1446124.		9	0	0	9	0	0	9	0	0			
10	2 2 4 2 2 0 0 0	975710.	1405203.	1445950.		4	0	0	4	0	0	4	0	0			
11	2 4 2 2 2 0 0 0	975674.	1405072.	1445901.		9	0	0	9	0	0	9	0	0			
12	4 2 2 2 2 0 0 0	975698.	1405115.	1445938.		2	0	0	8	0	0	2	0	0			
13	4 4 2 2 2 0 0 0	975572.	1404916.	1445692.		2	0	0	2	0	0	2	0	0			
14	4 2 4 2 2 0 0 0	975608.	1404971.	1445741.		12	0	0	3	0	0	12	0	0			
15	2 4 4 2 2 0 0 0	975584.	1404928.	1445704.		4	0	0	4	0	0	4	0	0			
16	4 4 4 2 2 0 0 0	975441.	1404744.	1445481.		12	0	0	12	0	0	12	0	0			
17	2 2 2 4 2 0 0 0	975733.	1405245.	1446041.		9	0	0	9	0	0	9	0	0			
18	2 2 4 4 2 0 0 0	975643.	1405101.	1445844.		4	0	0	4	0	0	4	0	0			
19	2 4 2 4 2 0 0 0	975607.	1404950.	1445782.		9	0	0	9	0	0	9	0	0			
20	4 2 2 4 2 0 0 0	975631.	1404953.	1445825.		2	0	0	8	0	0	8	0	0			
21	4 4 2 4 2 0 0 0	975464.	1404766.	1445559.		2	0	0	2	0	0	2	0	0			
22	4 2 4 4 2 0 0 0	975500.	1404821.	1445614.		12	0	0	3	0	0	3	0	0			
23	2 4 4 4 2 0 0 0	975476.	1404778.	1445571.		4	0	0	4	0	0	4	0	0			
24	4 4 4 4 2 0 0 0	975305.	1404576.	1445338.		12	0	0	12	0	0	12	0	0			
25	2 2 2 2 0 2 0 0	975749.	1405330.	1446155.		9	0	0	9	0	0	9	0	0			
26	2 2 4 2 0 2 0 0	975725.	1405230.	1445981.		4	0	0	4	0	0	4	0	0			
27	2 4 2 2 0 2 0 0	975689.	1405059.	1445933.		9	0	0	9	0	0	9	0	0			
28	4 2 2 2 0 2 0 0	975713.	1405142.	1445969.		2	0	0	8	0	0	2	0	0			
29	4 4 2 2 0 2 0 0	975587.	1404943.	1445724.		2	0	0	2	0	0	2	0	0			
30	4 2 4 2 0 2 0 0	975623.	1404958.	1445772.		12	0	0	3	0	0	12	0	0			
31	2 4 4 2 0 2 0 0	975599.	1404955.	1445736.		4	0	0	4	0	0	4	0	0			
32	4 4 4 2 0 2 0 0	975455.	1404772.	1445513.		12	0	0	12	0	0	12	0	0			
33	2 2 2 4 0 2 0 0	975747.	1405272.	1446071.		9	0	0	9	0	0	9	0	0			
34	2 2 4 4 0 2 0 0	975657.	1405128.	1445875.		4	0	0	4	0	0	4	0	0			
35	2 4 2 4 0 2 0 0	975621.	1404977.	1445813.		9	0	0	9	0	0	9	0	0			
36	4 2 2 4 0 2 0 0	975645.	1405020.	1445856.		2	0	0	8	0	0	8	0	0			
37	4 4 2 4 0 2 0 0	975478.	1404753.	1445590.		2	0	0	2	0	0	2	0	0			
38	4 2 4 4 0 2 0 0	975514.	1404848.	1445645.		12	0	0	3	0	0	3	0	0			
39	2 4 4 4 0 2 0 0	975490.	1404805.	1445602.		4	0	0	4	0	0	4	0	0			
40	4 4 4 4 0 2 0 0	975319.	1404603.	1445369.		12	0	0	12	0	0	12	0	0			
41	2 2 2 2 0 0 2 0	975721.	1405277.	1446109.		9	0	0	9	0	0	9	0	0			
42	2 2 4 2 0 0 2 0	975697.	1405178.	1445935.		4	0	0	4	0	0	4	0	0			
43	2 4 2 2 0 0 2 0	975661.	1405046.	1445887.		9	0	0	9	0	0	9	0	0			
44	4 2 2 2 0 0 2 0	975685.	1405089.	1445923.		2	0	0	8	0	0	2	0	0			
45	4 4 2 2 0 0 2 0	975559.	1404890.	1445678.		2	0	0	2	0	0	2	0	0			
46	4 2 4 2 0 0 2 0	975595.	1404945.	1445727.		12	0	0	3	0	0	12	0	0			
47	2 4 4 2 0 0 2 0	975571.	1404902.	1445690.		4	0	0	4	0	0	4	0	0			
48	4 4 4 2 0 0 2 0	975428.	1404719.	1445467.		12	0	0	12	0	0	12	0	0			
49	2 2 2 4 0 0 2 0	975719.	1405219.	1446027.		9	0	0	9	0	0	9	0	0			
50	2 2 4 4 0 0 2 0	975629.	1405075.	1445830.		4	0	0	4	0	0	4	0	0			
51	2 4 2 4 0 0 2 0	975593.	1404924.	1445767.		9	0	0	9	0	0	9	0	0			
52	4 2 2 4 0 0 2 0	975617.	1404967.	1445810.		2	0	0	8	0	0	8	0	0			
53	4 4 2 4 0 0 2 0	975450.	1404741.	1445544.		2	0	0	2	0	0	2	0	0			
54	4 2 4 4 0 0 2 0	975486.	1404756.	1445599.		12	0	0	3	0	0	3	0	0			
55	2 4 4 4 0 0 2 0	975462.	1404753.	1445556.		4	0	0	4	0	0	4	0	0			
56	4 4 4 4 0 0 2 0	975291.	1404550.	1445323.		12	0	0	12	0	0	12	0	0			
57	2 2 2 2 0 0 0 2	975736.	1405314.	1446142.		9	0	0	9	0	0	9	0	0			
58	2 2 4 2 0 0 0 2	975712.	1405215.	1445968.		4	0	0	4	0	0	4	0	0			
59	2 4 2 2 0 0 0 2	975676.	1405083.	1445920.		9	0	0	9	0	0	9	0	0			
60	4 2 2 2 0 0 0 2	975700.	1405126.	1445956.		2	0	0	8	0	0	2	0	0			
61	4 4 2 2 0 0 0 2	975574.	1404927.	1445712.		2	0	0	2	0	0	2	0	0			
62	4 2 4 2 0 0 0 2	975610.	1404982.	1445760.		12	0	0	3	0	0	12	0	0			
63	2 4 4 2 0 0 0 2	975586.	1404939.	1445724.		4	0	0	4	0	0	4	0	0			
64	4 4 4 2 0 0 0 2	975443.	1404756.	1445500.		12	0	0	12	0	0	12	0	0			
65	2 2 2 4 0 0 0 2	975735.	1405256.	1446060.		9	0	0	9	0	0	9	0	0			
66	2 2 4 4 0 0 0 2	975645.	1405113.	1445863.		4	0	0	4	0	0	4	0	0			
67	2 4 2 4 0 0 0 2	975609.	1404961.	1445801.		9	0	0	9	0	0	9	0	0			
68	4 2 2 4 0 0 0 2	975633.	1405004.	1445844.		2	0	0	8	0	0	8	0	0			
69	4 4 2 4 0 0 0 2	975466.	1404778.	1445578.		2	0	0	2	0	0	2	0	0			
70	4 2 4 4 0 0 0 2	975502.	1404833.	1445633.		12	0	0	3	0	0	3	0	0			
71	2 4 4 4 0 0 0 2	975478.	1404790.	1445590.		4	0	0	4	0	0	4	0	0			
72	4 4 4 4 0 0 0 2	975307.	1404588.	1445357.		12	0	0	12	0	0	12	0	0			
73	2 2 2 0 2 2 0 0	976810.	1406724.	1447889.		260	0	0	260	0	0	4	0	0			
74	2 2 4 0 2 2 0 0	977445.	1407788.	1448628.		4	0	0	4	0	0	4	0	0			
75	2 4 2 0 2 2 0 0	977199.	1407384.	1448268.		4	0	0	4	0	0	4	0	0			
76	4 2 2 0 2 2 0 0	976774.	1406613.	1447376.		183	0	0	183	0	0	183	0	0			
77	4 4 2 0 2 2 0 0	977097.	1407229.	1448059.		12	0	0	12	0	0	12	0	0			
78	4 2 4 0 2 2 0 0	977343.	1407555.	1448420.		12	0	0	3	0	0	12	0	0			
79	2 4 4 0 2 2 0 0	977319.	1407512.	1448383.		4	0	0	4	0	0	4	0	0			
80	4 4 4 0 2 2 0 0	977176.	1407329.	1448160.		12	0	0	12	0	0	12	0	0			
81	2 2 2 0 2 0 2 0	976996.	1407365.	1448014.		259	0	0	4	0	0	4	0	0			
82	2 2 4 0 2 0 2 0	977417.	1407735.	1448583.		4	0	0	4	0	0	4	0	0			
83	2 4 2 0 2 0 2 0	977172.	1407332.	1448222.		4	0	0	4	0	0	4	0	0			
84	4 2 2 0 2 0 2 0	976960.	1406799.	1447829.		182	0	0	182	0	0	12	0	0			
85	4 4 2 0 2 0 2 0	977070.	1407176.	1448014.		12	0	0	12	0	0	12	0	0			
86	4 2 4 0 2 0 2 0	977315.	1407503.	1448375.		12	0	0	3	0	0	12	0	0			
87	2 4 4 0 2 0 2 0	977291.	1407460.	1448338.		4	0	0	4	0	0	4	0	0			
88	4 4 4 0 2 0 2 0	977148.	1407276.	1448114.		12	0	0	12	0	0	12	0	0			
89	2 2 2 0 2 0 0 2	977255.	1407583.	1448473.		4	0	0	4	0	0	4	0	0			
90	2 2 4 0 2 0 0 2	977446.	1407753.	1448639.		4	0	0	4	0	0	4	0	0			
91	2 4 2 0 2 0 0 2	977195.	1407382.	1448271.		4	0	0	4	0	0	4	0	0			
92	4 2 2 0 2 0 0 2	977219.	1407425.	1448287.		12	0	0	3	0	0	12	0	0			
93	4 4 2 0 2 0 0 2	977093.	1407226.	1448062.		12	0	0	12	0	0	12	0	0			
94	4 2 4 0 2 0 0 2	977344.	1407561.	1448430.		12	0	0	3	0	0	12	0	0			
95	2 4 4 0 2 0 0 2	977320.	1407518.	1448393.		4	0	0	4	0	0	4	0	0			
96	4 4 4 0 2 0 0 2	977177.	1407334.	1448169.		12	0	0	12	0	0	12	0	0			
97	2 2 2 0 0 2 2 0	976644.	1406558.	1447905.		258	0	0	258	0	0	181	0	0			
98	2 2 4 0 0 2 2 0	979969.	1410259.	1450386.		181	0	0	181	0	0	181	0	0			
99	2 4 2 0 0 2 2 0	980281.	1410682.	1450734.		181	0	0	181	0	0	181	0	0			
100	4 2 2 0 0 2 2 0	976608.	1406447.	1447210.		181	0	0	181	0	0	181	0	0			

TABLE XVI (Continued)

101	4 4 2 0 0 2 2 0	981495.	1413571.	1455285.	12	0	0	12	0	0	12	0	0
102	4 2 4 0 0 2 2 0	981780.	1413961.	1455704.	12	0	0	3	0	0	12	0	0
103	2 4 4 0 0 2 2 0	981756.	1413918.	1455667.	4	0	0	4	0	0	4	0	0
104	4 4 4 0 0 2 2 0	981613.	1413735.	1455444.	12	0	0	12	0	0	12	0	0
105	2 2 2 0 0 2 0 2	977109.	1407680.	1448308.	269	0	0	76	0	0	76	0	0
106	2 2 4 0 0 2 0 2	980434.	1410785.	1451473.	192	0	0	192	0	0	84	0	0
107	2 4 2 0 0 2 0 2	980746.	1411207.	1451646.	192	0	0	192	0	0	84	0	0
108	4 2 2 0 0 2 0 2	977073.	1406972.	1448122.	192	0	0	192	0	0	84	0	0
109	4 4 2 0 0 2 0 2	981511.	1413608.	1455318.	12	0	0	12	0	0	12	0	0
110	4 2 4 0 0 2 0 2	981796.	1413959.	1455737.	12	0	0	3	0	0	12	0	0
111	2 4 4 0 0 2 0 2	981772.	1413956.	1455700.	4	0	0	4	0	0	4	0	0
112	4 4 4 0 0 2 0 2	981629.	1413712.	1455477.	12	0	0	12	0	0	12	0	0
113	2 2 2 0 0 0 2 2	977295.	1409335.	1450899.	268	262	0	4	0	0	4	0	0
114	2 2 4 0 0 0 2 2	979652.	1411091.	1452593.	4	0	0	4	0	0	4	0	0
115	2 4 2 0 0 0 2 2	979406.	1410681.	1452224.	4	0	0	4	0	0	4	0	0
116	4 2 2 0 0 0 2 2	977259.	1408582.	1450713.	191	185	0	132	185	0	12	0	0
117	4 4 2 0 0 0 2 2	979304.	1410525.	1452015.	12	0	0	12	0	0	12	0	0
118	4 2 4 0 0 0 2 2	979550.	1410859.	1452384.	12	0	0	3	0	0	12	0	0
119	2 4 4 0 0 0 2 2	979526.	1410816.	1452348.	4	0	0	4	0	0	4	0	0
120	4 4 4 0 0 0 2 2	979383.	1410632.	1452124.	12	0	0	12	0	0	12	0	0
121	2 2 2 2 2 2 0 0	976291.	1406073.	1446800.	4	0	0	4	0	0	4	0	0
122	2 2 4 2 2 2 0 0	979550.	1409730.	1449767.	12	0	0	12	0	0	12	0	0
123	2 4 2 2 2 2 0 0	979862.	1410152.	1450113.	12	0	0	12	0	0	12	0	0
124	4 2 2 2 2 2 0 0	976189.	1405917.	1446591.	12	0	0	12	0	0	12	0	0
125	2 2 2 4 2 2 0 0	976223.	1405970.	1446708.	4	0	0	4	0	0	4	0	0
126	2 2 4 4 2 2 0 0	979442.	1409600.	1449660.	12	0	0	12	0	0	12	0	0
127	2 4 2 4 2 2 0 0	979752.	1410022.	1450007.	12	0	0	12	0	0	12	0	0
128	4 2 2 4 2 2 0 0	976080.	1405787.	1446484.	12	0	0	12	0	0	12	0	0
129	2 2 2 2 2 0 2 0	976924.	1406763.	1448036.	40	0	0	40	0	0	48	0	0
130	2 2 4 2 2 0 2 0	980183.	1410420.	1450493.	48	0	0	48	0	0	48	0	0
131	2 4 2 2 2 0 2 0	980495.	1410842.	1450841.	48	0	0	48	0	0	48	0	0
132	4 2 2 2 2 0 2 0	976822.	1406607.	1447317.	48	0	0	48	0	0	48	0	0
133	2 2 2 4 2 0 2 0	976856.	1406641.	1447929.	40	0	0	40	0	0	48	0	0
134	2 2 4 4 2 0 2 0	980075.	1410270.	1450360.	48	0	0	48	0	0	48	0	0
135	2 4 2 4 2 0 2 0	980386.	1410653.	1450707.	48	0	0	48	0	0	48	0	0
136	4 2 2 4 2 0 2 0	976713.	1406457.	1447184.	48	0	0	48	0	0	48	0	0
137	2 2 2 2 2 0 0 2	976281.	1406061.	1446792.	4	0	0	4	0	0	4	0	0
138	2 2 4 2 2 0 0 2	979540.	1409718.	1449760.	12	0	0	12	0	0	12	0	0
139	2 4 2 2 2 0 0 2	979852.	1410140.	1450106.	12	0	0	12	0	0	12	0	0
140	4 2 2 2 2 0 0 2	976179.	1405905.	1446584.	12	0	0	12	0	0	12	0	0
141	2 2 2 4 2 0 0 2	976213.	1405958.	1446700.	4	0	0	4	0	0	4	0	0
142	2 2 4 4 2 0 0 2	979432.	1409588.	1449653.	12	0	0	12	0	0	12	0	0
143	2 4 2 4 2 0 0 2	979743.	1410010.	1450000.	12	0	0	12	0	0	12	0	0
144	4 2 2 4 2 0 0 2	976070.	1405775.	1446477.	12	0	0	12	0	0	12	0	0
145	2 2 2 2 0 2 2 0	976277.	1406047.	1446785.	4	0	0	4	0	0	4	0	0
146	2 2 4 2 0 2 2 0	979537.	1409704.	1449752.	12	0	0	12	0	0	12	0	0
147	2 4 2 2 0 2 2 0	979848.	1410127.	1450099.	12	0	0	12	0	0	12	0	0
148	4 2 2 2 0 2 2 0	976175.	1405851.	1446576.	12	0	0	12	0	0	12	0	0
149	2 2 2 4 0 2 2 0	976210.	1405945.	1446693.	4	0	0	4	0	0	4	0	0
150	2 2 4 4 0 2 2 0	979427.	1409573.	1449645.	12	0	0	12	0	0	12	0	0
151	2 4 2 4 0 2 2 0	979739.	1409956.	1449992.	12	0	0	12	0	0	12	0	0
152	4 2 2 4 0 2 2 0	976066.	1405761.	1446469.	12	0	0	12	0	0	12	0	0
153	2 2 2 2 0 2 0 2	976293.	1406084.	1446818.	4	0	0	4	0	0	4	0	0
154	2 2 4 2 0 2 0 2	979553.	1409741.	1449785.	12	0	0	12	0	0	12	0	0
155	2 4 2 2 0 2 0 2	979864.	1410164.	1450133.	12	0	0	12	0	0	12	0	0
156	4 2 2 2 0 2 0 2	976191.	1405928.	1446609.	12	0	0	12	0	0	12	0	0
157	2 2 2 4 0 2 0 2	976225.	1405982.	1446726.	4	0	0	4	0	0	4	0	0
158	2 2 4 4 0 2 0 2	979444.	1409611.	1449679.	12	0	0	12	0	0	12	0	0
159	2 4 2 4 0 2 0 2	979755.	1410033.	1450025.	12	0	0	12	0	0	12	0	0
160	4 2 2 4 0 2 0 2	976082.	1405758.	1446503.	12	0	0	12	0	0	12	0	0
161	2 2 2 2 0 0 2 2	976266.	1406032.	1446773.	4	0	0	4	0	0	4	0	0
162	2 2 4 2 0 0 2 2	979525.	1409688.	1449740.	12	0	0	12	0	0	12	0	0
163	2 4 2 2 0 0 2 2	979836.	1410111.	1450087.	12	0	0	12	0	0	12	0	0
164	4 2 2 2 0 0 2 2	976164.	1405876.	1446564.	12	0	0	12	0	0	12	0	0
165	2 2 2 4 0 0 2 2	976198.	1405929.	1446680.	4	0	0	4	0	0	4	0	0
166	2 2 4 4 0 0 2 2	979416.	1409558.	1449633.	12	0	0	12	0	0	12	0	0
167	2 4 2 4 0 0 2 2	979728.	1409981.	1449980.	12	0	0	12	0	0	12	0	0
168	4 2 2 4 0 0 2 2	976054.	1405745.	1446457.	12	0	0	12	0	0	12	0	0
169	2 2 2 0 2 2 2 0	976871.	1406765.	1447947.	261	0	0	280	0	0	4	0	0
170	2 2 4 0 2 2 2 0	980130.	1410422.	1450527.	184	0	0	204	0	0	184	0	0
171	2 4 2 0 2 2 2 0	980442.	1410845.	1450875.	184	0	0	204	0	0	184	0	0
172	4 2 2 0 2 2 2 0	976769.	1406609.	1447352.	184	0	0	204	0	0	184	0	0
173	2 2 2 0 2 2 0 2	977009.	1406908.	1448058.	260	0	0	260	0	0	4	0	0
174	2 2 4 0 2 2 0 2	980268.	1410560.	1450665.	183	0	0	203	0	0	183	0	0
175	2 4 2 0 2 2 0 2	980580.	1410983.	1451013.	183	0	0	203	0	0	183	0	0
176	4 2 2 0 2 2 0 2	976907.	1406747.	1447490.	183	0	0	203	0	0	183	0	0
177	2 2 2 0 0 2 2 2	976843.	1406737.	1448055.	258	0	0	277	0	0	181	0	0
178	2 2 4 0 0 2 2 2	980102.	1410394.	1450499.	181	0	0	201	0	0	181	0	0
179	2 4 2 0 0 2 2 2	980414.	1410817.	1450847.	181	0	0	201	0	0	181	0	0
180	4 2 2 0 0 2 2 2	976741.	1406581.	1447324.	181	0	0	201	0	0	181	0	0
181	2 2 2 2 2 2 2 0	976406.	1406151.	1446901.	4	0	0	4	0	0	4	0	0
182	2 2 4 2 2 2 2 0	979624.	1409820.	1449854.	12	0	0	12	0	0	12	0	0
183	2 4 2 2 2 2 2 0	979936.	1410243.	1450201.	12	0	0	12	0	0	12	0	0
184	4 2 2 2 2 2 2 0	976262.	1406007.	1446678.	12	0	0	12	0	0	12	0	0
185	2 2 2 4 2 2 2 0	976297.	1406041.	1446768.	4	0	0	4	0	0	4	0	0
186	2 2 4 4 2 2 2 0	979488.	1409652.	1449711.	12	0	0	12	0	0	12	0	0
187	2 4 2 4 2 2 2 0	979800.	1410075.	1450058.	12	0	0	12	0	0	12	0	0
188	4 2 2 4 2 2 2 0	976126.	1405839.	1446535.	12	0	0	12	0	0	12	0	0
189	2 2 2 0 2 2 2 2	977579.	1407947.	1448820.	4	0	0	4	0	0	4	0	0
190	2 2 4 0 2 2 2 2	981357.	1412409.	1452542.	12	0	0	12	0	0	12	0	0
191	2 4 2 0 2 2 2 2	981107.	1411957.	1452119.	12	0	0	12	0	0	12	0	0
192	4 2 2 0 2 2 2 2	977435.	1407763.	1448597.	12	0	0	12	0	0	12	0	0
193	2 2 2 2 2 2 2 2	976498.	1406303.	1447001.	4	0	0	4	0	0	4	0	0
194	2 2 4 2 2 2 2 2	979688.	1409913.	1449943.	12	0	0	12	0	0	12	0	0
195	2 4 2 2 2 2 2 2	980000.	1410336.	1450290.	12	0	0	12	0	0	12	0	0
196	4 2 2 2 2 2 2 2	976327.	1406101.	1446768.	12	0	0	12	0	0	12	0	0
197	2 2 2 4 2 2 2 2	976361.	1406121.	1446838.	4	0	0	4	0	0	4	0	0
198	2 2 4 4 2 2 2 2	979533.	1409718.	1449773.	12	0	0	12	0	0	12	0	0
199	2 4 2 4 2 2 2 2	979845.	1410141.	1450121.	12	0	0	12	0	0	12	0	0
200	4 2 2 4 2 2 2 2	976171.	1405906.	1445598.	12	0	0	12	0	0	12	0	0

TABLE XVII

OPTIMAL STAGE DECISION POLICIES

```
STAGE NUMBER =  4
        OPTIMAL DECISION = 3 0 0 0 0 0 0 0
        DECISION NUMBER =  4
        OPTIMAL COST =  1337571.

        NEW STATE = 4 2 2 0 0 0 0 0
        STATE NUMBER =  4

STAGE NUMBER =  3
        OPTIMAL DECISION = 0 0 0 0 0 0 0 0
        DECISION NUMBER = 12
        OPTIMAL COST =  1446180.

        NEW STATE = 4 2 2 0 0 0 0 0
        STATE NUMBER =  4

STAGE NUMBER =  2
        OPTIMAL DECISION = 0 3 3 2 2 0 0 0
        DECISION NUMBER =287
        OPTIMAL COST =  1405359.
                     THERE ARE MORE OPTIMAL DECISIONS IN THIS STAGE

        NEW STATE = 4 4 4 2 2 0 0 0
        STATE NUMBER = 16

STAGE NUMBER =  1
        OPTIMAL DECISION = 0 0 0 0 0 0 0 0
        DECISION NUMBER = 12
        OPTIMAL COST =   975441.

        NEW STATE = 4 4 4 2 2 0 0 0
        STATE NUMBER = 16
```

2-50 and 9-12 to four-lane roads. In addition, one constructs two new two-lane roads interconnecting nodes 4-33 and 17-46; 4) then in 1985 the optimal decision is D_1^* = 00000000 which corresponds to the do nothing decision. This produced an output state of $\tilde{S}_1$ = 44422000. These optimal decisions provide the minimum cost construction scheme for the budget limitations shown in Table XVIII. These optimal decisions minimize the sum of operators' costs, maintenance costs, and construction costs for the total planning period of 20 years.

TABLE XVIII

PERIOD BUDGET LIMITATIONS

Year	1970	1975	1980	1985
Construction Budget Limitation	$ 500	$ 800	$1200	$2000

A change in the interest rate from seven percent to four percent resulted in the following decision policies: 1) in 1970, the optimal decision D_4^* = 30000000 corresponds to converting the link between nodes 2-5 into a four-lane road. The optimal cost was $1,896,952; 2) in 1975, one changes the input state of 42200000 by D_3^* = 00300000 to give a resulting output state of 42400000. In other words, one converts the two-lane road connecting nodes 9-12 into a four-lane road. The optimal cost at this stage is $1,888,062; 3) at stage 2 (1980), the optimal decision, D_2^*, is 03042000 with a corresponding optimal cost of $1,689,274. Thus, one would convert the road between nodes 2-50 into a four-lane, build a new four-lane road between nodes 4-33, and build a new two-lane betweeen nodes 17-46; 4) in the final stage, 1985, the optimal decision is $D_1^* = 00000000$. The optimal cost is $1,088,798 and the final network output state is 4442000. As noted in stage one in Table XVII, the final output state was 44422000; therefore, the interest rate for discounting monies does play an important role in the resulting output decisions.

Reversing the budget constraints such that period 1, 2, 3, and 4 budgets are 500, 800, 1200, and 2000 also changed the optimal decisions. While using an interest rate of seven percent the optimal stage decisions and output states were the following: 1) in 1970, $D_4^* = 30000000$ and $\tilde{S}_4 = 42200000$; 2) in stage 3, 1975, $D_3^* = 00000000$ and $\tilde{S}_3 = 42200000$; 3) while in stage 2, 1980, $D_2^* = 00300000$ and $\tilde{S}_2 = 42400000$; 4) in 1985, $D_1^* = 00000000$ and $\tilde{S}_1 = 42400000$. This compares to the corresponding output in stage one in Table XVII of 44422000.

From these brief sensitivity runs, one would conclude that the last three roads, which connect nodes 19-44, 28-37, and 42-44, will need a major perturbation before they will be constructed. Also, the sensitivity due to the interest rate appears very reasonable because the smaller the discounting rate becomes the more monies provided for investing or building new roads. This is precisely what happens in the above examples. The major changes in the budget restrictions produced reasonably optimal decisions.

In essence, these two examples serve to illustrate the application of the regional planning model presented in this dissertation. The dynamic program required only 82k storage during the execution phase of the 53-node problem. The time requirements for the IBM 360 Model 65 Central Processor Unit (CPU) are shown in Table XIX. The total costs for the 23 runs of program 1 and program 2 and one run of the DP program are given. After seeing the total cost of $189.67 for the solution of the road investment problem, which includes 200 states and 326 decisions to be considered at the stages of the DP optimization, it reveals the minimum cost requirements for the implementation of this planning process. The ability of the BPR programs to handle large networks, up to 8,170 nodes with eight connectors per node, makes the method even more attractive. The DP program could handle from 600 to 800 input states and 1000 decisions with very few restrictions on computer storage requirements. Since the BPR programs are being utilized by several statewide planning agencies, the implementation of this regional planning method would require only small efforts to incorporate the three programs, FMTLKVOL, NTOPCOST, and DP, which have already been interfaced in this research.

This regional planning method for road investments has the characteristics of being application oriented, requires minimum computational times, requires only small efforts for implementation, and has the ability to plan road investments on a regional (statewide) basis. Smaller regions within a state should be elementary for this planning method. The ability of this planning method to cope with very large networks, a large number of alternative decisions, and a large number of feasible network states, while using relatively small amounts of computer times, distinguishes this regional planning method for road investments from other planning models.

TABLE XIX

IBM 360 MODEL 65 CENTRAL PROCESSOR TIME REQUIREMENTS
FOR 53-NODE PROBLEM

	CPU Time Required (Approximate)	Approximate Costs Including Printing
Program 1	1.67 minutes (23 runs)	$ 29.90
Program 2	4.95 minutes (23 runs)	$129.80
DP Program	2.67 minutes (one run)	$ 29.97
TOTAL	9.29 minutes	$189.67

CHAPTER VI

SUMMARY AND CONCLUSIONS

6.1 Summary. A method by which regional road investments can be justified without major biasing from political inputs has been presented. The purpose of this planning method is to order the priorities for new constructions. Because priorities change with time, it is felt that reapplication of this planning method very five years would be necessary. The subsequent planning periods would start with the network obtained in the first period of the first planning process and would continue for n stages with new alternatives. The requisite for reapplication is supported by the fact that more accurate data would be available for the generation of trips for the nearest planning periods. The primary reason for limiting the total planning period to twenty years was based upon the inaccuracies of trip generation estimates beyond that length of time. This planning method minimizes an objective function formulated as the sum of operators' costs, construction costs, and maintenance costs over a twenty year planning period. The optimal investment plan provides a network that satisfies the demands placed on it by the travelers and concomitantly minimizes the total costs encompassed by the objective function.

Although the objective function is not unique, any weighted combination of the operators' costs, construction costs, and maintenance costs could be implemented with only minor changes in the calculation procedure. The computer programs have been designed to use this particular objective function; however, the basic structure of the model would not change with the addition of an entirely new objective function.

As the review of literature progressed, it became very apparent that previous studies in considering alternative road investments over multiple-time periods were not desirable in most practical applications. Previous models were limited in scope by either considering less than ten nodes or less than five investment alternatives within a given network. Another severe limitation which has been characteristic of past modeling efforts has been the lack of planning on a multiple-time period basis. These static-type models are very inappropriate since the service functions of new and old roads within a given network extends over a time frame of many years. The ability to cope with dependencies among decisions made in different time periods is a trait that should be part of any regional road investment model. In general, all previous regional road investment models having the scope and capabilities of the model presented in Chapter III, either requires an excess of computer time for computations or is not easily implemented with current techniques.

Knowledge of the deficiencies of previous modeling efforts had led to the development of an integrated formulation for planning regional road investments. This integration has transpired from previous methods for solving the traffic generation, the traffic distribution, the traffic assignment, and the limited work on the road investment problem. These techniques made possible a marriage of Bergendahl's (4) basic approach with the BPR programs to achieve a practical model which may be implemented with only minor efforts by most highway planning agencies.

The BPR programs have been generated for studying large network problems while minimizing computational times required by the computer. The BPR programs permit a network to be constructed with a maximum of 8,170 nodes while using a maximum of eight link connections on each node or 16,362 nodes while using four or less link connections on each node. The BPR programs are very efficient as noted in Chapter V and have been employed mainly because of their distinctive features for the solution of the traffic generation, the traffic distribution, and the traffic assignment problems.

The utilization of these programs has been expanded for the consideration of multiple-time periods. The BPR programs were interfaced with two computer programs to provide a network operators' cost for each specified time period. The first program, FMTLKVOL, has been designed to convert the link volumes on a given spiderweb network into a Fortran readable form and to assign a sequencing index to each link in the network. The second program, NTOPCOST, performs the computations necessary to find a network operators' cost by summing the products of link travelers' costs and link volumes over the entire set of links within a specified network. The NTOPCOST and FMTLKVOL programs were coupled with the BPR programs to ascertain the network operators' costs for the constitutent time periods in the dynamic programming formulation. These costs are essential inputs for the road investment problem. This coupling of the above programs comprises the first portion of the regional planning model developed in Chapter III.

The final part of the model makes use of dynamic programming for the solution of the road investment problem. Construction costs were postulated for every feasible decision considered for investing in new roads. Maintenance costs for each network state were assumed different for each time period contained in the dynamic program. The duration of one time period was assumed to be five years. This was intuitively pleasing when considering the time delay necessary for the construction of new intercity roads; however, any reasonable length of time may be assumed for the duration of a time period without affecting the general modeling efforts employed by this planning method. The dynamic program solution technique as described in Chapter III was then applied to minimize the sum of construction costs, maintenance costs, and network operators' costs over multiple-time periods. The type of optimization problem encountered in this road investment problem is a serial-type problem with one fixed and one free end point. The fixed end point is the road network which exists today. The solution procedure normally begins at the free end point and progresses toward the fixed end point.

The planning model developed in this thesis has been implemented to illustrate the ease of considering a large number of network nodes, a large number of link connections, a large number of decisions or construction policies, and a large number of feasible network states.

6.2 Conclusions. The primary objective of the research reported in this dissertation was to develop an improved application oriented method for planning the construction of new roads on a regional, intercity, or statewide basis. Some of the important characteristics of this planning method which distinguishes it from previous methods are as follows:

1. The method was designed to use basic techniques already available to most statewide planning agencies; e.g., the BPR programs for solving the traffic generation, the traffic distribution, and the traffic assignment problems.

2. The planning method is application oriented and may be implemented with small effort by any statewide highway planning agency.

3. The model is capable of handling a very large network, a large number of specified construction policies, and a large number of specified network states.

4. This planning model allows for maintenance costs to change between periods. This is analogous to the deterioration of existing roads within a given network over the duration of one time period.

5. The decision for constructing new roads occurs at the beginning of a time period while the actual use of the new roads is assumed to occur five years later. The intent of this assumption is to improve the correlation between a mathematical modeling procedure and the existing real world situation.

6. Two ties between optimal decisions that lead to the same total objective function cost is denoted by the dynamic program output. In certain cases, the trade-offs between these optimal decisions can be very useful for the decision maker.

7. Budget limitations on construction decisions for each period are woven into the solution procedures.

8. The optimal decision policies are traced back through each stage of the problem to provide the optimal decision and minimum accumulated cost for the individual stages.

9. All monies are discounted to their present worth value for appropriate cost comparisons at one point in time.

10. The total required computer time for this planning method is very small and should make this planning model very attractive for practical applications.

The numbers postulated for the maintenance costs, construction costs, link operators' costs, and personal income projections may not be realistic; nevertheless, they do serve to demonstrate the regional planning method for highway investments over multiple-time periods. Prior to the implementation of this method which depends on the BPR programs, one must become familiar with each of these programs and realize their limitations. But regardless of their limitations, it can be concluded that a practical application oriented technique for planning the construction of new roads on a regional basis has been developed.

6.3 Recommendations for Further Study. There are a number of desirable investigations and extensions related to this research that should be considered. Some of the more important are: 1) testing the model with actual data being employed; 2) consider implementation of this method when assuming that trips are a random variable and may be formulated through a probability distribution; 3) modify the dynamic programming to include a lower limit on the budgetary constraints for new constructions. This modification would be imposed to force the expenditure of a certain percentage of each period budget; 4) consider the transfer of unused budgetary monies to the next planning period. This would ultimately reduce the constraints imposed by part of the period budgets; 5) perform several sensitivity analyses on the DP program to include period budgets, link costs, link volumes, maintenance costs, and interest rate sensitivities. It has been noted that the optimal decisions did indeed change when the interest rate was reduced. The most logical approach for continued efforts in this area is the testing of the model with real world input data. The incorporation of of real data will provide a meaningful sensitivity analysis.

BIBLIOGRAPHY

1. Almond, Joyce. "Traffic Assignment to a Road Network." Traffic Engineering and Control, Vol. 6, No. 10 (Feb., 1965), 616-617.

2. Basore, B., R. Rhoten, O. Osborne, E. Bula, N. Matthews, L. Chapman, and N. Moore. "Travel Demand in Oklahoma." Multidisciplinary Studies Group, Center for Systems Science, (Interim Report 4-28-71), Oklahoma State University, Stillwater, Oklahoma.

3. Beckmann, Martin J. "On the Theory of Traffic Flow in Networks." Traffic Quarterly, Vol. 21 (April, 1967), 109-117.

4. Bergendahl, G. "A Combined Linear and Dynamic Programming Model for Interdependent Road Investment Planning." Transportation Research, Vol. 3 (1969), 211-228.

5. Bergendahl, G. "An Extended Traffic Assignment Model." Research Report No. 10, Dept. of Business Administration, Stockholm, University, 1968.

6. Brokke, Glen E. "Nationwide Highway Travel." Paper presented at the Western Assn. of State Highway Officials Highway Planning Conf., Santa Fe, New Mexico (June 9, 1966).

7. Bellman, R. E. Dynamic Programming. Princeton, New Jersey: Princeton University Press, 1957.

8. Bellman, R. E. and S. E. Dreyfus. Applied Dynamic Programming. Princeton, New Jersey: Princeton University Press, 1962.

9. Deutschman, Harold. "Urban Transportation Planning." J. of Urban Transpn. Corp., ASCE, Sources of Information on Urban Transpn., Report No. 4 (1968).

10. Drake, Joseph S. and Lester A. Hoel. "Issues in Statewide Transportation Planning." Transpn. Engr. J., Proc. of ASCE, Vol. 96, No. TE3 (Aug., 1970), 379-402.

11. Fratar, T. J. "Vehicular Trip Distribution by Successive Approximations." Traffic Quarterly, Vol. 8, No. 1 (Jan., 1954), 53-65.

12. Funk, Monroe L. and Frank A. Tillman. "Optimal Construction Staging by Dynamic Programming." Journal of Highway Division, Proc. of ASCE, Vol. 94, No. HW2 (1968), 255-265.

13. Funk, Monroe L. and P. O. Roberts. "Toward Optimal Methods of Link Addition in Networks." Monograph, M.I.T. (1964).

14. Garrison, W. L. and D. F. Marble. "Analysis of Highway Networks: A Linear Programming Formulation." Highway Research Board Proc., Vol. 37 (1958), 1-17.

15. Gue, Ronald L. and Michael E. Thomas. Mathematical Methods in Operations Research, Collier-Macmillan Ltd. London: The Macmillan Company, 1970.

16. Guidelines for Trip Generation Analysis. U.S. Dept. of Transpn./ Federal Highway Adm., Bureau of Public Roads (June, 1967).

17. Jarema, F. E., C. E. Pyers, and H. A. Reed. "Evaluation of Trip Distribution and Calibration Procedures." Highway Research Record, No. 191 (1967), 106-129.

18. Jewell, W. S. "Models for Traffic Assignment." Transpn. Res., Vol. 1 (1967), 31-46.

19. Miller, Ronald E. "A Survey of Demand for Travel Studies." Studies in Travel Demand, Prepared by MATHEMATICA for the Dept. of Commerce under Contract No. C-247-65 (Neg) (Sept., 1965).

20. Morlok, Edward K. "A Goal-Directed Transportation Planning Model." Transpn. Res., Vol. 4 (1970), 199-213.

21. Neufville, Richard de and Yasuo Mori. "Optimal Highway Staging by Dynamic Programming." Transpn. Engr. J., Proc. of ASCE, Vol. 96, No. TE1 (1970), 11-25.

22. Osborne, Owen D. "The Application of Systems Engineering Techniques to the Analysis of Intercity Travel Impedance." (unpub. Ph.D. thesis, Oklahoma State University, 1971).

23. Overgaad, K. Rask. "Urban Transportation Planning: Traffic Estimation." Traffic Quarterly, Vol. 21 (Apr., 1967), 197-218.

24. Quandt, Richard E. and William J. Baumol. "The Abstract Mode Model: Theory and Measurement." Studies in Travel Demand, Prepared by MATHEMATICA for the Dept. of Commerce under Contract No. C-187-66 (1966).

25. Ridley, T. M. "An Investment Policy to Reduce the Travel Time in a Transportation Network." Transpn. Res., Vol. 2 (1968), 409-424.

26. Scott, A. J. "An Integer Program for the Optimization of a System of Chromatic Graphs." J. of Reg. Science, Vol. 7, No. 2 (1967), 291-296.

27. Scott, A. J. "The Optimal Network Problem: Some Computational Procedures." Transpn. Res., Vol. 3 (1969), 201-210.

28. Stairs, Sonia. "Selecting an Optimal Traffic Network." J. of Transport Economics and Policy, (1968), 219-231.

29. Thuesen, H. G. Engineering Economy. Englewood, N. J.: Prentice-Hall, 1957.

30. Tillman, Frank A. "Model for Planning a Transportation System." Transpn. Engr. J., Proc. of ASCE, (1970), 229-238.

31. Urban Transportation Planning System 360 ---- General Information. U. S. Bureau of Public Roads, (1970).

32. Urban Transportation Planning System 360 ---- Program Documentation. U. S. Bureau of Public Roads, (1970).

33. Wagner, Harvey M. Principles of Operations Research With Applications to Managerial Decision. Englewood Cliffs, N. J.: Prentice-Hall, 1969.

APPENDIX A

MATHEMATICAL DESCRIPTIONS OF TRAFFIC DISTRIBUTION MODELS

A.1 The Fratar Model. A mathematical representation of the Fratar Model which has been programmed and described by the BPR reference (31) is shown below:

$$T_{ij}(k+1) = (T_{ijk}F_{jk})F_{ik} \qquad \text{(A.1.1)}$$

where

$$F_{jk} = \frac{T_j}{\sum_{i=1}^{n} T_{ijk}} \qquad \text{(A.1.2)}$$

$$F_{ik} = \frac{T_i}{\sum_{j=1}^{n} (T_{ijk}F_{jk})} \qquad \text{(A.1.3)}$$

and where:

T_{ijk} = trips between i and j for iteration k (represents given trips when k = 1);

F_{jk} = destination (column) factor j;

F_{ik} = origin (row) factor i;

T_j = final desired total for destination j;

T_i = final desired total for origin;

i = origin zone number, i=1,2,...,n;

j = destination zone number, j=1,2,...,n;

n = number of zones;

k = iteration number, k=1,2,...,m; and

m = number of iterations.

It is evident that the computation of

$$T_{ijk}\ F_{nk}, \qquad j = 1,2,\ldots,n$$

must occur before F_{ik} can be obtained. Equation (A.1.1) represents a two-step programming process. The application of this process to all origin zones represents one iteration.

A.2 The Intervening Opportunities Model. The theory of the Intervening Opportunities Model was first published in 1960 by Morton Schneider as an appendix to an article entitled "Panel Discussion of Inter-Area Travel Formulas" in Highway Research Board Bulletin 253. The information concerning this model has been taken directly from the Bureau of Public Roads, General Information and Introduction to System/360 (31).

The Intervening Opportunities Model assumes that the trip interchange between an origin and a destination zone is equal to the total trips emanating from the origin multiplied by the probability that each trip origin will find an acceptable terminal at the destination. This is expressed as follows:

$$T_{ij} = O_i\ P(D_j) \qquad \text{(A.2.1)}$$

where:

T_{ij} = the trips between origin zone i and destination zone j;

O_i = the total trip origins produced at zone i;

D_j = the total trip destinations attracted to zone j;

$P(D_j)$ = the probability that each trip origin at i will find destination j an acceptable terminal.

$P(D_j)$, the probability that each trip origin at i will find destination j an acceptable terminal, is expressed as a function of D_j, which is the total trip destinations attracted to zone j. D_j is used because the model assumes that two zonal characteristics determine the probability that a destination will be acceptable. They are the size of the destination and the order in which it is encountered as trips proceed away from the origin.

$P(D_j)$ may also be expressed as the difference between the probability that the trip origins at i will find a suitable terminal in one of all destinations, ordered by closeness to i, up to and including j, and the probability that they will find a suitable terminal in all destinations up to but excluding j. Thus:

$$T_{ij} = O_i[P(A) - P(B)] \tag{A.2.2}$$

where:

A = the sum of all destinations for zones between, in terms of closeness, i and j and including j;

B = the sum of all destinations for zones between i and j but excluding j.

Note that

$$A = B + A_j \quad . \tag{A.2.3}$$

It is then possible to formulate the function P. The probability that a trip will terminate within some volume of destination points is equal to the product of two probabilities. These are (a) the probability that this volume contains an acceptable destination and (b) the probability that an acceptable destination closer to the origin of the trip has not been found. This may be expressed in differentials as follows:

$$dP = (1-P)\ LdV \tag{A.2.4}$$

where:

$$P = P(V)$$

and where:

V = volume of destination points (destination trip ends) within which the probability of a successful terminal is to be calculated;

L = the probability density (probability per destination) of destination acceptability at the point of consideration.

Assuming L to be constant, the solution to Equation (A.2.4) is:

$$P = 1-ke^{-LV} \tag{A.2.5}$$

where:

k = the constant of integration;

e = the constant base of natural logarithms, 2.71828... .

It can be shown that k=1 since P must be zero when V is zero. Equation (A.2.5) thus becomes:

$$P(V) = 1-e^{-LV} \quad . \tag{A.2.6}$$

The function thus derived for P(V) may be substituted into Equation (A.2.2) thus:

$$T_{ij} = O_i \; (e^{-LB}-e^{-LA}) \quad . \tag{A.2.7}$$

Equation (A.2.7) is the standard formulation of the Intervening Opportunities Model and is the one used by the BPR programs. The formulation requires that destination zones be ordered according to their nearness in time to the origin being considered. Thus, in the program, destinations are in sequence according to the contents of the skim tree associated with the origin. A skim tree is a sequence of records containing only the travel times between each pair of zones. This data is obtained during the tree building process.

Equation (A.2.7) is also of interest in that, unlike the Fratar and Gravity Model formulas, it is not insisted that the full number of trip origins be utilized. It is also significant that T_{ij} represents a curvilinear function of L which may obtain a maximum value.

Since the model is based upon the distribution of trip origins, an iterative technique similar to that employed in the Gravity Model is used to cause calculated destination (column) totals to approach the desired values. After each iteration adjusted destination totals are calculated by the following formula:

$$D_{jk} = \frac{D_j}{C_{j(k-1)}} D_{j(k-1)} \tag{A.2.8}$$

where:

D_{jk} = the adjusted destination total for destination zone (column) j, iteration k, $D_{jk} = D_j$ when k = 1;

C_{jk} = the actual destination (column) total for zone j, iteration k;

k = the iteration number, k = 1,2,...,m;

m = the number of iterations.

These adjusted destination totals are those to be employed in the next iteration of the model.

A.3 The Gravity Model. Since an Intuitive Gravity Model formulation has been presented in Chapter III, the author felt a more detailed description and formulation of the "Standard Gravity Model" should be presented. Again, the information concerning this more complex Gravity Model has been taken directly from the BPR reference (31).

The Gravity Model formulation is based upon the hypothesis that the trips produced at an origin and attracted to a destination are directly proportional to the total trip production at the origin, the total trip attraction at the destination, a calibrating term, and possibly a socio-economic adjustment factor. This relationship may be expressed as follows:

$$T_{ij} \propto P_i A_j F_{ij} K_{ij} \qquad \text{(A.3.1)}$$

where:

T_{ij} = trips produced at i and attracted at j;

P_i = total trip production at i;

A_j = total trip attraction at j;

F_{ij} = calibration term for interchange ij;

K_{ij} = socio-economic adjustment factor for interchange ij;

i = an origin zone number, i = 1,2,...,n; and

n = number of zones.

These terms will be further amplified below since they are basic to much of what follows.

From the Gravity Model formulation four separate parameters are required before the trip interchanges (T_{ij}) can be computed. Two of the basic parameters, the number of trips "produced" (P_i) and the number of trips "attracted" (A_j) by each traffic zone in the study area, are related to the use of the land and to the socio-economic characteristics of the people who make trips.

The Gravity Model distributes trips from production zone to attraction zone, while the other travel models in use distribute trips from origin zone to destination zone. To demonstrate the production and attraction definition, it is first necessary to class all trips as home based or nonhome based. Home based trips always have one end at the residence of the trip maker. Nonhome based trips have neither end at the residence of the trip maker.

Home based trips are always produced by the zone of residence of the trip maker whether the trip begins or ends in that zone. Home based trips are always attracted at the nonresidential end of the trip.

Nonhome based trips are always produced by the zone of origin and attracted by the zone of destination.

The spatial separation between zones can be measured by one of several parameters. To date, the most effective measure seems to be travel time.

The total travel time between zones is the sum of the minimum path driving time between zones plus the terminal times at both ends of the trip. Terminal times are added in order to allow for differences in parking and walking times in these zones, as caused by differences in congestion and parking facilities. This provides a more realistic measure of the actual spatial separation (in time) between zones as it is likely to influence automobile drivers in their decisions as to places to work, shop, etc. Terminal times are normally only considered in urban trip distribution and not intercity trip distribution.

The minimum path driving time between each pair of zones is obtained by the traffic assignment process. The traffic assignment process works with data showing the distance and travel speed over major routes of the transportation system. These data are used in preference to the trip times reported in the O-D home interview survey because people tend to report travel time to the nearest 15 minutes even when asked to specify time to the nearest minute.

Terminal times on the other hand, can be obtained from data on average walking distances, which are generally available from parking surveys. They can also be estimated by personal judgement. A reasonable estimate of the terminal time is better than omitting it completely.

Intrazonal driving times, the average driving times of those trips that start and end within the same zone, must also be estimated. Terminal times are added to intrazonal driving time to arrive at intrazonal travel time.

Travel time factors (F_{ij}) express the effect that spatial separation exerts on trip interchange. They indicate the impedance to interzonal travel due to spatial separation between zones. In effect, these factors measure the probability of tripmaking at each one-minute increment of travel time.

Today's travel time factors are usually assumed to remain the same into the future. The validity of this assumption has never been definitely proven, but evidence from studies of work trip travel patterns in Baltimore for the time period between 1926, 1946, and 1958, indicates that there is some basis for making this assumption.

The remaining input to the Gravity Model formula reflects the effect on travel patterns of social and economic characteristics of particular zones or portions of the study area. These are represented by the zone-to-zone adjustment factor (K_{ij}). These factors reflect the effects on travel patterns of social and economic characteristic which are not otherwise accounted for in the use of the model. If found to be necessary, they must be quantitatively related to socio-economic characteristics of the particular zones to which they apply. It is necessary to relate the adjustment factors to characteristics of the zones so that they may be forecast as a function of the socio-economic conditions estimated for the future land use plan. Although the gravity model provides for these adjustments very few cities have found it necessary to use them.

Relationship (A.3.1) may be written as an equation by introducing a constant term, C, as follows:

$$T_{ij} = CP_iA_jF_{ij}K_{ij} \qquad (A.3.2)$$

A value for constant C for any origin zone i, C_i, may be established when it is specified that the sum of all T_{ij}s for origin i must be equal to P_i:

$$P_i = \sum_{j=1}^{n} T_{ij} = \sum_{j=1}^{n} (C_i P_i A_j F_{ij} K_{ij})$$

$$= C_i P_i \sum_{j=1}^{n} (A_j F_{ij} K_{ij}), \qquad i = 1,2,\ldots,n$$

therefore,

$$C_i = \frac{1}{\sum_{j=1}^{n} (A_j F_{ij} K_{ij})}, \qquad i = 1,2,\ldots,n$$

and (A.3.2) becomes

$$T_{ij} = \frac{P_i A_j F_{ij} K_{ij}}{\sum_{j=1}^{n} (A_j F_{ij} K_{ij})}, \qquad i = 1,2,\ldots,n \qquad \text{(A.3.3)}$$

which is the Standard Gravity Model formula.

The calibrating term, F_{ij}, is generally found to be an inverse exponential function of impedance. However, it is not obligated to take that particular form. This elasticity is, perhaps, one of the major strengths of the model.

When all trip interchanges have been computed according to Equation (A.3.3), production (row) totals will be correct due to the structure of Equation (A.3.3), the Gravity Model formula. However, attraction (column) totals will not necessarily match their desired values. An iterative procedure is employed to refine calculated interchanges until actual attraction totals closely match the desired results.

After each iteration, adjusted attraction factors are calculated according to the following formula:

$$A_{jk} = \frac{A_j}{C_{j(k-1)}} A_{j(k-1)} \qquad \text{(A.3.4)}$$

where:

A_{jk} = adjusted attraction factor for attraction zone (column) j, iteration k. $A_{jk} = A_j$ when k=1;

C_{jk} = actual attraction (column) total for zone j, iteration k;

A_j = desired attraction total for attraction zone (column) j;

j = attraction zone number, j = 1,2,...,n;

n = number of zones;

k = iteration number, k = 1,2,...,m; and

m = number of iterations.

In each iteration, the Gravity Model formula is applied to calculate zonal trip interchanges using the adjusted attraction factors obtained from the preceding iteration. In practice, the Gravity Model formula thus becomes:

$$\left(T_{ijk} = \frac{P_i A_{jk} F_{ij} K_{ij}}{\sum_{j=1}^{n} (A_{jk} F_{ij} K_{ij})} \right)_p \qquad \text{(A.3.5)}$$

where T_{ijk} is the trip interchange between i and j for iteration k and $A_{jk} = A_j$ when k=1. Subscript j goes through one complete cycle every time k changes, and i goes through one complete cycle every time j changes. Formula (A.3.5) is enclosed in brackets which are subscripted p to indicate that the complete process is completed for each trip purpose. It is equivalent to placing a subscript p on every variable in Equation (A.3.5).

The calibration term, F_{ij}, is usually a function of trip time. Its usage is generalized, however, by using a table rather than a formula to obtain values for F_{ij}. The user thus supplies a table of F-factors (friction factors) for each trip purpose. Individual values are related to increments of trip time. Skim trees are supplied by the user to indicate interzonal travel times. The F-factor chosen for each interchange is thus a function of the trip time for that interchange.

It is quite evident, however, that the F-table supplied by the user for a particular trip purpose could easily represent something other than a continuous inverse exponential function. It is equally evident that the contents of the skim trees supplied by the user could reflect some other measures of impedance than time alone. This feature of the model makes it a very general technique.

The usual procedure, having chosen an appropriate measure of impedance, is to calibrate to base year data. An assumed set of F-factors is adjusted until a satisfactory approximation results. A detailed discussion of the calibration procedure can be found in the BPR documentation (31).

The adjustment term, K_{ij}, unlike the F-factor, is supplied only to interchange ij. If none is supplied by the user, $K_{ij} = 1$ is assumed by the program. K-factors should be resorted to only when a few extreme socio-economic variations can be distinguished. They are usually developed and applied to the aggregate.

APPENDIX B

COMPUTER PROGRAM LISTINGS AND SAMPLE OUTPUTS

B.1 Program 1. This program executed the programs BLDSPWB, FMTSPWB, and BLDSPTR. A listing of the execution cards and data cards for the 53-node problem is shown in Table XX. Disk data sets were employed to save the spiderweb and the spider trees found by the programs BLDSPWB and BLDSPTR.

TABLE XX

LISTING OF PROGRAM 1

```
//CHAP1  JOB (10652,441-40-6462,1,,,9001,3),'L D CHAPMAN'
//JOBLIB DD DISP=OLD,
//          DSN=OSU.ACT10652.CHAP
//STEP01 EXEC PGM=BLDSPWB,TIME=(,10),REGION=30K
//OPNTAPE DD SYSOUT=A
//NWRCDD DD DISP=OLD,
//          DSN=OSU.ACT10652.SPNET
//SYSIN DD *
PAR,53,53,8,0
GC
    1 1537 2447          OKLA CITY
    2 1625 2492          TULSA
    3 1488 2387          LAWTON
    4 1517 2511          ENID
    5 1625 2534          BARTLESVILLE
    6 1564 2530          PONCA CITY
    7 1660 2466          MUSKOGEE
    8 1514 2380          DUNCAN
    9 1560 2358          ARDMORE
   10 1565 2492          STILLWATER
   11 1572 2437          SHAWNEE
   12 1555 2399          PAULS VALLEY
   13 1436 2390          ALTUS
   14 1685 2544          MIAMI
   15 1625 2458          OKMULGEE
   16 1636 2410          MCALESTER
   17 1585 2399          ADA
   18 1432 2513          WOODWARD
   19 1702 2418          POTEAU
   20 1514 2418          CHICKASHA
   21 1496 2419          ANADARKO
   22 1457 2451          CLINTON
   23 1471 2539          ALVA
   24 1431 2443          ELK CITY
   25 1586 2432          SEMINOLE
   26 1662 2506          PRYOR
   27 1689 2339          IDABEL
   28 1457 2434          CORDELL
   29 1601 2344          DURANT
   30 1681 2480          TAHLEQUAH
   31 1645 2506          CLAREMORE
   32 1453 2373          FREDERICK
   33 1543 2475          GUTHRIE
   34 1573 2464          CHANDLER
   35 1452 2418          HOBART
   36 1487 2475          WATONGA
   37 1515 2475          KINGFISHER
   38 1665 2528          VINITA
   39 1690 2446          SALLISAW
   40 1551 2503          PERRY
   41 1658 2481          WAGONER
   42 1600 2420          HOLDENVILLE
   43 1650 2347          HUGO
   44 1616 2371          ATOKA
   45 1569 2381          SULPHER
   46 1580 2352          MADILL
   47 1646 2532          NOWATA
   48 1580 2508          PAWNEE
   49 1426 2408          MANGUM
   50 1650 2448          CHECOTAH
   51 1605 2443          OKEMAH
   52 1495 2371          WALTERS
   53 1403 2394          HOLLIS
END
```

TABLE XX (Continued)

```
        1   11   12   20   22   33   34   37
        2    5    7   15   31   34   41   48   50
        3    8   12   13   20   21   32   35   52
        4    6   18   23   33   36   37   40
        5    6   31   47   48
        6   10   23   40   48
        7   15   30   39   41   50
        8    9   12   20   45   52
        9   12   45   46   52
       10   15   33   34   40   48
       11   12   17   25   34   51
       12   17   20   45
       13   32   35   49   53
       14   30   38   47
       15   16   34   41   50   51
       16   17   19   27   42   43   44   50
       17   25   42   44   45
       18   22   23   24   36
       19   27   39   43   50
       20   21   37
       21   22   28   35   36   37
       22   23   24   28   36
       23   36
       24   28   35   49   53
       25   35   42   51
       26   30   31   38   41   47
       27   43
       28   35   37
       29   43   44   46
       30   39   41
       31   38   41   47
       32   35   52
       33   34   37   40
       34   51
       35   49
       36   37
       37   40
       38   47
       39   50
       40   48
       42   44   50   51
       43   44
       44   45   46
       45   46
       48   51
       49   53
       50   51
END
//STEPC2 EXEC PGM=FMTSPWR,TIME=(,10),REGION=20K
//OPRTAPE DD SYSOUT=A
//NWRCDI DD DISP=OLD,
//          DSN=OSU.ACT10652.SPNET
//STEP03 EXEC PGM=BLDSPTR,TIME=(,10),REGION=25K
//OPRTAPE DD DUMMY
//NWRCDI DD DISP=OLD,
//          DSN=OSU.ACT10652.SPNET
//PATHSO DD DISP=OLD,
//          DSN=OSU.ACT10652.TREES
//SYSIN DD *
TRALL
E0006
//
```

B.2 FMTSPWB Output. The sample output as shown in Table XXI is the standard output for this BPR program. The A node and each B node connected to the A node is provided for every node in the network. For example, in row 1 of Table XXI, node 1 is connected to node 11, 12, 20, 22, 33, 34, and 37. Also provided in the output of the FMTSPWB program is the link impedance, the link distance, the speed on this link, and the x-y coordinates of the A node.

TABLE XXI

SAMPLE OUTPUT OF PROGRAM FMTSPWB

NODE A		DESCRIPTIVE DATA			LEGS ZERO AND FOUR				LEGS ONE AND FIVE				LEGS TWO AND SIX				LEGS THREE AND SEVEN			
NUMBER	STATE #NAME	COUNTY NAME	COORDINATES X	COORDINATES Y	NODE B NO.LG	IMPECANCE	DISTANCE	SPD	NODE B NO.LG	IMPEDANCE	DISTANCE	SPD	NODE B NO.LG	IMPEDANCE	DISTANCE	SPD	NODE B NO.LG	IMPEDANCE	DISTANCE	SPD
1	0		1537	2447	11.0	36	36	60.0	12.0	51	51	60.0	20.0	37	37	60.0	22.0	80	80	60.0
					33.0	29	29	60.0	34.0	40	40	60.0	37.0	36	36	60.C				
2	0		1625	2492	5.0	42	42	60.0	7.0	44	44	60.0	15.0	34	34	60.0	31.0	24	24	60.0
					34.1	59	59	60.0	41.0	35	35	60.0	48.0	48	48	60.0	50.0	51	51	60.0
3	0		1488	2387	8.0	27	27	60.0	12.1	68	68	60.0	13.0	52	52	60.0	20.1	40	40	60.0
					21.C	33	33	60.0	32.0	38	38	60.0	35.0	48	48	60.0	52.0	17	17	60.0
4	0		1517	2511	6.0	51	51	60.0	18.0	85	85	60.0	23.0	54	54	60.0	33.1	44	44	60.0
					36.C	47	47	60.0	37.1	36	36	60.0	40.0	35	35	60.0				
5	0		1625	2534	2.0	42	42	60.0	6.1	61	61	60.0	31.1	34	34	60.0	47.0	21	21	60.0
					48.1	52	52	60.0												
6	0		1564	2530	4.C	51	51	60.0	5.1	61	61	60.0	10.0	38	38	60.0	23.1	93	93	60.0
					40.1	30	30	60.0	48.2	27	27	60.0								
7	0		1660	2466	2.1	44	44	60.0	15.1	36	36	60.0	30.0	25	25	60.0	39.0	36	36	60.0
					41.1	15	15	60.0	50.1	21	21	60.0								
8	0		1514	2380	3.C	27	27	60.0	9.0	51	51	60.0	12.2	45	45	60.0	20.2	38	38	60.0
					45.0	55	55	60.0	52.1	21	21	60.0								
9	0		1560	2358	8.1	51	51	60.0	12.3	41	41	60.0	45.1	25	25	60.0	46.0	21	21	60.0
					52.2	66	66	60.0												
10	0		1565	2492	6.2	38	38	60.0	15.2	69	69	60.0	33.2	28	28	60.0	34.2	29	29	60.0
					40.2	18	18	60.0	48.3	22	22	60.0								
11	0		1572	2437	1.C	36	36	60.0	12.4	42	42	60.0	17.0	40	40	60.0	25.0	15	15	60.0
					34.3	27	27	60.0	51.0	34	34	60.0								
12	0		1555	2399	1.1	51	51	60.0	3.1	68	68	60.0	8.2	45	45	60.0	9.1	41	41	60.0
					11.1	42	42	60.0	17.1	30	30	60.0	20.3	45	45	60.0	45.2	23	23	60.0
13	0		1436	2390	3.2	52	52	60.0	32.1	24	24	60.0	35.1	32	32	60.0	49.0	21	21	60.0
					53.0	33	33	60.0												
14	0		1685	2544	30.1	64	64	60.0	38.0	23	23	60.0	47.1	41	41	60.0				
15	0		1625	2458	2.2	34	34	60.0	7.1	36	36	60.0	10.1	69	69	60.0	16.0	49	49	60.0
					34.4	52	52	60.0	41.2	40	40	60.0	50.2	27	27	60.0	51.1	25	25	60.0
16	0		1636	2410	15.3	49	49	60.0	17.2	52	52	60.0	19.0	66	66	60.0	27.0	89	89	60.0
					42.0	37	37	60.0	43.0	65	65	60.0	44.0	44	44	60.0	50.3	41	41	60.0
17	0		1585	2399	11.2	40	40	60.0	12.5	30	30	60.0	16.1	52	52	60.0	25.1	33	33	60.0
					42.1	26	26	60.0	44.1	42	42	60.0	45.3	24	24	60.0				
18	0		1432	2513	4.1	85	85	60.0	22.1	67	67	60.0	23.2	47	47	60.0	24.0	70	70	60.0
					36.1	67	67	60.0												
19	0		1702	2418	16.2	66	66	60.0	27.1	80	80	60.0	39.1	30	30	60.0	43.1	88	88	60.0
					50.4	60	60	60.0												
20	0		1514	2418	1.2	37	37	60.0	3.3	40	40	60.0	8.3	38	38	60.0	12.6	45	45	60.0
					21.1	18	18	60.0	37.2	57	57	60.0								
21	0		1496	2419	3.4	33	33	60.0	20.4	18	18	60.0	22.2	50	50	60.0	28.0	42	42	60.0
					35.2	44	44	60.0	36.2	57	57	60.0	37.3	59	59	60.0				
22	0		1457	2451	1.3	80	80	60.0	18.1	67	67	60.0	21.2	50	50	60.0	23.3	89	89	60.0
					24.1	27	27	60.0	28.1	17	17	60.0	36.3	38	38	60.0				
23	0		1471	2539	4.2	54	54	60.0	6.3	93	93	60.0	18.2	47	47	60.0	22.3	89	89	60.0
					36.4	66	66	60.0												
24	0		1431	2443	18.3	7C	7C	60.0	22.4	27	27	60.0	28.2	28	28	60.0	35.3	33	33	60.0
					49.1	35	35	60.0	53.1	56	56	60.0								
25	0		1586	2432	11.3	15	15	60.0	17.3	33	33	60.0	35.4	135	135	60.0	42.2	18	18	60.0
					51.2	22	22	60.0												

TABLE XXI (Continued)

NODE A NUMBER	DESCRIPTIVE DATA: STATE NAME	COUNTY NAME	COORDINATES X	COORDINATES Y	LEGS ZERO AND FOUR: NODE B NO.LG	IMPEDANCE	DISTANCE	SPD	LEGS ONE AND FIVE: NODE B NO.LG	IMPEDANCE	DISTANCE	SPD	LEGS TWO AND SIX: NODE B NO.LG	IMPEDANCE	DISTANCE	SPD	LEGS THREE AND SEVEN: NODE B NO.LG	IMPEDANCE	DISTANCE	SPD
26	C		1662	2506	30.2	32	32	60.0	31.2	17	17	60.0	38.1	23	23	60.0	41.3	25	25	60.0
					47.2	31	31	60.0												
27	0		1689	2339	16.3	89	89	60.0	19.1	80	80	60.0	43.2	40	40	60.0				
28	0		1457	2434	21.3	42	42	60.0	22.5	17	17	60.0	24.2	28	28	60.0	35.5	17	17	60.0
					37.4	71	71	60.0												
29	0		1601	2344	43.3	49	49	60.0	44.2	31	31	60.0	46.1	22	22	60.0				
30	0		1681	2480	7.2	25	25	60.0	14.0	64	64	60.0	26.0	32	32	60.0	39.2	35	35	60.0
					41.4	23	23	60.0												
31	0		1645	2506	2.3	24	24	60.0	5.2	34	34	60.0	26.1	17	17	60.0	38.2	33	33	60.0
					41.5	28	28	60.0	47.3	26	26	60.0								
32	0		1453	2373	3.5	38	38	60.0	13.1	24	24	60.0	35.6	45	45	60.0	52.3	42	42	60.0
33	0		1543	2475	1.4	29	29	60.0	4.3	44	44	60.0	10.2	28	28	60.0	34.5	32	32	60.0
					37.5	28	28	60.0	40.3	29	29	60.0								
34	0		1573	2464	1.5	40	40	60.0	2.4	59	59	60.0	10.3	29	29	60.0	11.4	27	27	60.0
					15.4	52	52	60.0	33.3	32	32	60.0	51.3	38	38	60.0				
35	0		1452	2418	3.6	48	48	60.0	13.2	32	32	60.0	21.4	44	44	60.0	24.3	33	33	60.0
					25.2	135	135	60.0	28.3	17	17	60.0	32.2	45	45	60.0	49.2	28	28	60.0
36	0		1487	2475	4.4	47	47	60.0	18.4	67	67	60.0	21.5	57	57	60.0	22.6	38	38	60.0
					23.4	66	66	60.0	37.6	28	28	60.0								
37	0		1515	2475	1.6	36	36	60.0	4.5	36	36	60.0	20.5	57	57	60.0	21.6	59	59	60.0
					28.4	71	71	60.0	33.4	28	28	60.0	36.5	28	28	60.0	40.4	46	46	60.0
38	0		1669	2528	14.1	23	23	60.0	26.2	23	23	60.0	31.3	33	33	60.0	47.4	23	23	60.0
39	0		1690	2446	7.3	36	36	60.0	19.2	30	30	60.0	30.3	35	35	60.0	50.5	40	40	60.0
40	0		1551	2503	4.6	35	35	60.0	6.4	30	30	60.0	10.4	18	18	60.0	33.5	29	29	60.0
					37.7	46	46	60.0	48.4	29	29	60.0								
41	0		1658	2481	2.5	35	35	60.0	7.4	15	15	60.0	15.5	40	40	60.0	26.3	25	25	60.0
					30.4	23	23	60.0	31.4	28	28	60.0								
42	0		1600	2420	16.4	37	37	60.0	17.4	26	26	60.0	25.3	18	18	60.0	44.3	52	52	60.0
					50.6	57	57	60.0	51.4	24	24	60.0								
43	0		1650	2347	16.5	65	65	60.0	19.3	88	88	60.0	27.2	40	40	60.0	29.0	49	49	60.0
					44.4	42	42	60.0												
44	0		1616	2371	16.6	44	44	60.0	17.5	42	42	60.0	29.1	31	31	60.0	42.3	52	52	60.0
					43.4	42	42	60.0	45.4	48	48	60.0	46.2	41	41	60.0				
45	0		1569	2381	8.4	55	55	60.0	9.2	25	25	60.0	12.7	23	23	60.0	17.6	24	24	60.0
					44.5	48	48	60.0	46.3	31	31	60.0								
46	0		1580	2352	9.3	21	21	60.0	29.2	22	22	60.0	44.6	41	41	60.0	45.5	31	31	60.0
47	0		1646	2532	5.3	21	21	60.0	14.2	41	41	60.0	26.4	31	31	60.0	31.5	26	26	60.0
					38.3	23	23	60.0												
48	0		1580	2508	2.6	48	48	60.0	5.4	52	52	60.0	6.5	27	27	60.0	10.5	22	22	60.0
					40.5	29	29	60.0	51.5	70	70	60.0								
49	0		1426	2408	13.3	21	21	60.0	24.4	35	35	60.0	35.7	28	28	60.0	53.2	27	27	60.0
50	0		1650	2448	2.7	51	51	60.0	7.5	21	21	60.0	15.6	27	27	60.0	16.7	41	41	60.0
					19.4	60	60	60.0	39.3	40	40	60.0	42.4	57	57	60.0	51.6	45	45	60.0

TABLE XXI (Continued)

NODE A DESCRIPTIVE DATA					LEGS ZERO AND FOUR				LEGS ONE AND FIVE				LEGS TWO AND SIX				LEGS THREE AND SEVEN			
NUMBER	STATE #NAME	COUNTY NAME	COORDINATES X	COORDINATES Y	NODE B NO.LG	IMPEDANCE	DISTANCE	SPD	NODE B NO.LG	IMPEDANCE	DISTANCE	SPD	NODE B NO.LG	IMPEDANCE	DISTANCE	SPD	NODE B NO.LG	IMPEDANCE	DISTANCE	SPD
51	C		1605	2443	11.5	34	34	60.0	15.7	25	25	60.0	25.4	22	22	60.0	34.6	38	38	60.0
					42.5	24	24	60.0	48.5	70	70	60.0	50.7	45	45	60.0				
52	0		1495	2371	3.7	17	17	60.0	8.5	21	21	60.0	9.4	66	66	60.0	32.3	42	42	60.0
53	C		1403	2394	13.4	33	33	60.0	24.5	56	56	60.0	49.3	27	27	60.0				

FMTSPWB (06/19/69) CCMPLETE

B.3 FMTSPTR Program. The program FMTSPTR execution cards and data cards are shown in Table XXII. This is a typical execution setup for this program.

TABLE XXII

FORMAT SPIDER TREES (FMTSPTR) PROGRAM

```
//       JOB (10652,441-40-6462,1,,,9001,3),'L D CHAPMAN'
//STEP4 EXEC PGM=FMTSPTR,TIME=(,10),REGION=80K
//DPNTAPE DD SYSOUT=A
//PATHSI DD DISP=OLD,
//          DSN=OSU.ACT10652.TREES
//SYSIN DD *
C       53   53    8
T        1    2   40   53
E
//
```

B.4. FMTSPTR Output. Tables XXIII, XXIV, XXV, and XXVI are the typical outputs from the FMTSPTR program as shown in Appendix B.3. Tables XXIII, XXIV, XXV, and XXVI are the tree outputs for nodes 1, 2, 40, and 53, respectively.

TABLE XXIII

FMTSPTR OUTPUT FOR NODE 1

```
TREE FROM NODE      1

 2    34  HOME
 3    20  HOME
 4    37  HOME
 5    48     10     33  HOME
 6    40     33
     CONTINUED IN TRACE     5
 7    15     34
     CONTINUED IN TRACE     2
 8    20
     CONTINUED IN TRACE     3
 9    12  HOME
10   CONTINUED IN TRACE     5
11   HOME
12   CONTINUED IN TRACE     9
13     3
     CONTINUED IN TRACE     3
14    38     31      2
     CONTINUED IN TRACE     2
15   CONTINUED IN TRACE     7
16    42     25     11
     CONTINUED IN TRACE    11
17    11
     CONTINUED IN TRACE    11
18    36     37
     CONTINUED IN TRACE     4
19    16
     CONTINUED IN TRACE    16
20   CONTINUED IN TRACE     3
21    20
     CONTINUED IN TRACE     3
22   HOME
23     4
     CONTINUED IN TRACE     4
24    22
     CONTINUED IN TRACE    22
25   CONTINUED IN TRACE    16
26    31
     CONTINUED IN TRACE    14
27    16
     CONTINUED IN TRACE    16
28    21
     CONTINUED IN TRACE    21
29    46     45     12
     CONTINUED IN TRACE     9
30     7
     CONTINUED IN TRACE     7
31   CONTINUED IN TRACE    14
32     3
     CONTINUED IN TRACE     3
33   CONTINUED IN TRACE     5
34   CONTINUED IN TRACE     2
35    21
     CONTINUED IN TRACE    21
36   CONTINUED IN TRACE    18
37   CONTINUED IN TRACE     4
38   CONTINUED IN TRACE    14
39    50     51     11
     CONTINUED IN TRACE    11
40   CONTINUED IN TRACE     6
41    15
     CONTINUED IN TRACE     7
42   CONTINUED IN TRACE    16
43    44     17
     CONTINUED IN TRACE    17
44   CONTINUED IN TRACE    43
45   CONTINUED IN TRACE    29
46   CONTINUED IN TRACE    29
47    31
     CONTINUED IN TRACE    14
48   CONTINUED IN TRACE     5
49    35
     CONTINUED IN TRACE    35
50   CONTINUED IN TRACE    39
51   CONTINUED IN TRACE    39
52     3
     CONTINUED IN TRACE     3
53    49
     CONTINUED IN TRACE    49
```

TABLE XXIV

FMTSPTR OUTPUT FOR NODE 2

```
TREE FROM NODE      2

     1    34  HOME

     3    20      1
        CONTINUED IN TRACE      1

     4    40     48  HOME

     5  HOME

     6    48
        CONTINUED IN TRACE      4

     7  HOME

     8    12     11     34
        CONTINUED IN TRACE      1

     9    45     17     42     51     15  HOME

    10    48
        CONTINUED IN TRACE      4

    11  CONTINUED IN TRACE      8

    12  CONTINUED IN TRACE      8

    13     3
        CONTINUED IN TRACE      3

    14    38     31  HOME

    15  CONTINUED IN TRACE      9

    16    15
        CONTINUED IN TRACE      9

    17  CONTINUED IN TRACE      9

    18     4
        CONTINUED IN TRACE      4

    19    39      7
        CONTINUED IN TRACE      7

    20  CONTINUED IN TRACE      3

    21    20
        CONTINUED IN TRACE      3

    22     1
        CONTINUED IN TRACE      1

    23     4
        CONTINUED IN TRACE      4

    24    22
        CONTINUED IN TRACE     22

    25    51
        CONTINUED IN TRACE      9

    26    31
        CONTINUED IN TRACE     14

    27    16
        CONTINUED IN TRACE     16

    28    37     33     34
        CONTINUED IN TRACE      1

    29    44     16
        CONTINUED IN TRACE     16

    30    41  HOME

    31  CONTINUED IN TRACE     14

    32     3
        CONTINUED IN TRACE      3

    33  CONTINUED IN TRACE     28

    34  CONTINUED IN TRACE      1

    35    21
        CONTINUED IN TRACE     21

    36    37
        CONTINUED IN TRACE     28

    37  CONTINUED IN TRACE     28

    38  CONTINUED IN TRACE     14

    39  CONTINUED IN TRACE     19

    40  CONTINUED IN TRACE      4

    41  CONTINUED IN TRACE     30

    42  CONTINUED IN TRACE      9

    43    16
        CONTINUED IN TRACE     16

    44  CONTINUED IN TRACE     29

    45  CONTINUED IN TRACE      9

    46    45
        CONTINUED IN TRACE      9

    47    31
        CONTINUED IN TRACE     14

    48  CONTINUED IN TRACE      4

    49    35
        CONTINUED IN TRACE     35

    50  HOME

    51  CONTINUED IN TRACE      9

    52     3
        CONTINUED IN TRACE      3

    53    49
        CONTINUED IN TRACE     49
```

TABLE XXV

FMTSPTR OUTPUT FOR NODE 40

```
TREE FROM NODE    40

 1    33  HOME

 2    48  HOME

 3    20     1
     CONTINUED IN TRACE     1

 4   HOME

 5    48
     CONTINUED IN TRACE     2

 6   HOME

 7     2
     CONTINUED IN TRACE     2

 8    20
     CONTINUED IN TRACE     3

 9    12     1
     CONTINUED IN TRACE     1

10   HOME

11    34    10
     CONTINUED IN TRACE    10

12   CONTINUED IN TRACE     9

13    35    28    37  HOME

14    47     5
     CONTINUED IN TRACE     5

15    10
     CONTINUED IN TRACE    10

16    15
     CONTINUED IN TRACE    15

17    11
     CONTINUED IN TRACE    11

18     4
     CONTINUED IN TRACE     4

19    50    15
     CONTINUED IN TRACE    15

20   CONTINUED IN TRACE     3

21    37
     CONTINUED IN TRACE    13

22    36    37
     CONTINUED IN TRACE    13

23     4
     CONTINUED IN TRACE     4

24    22
     CONTINUED IN TRACE    22

25    11
     CONTINUED IN TRACE    11

26    31     2
     CONTINUED IN TRACE     2

27    16
     CONTINUED IN TRACE    16

28   CONTINUED IN TRACE    13

29    46    45    12
     CONTINUED IN TRACE     9

30    41     2
     CONTINUED IN TRACE     2

31   CONTINUED IN TRACE    26

32     3
     CONTINUED IN TRACE     3

33   CONTINUED IN TRACE     1

34   CONTINUED IN TRACE    11

35   CONTINUED IN TRACE    13

36   CONTINUED IN TRACE    22

37   CONTINUED IN TRACE    13

38    47
     CONTINUED IN TRACE    14

39    50
     CONTINUED IN TRACE    19

41   CONTINUED IN TRACE    30

42    25
     CONTINUED IN TRACE    25

43    44    17
     CONTINUED IN TRACE    17

44   CONTINUED IN TRACE    43

45   CONTINUED IN TRACE    29

46   CONTINUED IN TRACE    29

47   CONTINUED IN TRACE    14

48   CONTINUED IN TRACE     2

49    35
     CONTINUED IN TRACE    13

50   CONTINUED IN TRACE    19

51    34
     CONTINUED IN TRACE    11

52     3
     CONTINUED IN TRACE     3

53    49
     CONTINUED IN TRACE    49
```

TABLE XXVI

FMTSPTR OUTPUT FOR NODE 53

```
TREE FROM NODE     53

     1    20    21    35    49   HOME

     2    34     1
       CONTINUED IN TRACE     1

     3    13   HOME

     4    36    22    24   HOME

     5    48    40    37    28    35
       CONTINUED IN TRACE     1

     6     4
       CONTINUED IN TRACE     4

     7    15    51    25    35
       CONTINUED IN TRACE     1

     8     3
       CONTINUED IN TRACE     3

     9     8
       CONTINUED IN TRACE     8

    10    33    37
       CONTINUED IN TRACE     5

    11     1
       CONTINUED IN TRACE     1

    12     3
       CONTINUED IN TRACE     3

    13 CONTINUED IN TRACE     3

    14    47     5
       CONTINUED IN TRACE     5

    15 CONTINUED IN TRACE     7

    16    17    12
       CONTINUED IN TRACE    12

    17 CONTINUED IN TRACE    16

    18    24
       CONTINUED IN TRACE     4

    19    16
       CONTINUED IN TRACE    16

    20 CONTINUED IN TRACE     1

    21 CONTINUED IN TRACE     1

    22 CONTINUED IN TRACE     4

    23    22
       CONTINUED IN TRACE     4

    24 CONTINUED IN TRACE     4

    25 CONTINUED IN TRACE     7

    26    31     2
       CONTINUED IN TRACE     2

    27    43    29    46     9
       CONTINUED IN TRACE     9

    28 CONTINUED IN TRACE     5

    29 CONTINUED IN TRACE    27

    30     7
       CONTINUED IN TRACE     7

    31 CONTINUED IN TRACE    26

    32    13
       CONTINUED IN TRACE     3

    33 CONTINUED IN TRACE    10

    34 CONTINUED IN TRACE     2

    35 CONTINUED IN TRACE     1

    36 CONTINUED IN TRACE     4

    37 CONTINUED IN TRACE     5

    38    31
       CONTINUED IN TRACE    26

    39    50    51
       CONTINUED IN TRACE     7

    40 CONTINUED IN TRACE     5

    41    15
       CONTINUED IN TRACE     7

    42    25
       CONTINUED IN TRACE     7

    43 CONTINUED IN TRACE    27

    44    45     8
       CONTINUED IN TRACE     8

    45 CONTINUED IN TRACE    44

    46 CONTINUED IN TRACE    27

    47 CONTINUED IN TRACE    14

    48 CONTINUED IN TRACE     5

    49 CONTINUED IN TRACE     1

    50 CONTINUED IN TRACE    39

    51 CONTINUED IN TRACE     7

    52    32
       CONTINUED IN TRACE    32

FMTSPTR (07/31/69) COMPLETE.
```

B.5 BLDTT Program. In Table XXVII is the program listing for the BLDTT program. This program computes the two-way trips between each pair of nodes for a given network and writes an output that is compatable with the LDSPWB program. The Intuitive Gravity Model in Equation (3.2.1) has been used in this program. The program will read trip table entries directly from data cards and format the proper input requirements for the LDSPWB program.

TABLE XXVII

BUILD TRIP TABLE (BLDTT) LISTING

```
//BLCTT  JOB (10652,441-40-6462,1,,,9001,3),'L D CHAPMAN'
// EXEC FORTGCLG,REGION.GO=89K
//FORT.SYSIN DD *
      COMMON NR,N,NP,NN( 60),IDEN1( 60),IDEN2( 60),N1,N2,M1,M2,G1,G2,
     # A1,A2,C(60,60),F(60,60),F1(60,60),D(60),P2(60),
     $S(60),S2(60),W1(60),W2(60)
      DIMENSION A(60),B(60),AR(60),BR(60),MTAB(60)
      NP=0
      G1=440.
      M1=1
      N1=1
      A1=2.78
      READ(5,10)N,JKL,NR,NOTTO,TRPMU
   10 FORMAT(4I5,F10.3)
      GO TO (12,60,71),JKL
   12 WRITE(6,13)
   13 FORMAT(//1X,'GRAVITY MODEL USED TO GENERATE TRIP TABLE')
      WRITE (6,20)
   20 FORMAT (1H ,4HNODE,2X,8HLATITUDE,2X,9HLONGITUDE,2X,18HPERSONAL INC
     #OME M$)
      DO 30 KK=1,N
      READ(5,25) NO,DLAT,OLAT,DLON,OLON,EP,ID1,ID2
   25 FORMAT (I5,4X,F2.0,1X,F2.0,5X,F3.0,1X,F2.0,2X,F12.4,33X,2A4)
      WRITE(6,35)NO,DLAT,OLAT,DLON,OLON,EP,ID1,ID2
   35 FORMAT(1H ,I3,5X,F3.0,1X,F3.0,5X,F3.0,1X,F3.0,5X,F9.1,8X,2A4)
      DOLAT = OLAT/60.0
      A(KK) = DLAT + DOLAT
      AR(KK) = A(KK)*3.14159/180.0
      DOLON = OLON/60.0
      B(KK) = DLON + DOLON
      BR(KK) = B(KK)*3.14159/180.0
      D(KK) = EP
      NN(KK) = NO
      IDEN1(KK)=ID1
      IDEN2(KK)=ID2
   30 CONTINUE
      DO 50 I=1,N
      AI = AR(I)
      BI = BR(I)
      DO 50 J=1,I
      IF(I-J) 40,45,40
   40 CONTINUE
      AJ = AR(J)
      BJ = BR(J)
      X=SIN(AI)*SIN(AJ)+COS(AI)*COS(AJ)*COS(BI-BJ)
      CIJ=3960.0*ATAN(SQRT(1.0-X**2)/X)
      GO TO 46
   45 CONTINUE
      CIJ = 100.0
   46 CONTINUE
      C(J,I)=CIJ
   50 F(J,I)=440.*(D(I)*D(J))/CIJ**2.78
      S(1)=0.0
      DO 51 J=2,N
   51 S(1)=S(1)+F(1,J)
      K=N-1
      DO 53 I=2,K
      S(I)=0.0
```

TABLE XXVII (Continued)

```
      L=I+1
      DO 52 J=L,N
 52   S(I)=S(I)+F(I,J)
      M=I-1
      DO 53 J=1,M
 53   S(I)=S(I)+F(J,I)
      S(N)=0.0
      MM=N-1
      DO 54 J=1,MM
 54   S(N)=S(N)+F(J,N)
      IF(NOTTO.EQ.1)  GO TO 55
  55  DO 120  I=1,N
      DO 115 J=1,I
      F(J,I)=F(J,I)/2.
115   F(I,J)=F(J,I)
      F(I,I)=0.
 120  CONTINUE
 600  FORMAT('1',1X,20I6)
      WRITE(6,600) (J,J=1,20)
      DO 800 I=1,53
 750  WRITE(6,701) I,(F(I,J),J=1,20)
 701  FORMAT(I3,20F6.0)
 800  CONTINUE
      L1 = 21
      L2 = 40
      DO 890 I1=1,2
      WRITE(6,600) (J,J=L1,L2)
      DO 850 I=1,53
      WRITE(6,701) I,(F(I,J),J=L1,L2)
 850  CONTINUE
      L1 = 41
      L2 = 53
 890  CONTINUE
      STOP
 140   NPURP=0
      DO 150 I=1,N
      K=1
      DO 145 J=1,N
      M1=F(I,J)
      IF(M1) 123,122,123
 123  K=K+1
 121  MTAB(K)=(J*262144)+M1
      NWD=K
 122  CONTINUE
 145  CONTINUE
      NZON=I*65536
      WRITE(8,125) NWD,NPURP,NZON,(MTAB(L),L=2,NWD)
 125  FORMAT(A4,2A2,95A4)
 150  CONTINUE
      WRITE(6,200)
200   FORMAT(////1X,'BUILD TRIP TABLE COMPLETE')
      GO TO 64
  60  READ(5,62)((F(I,J),J=1,N),I=1,N)
  62  FORMAT(8F10.0)
      WRITE(6,63)
  63  FORMAT(//1X,'TRIP TABLE HAS BEEN READ DIRECTLY FROM DATA CARDS')
  66  GO TO 140
  71  DO 75  I=1,N
      DO 75  J=1,N
```

TABLE XXVII (Continued)

```
      F(I,J)=TRPMU*F(I,J)
   75 CONTINUE
      WRITE(6,72)
   72 FORMAT(//1X,'ALL TRIPS MULTIPLIED BY TRPMU')
      WRITE(6,73) TRPMU
   73 FORMAT(//10X,'TRPMU = ',F10.3)
      IF(NOTTO.EQ.1)  GO TO 74
      CALL OUTPUT
   74 GO TO 140
   64 CALL EXIT
      END
      SUBROUTINE OUTPUT
      REAL LAF,LAP
      LOGICAL MAU2,MAU3,MAU4,MAV1,MAV2,MAV3,MAV4
      COMMON NR,N,NP,NN( 60),IDEN1( 60),IDEN2( 60),N1,N2,M1,M2,G1,G2,
     # A1,A2,D(60,60),F1(60,60),F2(60,60),P1(60),P2(60),
     # S1(60),S2(60),W1(60),W2(60)
      DIMENSION TA(20),UA2(60,60),UA3(60,60),UA4(60,60),VA1(60),
     #VA2(60),VA3(60),VA4(60),FMTN(10),FMTA(10),HAS(10)
      EQUIVALENCE (UA2,D),(UA3,F1),(UA4,F2),(VA1,P1,VA2,VA3,VA4)
      DATA FMTN/'(4X,','I3 ,',',2X,',' ','X,','10(','F',' 12.','4',')'/
      DATA FMTA/'(1X,','2A4,','   ',' ','X,','10(','F',' 12.','4',')'/
      DATA HAS/'2','14','26','38','50','62','74','86','98','110'/
      DATA HAB, HA1, HA2, HA4/' ','1','2','4'/
      DATA HD, HF1, HF2, HP1, HP2, HW1, HW2, HS1, HS2, HE, HF
     #/    'D','F1','F2','P1','P2','W1','W2','S1','S2','E','F'/
    1 FORMAT(' ')
    2 FORMAT('0')
    3 FORMAT('-')
    9 FORMAT('1')
      READ (5,10) LAR,LAF,LAP,MAU2,MAU3,MAU4,MAV1,MAV2,MAV3,MAV4,NAT
   10 FORMAT(I1,2A1,4X,3L1,1X,4L1,I2)
      WRITE(6,9)
      WRITE(6,3)
      IF(NAT.LE.0)GO TO 140
      DO 139 IA=1,NAT
      READ (5,13) (TA(JA),JA=1,20)
   13 FORMAT (20A4)
      WRITE(6,14)(TA(JA),JA=1,20)
   14 FORMAT(26X,20A4)
  139 CONTINUE
  140 WRITE(6,3)
      IF(NP.EQ.0) NP=N
      WRITE(6,15)
   15 FORMAT (5X,'PROGRAM CONSTANTS:')
      WRITE (6,16) NR,N,NP
   16 FORMAT('-',10X,'RUN',I3,';',I5,' NODES;'I5,' PRIMARY NODES')
      WRITE (6,17) HA1,G1,M1,N1,A1
   17 FORMAT('-',10X,'EQUATION ',A1,': G=',E15.8,', M=', I5,', N='I5,
     # ', A=',F5.2)
      IF(LAR.LE.0)GO TO 180
      WRITE (6,17) HA2,G2,M2,N2,A2
  180 HAU1=HAB
      HAU2=HD
      HAU3=HF1
      HAU4=HF2
      HAV1=HP1
      LAR=LAR+1
      GOTO(210,215,220),LAR
```

TABLE XXVII (Continued)

```
210 DO 214 IA=1,N
    VA2(IA)=P1(IA)
    VA3(IA)=W1(IA)
214 VA4(IA)=S1(IA)
    MAV1=.TRUE.
    MAU4=.TRUE.
    HAV2=HP1
    HAV3=HW1
    HAV4=HS1
    GO TO 225
215 DO 219 IA=1,N
    VA2(IA)=W1(IA)
    VA3(IA)=P2(IA)
219 VA4(IA)=W2(IA)
    HAV2=HW1
    HAV3=HP2
    HAV4=HW2
    GO TO 225
220 DO 224 IA=1,N
    VA2(IA)=S2(IA)
    VA3(IA)=P2(IA)
224 VA4(IA)=W2(IA)
    HAV2=HS1
    HAV3=HP2
    HAV4=HS2
225 IF(MAU2) HAU2=HAB
    IF(MAU3) HAU3=HAB
    IF(MAU4) HAU4=HAB
    IF(MAV1) HAV1=HAB
    IF(MAV2) HAV2=HAB
    IF(MAV3) HAV3=HAB
    IF(MAV4) HAV4=HAB
    IF(LAF.NE.HE) LAF=HF
    IF(LAP.EQ.HAB) LAP=HA4
    FMTN(7)=LAF
    FMTA(7)=LAF
    FMTN(9)=LAP
    FMTA(9)=LAP
    KAC2=0
    JAC2=(N+9)/10
    DO 890 JAC=1,JAC2
    FMTN(4)=HAS(1)
    FMTA(4)=HAS(1)
    KAC1=KAC2+1
    KAC2=KAC1+9
    IF(KAC2.GT.N)KAC2=N
    KAR2=0
    IF(JAC.LE.1)GO TO 700
    JACM1=JAC-1
    DO 599JAR=1,JACM1
    KAR1=KAR2+1
    KAR2=KAR1+9
    IF(JAR.GT.(NP+9)/10) GO TO 700
    WRITE(6,41) JAR,JAC,NR
 41 FORMAT('1PAGE ',I2,',',I2,5X,'RUN',I3)
    WRITE (6,42) HAU1,HAV1
 42 FORMAT(2X,A2,3X,A2,2X,10(5X,I3,4X) )
    WRITE(6,42) HAU2,HAV2,(NN(KAC),KAC=KAC1,KAC2)
    WRITE(6,43) HAU3,HAV3,(IDEN1(KAC),IDEN2(KAC),KAC=KAC1,KAC2)
```

TABLE XXVII (Continued)

```
 43 FORMAT(2X,A2,3X,A2,2X,10(2X,2A4,2X) )
    WRITE(6,43) HAU4,HAV4
    DO 589 KAR=KAR1,KAR2
    WRITE(6,2)
    IF(MAU2)GO TO 525
    WRITE(6,FMTN) NN(KAR),(UA2(KAR,KAC),KAC=KAC1,KAC2)
    GO TO 530
525 WRITE (6,FMTN)NN(KAR)
530 IF(MAU3) GO TO 535
    WRITE(6,FMTA) IDEN1(KAR),IDEN2(KAR),(UA3(KAR,KAC),KAC=KAC1,KAC2)
    GO TO 540
535 WRITE(6,FMTA)IDEN1(KAR),IDEN2(KAR)
540 IF(MAU4)GO TO 545
    WRITE(6,FMTA)HAV,HAB,(UA4(KAR,KAC),KAC=KAC1,KAC2)
    GO TO 550
545 WRITE(6,1)
550 CONTINUE
589 CONTINUE
599 CONTINUE
700 WRITE(6,41) JAC,JAC,NR
    WRITE (6,42) HAU1,HAV1
    WRITE(6,42) HAU2,HAV2,(NN(KAC),KAC=KAC1,KAC2)
    WRITE(6,43) HAU3,HAV3,(IDEN1(KAC),IDEN2(KAC),KAC=KAC1,KAC2)
    WRITE(6,43) HAU4,HAV4
    LAM2=1
    IF(MAV2)LAM2=2
    IF(MAU2)LAM2=LAM2+2
    LAM3=1
    IF(MAV3) LAM3=2
    IF(MAU3)LAM3=LAM3+2
    LAM4=1
    IF(MAV4)LAM4=2
    IF(MAU4)LAM4=LAM4+2
    KA=0
    DO 889 KAR=KAC1,KAC2
    KA=KA+1
    KARP1=KAR+1
    IF(KAR.LT.KAC2)GO TO 809
    IF(LAM2.LE.2)LAM2=LAM2+2
    IF(LAM3.LE.2) LAM3=LAM3+2
    IF(LAM4.LE.2) LAM4=LAM4+2
809 FMTN(4)=HAS(KA)
    FMTA(4)=HAS(KA)
    WRITE(6,1)
    IF(MAV1) GO TO 815
    WRITE(6,FMTA) HAB,HAB,VA1(KAR)
    GO TO 820
815 WRITE(6,1)
820 GO TO (821,822,823,824),LAM2
821 WRITE(6,FMTN)NN(KAR),VA2(KAR),(UA2(KAR,KAC),KAC=KARP1,KAC2)
    GO TO 830
822 FMTN(4)=HAS(KA+1)
    WRITE(6,FMTN)NN(KAR),(UA2(KAR,KAC),KAC=KARP1,KAC2)
    FMTN(4)=HAS(KA)
    GO TO 830
823 WRITE(6,FMTN)NN(KAR),VA2(KAR)
    GO TO 830
824 WRITE(6,FMTN)NN(KAR)
830 GO TO(831,832,833,834),LAM3
```

TABLE XXVII (Continued)

```
 831 WRITE(6,FMTA)IDEN1(KAR),IDEN2(KAR),VA3(KAR),
   #       (UA3(KAR,KAC), KAC = KARP1, KAC2)
     GO TO 840
 832 FMTA(4)=HAS(KA+1)
     WRITE(6,FMTA) IDEN1(KAR),IDEN2(KAR),(UA3(KAR,KAC),KAC=KARP1,KAC2)
     FMTA(4)=HAS(KA)
     GO TO 840
 833 WRITE(6,FMTA)IDEN1(KAR),IDEN2(KAR),VA3(KAR)
     GO TO 840
 834 WRITE(6,FMTA)     IDEN1(KAR),IDEN2(KAR)
 840 GO TO (841,842,843,844),LAM4
 841 WRITE(6,FMTA)HAB,HAB,VA4(KAR),(UA4(KAR,KAC),KAC=KARP1,KAC2)
     GO TO 850
 842 FMTA(4)=HAS(KA+1)
     WRITE(6,FMTA)HAB,HAB,(UA4(KAR,KAC),KAC=KARP1,KAC2)
     FMTA(4)=HAS(KA)
     GO TO 850
 843 WRITE(6,FMTA) HAB,HAB,VA4(KAR)
     GO TO 850
 844 WRITE(6,1)
 850 CONTINUE
 889 CONTINUE
 890 CONTINUE
     NCELL=N-10*(N/10)
     NROW=5*NCELL+10
     DO 900 K=1,NROW
 900 WRITE(6,1)
     RETURN
     END
//GO.SYSIN DD *
```

B.6 Program 2. This program executes the following programs: 1) LDSPWB; 2) FMTLKVOL; and 3) NTOPCOST. A listing for the four time periods used in the 53-node problem is shown in Table XXVIII. Each of the programs are repeated for each of the four different time periods.

TABLE XXVIII

PROGRAM 2 LISTING

```
//CHAP2  JOB (10652,441-40-6462,1,,,9001,3),'L D CHAPMAN'
//JOBLIB DD DISP=OLD,
//         DSN=OSU.ACT10652.CHAP
//        DD DSN=OSU.ACT10652.CHAP2,DISP=OLD
//STEP1 EXEC PGM=LDSPWB,TIME=(,10),REGION=33K
//CPNTAPE CD DUMMY
//PATHSI DD DISP=OLD,
//         DSN=OSU.ACT10652.TREES
//TRIPSI DD DISP=OLD,DSN=OSU.ACT10652.TT1
//NWRCDI DD DISP=CLD,
//         DSN=OSU.ACT10652.SPNET
//NWRCCO DD DISP=(NEW,PASS),
//         DSN=&&LDNET,UNIT=DISK,SPACE=(TRK,10),
//         CCB=(RECFM=VBS,LRECL=84,BLKSIZE=1000)
//SYSIN DD *,DCB=BLKSIZE=80
NL07      VOLAB       10  1-4       12       4   VOL  A  TO  B  FIRST  LOAD
NL08      VCLBA       11  1-4       16       4   VOL  B  TO  A  FIRST  LOAD
V1A-A          12   16
V2         LCCVAB1    12    92    2 VOLUME A-B FIRST LOAB
V2         LCCVAB2    16    94    2 VOLUME B-A FIRST LOAB
V2         RESERVE          96  76   19 WORDS
E
//STEP2 EXEC PGM=FMTLKVCL,TIME=(,10),REGION=43K
//INCATA CD DISP=(OLD,DELETE),
//         DSN=&&LDNET
//OUTDATA DD DISP=(NEW,PASS),
//         DSN=&&LKNVOL,UNIT=DISK,SPACE=(TRK,10),
//               DCB=(RECFM=VB,LRECL=50,BLKSIZE=444)
//SYSPRINT DD SYSOUT=A
//STEP3 EXEC PGM=NTOPCOST,TIME=(,10),REGION=50K
//FTC5F001 DD *
153  8  6  1  8
          8          15          27          49          123          144
 3.7 5.2  3.5  8.1  3.0   3.7 3.6  4.3  3.3  2.5  5.6  3.4  4.7  2.8  6.8  5.3
3.9  3.3  3.9  4.8  1.8  5.0  8.4  5.3  4.7  3.6  3.4  6.0  3.4  2.1  5.1  3.9
9.2  3.1  2.8  3.7  2.6  3.6  1.5  2.1  5.2  4.5  3.8  5.6  2.2  2.5  2.2  6.8
6.8  2.7  2.9  1.8  2.3  4.3  4.0  1.5  2.7  3.3  3.1  4.5  2.3  2.4  3.2  2.0
3.3  6.5  2.3  4.1  5.0  5.2  4.1  2.7  2.5  5.3  6.7  8.9  3.8  6.5  4.4  4.0
3.2  2.6  4.3  2.5  6.7  4.6  7.0  6.7  8.1  3.0  8.8  6.1  1.8  5.8  5.2  4.3
4.5  5.6  6.0  8.9  2.7  1.7  3.9  6.6  2.7  3.3  3.6  5.7  3.5  1.8  2.2  3.2
1.7  2.4  2.5  3.1  4.0  1.7  5.1  3.1  2.3  3.6  2.3  3.3  2.9  2.5  4.5  4.3
3.2  2.8  2.9  3.8  2.8  2.8  4.5  2.3  4.0  3.0  5.7  2.4  4.3  4.9  4.1  3.1
6.9  2.7  4.5
4.3  4.4  4.7  5.0
5.0  5.3  5.6  5.7
4.1  4.2  4.6  4.9
4.4  4.6  4.8  5.0
4.8  5.0  5.4  5.5
9.1  9.3  9.5  9.9
7.2  7.7  7.8  7.9
5.2  5.4  5.6  5.8
4.1  4.2  4.3  4.4
4.6  4.7  4.9  5.1
3.7  3.9  4.0  4.3
3.8  4.2  4.2  4.3
4.5  4.5  4.7  4.8
8.0  8.2  8.7  9.0
6.5  6.7  7.0  7.1
4.6  4.7  4.8  5.0
  161 2 2 2 2 0 0 2 2
  162 2 2 4 2 0 0 2 2
  163 2 4 2 2 0 0 2 2
  164 4 2 2 2 0 0 2 2
  165 2 2 2 4 0 0 2 2
  166 2 2 4 4 0 0 2 2
  167 2 4 2 4 0 0 2 2
  168 4 2 2 4 0 0 2 2
//FTC6F001 DD SYSOUT=A
//FTC8F001 DD DISP=(OLD,DELETE),
//         DSN=&&LKNVCL
//FTC9F001 DD DISP=MOD,
//         DSN=OSU.ACT10652.NTOC,
//         DCB=(RECFM=VB,LRECL=99,BLKSIZE=500)
```

TABLE XXVIII (Continued)

```
//STEP4 EXEC PGM=LDSPWB,TIME=(,10),REGION=33K
//CPATAPE DD DUMMY
//PATHSI DD DISP=OLD,
//            DSN=OSU.ACT10652.TREES
//TRIPSI DD DISP=OLD,DSN=OSU.ACT10652.TT2
//NWRCDI DD DISP=OLD,
//            DSN=OSU.ACT10652.SPNET
//NWRCDO DD DISP=(NEW,PASS),
//            DSN=&&LDNET,UNIT=DISK,SPACE=(TRK,10),
//            DCB=(RECFM=VBS,LRECL=84,BLKSIZE=1000)
//SYSIN DD *,DCB=BLKSIZE=80
NL07      VCLAB      10  1-4      12      4  VOL A TO B FIRST LOAD
NL08      VCLBA      11  1-4      16      4  VOL B TO A FIRST LOAD
V1A-A          12   16
V2         LCCVAB1   12    92    2 VOLUME A-B FIRST LOAB
V2         LCCVAB2   16    94    2 VOLUME B-A FIRST LOAB
V2         RESERVE         96   76   19 WORDS
E
//STEP5 EXEC PGM=FMTLKVCL,TIME=(,10),REGION=43K
//INDATA DD DISP=(OLD,DELETE),
//            DSN=&&LDNET
//OUTDATA DD DISP=(NEW,PASS),
//            DSN=&&LKNVOL,UNIT=DISK,SPACE=(TRK,10),
//                DCB=(RECFM=VB,LRECL=50,BLKSIZE=444)
//SYSPRINT DD SYSOUT=A
//STEP6 EXEC PGM=NTOPCOST,TIME=(,10),REGION=50K
//FTC5F001 DD *
153  8  6  2  8
         8          15          27          49          123          144
3.9  5.4  3.6  8.4  3.2  3.9  3.9  4.5  3.5  2.7  5.8  3.6  4.8  3.0  7.0  5.6
4.1  3.5  4.3  5.0  2.0  5.2  8.6  5.4  5.0  3.7  3.8  6.3  3.5  2.5  5.5  4.1
9.3  3.2  3.0  4.0  2.7  3.7  1.7  2.4  5.5  4.6  4.2  5.8  2.4  2.7  2.4  7.3
7.0  3.0  3.1  1.9  2.4  4.6  4.4  1.7  3.0  3.5  3.3  4.9  2.5  2.7  3.4  2.1
3.5  6.9  2.5  4.5  5.2  5.5  4.2  2.8  2.6  5.5  7.1  9.1  4.2  7.0  4.8  4.0
3.6  3.0  4.6  2.7  7.2  5.0  7.3  7.0  8.3  3.2  9.3  6.3  2.0  5.9  5.3  4.7
4.8  5.7  6.3  9.2  2.9  2.0  4.2  7.2  3.0  3.7  4.0  6.0  3.8  2.3  2.5  3.4
2.0  2.8  2.5  3.4  4.1  2.0  5.4  3.1  2.5  4.0  2.4  3.5  3.3  2.6  5.1  4.4
3.5  3.0  3.1  4.2  3.0  3.0  5.0  2.3  4.0  3.2  6.2  2.5  4.4  5.2  4.3  3.2
7.2  3.0  4.8
4.3  4.4  4.7  5.0
5.0  5.3  5.6  5.7
4.1  4.2  4.6  4.9
4.4  4.6  4.8  5.0
4.8  5.0  5.4  5.5
9.1  9.3  9.5  9.9
7.2  7.7  7.8  7.9
5.2  5.4  5.6  5.8
4.1  4.2  4.3  4.4
4.6  4.7  4.9  5.1
3.7  3.9  4.0  4.3
3.8  4.2  4.2  4.3
4.5  4.5  4.7  4.8
8.0  8.2  8.7  9.0
6.5  6.7  7.0  7.1
4.6  4.7  4.8  5.0
  161 2 2 2 2 0 0 2 2
  162 2 2 4 2 0 0 2 2
  163 2 4 2 2 0 0 2 2
  164 4 2 2 2 0 0 2 2
  165 2 2 2 4 0 0 2 2
  166 2 2 4 4 0 0 2 2
  167 2 4 2 4 0 0 2 2
  168 4 2 2 4 0 0 2 2
//FTC6F001 DD SYSOUT=A
//FT08F001 DD DISP=(OLD,DELETE),
//            DSN=&&LKNVCL
//FTC9F001 DD DISP=MOD,
//            DSN=OSU.ACT10652.NTOC,
//            DCB=(RECFM=VB,LRECL=99,BLKSIZE=500)
```

TABLE XXVIII (Continued)

```
//STEP7 EXEC PGM=LDSPWB,TIME=(,10),REGION=33K
//CPNTAPE CC CUMMY
//PATHSI DD DISP=OLD,
//          DSN=OSU.ACT10652.TREES
//TRIPSI DD DISP=OLD,DSN=OSU.ACT10652.TT3
//NWRCDI DD DISP=OLD,
//          DSN=OSU.ACT10652.SPNET
//NWRCDO DD DISP=(NEW,PASS),
//          DSN=&&LDNET,UNIT=DISK,SPACE=(TRK,10),
//          DCB=(RECFM=VBS,LRECL=84,BLKSIZE=1000)
//SYSIN DD *,DCB=BLKSIZE=80
NL07      VOLAB      10  1-4      12      4  VOL A TO B FIRST LOAD
NL08      VOLBA      11  1-4      16      4  VOL B TO A FIRST LOAD
V1A-A          12   16
V2         LOCVAB1    12    92    2 VOLUME A-B FIRST LOAB
V2         LOCVAB2    16    94    2 VOLUME B-A FIRST LOAB
V2         RESERVE          96   76   19 WORDS
E
//STEP8 EXEC PGM=FMTLKVCL,TIME=(,10),REGION=43K
//INDATA CD DISP=(OLD,DELETE),
//          CSN=&&LDNET
//OUTDATA DD DISP=(NEW,PASS),
//          DSN=&&LKNVOL,UNIT=DISK,SPACE=(TRK,10),
//               DCB=(RECFM=VB,LRECL=50,BLKSIZE=444)
//SYSPRINT DD SYSOUT=A
//STEP9 EXEC PGM=NTOPCOST,TIME=(,10),REGION=50K
//FTC5F001 DD *
153  8  6  3  8
        8          15          27          49          123         144
4.0  5.7  3.8  8.5  3.6  4.0  4.2  4.8  3.8  3.0  5.9  3.9  5.0  3.4  7.3  5.7
4.2  3.8  4.4  5.3  2.3  5.4  8.8  5.6  5.2  3.8  3.9  6.4  3.6  2.6  5.6  4.4
9.8  3.6  3.4  4.2  3.2  3.8  1.9  2.6  5.7  5.0  4.3  6.0  2.6  2.8  2.6  7.4
7.4  3.2  3.4  2.0  2.7  4.8  4.5  1.8  3.2  3.7  3.7  5.2  2.8  3.0  3.6  2.3
3.8  7.3  2.7  4.6  5.4  5.6  4.3  3.1  2.8  5.6  7.3  9.5  4.3  7.1  5.0  4.2
3.6  3.0  4.8  2.8  7.4  5.1  7.6  7.2  8.4  3.4  9.7  6.4  2.1  6.0  5.4  4.8
4.9  5.8  6.5  9.7  3.2  2.2  4.5  7.3  3.3  3.8  4.2  6.2  3.9  2.5  2.7  3.6
2.3  2.9  2.6  3.5  4.2  2.3  5.6  3.4  2.7  4.2  2.6  3.7  3.4  2.8  5.1  4.9
3.7  3.3  3.5  4.5  3.3  3.6  5.2  2.5  4.3  3.4  6.5  2.7  4.6  5.2  4.6  3.4
7.6  3.1  5.2
4.3  4.4  4.7  5.0
5.0  5.3  5.6  5.7
4.1  4.2  4.6  4.9
4.4  4.6  4.8  5.0
4.8  5.0  5.4  5.5
9.1  9.3  9.5  9.9
7.2  7.7  7.8  7.9
5.2  5.4  5.6  5.8
4.1  4.2  4.3  4.4
4.6  4.7  4.9  5.1
3.7  3.9  4.0  4.3
3.8  4.2  4.2  4.3
4.5  4.5  4.7  4.8
8.0  8.2  8.7  9.0
6.5  6.7  7.0  7.1
4.6  4.7  4.8  5.0
  161 2 2 2 2 0 0 2 2
  162 2 2 4 2 0 0 2 2
  163 2 4 2 2 0 0 2 2
  164 4 2 2 2 0 0 2 2
  165 2 2 2 4 0 0 2 2
  166 2 2 4 4 0 0 2 2
  167 2 4 2 4 0 0 2 2
  168 4 2 2 4 0 0 2 2
//FTC6F001 DD SYSOUT=A
//FTC8F001 DD DISP=(OLD,DELETE),
//          CSN=&&LKNVCL
//FTC9F001 DD DISP=MOD,
//          DSN=OSU.ACT10652.NTOC,
//          DCB=(RECFM=VB,LRECL=99,BLKSIZE=500)
```

TABLE XXVIII (Continued)

```
//STEP10 EXEC PGM=LDSPWB,TIME=(,10),REGION=33K
//DPNTAPE DD DUMMY
//PATHSI DD DISP=OLD,
//          DSN=OSU.ACT10652.TREES
//TRIPSI DD DISP=OLD,DSN=OSU.ACT10652.TT4
//NWRCDI DD DISP=OLD,
//          DSN=OSU.ACT10652.SPNET
//NWRCDO DD DISP=(NEW,PASS),
//          DSN=&&LDNET,UNIT=DISK,SPACE=(TRK,10),
//          DCB=(RECFM=VBS,LRECL=84,BLKSIZE=1000)
//SYSIN DD *,DCB=BLKSIZE=80
NLC7      VOLAB       10  1-4      12       4  VOL A TO B FIRST LOAD
NLO8      VOLBA       11  1-4      16       4  VOL B TO A FIRST LOAD
V1A-A         12   16
V2         LOCVAB1    12    92    2 VOLUME A-B FIRST LOAB
V2         LCCVAB2    16    94    2 VOLUME B-A FIRST LOAB
V2         RESERVE          96   76   19 WORDS
E
//STEP11 EXEC PGM=FMTLKVOL,TIME=(,10),REGION=43K
//INDATA DD DISP=(OLD,DELETE),
//          DSN=&&LDNET
//OUTDATA DD DISP=(NEW,PASS),
//          DSN=&&LKNVOL,UNIT=DISK,SPACE=(TRK,10),
//              DCB=(RECFM=VB,LRECL=50,BLKSIZE=444)
//SYSPRINT DD SYSOUT=A
//STEP12 EXEC PGM=NTOPCOST,TIME=(,10),REGION=50K
//FTC5F001 DD *
153  8  6  4  8
         8         15         27         49         123         144
4.2  6.0  3.9  8.7  3.7  4.1  4.5  5.0  4.1  3.2  6.0  4.0  5.2  3.5  7.4  5.9
4.4  3.9  4.6  5.4  2.4  5.5  8.9  5.8  5.3  4.2  4.1  6.5  3.7  2.8  5.7  4.5
9.9  3.8  3.5  4.3  3.3  4.3  2.3  2.7  5.8  5.2  4.4  6.2  2.8  2.9  2.7  7.5
7.5  3.3  3.5  2.3  2.9  4.9  4.6  2.0  3.3  3.9  3.8  5.4  3.0  3.1  3.8  2.5
4.0  7.4  2.8  4.8  5.6  5.7  4.4  3.4  3.1  5.7  7.5  9.6  4.4  7.2  5.3  4.7
3.8  3.2  5.0  2.9  7.6  5.2  7.7  7.4  8.5  3.6  9.8  6.5  2.2  6.3  5.8  4.9
5.0  6.3  6.6  9.9  3.2  2.2  4.6  7.4  3.4  3.9  4.3  6.4  4.0  2.6  2.9  3.8
2.3  3.0  2.7  3.8  4.4  2.4  5.9  3.6  3.0  4.5  2.7  4.0  3.6  3.0  5.2  5.0
4.0  3.6  3.8  4.7  3.5  3.6  5.2  2.5  4.5  3.8  6.6  3.0  4.8  5.4  4.6  3.5
7.7  3.2  5.3
4.3  4.4  4.7  5.0
5.0  5.3  5.6  5.7
4.1  4.2  4.6  4.9
4.4  4.6  4.8  5.0
4.8  5.0  5.4  5.5
9.1  9.3  9.5  9.9
7.2  7.7  7.8  7.9
5.2  5.4  5.6  5.8
4.1  4.2  4.3  4.4
4.6  4.7  4.9  5.1
3.7  3.9  4.0  4.3
3.8  4.2  4.2  4.3
4.5  4.5  4.7  4.8
8.0  8.2  8.7  9.0
6.5  6.7  7.0  7.1
4.6  4.7  4.8  5.0
  161 2 2 2 2 0 0 2 2
  162 2 2 4 2 0 0 2 2
  163 2 4 2 2 0 0 2 2
  164 4 2 2 2 0 0 2 2
  165 2 2 2 4 0 0 2 2
  166 2 2 4 4 0 0 2 2
  167 2 4 2 4 0 0 2 2
  168 4 2 2 4 0 0 2 2
//FTC6F001 DD SYSOUT=A
//FTC8F001 DD DISP=(OLD,DELETE),
//          DSN=&&LKNVCL
//FTC9F001 DD DISP=MOD,
//          DSN=OSU.ACT10652.NTOC,
//          DCB=(RECFM=VB,LRECL=99,BLKSIZE=500)
//
```

B.7 FMTLKVOL Program. This program was written to convert the output of the LDSPWB program into a readable Fortran form. The program was written in the PL1 language by Fred Witz. The listing of the program is shown in Table XXIX. The output of the program is the link volumes for each link in the spiderweb network with a sequencing index which is assigned to every link for identification purposes.

TABLE XXIX

FORMAT LINK VOLUME (FMTLKVOL) PROGRAM

```
$LIST
//FMLKV  JOB (10652,441-40-6462,1,,,9001,3),'L D CHAPMAN'
//LOAD EXEC PL1LFCL
//PL1L.SYSIN DD *
                              /*        LEVEL=1971.0202.1630              */

        /*
        PROGRAM   TO        CONVERT OUTPUT OF LDSPWB
                                      TO FORTRAN READABLE FORMAT
                  BY        FRED WITZ
                  FOR       LEON CHAPMAN
        INPUT
                  RECORDS   VARIABLE BLOCKED OS/360 STANDARD
                  LENGTH    VARIES
                  FORMAT    AS SPECIFIED BY LCSPWB
        OUTPUT
                  RECORDS   VARIABLE BLOCKED OS/360 STANDARD
                  LENGTH    21 BYTES
                  FORMAT    AS FOLLOWS
        OFFSET IN RECORD
        |   LENGTH
        V         INFORMATION
         0    1  CARRIAGE CONTROL CHARACTER
         1   10  SEQUENCE NUMBER
        11   10  TOTAL VOLUME ON LINK
*/

CCNV :  PROC OPTIONS(MAIN) ;

        DCL
                              /*        STRUCTURES FOR INPUT RECORDS:   */
                                        /*                              */
                                        /*  PARAMETER RECORD            */
        1    PARM             BASED(APARM),
             2   UNKNOWN      CHAR(1),
                                        /*  NUMBER OF LAST NODE         */
/*  0*/      2   CLAST        CHAR(4),
/*  4*/      2   FILL         CHAR(168),
/*172*/
                                        /*  NODE INFORMATION RECORD     */
        1    REC              BASED(AREC),
                                        /*  NUMBER OF LINKS             */
/*  0*/      2   CNUM         CHAR(1),
/*  1*/      2   FILL         CHAR(23),
                                        /*  LINK INFORMATION            */
/* 24*/      2   LINK(8),
                                        /*  FROM-NODE NUMBER*8+LINK #   */
/*  0*/          3   IA       FIXED BIN(15),
/*  2*/          3   FILL     CHAR(2),
                                        /*  TO-NODE NUMBER*8 + LINK #   */
/*  4*/          3   IB       FIXED BIN(15),
/*  6*/          3   FILL2    CHAR(6),
                                        /*  VOLUMES                     */
/* 12*/          3   VOLA     FIXED BIN(31),
/* 16*/          3   VOLB     FIXED BIN(31),
/* 20*/

                                        /*  WORK VARIABLES              */
```

TABLE XXIX (Continued)

```
        LAST             FIXED BIN(31),
        NUM              FIXED BIN(31),
        (I, ILINK, IREC,
        IAI, IBI)        FIXED BIN(31),

                                   /*  DATA SETS                    */
        INDATA           FILE SEQL INPUT,
        OUTDATA          FILE PRINT,
        SYSPRINT         FILE PRINT ;

    OPEN
        FILE(INDATA),
        FILE(OUTDATA),
        FILE(SYSPRINT) LINESIZE(125) ;

    PUT FILE(SYSPRINT) LIST('$$$$$$$$$ CONVERTER PROGRAM') ;
                                   /*  SKIP RECORDS LISTING         */
                                   /*  FORMAT IN ENGLISH            */
    READ FILE(INDATA) IGNORE(51) ;
                                   /*  READ PARAMETER RECORD        */
                                   /*  (IN BUFFER)                  */
    READ FILE(INDATA) SET(APARM) ;
                                   /*  MOVE LAST TO ALIGN           */
    UNSPEC(LAST) = UNSPEC(CLAST) ;
    ILINK = 0 ;
                                   /*  ITERATE FOR ALL NODES        */
    DO IREC = 1 TO LAST ;
                                   /*  READ NODE INFO. RECORD       */
        READ FILE(INDATA) SET(AREC) ;
                                   /*  CONVERT NUM TO FULLWORD      */
        UNSPEC(NUM) = (24)'0'B || UNSPEC(CNUM) ;
        IAI = IA(1)/8 ;
        PUT FILE(SYSPRINT) SKIP(1) ;
                                   /*  ITERATE LINKS                */
        DO I = 1 TO NUM ;
                                   /*  ELIMINATE "LOWER TRIANGLE"   */
          IBI = IB(I)/8 ;
          IF IBI > IAI THEN DO ;
            ILINK = ILINK + 1 ;
                                   /*  PRINT INFORMATION            */
            PUT FILE(SYSPRINT)
                EDIT(ILINK, IAI, IBI, VOLA(I) + VOLB(I) )
                ( (3)F(5), F(10) ) ;
                                   /*  COPY INFO. TO DISK           */
            PUT FILE(OUTDATA)
                EDIT(ILINK, VOLA(I) + VOLB(I) )
                (COL(1), (4)F(10) ) ;
            END ;
          END ;
        IF IAI = LAST THEN GO TO EXIT ;
        END ;
 EXIT :
    PUT FILE(SYSPRINT) SKIP(3) LIST('$$$$$$$$$ STANDARD END') ;
    END CONV ;
//LKED.SYSLMOD DD DISP=OLD,
//        DSN=OSU.ACT10652.CHAP2(FMTLKVOL)
//
$ENDLIST
```

B.8. NTOPCOST Program. Table XXX is the computer listing of the NTOPCOST program. This program calculates the network operators' cost for a fixed network configuration. Two to four-lane changes which construct new feasible networks are calculated by this program. This permits the user to calculate the network operators' cost of several networks with only one run of the BPR programs.

TABLE XXX

NETWORK OPERATORS' COST PROGRAM

```
$LIST
//NTCCST JOB (10652,441-40-6462,1,,,9001,3),'L D CHAPMAN'
//LOAD EXEC FORTGCL
//FORT.SYSIN DD *
      INTEGER P,PA(20)
      REAL NOC(205,6),LVOL,LKC2L(8,4),LKC4L(8,4)
      DIMENSION L(300),LKVOL(300),LC(300),LKVOLS(300),
     #LVCL(300),OCL(300)
      COMMON OCL,PA,LKC2L,LKC4L
C
C     M = NUMBER OF LKS IN NETWORK
C     LA = NUMBER OF POSSIBLE LK ADDITIONS IN COMPLETE NETWORK
C     LACN = NUMBER OF POSSIBLE LK ADDITIONS IN CURRENT NETWORK
C     L(I) = LK NUMBER
C     LKVOL(I) = LK VOLUME ON LK I
C     LC(I) = LK CHANGE NUMBER
C     PA(I)=POSSIBLE ALTERNATIVE INVESTMENT IN ROAD I
C     OCL(I)=OPERATORS COST FOR LINK I
C     NOC(NN)=NETWORK OPERATORS COST FOR NETWORK NN
C     NN=NETWORK NUMBER
C     P=PERIOD
C     NRUNS = NUMBER OF RUNS WITH THIS NETWORK
C
      READ(5,1) M,LA,LACN,P,NRUNS
    1 FORMAT(5I3)
C
C     READ LK NUMBER AND LK VOLUME
C
      READ(8,3) (L(I),LKVOL(I),I=1,M)
    3 FORMAT(1X,2I10)
C
C     READ LINK NUMBERS THAT MAY CHANGE
C
      READ(5,5) (LC(I),I=1,LACN)
    5 FORMAT(8I10)
      WRITE(6,7)
    7 FORMAT(30X,'L(I)',3X,'LKVOL(I)',5X,'ORIGINAL')
      WRITE(6,8)  (L(I),LKVOL(I),I=1,M)
    8 FORMAT(1X,I5,11I10)
      WRITE(6,9) (LC(I),I=1,LACN)
    9 FORMAT(///1X,'LK CHANGE NUMBERS =',20I3)
      NW=M-LACN+1
      JJ=M-LACN
C
C     TEST FOR LK CHANGES
C
      IF(LC(I).EQ.0) GO TO 100
C
C     STATEMENTS THRU 40 REORDERS LKVOL SUCH THAT THE
C     LINKS THAT MAY CHANGE ARE PLACED AT BOTTOM OF LIST
C
C     STORE LKVOL THAT MAY CHANGE
C
      DO 10 JK=1,LACN
   10 LKVOLS(JJ+JK)=LKVOL(LC(JK))
C
C     MOVE ALL LKVOL UP TO FILL GAPS CREATED BY
C     LKVOL THAT CHANGE
```

TABLE XXX (Continued)

```
C
      JE=1
      LNC=LC(1)
      LLA = LACN-1
C
C     IF THERE IS ONLY ONE LK ADDITION POSSIBLE, GO TO STEP  25
C
      IF(LLA.LT.1) GO TO 25
      DO  20 J=1,LLA
      JE=JE+1
      N=LC(J+1)-(J+1)
C
C     INCREMENT VARIABLES IF LC(I) AND LC(I+1) ARE TWO
C     NUMBERS IN SEQUENCE
C
      IF(N.LT.LNC) GO TO  20
      DO  15 I=LNC,N
   15 LKVOL(I)=LKVOL(I+J)
   20 LNC=LC(J+1)-J
   25 NE=M-JE
C
C     IF LK CHANGE NUMBER IS CONTAINED IN LAST LA LK CHANGE
C     NUMBERS, THEN PLACE LKVOL THAT CHANGE AT END OF LIST
C
      IF(NE.LT.LNC) GO TO  35
      DO  30 II=LNC,NE
   30 LKVOL(II)=LKVOL(II+JE)
C
C     PLACE THE LKVCL THAT CHANGE AT END CF LIST
C
   35 CONTINUE
      DO 36 IB =NW,M
   36 LKVOL(IB) = LKVOLS(IB)
C
C     IF LACN IS GE 4, RECRDER LKVOL, PLACING LKVCL 9-12 AHEAD
C     CF LKVOL 4-33
C
      IF(LACN.LE.3) GO TO 41
      NABC = LKVOL(NW+2)
      LKVCL(NW+2) = LKVCL(NW+3)
      LKVOL(NW+3) = NABC
C
C     READ FIXED OCL(I) FOR THIS NETWORK ------MUST BE IN THE
C     CORRECT LK NUMBER SEQUENCE AFTER THE LKS THAT MAY CHANGE
C     ARE PLACED AT THE BOTTOM OF ORDERING SEQUENCE
C
   41 READ(5,42)  (OCL(I),I=1,JJ)
   42 FORMAT(16F5.1)
C
C     READ 2L AND 4L LK COST MATRIX THAT MAY CHANGE IN COMPLETE
C     NETWORK FOR EACH PERIOD
C
   43 READ(5,44)  ((LKC2L(I,J),J=1,4),I=1,LA)
   44 FORMAT(4F5.1)
      READ(5,44)  ((LKC4L(I,J),J=1,4),I=1,LA)
C
C     READ NETWORK NUMBER AND STATE
C
      READ(5,76) NN,(PA(NA),NA=1,LA)
```

TABLE XXX (Continued)

```
C
C     FIND OCL(I) THAT CHANGE FOR THIS NETWORK
C
      CALL OCLCHN(NW,LA,P)
      WRITE(6,45)
   45 FORMAT(//30X,'L(I)',3X,'LKVOL(I)',5X,'OCL(I)',5X,'REORDERED')
   47 WRITE(6, 50) (L(I),LKVOL(I),OCL(I),I=1,M)
   50 FORMAT(1X,I5,I10,F5.1,2I10,F5.1,2I10,F5.1,2I10,F5.1,2I10,F5.1)
   55 CONTINUE
   57 FORMAT(///1X,'NOC( NETWORK NUMBER, PERIOD)')
      WRITE(6,57)
      DO 60 I=1,M
   60 LVOL(I)=LKVOL(I)
C     CALCULATE NETWORK OPERATORS COST
C
      A=0.
      DO 75 I=1,JJ
   75 A=LVOL(I)*OCL(I)+A
C
C     CALCULATE NOC FOR EACH NETWORK THAT IS FEASIBLE
C
      DO 99  NNET=1,NRUNS
   76 FORMAT(I5,20I2)
   80 NOC(NN,P)=A
      DO 85 I=NW,M
   85 NOC(NN,P)=LVOL(I)*OCL(I)+NOC(NN,P)
      WRITE(6,90) NN,P,NOC(NN,P),(PA(I),I=1,LA),(OCL(I),I=NW,M)
   90 FORMAT(10X,'NOC(',I3,',',I2,') =',E14.7,
     #5X,'LK INVEST =',8I2,5X,'OCLS =',8F5.1)
      WRITE(9,92) NN,P,NOC(NN,P),(PA(I),I=1,LA)
   92 FORMAT(2I5,E14.7,20I2)
      IF(NNET.EQ.NRUNS) GO TO 99
C
C     READ NETWORK NUMBER AND STATE
C
      READ(5,76) NN,(PA(NA),NA=1,LA)
C
C     FIND OCL(I) THAT CHANGE FOR THIS NETWORK
C
      CALL OCLCHN(NW,LA,P)
   99 CONTINUE
      STOP
  100 WRITE(6,105)
  105 FORMAT(///9X,'L(I)',3X,'LKVOL(I)',5X,
     #'REORDERED SAME AS ORIGINAL')
      GO TO 41
      END
      SUBROUTINE OCLCHN(NW,LA,P)
      REAL LKC2L(8,4),LKC4L(8,4)
      INTEGER P,PA(20)
      COMMON OCL(300),PA,LKC2L,LKC4L
      LIN = NW
      DO 470 I=1,LA
      IF(PA(I).EQ.0) GO TO 470
      IF(PA(I).EQ.2) GO TO 441
      OCL(LIN) = LKC4L(I,P)
      GO TO 442
  441 OCL(LIN) = LKC2L(I,P)
  442 LIN = LIN + 1
  470 CONTINUE
      RETURN
      END
//LKED.SYSLMOD DD DISP=OLD,
//         DSN=OSU.ACT10652.CHAP2(NTOPCOST)
//
$ENDLIST
```

B.9 Four Period Output From FMTLKVOL and NTOPCOST Programs.

Table XXXI represents the output of the FMTLKVOL program for the periods of 1970, 1975, 1980, and 1985. Table XXXII shows the corresponding output for the four periods of the NTOPCOST program. The output of the FMTLKVOL program is formated in rows of a maximum of five sets of numbers each containing four distinct numbers. Within a given set of numbers, the first number is the link number, the next two represent the node connections for this particular link, and the fourth number is the volume on the given link. The formated output of the NTOPCOST programs contains the original set of link numbers and corresponding link volumes. The links that change in this network are shown to be links 8, 15, 27, 49, 123, and 144. These link volumes are placed at the end of the list and the corresponding link operators' costs for each link are shown. The link operators' costs that change are shown beside each network operators' cost and link investment scheme.

TABLE XXXI

1970, 1975, 1980, AND 1985 OUTPUT FROM FMTLKVOL PROGRAM

$$$$$$$$$ CCNVERTER PROGRAM

1	1	11	4534	2	1	12	1780	3	1	20	6587	4	1	22	485	5	1	33	5016
6	1	34	5536	7	1	37	3C80												
8	2	5	3458	9	2	7	3274	10	2	15	4180	11	2	31	5880	12	2	34	5104
13	2	41	1382	14	2	48	2396	15	2	50	228								
16	3	8	1844	17	3	12	224	18	3	13	422	19	3	20	2390	20	3	21	668
21	3	32	288	22	3	35	330	23	3	52	878								
24	4	6	332	25	4	18	70	26	4	23	242	27	4	33	152	28	4	36	124
29	4	37	1686	30	4	40	700												
31	5	6	222	32	5	31	358	33	5	47	428	34	5	48	328				
35	6	10	362	36	6	23	20	37	6	40	770	38	6	48	936				
39	7	15	524	40	7	30	416	41	7	39	496	42	7	41	1224	43	7	50	376
44	8	9	248	45	8	12	228	46	8	20	922	47	8	45	32	48	8	52	158
49	9	12	588	50	9	45	350	51	9	46	326	52	9	52	10				
53	10	15	94	54	10	33	1938	55	10	34	336	56	10	40	596	57	10	48	1346
58	11	12	460	59	11	17	666	60	11	25	2298	61	11	34	996	62	11	51	278
63	12	17	328	64	12	2C	160	65	12	45	356								
66	13	32	198	67	13	35	150	68	13	49	420	69	13	53	36				
70	14	30	20	71	14	38	598	72	14	47	102				.				
73	15	16	410	74	15	34	524	75	15	41	96	76	15	50	84	77	15	51	1042
78	16	17	90	79	16	19	80	80	16	27	70	81	16	42	304	82	16	43	20
83	16	44	86	84	16	50	106												
85	17	25	122	86	17	42	460	87	17	44	96	88	17	45	366				
89	18	22	50	90	18	23	56	91	18	24	20	92	18	36	120				
93	19	27	18	94	19	39	326	95	19	43	4	96	19	50	20				
97	20	21	2219	98	20	37	44												
99	21	22	72	100	21	28	151	101	21	35	234	102	21	36	56	103	21	37	90
104	22	23	10	105	22	24	310	106	22	28	709	107	22	36	130				
108	23	36	20																
109	24	28	118	110	24	35	92	111	24	49	90	112	24	53	6				
113	25	35	0	114	25	42	648	115	25	51	380								
116	26	30	122	117	26	31	1830	118	26	38	208	119	26	41	294	120	26	47	30
121	27	43	38																
122	28	35	958	123	28	37	28												
124	29	43	22	125	29	44	90	126	29	46	226								
127	30	39	72	128	30	41	486												
129	31	38	650	130	31	41	218	131	31	47	392								
132	32	35	28	133	32	52	30												
134	33	34	348	135	33	37	176	136	33	40	924								
137	34	51	16																
138	35	49	156																
139	36	37	546																
140	37	40	44																
141	38	47	98																
142	39	50	70																
143	40	48	540																
144	42	44	8	145	42	50	58	146	42	51	442								
147	43	44	26																
148	44	45	16	149	44	46	22												
150	45	46	164																
151	48	51	6																
152	49	53	38																
153	50	51	64																

$$$$$$$$$ STANCARC END

TABLE XXXI (Continued)

88888888 CONVERTER PROGRAM

1	1	11	6220	2	1	12	23C0	3	1	20	8603	4	1	22	715	5	1	33	6604
6	1	34	71C6	7	1	37	3700												
8	2	5	4524	9	2	7	4488	10	2	15	6200	11	2	31	8840	12	2	34	6618
13	2	41	1982	14	2	48	3216	15	2	50	332								
16	3	8	3146	17	3	12	326	18	3	13	536	19	3	20	3190	20	3	21	910
21	3	32	402	22	3	35	512	23	3	52	1152								
24	4	6	404	25	4	18	96	26	4	23	288	27	4	33	202	28	4	36	176
29	4	37	1948	30	4	40	890.												
31	5	6	280	32	5	31	504	33	5	47	524	34	5	48	408				
35	6	10	498	36	6	23	24	37	6	40	978	38	6	48	1236				
39	7	15	774	40	7	30	704	41	7	39	662	42	7	41	1676	43	7	50	546
44	8	9	418	45	8	12	380	46	8	20	1426	47	8	45	52	48	8	52	240
49	9	12	878	50	9	45	562	51	9	46	428	52	9	52	14				
53	10	15	140	54	10	33	2508	55	10	34	488	56	10	40	836	57	10	48	1804
58	11	12	700	59	11	17	754	60	11	25	3298	61	11	34	1418	62	11	51	382
63	12	17	432	64	12	20	236	65	12	45	486								
66	13	32	238	67	13	35	186	68	13	49	502	69	13	53	50				
70	14	30	30	71	14	38	788	72	14	47	128								
73	15	16	508	74	15	34	734	75	15	41	152	76	15	50	136	77	15	51	1514
78	16	17	112	79	16	19	100	80	16	27	100	81	16	42	360	82	16	43	20
83	16	44	1C6	84	16	50	132												
85	17	25	170	86	17	42	636	87	17	44	110	88	17	45	536				
89	18	22	84	90	18	23	70	91	18	24	28	92	18	36	152				
93	19	27	30	94	19	39	428	95	19	43	4	96	19	50	26				
97	20	21	2837	98	20	37	60												
99	21	22	120	100	21	28	221	101	21	35	284	102	21	36	76	103	21	37	116
104	22	23	20	105	22	24	500	106	22	28	1187	107	22	36	216				
108	23	36	24																
109	24	28	192	110	24	35	140	111	24	49	130	112	24	53	8				
113	25	35	0	114	25	42	862	115	25	51	582								
116	26	30	206	117	26	31	2602	118	26	38	282	119	26	41	414	120	26	47	42
121	27	43	50																
122	28	35	848	123	28	37	50												
124	29	43	34	125	29	44	126	126	29	46	314								
127	30	39	116	128	30	41	802												
129	31	38	852	130	31	41	320	131	31	47	510								
132	32	35	42	133	32	52	46												
134	33	34	508	135	33	37	266	136	33	40	1182								
137	34	51	24																
138	35	49	198																
139	36	37	742																
140	37	40	64																
141	38	47	116																
142	39	50	102																
143	40	48	706																
144	42	44	16	145	42	50	98	146	42	51	612								
147	43	44	28																
148	44	45	18	149	44	46	24												
150	45	46	200																
151	48	51	12																
152	49	53	46																
153	50	51	88																

88888888 STANDARD END

TABLE XXXI (Continued)

$$$$$$$$$ CCNVERTER PRCGRAM

1	1	11	8316	2	1	12	3132	3	1	20	12897	4	1	22	928	5	1	33	8930
6	1	34	10855	7	1	37	5536												
8	2	5	6606	9	2	7	6430	10	2	15	9012	11	2	31	12554	12	2	34	10236
13	2	41	2536	14	2	48	4840	15	2	50	536								
16	3	8	5148	17	3	12	575	18	3	13	856	19	3	20	5823	20	3	21	1672
21	3	32	680	22	3	35	876	23	3	52	1958								
24	4	6	642	25	4	18	144	26	4	23	394	27	4	33	318	28	4	36	262
29	4	37	3210	30	4	40	1454												
31	5	6	382	32	5	31	636	33	5	47	688	34	5	48	564				
35	6	10	640	36	6	23	28	37	6	40	1344	38	6	48	1842				
39	7	15	1002	40	7	30	926	41	7	39	892	42	7	41	2096	43	7	50	744
44	8	9	578	45	8	12	474	46	8	20	1770	47	8	45	76	48	8	52	280
49	9	12	1094	50	9	45	664	51	9	46	534	52	9	52	16				
53	10	15	200	54	10	33	3308	55	10	34	638	56	10	40	1062	57	10	48	2520
58	11	12	977	59	11	17	984	60	11	25	3968	61	11	34	1992	62	11	51	533
63	12	17	580	64	12	20	300	65	12	45	630								
66	13	32	308	67	13	35	256	68	13	49	634	69	13	53	68				
70	14	30	38	71	14	38	1040	72	14	47	164								
73	15	16	712	74	15	34	1011	75	15	41	198	76	15	50	198	77	15	51	2059
78	16	17	144	79	16	19	128	80	16	27	134	81	16	42	454	82	16	43	30
83	16	44	156	84	16	50	168												
85	17	25	184	86	17	42	840	87	17	44	150	88	17	45	662				
89	18	22	116	90	18	23	84	91	18	24	36	92	18	36	212				
93	19	27	32	94	19	39	958	95	19	43	4	96	19	50	42				
97	20	21	3830	98	20	37	86												
99	21	22	142	100	21	28	290	101	21	35	396	102	21	36	114	103	21	37	222
104	22	23	22	105	22	24	612	106	22	28	1512	107	22	36	288				
108	23	36	36																
109	24	28	238	110	24	35	208	111	24	49	152	112	24	53	8				
113	25	35	2	114	25	42	1052	115	25	51	734								
116	26	30	262	117	26	31	3422	118	26	38	334	119	26	41	500	120	26	47	52
121	27	43	66																
122	28	35	1228	123	28	37	78												
124	29	43	44	125	29	44	168	126	29	46	400								
127	30	39	148	128	30	41	1194												
129	31	38	1198	130	31	41	410	131	31	47	762								
132	32	35	60	133	32	52	54												
134	33	34	746	135	33	37	376	136	33	40	1610								
137	34	51	36																
138	35	49	254																
139	36	37	986																
140	37	40	92																
141	38	47	150																
142	39	50	136																
143	40	48	1146																
144	42	44	24	145	42	50	132	146	42	51	832								
147	43	44	38																
148	44	45	34	149	44	46	36												
150	45	46	270																
151	48	51	16																
152	49	53	60																
153	50	51	130																

$$$$$$$$$ STANDARD END

TABLE XXXI (Continued)

$$$$$$$$$ CCNVERTER PRCGRAM

1	1	11	1C890	2	1	12	4000	3	1	20	16000	4	1	22	1182	5	1	33	11896
6	1	34	12886	7	1	37	6776												
8	2	5	9266	9	2	7	8228	10	2	15	11589	11	2	31	15544	12	2	34	12202
13	2	41	4202	14	2	48	6169	15	2	50	680								
16	3	8	6776	17	3	12	746	18	3	13	1022	19	3	20	7038	20	3	21	2118
21	3	32	852	22	3	35	1C84	23	3	52	2374								
24	4	6	906	25	4	18	196	26	4	23	508	27	4	33	466	28	4	36	364
29	4	37	4198	30	4	40	1598												
31	5	6	572	32	5	31	964	33	5	47	1090	34	5	48	830				
35	6	10	894	36	6	23	40	37	6	40	1742	38	6	48	2300				
39	7	15	1414	40	7	30	1354	41	7	39	1144	42	7	41	3362	43	7	50	1070
44	8	9	754	45	8	12	662	46	8	20	2364	47	8	45	112	48	8	52	376
49	9	12	14C8	50	9	45	898	51	9	46	656	52	9	52	22				
53	10	15	286	54	10	33	4490	55	10	34	890	56	10	40	1598	57	10	48	3388
58	11	12	1328	59	11	17	1178	60	11	25	5154	61	11	34	2718	62	11	51	728
63	12	17	746	64	12	20	418	65	12	45	886								
66	13	32	380	67	13	35	304	68	13	49	826	69	13	53	78				
70	14	30	58	71	14	38	1420	72	14	47	256								
73	15	16	921	74	15	34	1364	75	15	41	308	76	15	50	282	77	15	51	2604
78	16	17	188	79	16	19	164	80	16	27	164	81	16	42	595	82	16	43	36
83	16	44	212	84	16	50	248												
85	17	25	218	86	17	42	1C74	87	17	44	192	88	17	45	866				
89	18	22	158	90	18	23	110	91	18	24	48	92	18	36	280				
93	19	27	44	94	19	39	704	95	19	43	8	96	19	50	52				
97	20	21	4938	98	20	37	118												
99	21	22	206	100	21	28	368	101	21	35	488	102	21	36	150	103	21	37	290
104	22	23	30	105	22	24	806	106	22	28	1974	107	22	36	396				
108	23	36	48																
109	24	28	304	110	24	35	260	111	24	49	206	112	24	53	14				
113	25	35	6	114	25	42	1398	115	25	51	894								
116	26	30	372	117	26	31	4156	118	26	38	478	119	26	41	716	120	26	47	76
121	27	43	84																
122	28	35	1524	123	28	37	104												
124	29	43	60	125	29	44	216	126	29	46	504								
127	30	39	208	128	30	41	1632												
129	31	38	1570	130	31	41	622	131	31	47	1006								
132	32	35	78	133	32	52	76												
134	33	34	1002	135	33	37	468	136	33	40	2100								
137	34	51	50																
138	35	49	340																
139	36	37	1260																
140	37	40	120																
141	38	47	232																
142	39	50	192																
143	40	48	1514																
144	42	44	34	145	42	50	194	146	42	51	1057								
147	43	44	52																
148	44	45	52	149	44	46	48												
150	45	46	330																
151	48	51	23																
152	49	53	76																
153	50	51	170																

$$$$$$$$$ STANCARC END

TABLE XXXII

1970, 1975, 1980, AND 1985 OUTPUT FROM NTOPCOST PROGRAM

			L(I)	LKVCL(I)	ORIGINAL						
1	4934	2	1780	3	6587	4	485	5	5016	6	5536
7	3080	8	3498	9	3274	10	4180	11	5880	12	5104
13	1382	14	2396	15	228	16	1844	17	224	18	422
19	2390	20	668	21	288	22	330	23	878	24	332
25	70	26	242	27	152	28	124	29	1686	30	700
31	222	32	358	33	428	34	328	35	362	36	20
37	770	38	936	39	524	40	416	41	496	42	1224
43	376	44	248	45	228	46	922	47	32	48	158
49	588	50	350	51	326	52	10	53	94	54	1938
55	336	56	596	57	1346	58	460	59	666	60	2298
61	996	62	278	63	328	64	160	65	356	66	198
67	150	68	420	69	36	70	20	71	598	72	102
73	410	74	524	75	96	76	84	77	1042	78	90
79	80	80	70	81	304	82	20	83	86	84	106
85	122	86	460	87	96	88	366	89	50	90	56
91	20	92	120	93	18	94	326	95	4	96	20
97	2219	98	44	99	72	100	151	101	234	102	56
103	90	104	10	105	310	106	709	107	130	108	20
109	118	110	92	111	90	112	6	113	0	114	648
115	380	116	122	117	1830	118	208	119	294	120	30
121	38	122	558	123	28	124	22	125	90	126	226
127	72	128	486	129	650	130	218	131	392	132	28
133	30	134	348	135	176	136	924	137	16	138	156
139	546	140	44	141	98	142	70	143	540	144	8
145	58	146	442	147	26	148	16	149	22	150	164
151	6	152	38	153	64						

LK CHANGE NUMBERS = 8 15 27 49123144

			L(I)	LKVOL(I)		OCL(I)	REORDERED							
1	4934	3.7	2	1780	5.2	3	6587	3.5	4	485	8.1	5	5016	3.0
6	5536	3.7	7	3080	3.6	8	3274	4.3	9	4180	3.3	10	5880	2.5
11	5104	5.6	12	1382	3.4	13	2396	4.7	14	1844	2.8	15	224	6.8
16	422	5.3	17	2390	3.9	18	668	3.3	19	288	3.9	20	330	4.8
21	878	1.8	22	332	5.0	23	70	8.4	24	242	5.3	25	124	4.7
26	1686	3.6	27	700	3.4	28	222	6.0	29	358	3.4	30	428	2.1
31	328	5.1	32	362	3.9	33	20	9.2	34	770	3.1	35	936	2.8
36	524	3.7	37	416	2.6	38	496	3.6	39	1224	1.5	40	376	2.1
41	248	5.2	42	228	4.5	43	922	3.8	44	32	5.6	45	158	2.2
46	350	2.5	47	326	2.2	48	10	6.8	49	94	6.8	50	1938	2.7
51	336	2.9	52	596	1.8	53	1346	2.3	54	460	4.3	55	666	4.0
56	2298	1.5	57	996	2.7	58	278	3.3	59	328	3.1	60	160	4.5
61	356	2.3	62	198	2.4	63	150	3.2	64	420	2.0	65	36	3.3
66	20	6.5	67	598	2.3	68	102	4.1	69	410	5.0	70	524	5.2
71	96	4.1	72	84	2.7	73	1042	2.5	74	90	5.3	75	80	6.7
76	70	8.9	77	304	3.8	78	20	6.5	79	86	4.4	80	106	4.0
81	122	3.2	82	460	2.6	83	96	4.3	84	366	2.5	85	50	4.7
86	56	4.4	87	20	7.0	88	120	6.7	89	18	8.1	90	326	3.0
91	4	8.8	92	20	6.1	93	2219	1.8	94	44	5.8	95	72	5.2
96	151	4.3	97	234	4.5	98	56	5.6	99	90	6.0	100	10	8.9
101	310	2.7	102	709	1.7	103	130	3.9	104	20	6.6	105	118	2.7
106	92	3.3	107	90	3.6	108	6	5.7	109	0	3.5	110	648	1.8
111	380	2.2	112	122	3.2	113	1830	1.7	114	208	2.4	115	294	2.5
116	30	3.1	117	38	4.0	118	558	1.7	119	22	5.1	120	90	3.1
121	226	2.3	122	72	3.6	123	486	2.3	124	650	3.3	125	218	2.9
126	392	2.5	127	28	4.5	128	30	4.3	129	348	3.2	130	176	2.8
131	924	2.9	132	16	3.8	133	156	2.8	134	546	2.8	135	44	4.5
136	98	2.3	137	70	4.0	138	540	3.0	139	58	5.7	140	442	2.4
141	26	4.3	142	16	4.9	143	22	4.1	144	164	3.1	145	6	6.9
146	38	2.7	147	64	4.5	148	3498	4.3	149	228	5.0	150	588	4.1
151	152	4.4	152	28	7.2	153	8	5.2						

```
NOC( NETWORK NUMBER, PERIOD)
        NOC(161, 1) = 0.3551986E 06    LK INVEST = 2 2 2 2 0 0 2 2    OCLS = 4.3 5.0 4.1 4.4 7.2 5.2
        NOC(162, 1) = 0.3549234E 06    LK INVEST = 2 2 4 2 0 0 2 2    OCLS = 4.3 5.0 3.7 4.4 7.2 5.2
        NOC(163, 1) = C.3550674E 06    LK INVEST = 2 4 2 2 0 0 2 2    OCLS = 4.3 4.6 4.1 4.4 7.2 5.2
        NOC(164, 1) = 0.3544590E 06    LK INVEST = 4 2 2 2 0 0 2 2    OCLS = 4.1 5.0 4.1 4.4 7.2 5.2
        NOC(165, 1) = 0.3550674E 06    LK INVEST = 2 2 2 4 0 0 2 2    OCLS = 4.3 5.0 4.1 3.8 7.2 5.2
        NOC(166, 1) = 0.3548323E 06    LK INVEST = 2 2 4 4 0 0 2 2    OCLS = 4.3 5.0 3.7 3.8 7.2 5.2
        NOC(167, 1) = 0.3549762E 06    LK INVFST = 2 4 2 4 0 0 2 2    OCLS = 4.3 4.6 4.1 3.8 7.2 5.2
        NOC(168, 1) = 0.3543678E 06    LK INVEST = 4 2 2 4 0 0 2 2    OCLS = 4.1 5.0 4.1 3.8 7.2 5.2
```

TABLE XXXII (Continued)

			L(I)	LKVOL(I)	ORIGINAL						
1	6220	2	2300	3	8603	4	715	5	6604	6	7106
7	3700	8	4524	9	4488	10	6200	11	8840	12	6618
13	1982	14	3216	15	332	16	3146	17	326	18	536
19	3190	20	910	21	402	22	512	23	1152	24	404
25	96	26	288	27	202	28	176	29	1948	30	890
31	280	32	504	33	524	34	408	35	498	36	24
37	978	38	1236	39	774	40	704	41	662	42	1676
43	546	44	418	45	380	46	1426	47	52	48	240
49	878	50	562	51	428	52	14	53	140	54	2508
55	488	56	836	57	1804	58	700	59	754	60	3298
61	1418	62	382	63	432	64	236	65	486	66	238
67	186	68	502	69	50	70	30	71	788	72	128
73	508	74	734	75	152	76	136	77	1514	78	112
79	100	80	190	81	360	82	20	83	106	84	132
85	170	86	636	87	110	88	536	89	84	90	70
91	28	92	152	93	30	94	428	95	4	96	26
97	2837	98	60	99	120	100	221	101	284	102	76
103	116	104	20	105	500	106	1187	107	216	108	24
109	192	110	140	111	130	112	8	113	0	114	862
115	582	116	206	117	2602	118	282	119	414	120	42
121	50	122	848	123	50	124	34	125	126	126	314
127	116	128	802	129	852	130	320	131	510	132	42
133	46	134	508	135	266	136	1182	137	24	138	198
139	742	140	64	141	116	142	102	143	706	144	16
145	98	146	612	147	28	148	18	149	24	150	20C
151	12	152	46	153	88						

LK CHANGE NUMBERS = 8 15 27 49123144

			L(I)	LKVOL(I)		OCL(I)	REORDERED							
1	6220	3.9	2	2300	5.4	3	8603	3.6	4	715	8.4	5	6604	3.2
6	7106	3.9	7	3700	3.9	8	4488	4.5	9	6200	3.5	10	8840	2.7
11	6618	5.8	12	1982	3.6	13	3216	4.8	14	3146	3.0	15	326	7.0
16	536	5.6	17	3190	4.1	18	910	3.5	19	402	4.3	20	512	5.0
21	1152	2.0	22	404	5.2	23	96	8.6	24	288	5.4	25	176	5.0
26	1948	3.7	27	890	3.8	28	280	6.3	29	504	3.5	30	524	2.5
31	408	5.5	32	498	4.1	33	24	9.3	34	978	3.2	35	1236	3.0
36	774	4.0	37	704	2.7	38	662	3.7	39	1676	1.7	40	546	2.4
41	418	5.5	42	380	4.6	43	1426	4.2	44	52	5.8	45	240	2.4
46	562	2.7	47	428	2.4	48	14	7.3	49	140	7.0	50	2508	3.0
51	488	3.1	52	836	1.9	53	1804	2.4	54	700	4.6	55	754	4.4
56	3298	1.7	57	1418	3.0	58	382	3.5	59	432	3.3	60	236	4.9
61	486	2.5	62	238	2.7	63	186	3.4	64	502	2.1	65	50	3.5
66	30	6.9	67	788	2.5	68	128	4.5	69	508	5.2	70	734	5.5
71	152	4.2	72	136	2.8	73	1514	2.6	74	112	5.5	75	100	7.1
76	100	9.1	77	360	4.2	78	20	7.0	79	106	4.8	80	132	4.0
81	170	3.6	82	636	3.0	83	110	4.6	84	536	2.7	85	84	7.2
86	70	5.0	87	28	7.3	88	152	7.0	89	30	8.3	90	428	3.2
91	4	9.3	92	26	6.3	93	2837	2.0	94	60	5.9	95	120	5.3
96	221	4.7	97	284	4.8	98	76	5.7	99	116	6.3	100	20	9.2
101	500	2.9	102	1187	2.0	103	216	4.2	104	24	7.2	105	192	3.0
106	140	3.7	107	130	4.0	108	8	6.0	109	0	3.8	110	862	2.3
111	582	2.5	112	206	3.4	113	2602	2.0	114	282	2.8	115	414	2.5
116	42	3.4	117	50	4.1	118	848	2.0	119	34	9.4	120	126	3.1
121	314	2.5	122	116	4.0	123	802	2.4	124	852	3.5	125	320	3.3
126	510	2.6	127	42	5.1	128	46	4.4	129	508	3.5	130	266	3.0
131	1182	3.1	132	24	4.2	133	198	3.0	134	742	3.0	135	64	5.0
136	116	2.3	137	102	4.0	138	706	3.2	139	98	6.2	140	612	2.5
141	28	4.4	142	18	5.2	143	24	4.3	144	200	3.2	145	12	7.2
146	46	3.0	147	88	4.8	148	4524	4.4	149	332	5.3	150	878	4.2
151	202	4.6	152	50	7.7	153	16	5.4						

```
NOC( NETWORK NUMBER, PERIOD)
        NOC(161, 2) = 0.5089588E 06    LK INVEST = 2 2 2 2 0 0 2 2    OCLS =  4.4  5.3  4.2  4.6  7.7  5.4
        NOC(162, 2) = 0.5086954E 06    LK INVEST = 2 2 4 2 0 0 2 2    OCLS =  4.4  5.3  3.9  4.6  7.7  5.4
        NOC(163, 2) = 0.5087596E 06    LK INVEST = 2 4 2 2 0 0 2 2    OCLS =  4.4  4.7  4.2  4.6  7.7  5.4
        NOC(164, 2) = 0.5080539E 06    LK INVEST = 4 2 2 2 0 0 2 2    OCLS =  4.2  5.3  4.2  4.6  7.7  5.4
        NOC(165, 2) = 0.5088779E 06    LK INVEST = 2 2 2 4 0 0 2 2    OCLS =  4.4  5.3  4.2  4.2  7.7  5.4
        NOC(166, 2) = C.5086146E 06    LK INVEST = 2 2 4 4 0 0 2 2    OCLS =  4.4  5.3  3.9  4.2  7.7  5.4
        NOC(167, 2) = 0.5086788E 06    LK INVEST = 2 4 2 4 0 0 2 2    OCLS =  4.4  4.7  4.2  4.2  7.7  5.4
        NOC(168, 2) = 0.5079731E 06    LK INVEST = 4 2 2 4 0 0 2 2    OCLS =  4.2  5.3  4.2  4.2  7.7  5.4
```

TABLE XXXII (Continued)

			L(I)	LKVOL(I)	ORIGINAL						
1	8316	2	3132	3	12897	4	928	5	8930	6	10855
7	5536	8	6606	9	6430	10	9012	11	12554	12	10236
13	2936	14	4840	15	536	16	5148	17	575	18	856
19	5823	20	1672	21	680	22	876	23	1958	24	642
25	144	26	394	27	318	28	262	29	3210	30	1454
31	382	32	636	33	688	34	564	35	640	36	28
37	1344	38	1842	39	1002	40	926	41	892	42	2096
43	744	44	578	45	474	46	1770	47	76	48	280
49	1094	50	664	51	534	52	16	53	2C0	54	3308
55	638	56	1062	57	2520	58	977	59	984	60	3968
61	1992	62	533	63	580	64	300	65	630	66	308
67	256	68	634	69	68	70	38	71	1040	72	164
73	712	74	1011	75	198	76	198	77	2059	78	144
79	128	80	134	81	454	82	30	83	156	84	168
85	184	86	840	87	150	88	662	89	116	90	84
91	36	92	212	93	32	94	558	95	4	96	42
97	3830	98	86	99	142	100	290	101	396	102	114
103	222	104	22	105	612	106	1512	107	288	108	36
109	238	110	208	111	152	112	8	113	2	114	1052
115	734	116	262	117	3422	118	334	119	500	120	52
121	66	122	1228	123	78	124	44	125	168	126	400
127	148	128	1194	129	1198	130	410	131	762	132	60
133	54	134	746	135	376	136	1610	137	36	138	254
139	586	140	92	141	150	142	136	143	1146	144	24
145	132	146	832	147	38	148	34	149	36	150	270
151	16	152	60	153	130						

LK CHANGE NUMBERS = 8 15 27 49123144

			L(I)	LKVCL(I)		OCL(I)	REORDERED							
1	8316	4.0	2	3132	5.7	3	12897	3.8	4	928	8.5	5	8930	3.6
6	10855	4.0	7	5536	4.2	8	6430	4.8	9	9012	3.8	10	12554	3.C
11	10236	5.9	12	2936	3.9	13	4840	5.0	14	5148	3.4	15	575	7.3
16	856	5.7	17	5823	4.2	18	1672	3.8	19	680	4.4	20	876	5.3
21	1958	2.3	22	642	5.4	23	144	8.8	24	394	5.6	25	262	5.2
26	3210	3.8	27	1454	3.9	28	382	6.4	29	636	3.6	30	688	2.6
31	564	5.6	32	640	4.4	33	28	9.8	34	1344	3.6	35	1842	3.4
36	1002	4.2	37	926	3.2	38	892	3.8	39	2096	1.9	40	744	2.6
41	578	5.7	42	474	5.0	43	1770	4.3	44	76	6.0	45	280	2.6
46	664	2.8	47	534	2.6	48	16	7.4	49	200	7.4	50	3308	3.2
51	638	3.4	52	1062	2.0	53	2520	2.7	54	977	4.8	55	984	4.5
56	3968	1.8	57	1992	3.2	58	533	3.7	59	580	3.7	60	300	5.2
61	630	2.8	62	308	3.0	63	256	3.6	64	634	2.3	65	68	3.8
66	38	7.3	67	1040	2.7	68	164	4.6	69	712	5.4	70	1011	5.6
71	198	4.3	72	198	3.1	73	2059	2.8	74	144	5.6	75	128	7.3
76	134	9.5	77	454	4.3	78	30	7.1	79	156	5.0	80	168	4.2
81	184	3.6	82	840	3.0	83	150	4.8	84	662	2.8	85	116	7.4
86	84	5.1	87	36	7.6	88	212	7.2	89	32	8.4	90	558	3.4
91	4	9.7	92	42	6.4	93	3830	2.1	94	86	6.0	95	142	5.4
96	290	4.8	97	396	4.9	98	114	5.8	99	222	6.5	100	22	9.7
101	612	3.2	102	1512	2.2	103	288	4.5	104	36	7.3	105	238	3.3
106	208	3.8	107	152	4.2	108	8	6.2	109	2	3.9	110	1052	2.5
111	734	2.7	112	262	3.6	113	3422	2.3	114	334	2.9	115	500	2.6
116	52	3.5	117	66	4.2	118	1228	2.3	119	44	5.6	120	168	3.4
121	400	2.7	122	148	4.2	123	1194	2.6	124	1198	3.7	125	410	3.4
126	762	2.8	127	60	5.1	128	54	4.9	129	746	3.7	130	376	3.3
131	1610	3.5	132	36	4.5	133	254	3.3	134	986	3.6	135	92	5.2
136	150	2.5	137	136	4.3	138	1146	3.4	139	132	6.5	140	832	2.7
141	38	4.6	142	34	5.2	143	36	4.6	144	27C	3.4	145	16	7.6
146	60	3.1	147	130	5.2	148	6606	4.7	149	536	5.6	150	1094	4.6
151	318	4.8	152	78	7.8	153	24	5.6						

NOC(NETWORK NUMBER, PERIOD)

NOC(161, 3) = C.7784956E 06 LK INVEST = 2 2 2 2 0 0 2 2 OCLS = 4.7 5.6 4.6 4.8 7.8 5.6
NOC(162, 3) = C.7778393E 06 LK INVEST = 2 2 4 2 0 0 2 2 OCLS = 4.7 5.6 4.0 4.8 7.8 5.6
NOC(163, 3) = C.77812C4E 06 LK INVEST = 2 4 2 2 0 C 2 2 OCLS = 4.7 4.9 4.6 4.8 7.8 5.6
NOC(164, 3) = 0.7758532E 06 LK INVEST = 4 2 2 2 0 0 2 2 OCLS = 4.3 5.6 4.6 4.8 7.8 5.6
NOC(165, 3) = C.7783048E 06 LK INVEST = 2 2 2 4 0 0 2 2 OCLS = 4.7 5.6 4.6 4.2 7.8 5.6
NOC(166, 3) = 0.7776484E 06 LK INVEST = 2 2 4 4 0 0 2 2 OCLS = 4.7 5.6 4.0 4.2 7.8 5.6
NOC(167, 3) = 0.7779256E 06 LK INVEST = 2 4 2 4 0 0 2 2 OCLS = 4.7 4.9 4.6 4.2 7.8 5.6
NOC(168, 3) = 0.7756624E 06 LK INVEST = 4 2 2 4 0 0 2 2 OCLS = 4.3 5.5 4.6 4.2 7.8 5.5

TABLE XXXII (Continued)

			L(I)	LKVOL(I)	ORIGINAL						
1	10890	2	4000	3	16000	4	1182	5	11896	6	12886
7	6776	8	9266	9	8228	10	11589	11	15544	12	12202
13	4202	14	6169	15	680	16	6776	17	746	18	1022
19	7038	20	2118	21	852	22	1084	23	2374	24	906
25	196	26	508	27	466	28	364	29	4198	30	1998
31	572	32	964	33	1090	34	830	35	894	36	40
37	1742	38	2300	39	1414	40	1354	41	1144	42	3362
43	1070	44	754	45	662	46	2364	47	112	48	376
49	1408	50	898	51	656	52	22	53	286	54	4490
55	890	56	1598	57	3388	58	1328	59	1178	60	5154
61	2718	62	728	63	746	64	418	65	886	66	380
67	304	68	826	69	78	70	58	71	1420	72	256
73	921	74	1364	75	308	76	282	77	2604	78	188
79	164	80	164	81	595	82	36	83	212	84	248
85	218	86	1074	87	192	88	866	89	158	90	110
91	48	92	280	93	44	94	704	95	8	96	52
97	4938	98	118	99	206	100	368	101	488	102	150
103	290	104	30	105	806	106	1974	107	396	108	48
109	304	110	260	111	206	112	14	113	6	114	1398
115	894	116	372	117	4156	118	478	119	716	120	76
121	84	122	1524	123	104	124	60	125	216	126	504
127	208	128	1632	129	1570	130	622	131	1006	132	78
133	76	134	1002	135	468	136	2100	137	50	138	340
139	1260	140	120	141	232	142	192	143	1514	144	34
145	194	146	1057	147	52	148	52	149	48	150	330
151	23	152	76	153	170						

LK CHANGE NUMBERS = 8 15 27 49123144

			L(I)	LKVOL(I)		OCL(I)	REORDERED										
1	10890	4.2	2	4000	6.0	3	16000	3.9	4	1182	8.7	5	11896	3.7			
6	12886	4.1	7	6776	4.5	8	8228	5.0	9	11589	4.1	10	15544	3.2			
11	12202	6.0	12	4202	4.0	13	6169	5.2	14	6776	3.5	15	746	7.4			
16	1022	5.9	17	7038	4.4	18	2118	3.9	19	852	4.6	20	1084	5.4			
21	2374	2.4	22	906	5.5	23	196	8.9	24	508	5.8	25	364	5.3			
26	4198	4.2	27	1998	4.1	28	572	6.5	29	964	3.7	30	1090	2.8			
31	830	5.7	32	894	4.5	33	40	9.9	34	1742	3.8	35	2300	3.5			
36	1414	4.3	37	1354	3.3	38	1144	4.3	39	3362	2.3	40	1070	2.7			
41	754	5.8	42	662	5.2	43	2364	4.4	44	112	6.2	45	376	2.8			
46	898	2.9	47	656	2.7	48	22	7.5	49	286	7.5	50	4490	3.3			
51	890	3.5	52	1558	2.3	53	3388	2.9	54	1328	4.9	55	1178	4.6			
56	5154	2.0	57	2718	3.3	58	728	3.9	59	746	3.8	60	418	5.4			
61	886	3.0	62	380	3.1	63	304	3.8	64	826	2.5	65	78	4.0			
66	58	7.4	67	1420	2.8	68	256	4.8	69	921	5.6	70	1364	5.7			
71	308	4.4	72	282	3.4	73	2604	3.1	74	188	5.7	75	164	7.5			
76	164	9.6	77	595	4.4	78	36	7.2	79	212	5.3	80	248	4.7			
81	218	3.8	82	1074	3.2	83	192	5.0	84	866	2.9	85	158	7.6			
86	110	5.2	87	48	7.7	88	280	7.4	89	44	8.5	90	704	3.6			
91	8	9.8	92	52	6.5	93	4938	2.2	94	118	6.3	95	206	5.8			
96	368	4.9	97	488	5.0	98	150	6.3	99	290	6.6	100	30	9.9			
101	806	3.2	102	1974	2.2	103	396	4.6	104	48	7.4	105	304	3.4			
106	260	3.9	107	206	4.3	108	14	6.4	109	6	4.0	110	1398	2.6			
111	894	2.9	112	372	3.8	113	4156	2.3	114	478	3.0	115	716	2.7			
116	76	3.8	117	84	4.4	118	1524	2.4	119	60	5.9	120	216	3.6			
121	504	3.0	122	208	4.5	123	1632	2.7	124	1570	4.0	125	622	3.6			
126	1006	3.0	127	78	5.2	128	76	5.0	129	1002	4.0	130	468	3.6			
131	2100	3.8	132	50	4.7	133	340	3.5	134	1260	3.6	135	120	5.2			
136	232	2.5	137	192	4.5	138	1514	3.8	139	194	6.6	140	1057	3.0			
141	52	4.8	142	52	5.4	143	48	4.6	144	330	3.5	145	23	7.7			
146	76	3.2	147	170	5.3	148	9266	5.0	149	680	5.7	150	1408	4.9			
151	466	5.0	152	104	7.9	153	34	5.8									

NOC(NETWORK NUMBER, PERIOD)

NOC(161, 4) = 0.1048869E 07	LK INVEST = 2 2 2 2 0 0 2 2	OCLS = 5.0 5.7 4.9 5.0 7.9 5.8
NOC(162, 4) = 0.1048025E 07	LK INVEST = 2 2 4 2 0 0 2 2	OCLS = 5.0 5.7 4.3 5.0 7.9 5.8
NOC(163, 4) = C.1048462E 07	LK INVFST = 2 4 2 2 0 0 2 2	OCLS = 5.0 5.1 4.9 5.0 7.9 5.8
NOC(164, 4) = 0.1043311E 07	LK INVEST = 4 2 2 2 0 0 2 2	OCLS = 4.4 5.7 4.9 5.0 7.9 5.8
NOC(165, 4) = 0.1048544E 07	LK INVEST = 2 2 2 4 0 0 2 2	OCLS = 5.0 5.7 4.9 4.3 7.9 5.8
NOC(166, 4) = 0.1047699E 07	LK INVEST = 2 2 4 4 0 0 2 2	OCLS = 5.0 5.7 4.3 4.3 7.9 5.8
NOC(167, 4) = C.1048136E 07	LK INVEST = 2 4 2 4 0 0 2 2	OCLS = 5.0 5.1 4.9 4.3 7.9 5.8
NOC(168, 4) = 0.1042584E 07	LK INVEST = 4 2 2 4 0 0 2 2	OCLS = 4.4 5.7 4.9 4.3 7.9 5.8

B.10 Dynamic Programming (DP) Program. Table XXXIII is the listing of the DP program. Two subroutines were employed within this program which were used in generating construction costs and maintenance costs. These subroutines would not be adequate for practical applications.

TABLE XXXIII

DYNAMIC PROGRAMMING PROGRAM

```
//DP      JOB (10652,441-40-6462,5,,,9001,3),'L D CHAPMAN'
//   EXEC FORTGCLG,REGION.GO=85K,TIME=(5)
//FCRT.SYSIN CD *
      REAL MC(200,4),IR
      INTEGER S(200, 8),D(330, 8),DNO(330),DEC(330,3,4),ST,STATE,DN
      DIMENSION OC(200,4),CC(330),NS(8),F(200,5),NETST(8),BUDG(4)
      CCMMON CC,MC,CC,D,S
      CATA CEC/3960*0/,F/1000*0./
C
C        NPA = NUMBER OF POSSIBLE ALTERNATIVE ROADS (DIGITS)
C        NOPER = NUMBER OF PERIODS
C        NOSTAT = NUMBER OF STATES
C        NODEC = NUMBER OF POSSIBLE DECISIONS
C        NPYR = NUMBER OF PERIOD YEARS
C        NOYR = NUMBER OF YEARS
C        IR = INTEREST RATE
C        BUDG(I) = CONSTRUCTION BUDGET  LIMITATION IN PERIOD I
C        S(I,4) = STAT OF NETWORK I IN PERIOD J
C        NS = NEW STATE
C        NETST = NETWORK STATE
C        DN = DECISION NUMBER
C        D(I,J) = DECISION I IN PERIOD J
C        DNO = DECISION NUMBER
C        F(I,J) = OPTIMAL COST FUNCTION FCR NETWORK I THRU PERIOD J
C        CC(I) = CONSTRUCTION COST FOR DECISION I
C        OC(I,J) = NETWORK OPERATORS' COST FOR NETWORK I DURING PERIOD J
C        DEC(I,J,K) = DECISION NUMBER I IN PERIOD K; J PROVIDES 3
C        POSSIBLE TIES FOR OPTIMAL DECISIONS
C
      NPA = 8
      NOPER = 4
      NCSTAT = 200
      NODEC = 326
      NPYR = 5
      BUDG(1) = 2000.
      BUDG(2) = 1200.
      BUDG(3) = 800.
      BUDG(4) = 500.
      IR = 0.07
      PS = 1./(1.+IR)**NPYR
C
C        READ OPERATORS COST AND NETWORK STATES FOR EACH NETWORK AND
C        PERIOD FROM DISK SAVING ONLY THE LAST OCCURING VALUES
C
      DO 10 JKL=1,1500
      READ(8,4,END=30) I,J,X,(NETST(L),L=1,NPA)
    4 FORMAT(2I5,E14.7,8I2)
C
C        CHANGE PERIOD ORDERING FROM LEFT-TO-RIGHT TO REGHT-TO-LEFT
C        TO CORRESPOND WITH ORDERING DEFINITION AS THE NUMBER OF
C        PERICDS REMAINING IN THE TOTAL PLANNING PERIOD
C
      GO TO (5,6,7,8),J
    5 J=4
      GO TO 9
    6 J=3
      GO TO 9
    7 J=2
```

TABLE XXXIII (Continued)

```
      GO TO 9
    8 J=1
C
C         STORE OPERATORS COST FOR NETWORK I AND PERIOD J
C
    9 OC(I,J) = X
C
C         STORE NETWORK STATE
C
      DO 10 KL=1,NPA
   10 S(I,KL)=NETST(KL)
      WRITE(6,15)
   15 FORMAT(///1X,'DID NOT FIND END OF FILE',///1X,'INCREASE JKL
     #LIMIT IN FIRST DO LOOP')
      STOP
   30 WRITE(6,31)
   31 FORMAT(///1X,'FOUND END OF FILE')
C
C         READ DECISION NUMBER AND DECISION DIGITS
C
      READ(5,32)  (DNO(K),(D(K,L),L=1,NPA),K=1,NODEC)
   32 FORMAT(I5,8I2)
C
C         CALL SUBROUTINE TO CALCULATE ALL CONSTRUCTION COSTS
C
      CALL CONSTC(NPA,NODEC)
C
C         WRITE STATE NUMBER, STATE DIGITS, AND CONSTRUCTION COSTS
C         BY COLUMNS
C
      WRITE(6,33)
   33 FORMAT('1',//1X,'DNO',9X,'DEC',11X,'CC',4X,'DNO',9X,'DEC',11X,
     #'CC',4X,'DNO',9X,'DEC',11X,'CC',4X,'DNO',9X,'DEC',11X,'CC')
      LNCOL = NODEC/4
      L2COL = LNCOL
      L3COL = L2COL + LNCOL
      L4COL = L3COL + LNCOL
      DO 34 I= 1,LNCOL
      I2 = I + L2COL
      I3 = I + L3COL
      I4 = I + L4COL
      WRITE(6,35) I,(D(I,L),L=1,NPA),CC(I),I2,(D(I2,L),L=1,NPA),
     1CC(I2),I3,(D(I3,L),L=1,NPA),CC(I3),I4,(D(I4,L),L=1,NPA),CC(I4)
   34 CONTINUE
   35 FORMAT(I4,2X,8I2,2X,F6.0,I6,2X,8I2,2X,F6.0,I6,2X,8I2,2X,F6.0,
     #I6,2X,8I2,2X,F6.0)
      IF(4*LNCOL.EQ.NODEC) GO TO 42
      I5 = I4 + 1
      WRITE(6,36) (I,(D(I,L),L=1,NPA),CC(I),I=I5,NODEC)
   36 FORMAT(94X,I6,2X,8I2,2X,F6.0)
C
C         CALL SUBROUTINE TO CALCULATE ALL MAINTENANCE COSTS
C
      CALL MAINC(NOSTAT,NOPER,NPA)
C
C         WRITE STATE NUMBER, STATE DIGITS, NETWORK OPERATORS'
C         COSTS, AND MAINTENANCE COSTS
C
   42 WRITE(6,43)
```

TABLE XXXIII (Continued)

```
   43 FORMAT('1',//1X,'NETNU',7X,'STATE',22X,'OPCOST,PERIOD=1,2,3,4',
     #27X,'MAINTENANCE COST,PERIOD=1,2,3,4')
      DO 45 I=1,NOSTAT
      WRITE(6,44)  I,(S(I,IK),IK=1,NPA),(OC(I,J),J=1,NOPER),
     #(MC(I,J),J=1,NOPER)
   44 FORMAT(I5,3X,8I2,3X,4F11.0,8X,4F11.0)
   45 CONTINUE
C
C        START AT LAST STAGE
C
   46 NP = 1
C
C        TAKE 1ST NETWORK
C
   48 NN=1
C
C        TAKE FIRST DECISION
C
   50 DN=1
C
C        IS CONSTRUCTION COST BELOW BUDGET IN THIS PERIOD
C
   55 IF(CC(DN).GT.BUDG(NP)) GO TO 121
C
C        TEST FOR PERMISSIBLE INVESTMENT
C        CHECK DIGIT BY DIGIT
C
      DO 80 M=1,NPA
      IF(S(NN,M)*D(DN,M).EQ.0) GO TO 80
      IF(S(NN,M).EQ.D(DN,M)) GO TO  121
      X1=S(NN,M)+D(DN,M)
      IF(X1.EQ.3) GO TO  121
      IF(X1.EQ.6) GO TO 121
      IF(X1.EQ.7) GO TO 121
   80 CONTINUE
C
C        CONSTRUCT NEW STATE AFTER INVESTMENT
C
      DO 100 L=1,NPA
      NS(L) = S(NN,L)+D(DN,L)
      IF(NS(L).EQ.5) NS(L)=4
  100 CONTINUE
C
C        TEST FOR EXISTANCE OF THE NEW STATE BY CHECKING ALL STATES
C        TAKE ONE STATE
C
      DO 110 ST=1,NOSTAT
      STATE=ST
C
C        MAKE A DIGIT BY DIGIT COMPARISON
C
      DO 105 M=1,NPA
      IF(NS(M)-S(ST,M).NE.0) GO TO 110
  105 CONTINUE
      GO TO 115
  110 CONTINUE
      GO TO 121
C
C        NEW STATE INDEX
```

TABLE XXXIII (Continued)

```
C
  115 NSI=STATE
C
C        CALCULATE TOTAL COST FOR THIS PERIOD AND NETWORK
C
      CJ = PS*OC(NSI,NP) + CC(DN) + MC(NN,NP)
C
C        CALCULATE TEMPORARY TOTAL ACCUMULATED COST
C
      TEMPF = CJ + PS*F(NSI,NP)
C
C        HAS A COST FOR THIS STATE BEEN CALCULATED BEFORE---IF NOT,
C        STORE THIS VALUE
C
      IF(F(NN,NP+1).EQ.0.) GO TO 118
C
C        TEST FOR MORE THAN ONE OPTIMAL PT
C
      IF(F(NN,NP+1).EQ.TEMPF) GO TO 150
C
C        TEST FOR MINIMUM INVESTMENT
C
      IF(F(NN,NP+1).LT.TEMPF) GO TO 121
  118 F(NN,NP+1) = TEMPF
      IB=1
C
C        NOTE DECISION THAT PRODUCED BETTER INVESTMENT
C
  119 DEC(NN,IB,NP) = DN
  121 IF(DN.LT.NODEC) GO TO 200
      IF(NP.EQ.NOPER) GO TO 124
  122 IF(NN.LT.NOSTAT) GO TO 210
  123 IF(NP.NE.NOPER) GO TO 220
C
C        WRITE STATE NUMBER, STATE DIGITS, TOTAL OPTIMAL COSTS, AND
C        OPTIMAL DECISION NUMBERS
C
  124 WRITE(6,125)
  125 FORMAT('1',//2X,'NN',9X,'STATE',18X,'TOTAL PERIOD COSTS(1,2,3,4)',
     # 8X,'PERIOD (1,2,3,4) DECISIONS---COL 1 = DNO, COL 2&3=ALT DNO')
  126 LJK=NOPER+1
      LLL= NOPER - 1
      WRITE(6,127) DNO(1),(S(1,J),J=1,NPA),(F(1,J),J=2,LJK),
     #((DEC(1,IB,J),IB=1,3),J=1,NOPER)
  127 FORMAT(1X,I3,3X,8I2,4F11.0,2X,3I4,5X,3I4,5X,3I4,5X,3I4)
      DO 129 I = 2,NOSTAT
      WRITE(6,128) I,(S(I,J),J=1,NPA),(F(I,J),J=2,NOPER),
     #((DEC(I,IB,J),IB=1,3),J=1,LLL)
  128 FORMAT(1X,I3,3X,8I2,3F11.0,13X,3I4,5X,3I4,5X,3I4)
  129 CONTINUE
C
C        TRACE OPTIMAL DECISIONS BACK THRU THE STAGES.
C
      NSI = 1
      IB = 1
      IL = NOPER
  130 JKN = DEC(NSI,IB,IL)
C
C        WRITE OPTIMAL DECISION,DECISION NUMBER, AND OPTIMAL COST
```

TABLE XXXIII (Continued)

```
C
      WRITE(6,280) IL,(D(JKN,L),L=1,NPA),JKN,F(NSI,LJK)
      IF(DEC(NSI,2,IL).NE.0) WRITE(6,271)
C
C        FORM NEW STATE
C
      DO 131 L= 1,NPA
      NS(L) = S(NSI,L) + D(JKN,L)
      IF(NS(L).EQ.5) NS(L) = 4
  131 CONTINUE
C
C        SEARCH FOR STATE NUMBER
C
      DO 133 ST=1,NOSTAT
      STATE = ST
      DO 132 M=1,NPA
      IF(NS(M).NE.S(ST,M)) GO TO 133
  132 CONTINUE
      GO TO 134
  133 CONTINUE
  134 NSI = STATE
C
C        WRITE NEW STATE DIGITS AND STATE NUMBER
C
      WRITE(6,275) (NS(I),I=1,NPA),NSI
      IF(IL.EQ.1) GO TO 135
      LJK = LJK - 1
      IL = IL-1
      GO TO 130
  135 STOP
  150 IB=IB+1
      IF(IB.GT.3) GO TO 270
      GO TO 119
C
C        INCREMENT DECISION NUMBER
C
  200 CN=CN+1
      GO TO 55
C
C        INCREMENT NETWORK NUMBER
C
  210 NN = NN + 1
      GO TO 50
C
C        INCREMENT STAGE NUMBER
C
  220 NP = NP + 1
      GO TO 48
  270 WRITE(6,271)
  271 FORMAT(/1X,20X,'THERE ARE MORE OPTIMAL DECISIONS IN THIS STAGE')
      GO TO 121
  275 FORMAT(//8X,'NEW STATE =',8I2,//8X,'STATE NUMBER =',I3)
  280 FORMAT(///1X,'STAGE NUMBER =',I3,//8X,'OPTIMAL DECISION =',
     #8I2,//8X,'DECISION NUMBER =',I3,//8X,'OPTIMAL COST =',F10.0)
      END
```

TABLE XXXIII (Continued)

```
C
      SUBROUTINE CONSTC(NPA,NODEC)
      INTEGER D(330,8),S(200,8)
      REAL MC(200,4)
      DIMENSION CST2L(8),CST3L(8),CST4L(8),CC(330),OC(200,4)
      COMMON OC,MC,CC,D,S
C
C        READ CONSTRUCTION COST FOR CONSTRUCTING 2L,
C        3L (CONVERT 2L TO 4L), AND 4L ROADS FOR EACH LINK ADDITION
C
      READ(5,5)  (CST2L(I),CST3L(I),CST4L(I),I=1,NPA)
    5 FORMAT(3F10.0)
C
C        CALCULATE CONSTRUCTION COST FOR EACH FEASIBLE DECISION
C
      DO 40 I=1,NODEC
      CC(I) = 0.
      DO 40 J=1,NPA
      IF(D(I,J).EQ.0) GO TO 40
      IF(D(I,J).EQ.2) GO TO 10
      IF(D(I,J).EQ.3) GO TO 20
      CCTEM = CST4L(J)
      GO TO 30
   10 CCTEM = CST2L(J)
      GO TO 30
   20 CCTEM = CST3L(J)
   30 CC(I) = CC(I) + CCTEM
   40 CONTINUE
      RETURN
      END
C
      SUBROUTINE MAINC(NOSTAT,NOPER,NPA)
      REAL MC(200,4)
      INTEGER D(330,8),S(200,8)
      DIMENSION CMC(4),OC(200,4),CC(330)
      COMMON OC,MC,CC,D,S
      MC(1,1) = 230975.
      MC(1,2) = 155818.
      MC(1,3) = 81489.
      MC(1,4) = 53308.
      DO 30 I=2,NOSTAT
      NDSUM = 0
      DO 10 J=1,NPA
   10 NDSUM = NDSUM + S(I,J)
      SUM = NDSUM
      X = (SUM-6.)/SUM
      DO 20 J=1,NOPER
      Y = (X/100.)*MC(1,J)
   20 MC(I,J) = Y + MC(1,J)
   30 CONTINUE
      RETURN
      END
//GO.FTC8FC01 DD DISP=OLD,
//    DSN=OSU.ACT10652.NTOC
//GO.SYSIN DD *
```

VITA

Leon Darrel Chapman

Candidate for the Degree of

Doctor of Philosophy

Thesis: A PLANNING METHOD FOR REGIONAL ROAD INVESTMENTS

Major Field: Electrical Engineering

Biographical:

Personal Data: Born in Arkansas City, Kansas, October 27, 1941, the son of Mr. and Mrs. Theodore D. Chapman; married with one child.

Education: Graduated from Newkirk High School, Newkirk, Oklahoma, in May, 1959; received the Bachelor of Science degree in Electrical Engineering from Oklahoma State University in January, 1964; received the Master of Science degree with a major in Electrical Engineering from Oklahoma State University in May, 1968; completed the requirements for the Doctor of Philosophy degree at Oklahoma State University in July, 1971.

Professional Experience: Design Engineer for Continental Oil Company, Geophysical Department for three years between 1962 and 1968; Lieutenant in United States Army, Air Defense Artillery, from June, 1964, to April, 1966; Company Commander with a final rank of Captain in the United States Army Reserves from May, 1966, to December, 1969; Part-time Instructor, Technical Institute, Oklahoma State University, September, 1967 to June, 1969; Research Assistant in Multidisciplinary Studies Group, Center for Systems Science, Oklahoma State University, June, 1969, to July, 1971.