TECHNICAL REPORT STANDARD TITLE PAGE

| | | |
|---|---|---|
| 1. Report No. | 2. Government Accession No. | 3. Recipient's Catalog No. |

4. Title and Subtitle

Rehabilitation and Maintenance System
District Time Optimization

5. Report Date

January 1981

6. Performing Organization Code

7. Author(s)

Don T. Phillips, Chiyyarath V. Shanmugham,
Shashikant Sathaye, Robert L. Lytton

8. Performing Organization Report No.

Research Report 239-3

9. Performing Organization Name and Address

Texas Transportation Institute
The Texas A&M University System
College Station, Texas 77843

10. Work Unit No.

11. Contract or Grant No.

Research Study 2-18-79-239

12. Sponsoring Agency Name and Address

Texas State Department of Highways and Public
Transportation; Transportation Planning Division
P. O. Box 5051
Austin, Texas 78763

13. Type of Report and Period Covered

Interim - September, 1978
- January, 1981

14. Sponsoring Agency Code

15. Supplementary Notes

Research Study Title: Pavement Rehabilitation Fund Allocation

16. Abstract

The formulation and solution of the multi-year District Rehabilitation And Maintenance problem is presented. The problem is formulated as a 0-1 integer nonlinear programming problem (INLP) and the solution methodology uses the concepts of relaxation, decomposition, and network formulation to convert the 0-1 INLP problem into 0-1 integer linear programming problems (ILP).

A highway maintenance problem is formulated using real data from District 17 and is solved by the computer program based on the proposed solution methodology. The sample problem had 15 highway segments, 6 distress types, 9 rehabilitation and maintenance strategies and a 10 year planning period. The INLP had 1350 0-1 variables and the problem was solved in approximately 35 seconds of CPU time on the AMDAHL 470V/6 computer at the Texas A&M University Campus.

It was concluded that the proposed mathematical model and the algorithm is a good tool for solving the time optimization problems involved in the rehabilitation and maintenance of highway segments at the District level.

17. Key Words

Pavements, Rehabilitation and Maintenance, District Time Optimization

18. Distribution Statement

No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161

| | | | |
|---|---|---|---|
| 19. Security Classif. (of this report) | 20. Security Classif. (of this page) | 21. No. of Pages | 22. Price |
| Unclassified | Unclassified | 111 | |

# REHABILITATION AND MAINTENANCE SYSTEM
# DISTRICT TIME OPTIMIZATION
# (RAMS-DTO-1)

by

Don T. Phillips

Chiyyarath V. Shanmugham

Shashikant Sathaye

Robert L. Lytton

Research Report Number 239-3

Pavement Rehabilitation Fund Allocation

Research Project 2-18-79-239

conducted for
The Texas State Department of Highways and
Public Transportation

by the

Texas Transportation Institute
The Texas A&M University System

September 1980

# ABSTRACT

The formulation and solution of the multi-year District Rehabilitation And Maintenance problem is presented. The problem is formulated as a 0-1 integer nonlinear programming problem (INLP) and the solution methodology uses the concepts of relaxation, decomposition, and network formulation to convert the 0-1 INLP problem into 0-1 integer linear programming problems (ILP).

A highway maintenance problem is formulated using real data from District 17 and is solved by the computer program based on the proposed solution methodology. The sample problem had 15 highway segements, 6 distress types, 9 rehabilitation and maintenance strategies and a 10 year planning period. The INLP had 1350 0-1 variables and the problem was solved in approximately 35 seconds of CPU time on the AMDAHL 470V/6 computer at the Texas A&M University Campus.

It was concluded that the proposed mathematical model and the algorithm is a good tool for solving the time optimization problems involved in the rehabilitation and maintenance of highway segments at the District level.

# SUMMARY

This report describes in detail the District Time Optimization Model of the Rehabilitation And Maintenance System family of computer programs. The model, the solution methodology and the computer programs were developed by the Texas Transportation Institute to assist the District offices in determining the funds required for every year of a finite (5, 10, or 15 years) planning horizon to maintain the segments of the District road network at a specified pavement quality. This in turn will help the state to assess the needs and requirements in planning the rehabilitation and maintenance of highways in the state in future years.

The report contains a description of the mathematical model and the solution methodology developed to optimize the highway maintenance problem. A problem formulated using a few segments from District 17 is solved and presented. The complete details of the problem such as highway segment data, pavement quality requirements and resource availabilities are also described.

A user's manual of the program is provided: it contains the description of subroutines and input data (Appendix B). A listing of the input data and output (solution) of the example problem are presented (Appendix C). The listing of the program is given in Appendix D.

## IMPLEMENTATION STATEMENT

RAMS-DTO-1 is a computer program which has been developed by the Texas Transportation Institute for use by the Highway District offices in the State of Texas to optimally schedule current maintenance of segments of a highway network within the constraints imposed by the resource availability in a finite planning period and to determine the funds required for every year of a finite planning horizon to maintain the highway segments at a specified quality level. This report gives a detailed description of the mathematical model and the solution methodology. This report is intended as a working document which can be used by implementation workshops to train Texas State Department of Highways and Public Transportation personnel in the use of RAMS-DTO-1 program.

## DISCLAIMER

The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Federal Highway Administration. This report does not constitute a standard, specification or regulation.

# TABLE OF CONTENTS

## LIST OF FIGURES

# LIST OF TABLES

# CHAPTER 1

# INTRODUCTION

Until recently, even though millions of dollars are spent each year on highway pavement rehabilitation and maintenance, a negligible amount of research effort has been directed towards developing systematic cost effective procedures for both selecting and scheduling highway rehabilitation and maintenance. Most of the applications of systems analysis operations research have been in the areas of expansion of the existing highway network or construction of new ones. This is due in part to the significant amount of capital required for highway network expansion as compared to highway network maintenance. However, this trend is reversing as the present highways become older and traffic intensity (load and frequency) increases (2). A corresponding shift in fund allocation is due to the increasingly high costs of new highway construction, and the rising costs of rehabilitation and maintenance of the present network. Decisions concerning fund allocation are made more difficult by the fact the funds that are available are rarely sufficient to accomplish all of the work that needs to be done to satisfy the transportation needs of the public.

Few systematic procedures have been developed for generating highway rehabilitation and maintenance schedules and, at present, these schedules are being determined by highway engineers using intuitive rules and formulas for fund allocation. One of the first analytical procedures for generating highway maintenance schedules was recently developed by Ahmed, et al. (2) using a methematical programming model developed by Lu and Lytton (8), the solution technique for which was later described by Phillips and

Lytton (13). It is a single period model and does not consider the effects of future resource supplies on the present maintenance schedule. This single period model (2) will be expanded to a multiperiod model in this report and a procedure for solving the model will be developed.

Mathematical modelling normally consists of three phases: (1) data collection and analysis, (2) mathematical model development, and (3) the development of a solution procedure. The first phase of data collection, analysis, and problem identification has already been completed by the research staff of the Texas Transportation Institute (2). Hence, this report concentrates on the other two areas. The proposed model is a 0-1 integer nonlinear program with thousands of variables and constraints. This is a problem which cannot be solved using existing solution procedures, hence heuristic techniques in conjunction with other state of art concepts will be used to derive an efficient computational algorithm.

## 1.1. Problem Definition

The Texas State Department of Highways and Public Transportation (SDHPT) maintains a network of highway pavements within the state of Texas. The Texas highway network is divided into a number of regions called districts for the purpose of highway construction, maintenance and rehabilitation. Each District is allocated a certain fraction (undetermined) of the yearly state budget depending on its needs. Each district contains a number of highways that are further divided into sections, called highway segments, such that ideally a highway segment has uniform design, environmental conditions and traffic intensity. The District maintains highways by segments rather than by entire portions of the highway within a district. A District highway rehabiliation and maintenance system is defined as a systematic procedure which can analyze highway condition data to generate a 'good' maintenance schedule within the constraints of available resources.

There is a set of about fifteen alternatives, called maintenance strategies, that can be used to maintain or rehabilitate a highway segment. Not all of the maintenance strategies may be feasible for a particular highway segment at any given time for that depends on the highway pavement type and its condition. A highway can deteriorate in about 10-15 different ways, called distress types. A point scale is used to quantify the wear in a highway segment, and under this point system a highway segment in ideal condition is given "$RMAX_i$' rating points for each distress type 'i'. A highway segment's condition is determined by deducting points from '$RMAX_i$' for each distress type 'i'; depending on its condition. Note that not all distress types may be considered important for a particular highway segment, and hence may not be used to determine the highway segment's condition. When a maintenance strategy is applied to a highway segment, points are added to the highway segment's condition and the improvement is a function of highway pavement type and the corrective alternative used. A few of the distress types are: (1) alligator cracking, (2) transverse cracking, (3) rutting, (4) longitudinal cracking, (5) failures per unit length, and (6) low serviceability index. A highway segment's probability of surviving beyond a certain number of years is a function of maintenance strategies through time and the pavement condition when the strategies are implemented. A highway segment may deteriorate at different rates for each distress type. A set of survival curves for each has been obtained from an analysis of highway maintenance data (1). The probability of a highway segment deteriorating below a predefined minimum level at some predictable time in the future can be obtained from these curves. A typical set of survival curves for a particular highway segment is shown in Figure 1.

The total problem can be loosely stated as follows: develop a procedure that can be used by district maintenance supervisors to optimally schedule current rehabilitation and maintenance of highway segments, within the constraints imposed by resource availability and by specified minimum highway segment condition rating requirements over the planning period.

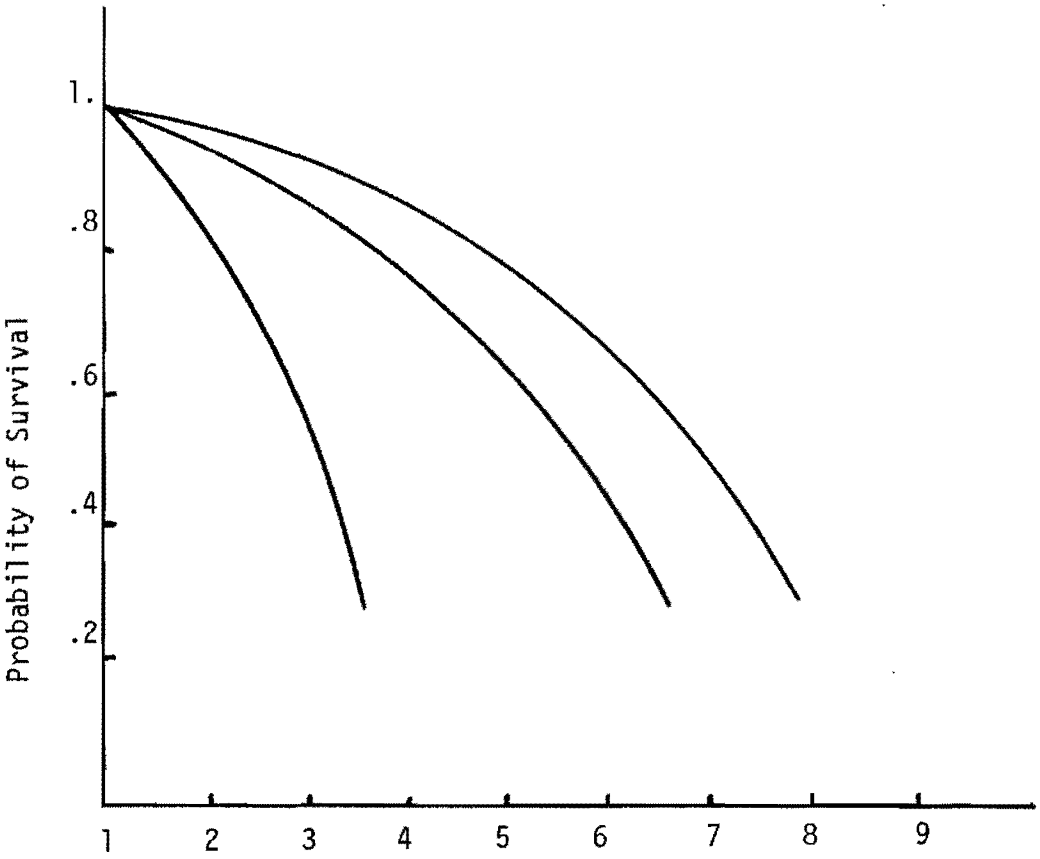


Figure 1 - A Typical Set of Survival Curves for a Particular Highway Segment

# CHAPTER 2

## DEVELOPMENT OF THE MATHEMATICAL MODEL

In operations research, two types of strategies are normally used for optimal resource allocation: (1) maximize a given criterion or goal within the constraints imposed by limited resources, and (2) minimize the use of resources to achieve a certain goal or criterion. A maintenance strategy model based upon the former strategy is developed and solved in this report. The amount of resources available to a District are assumed to be fixed, thus the former strategy was considered more appropriate. The number of years to be considered in future planning is called the planning horizon and could vary anywhere from 5 to 15 years. Although the problem statement only requires the current rehabilitation and maintenance schedule, it is necessary to consider the effects of available resources over the entire planning period, and maintenance schedules for every year in the planning period must be generated. Thus, the problem statement can be interpreted and restated as follows: generate a sequence of interrelated maintenance strategies over a fixed planning horizon for each highway segment so as to maximize the overall highway pavement quality level within the constraints imposed by resources and the required highway conditions. This sequence of interrelated maintenance strategies for a highway segment will be called a maintenance policy.

A quantitative measure of the highway pavement quality can be used as the objective criterion. The highway pavement survival probability curves and the highway pavement point system defined by distress types are used to measure pavement quality levels. A highway is assumed to deteriorate at the same rate as the survival probability curve, and the quantitative measure of ideal highway condition is equal to the maximum rating points available by distress types. This set of highway quality level curves will be called the pavement deterioration curves. A set of pavement deterioration curves for a particular highway segment is illustrated in Figure 2. The objective function value for a rehabilitation or maintenance policy applied to a particular highway segment is determined from the highway pavement quality curves in the following manner. Suppose a maintenance policy for a typical highway segment is as shown in Table I. (Note: Strategy number 1 is a 'do nothing' policy alternative).

TABLE I

A Maintenance Policy for A Typical Highway Segment

| Time Period | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Maintenance Strategy Used | j | 1 | k | 1 | 1 | 1 | 1 | ℓ | 1 | 1 |

The corresponding highway pavement quality level curve for this maintenance policy for distress type 'j' is shown in Figure 3. The objective function value, called policy benefit for a given policy set, is the sum of areas under the corresponding highway quality level curve for each distress type. The objective function coefficient for a strategy 'j' in time period 'l' (See Figure 3) is the sum of the areas between the "RMIN" level and the highway pavement quality level in period 'l', for each distress type. The total benefit of a maintenance policy for a segment is the sum of maintenance effectiveness obtained by implementing the various strategies at the appropriate time in the planning period, for all distress types. For example, the benefit (maintenance

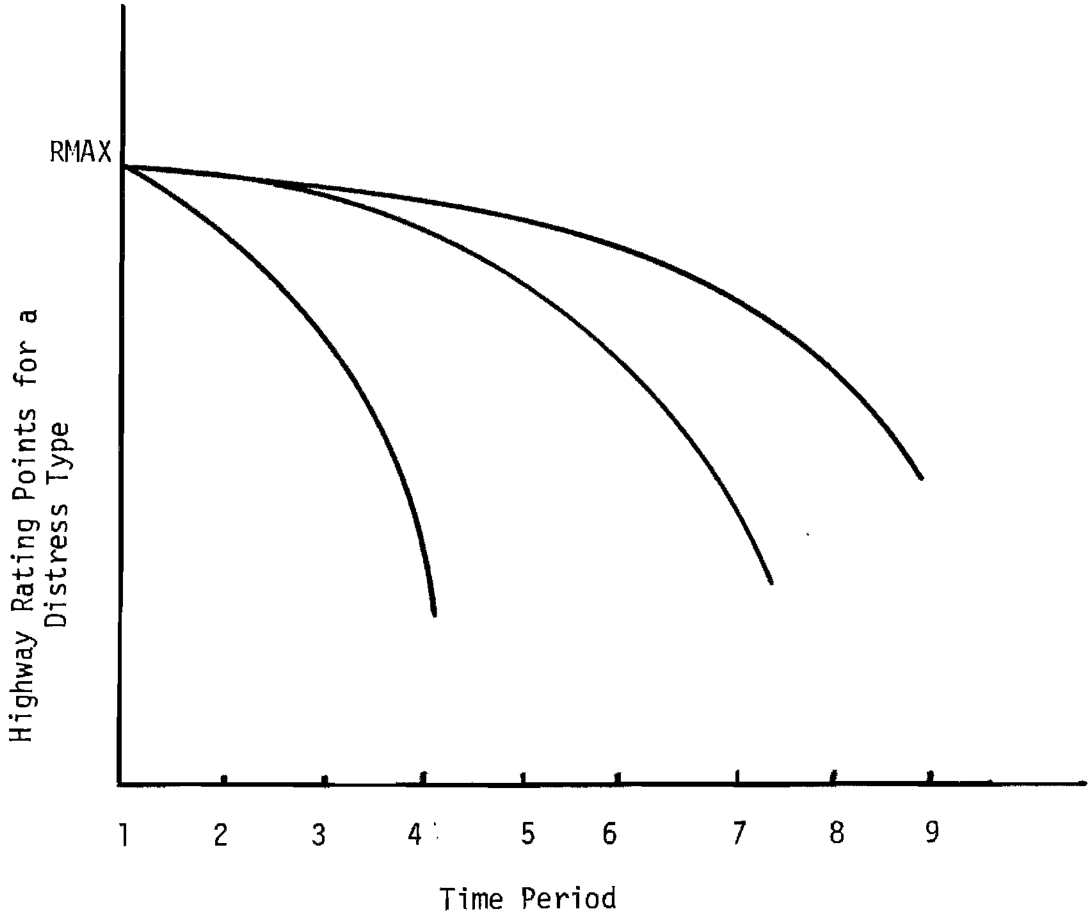
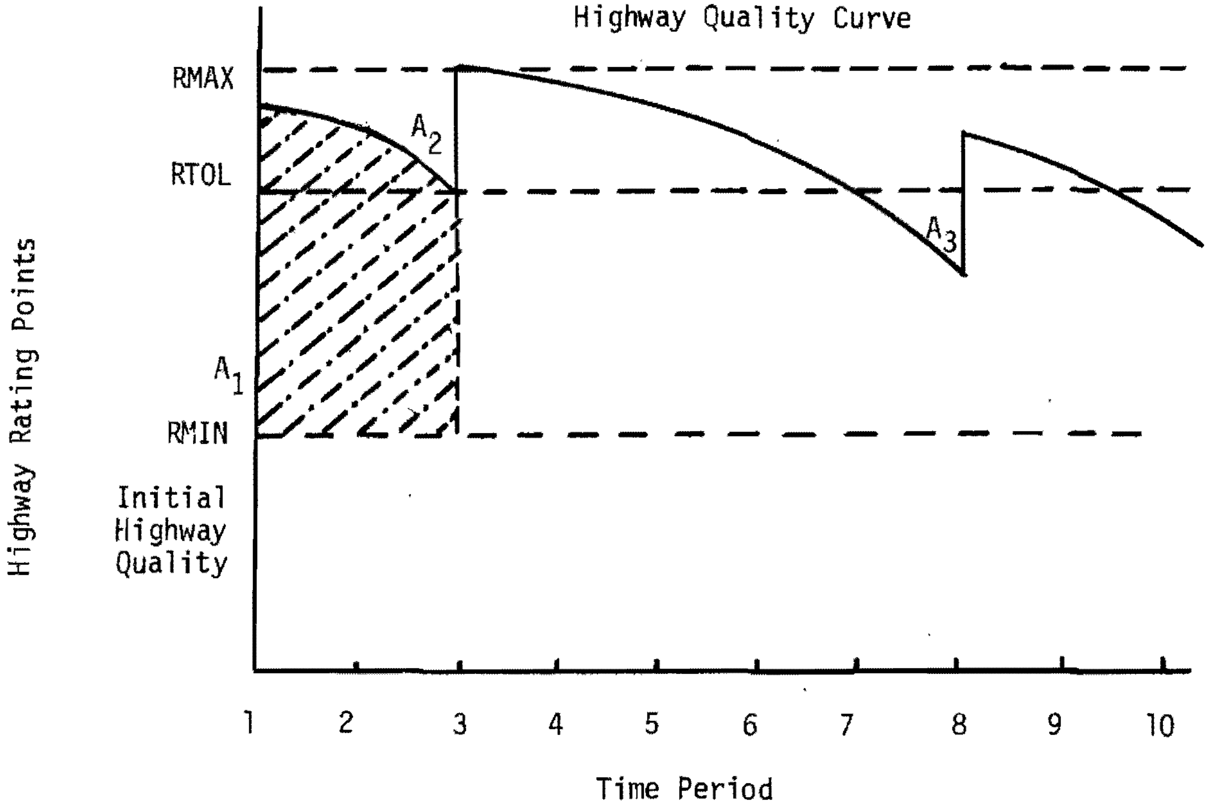


Figure 2 - Typical Set of Pavement Deterioration Curves for a Particular Highway Segment



$A_1$ = Improvement obtained by implementing maintenance strategy 'j'.

$A_2$ = Improvement obtained by implementing maintenance strategy 'k'.

$A_3$ = Improvement obtained by implementing maintenance strategy '$\ell$'.

Figure 3 - A Particular Highway Segment's Quality Curve for a Distress Type

effectiveness) obtained by implementing strategy 'j' during time period 1 is equal to the shaded area ($A_1$) in Figure 3. According to the policy, strategy 'k' is implemented at the end of period 2, and strategy 'l' is implemented at the end of period 7. The corresponding effectiveness for a particular distress type are represented by areas $A_2$ and $A_3$. Hence, the maintenance effectiveness of the policy shown in Table I is the sum of $A_1$, $A_2$ and $A_3$ for a particular distress type.

There are two types of constraints imposed on this problem: (1) resource constraints, and (2) minimum highway quality level constraints and highway feasibility constraints. The resource constraints consist of budget, manpower, material and equipment restrictions. Quality level constraints consist of minimum highway quality requirements in each time period for each distress type and additional restrictions required to determine the feasible strategies in each time period. A strategy is not considered feasible if the improvement obtained by implementation of the strategy is not sufficient to meet the minimum highway pavement condition requirements. If in any time period a highway segment's condition is better than a predefined tolerance level (RTOL), then the highway segment is not to be considered for maintenance in that time period. This effectively forces all the strategies except the 'do nothing' strategy to be infeasible.

The mathematical model that can be used to generate a multiperiod resource-effective highway rehabilitation schedule for a District highway system can be stated as follows:

Problem A

$$\text{Max } Z = \sum_{i=1}^{N} \sum_{j=1}^{S} \sum_{t=1}^{T} B_{ijt} (X_{i11}, \ldots, X_{i,S,t-1}, X_{ijt}) \cdot X_{ijt} \qquad (1)$$

subject to

$$\sum_{j=1}^{S} X_{ijt} = 1, \qquad \text{for } i = 1, 2, \ldots, N \quad t = 1, 2, \ldots, T \qquad (2)$$

$$\sum_{i=1}^{N} \sum_{j=1}^{S} CR_{ijt} \cdot X_{ijt} \leq C_t, \qquad \text{for } t = 1, 2, \ldots, T \qquad (3)$$

$$\sum_{i=1}^{N} \sum_{j=1}^{S} MR_{ijtm} \cdot X_{ijt} \leq AM_{tm}, \qquad \text{for } t = 1, 2, \ldots, T \quad m = 1, 2, \ldots, M \qquad (4)$$

$$\sum_{i=1}^{N} \sum_{j=1}^{S} ER_{ijte} \cdot X_{ijt} \leq AE_{te}, \qquad \text{for } t = 1, 2, \ldots, T \quad e = 1, 2, \ldots, E \qquad (5)$$

$$\sum_{i=1}^{N} \sum_{j=1}^{S} RM_{ijto} \cdot X_{ijt} \leq MA_{to}, \qquad \text{for } t = 1, 2, \ldots, T \quad o = 1, 2, \ldots, O \qquad (6)$$

$$PQ_{it\ell} \geq RMIN_{t\ell}, \qquad \text{for } t = 1, 2, \ldots, T \quad i = 1, 2, \ldots, N \quad \ell = 1, 2, \ldots, D \qquad (7)$$

$$PQ_{it\ell} \geq RTOL_{t\ell} \qquad \text{for all } \ell = 1, 2, \ldots, D$$

$$\sum_{j=2}^{S} X_{ijt} = 0, \qquad \text{for } i = i, 2, \ldots, N \qquad (8)$$

$$PQ_{it} = PQ_{i,t-1,\ell} + \sum_{j=1}^{S} X_{ijt} \cdot RIMP_{j\ell}, \text{for } i = 1, 2, \ldots, N \quad t = 1, 2, \ldots, T \quad \ell = 1, 2, \ldots, D \qquad (9)$$

$$X_{ijt} = 0 \text{ or } 1, \qquad \text{for } i = 1, 2, \ldots, N \quad t = 1, 2, \ldots, T \quad j = 1, 2, \ldots, S \tag{10}$$

where,

$X_{ijt}$ = is a 0-1 decision variable and represents alternative 'j' for highway segment 'i' during time period 't';

$B_{ijt}(\ldots)$ = benefit coefficient for variable $X_{ijt}$ and is a function of the decisions in the prior period;

$N$ = number of highway segments;

$S$ = number of maintenance strategies;

$T$ = length of planning horizon;

$D$ = number of distress types;

$M$ = types of manpower resources;

$E$ = types of equipments;

$O$ = types of materials used;

$C_t$ = the district budget for time period 't';

$CR_{ijt}$ = the capital required to implement alternative 'j' on highway segment 'i' during time period 't';

$AM_{tm}$ = manpower of type 'm' available during time period 't', in man-days;

$RM_{ijtm}$ = manpower required to implement alternative 'j' highway segment 'i' during time period 't', in man-days

$AE_{te}$ = equipment type 'e' available during time period 't', in man-days;

$RE_{ijte}$ = equipment type 'e' required to implement alternative 'j' on highway segment 'i' during time period 't', in man-days;

$MA_{to}$ = material type 'o' available in time period 't';

$MR_{ijto}$ = the amount of material type 'o' required to implement alternative 'j' on highway segment 'i' during time period 't';

$PQ_{it\ell}$ = pavement quality level of highway segment 'i' during time period 't' for distress type '$\ell$';

$RMIN_{t\ell}$ = minimum pavement quality level acceptable for distress type '$\ell$' during time period 't';

$RTOL_{t\ell}$ = tolerable quality level such that if the pavement quality level is above this level in any time period 't', then the highway segment is not considered for maintenance in that particular time period.

This formulation of the highway maintenance problem results in a binary nonlinear integer program (0-1 INLP). The nonlinearity in the problem is in the objective function as well as in the constraints. The benefit function is calculated as the area under pavement quality level curves during any single time period. It is a function of the initial condition and the rehabilitation and maintenance strategies selected in the preceding time periods. The constraints can be classified into two types: 1) the resource constraints (constraint sets (3), (4), (5), and (6)), and 2) strategy feasibility constraints (constraints (2), (7), (8), and (9)). The resource constraints consist of four types of resources: budget, manpower, equipment and material. The strategy feasibility constraints are used to determine the feasible strategies for a highway segment during any single time period. Constraints (2) force the problem to choose one and only one strategy for each highway segment in any time period. (Note:

Strategy '1' is a 'do nothing' strategy). Constraint (7) is used to eliminate any alternative that does not meet the minimum highway pavement quality level requirements for a highway segment in some time period 't'. Constraint (8) ensures that a highway segment is not considered for maintenance if its condition is better than a predefined tolerance level '$RTOL_{t\ell}$', during a given time period 't'.

As previously stated, this mathematical formulation of the highway maintenance problem is a 0-1 INLP problem. In general, a nonlinear programming problem is much more difficult to solve than a linear programming problem and the integer nature of the variable compounds the difficulty. This INLP formulation of the highway maintenance scheduling problem has (N) X (S) X (T) variables. Normally a district has about 150-200 highway segments, 10-15 maintenance strategies, and a planning period around 10 years. This means that the number of binary variables in the problem is around 30,000 and the number of constraints in the neighborhood of 10,000. This INLP problem is not only nonlinear, but the number of variables is extremely large for this class of problems. Hence, a solution procedure for a 0-1 INLP problem of this type is a significant contribution to mathematical programming.

# CHAPTER 3

## OPTIMIZATION OF THE MULTIPERIOD HIGHWAY MAINTENANCE PROBLEM

In this chapter, an algorithm is developed to solve the multiperiod highway maintenance problem is described. The solution procedure is based upon relaxation, decomposition, network formulation, and heuristic techniques. Problem B is the general form (Integer Nonlinear Programming Problem) of Problem A which was presented in the previous chapter. Problem B is as follows:

Problem B:

$$\text{Max } Z = \sum_{i=1}^{N} \sum_{j=1}^{S} \sum_{t=1}^{T} B_{ijt}(X_{i11},\ldots,X_{ijt}) \cdot X_{ijt} \qquad (11)$$

subject to

$$\sum_{j=1}^{S} X_{ijt} = 1, \qquad \text{for } i = 1, 2, \ldots, N \quad t = 1, 2, \ldots, T \qquad (12)$$

$$\sum_{i=1}^{N} \sum_{j=1}^{S} A_{ijtm} \cdot X_{ijt} \leq b_{tm}, \qquad \text{for } m = 1, 2, \ldots, M \quad t = 1, 2, \ldots, T \qquad (13)$$

$$C_{ipt}(X_{i11},\ldots,X_{iSt}) \leq D_{ipt} \qquad \text{for } i = 1, 2, \ldots, N \quad p = 1, 2, \ldots, P \quad t = 1, 2, \ldots, T \qquad (14)$$

$$X_{ijt} = 0, 1 \qquad \text{for } i = 1, 2, \ldots, N \quad j = 1, 2, \ldots, S \quad t = 1, 2, \ldots, T \qquad (15)$$

A relaxed subproblem (C.i) is obtained by relaxing constraints (13) and decomposing the problem as follows:

Problem C.i

$$\text{Max} \sum_{j=1}^{S} \sum_{t=1}^{T} B_{ijt}(X_{i11},\ldots,X_{i,S,t-1},X_{ijt}) \cdot X_{ijt} \qquad (16)$$

subject to

$$\sum_{j-1}^{S} X_{ijt} = 1, \qquad \text{for } t = 1, 2, \ldots, T \qquad (17.t)$$

$$C_{ipt}(X_{i11},\ldots,X_{iSt}) \leq D_{ipt}, \qquad \text{for } p = 1, 2, \ldots, P \quad t = 1, 2, \ldots, T \qquad (18)$$

$$X_{ijt} = 0, 1, \qquad \text{for } j = 1, 2, \ldots, S \quad t = 1, 2, \ldots, T \qquad (19)$$

The algorithmic procedure consists of four main steps: (1) construction of a network model for each of the subproblems (C), (2) solution of the network models of (C.i), (3) synthesis of solutions to the network models, and (4) improvement of the solution obtained in the previous step using a greedy heuristic.

### 3.1: Construction of a Network Model for Problem (C)

A network model is constructed in a stagewise fashion, where each stage corresponds to a value of 't', and there are a total of T+1 stages. Variables considered for network generation at each stage 't' consist of the variables in corresponding GUB (Generalized Upper Bound) constraint sets (17.t). For example, at stage t, the variables $X_{i1t}$, $X_{i2t}$, $\ldots$,$X_{iSt}$ are considered. Node 1 is a source node and is assumed to represent stage '1'.

A feasible set of variables in GUB set (17.1) is determined from constraints (18) with t=1, and arcs are added for each feasible strategy $X_{ij1}$, contained in constraints (17.1), from the source node to nodes 2, 3, ..., L, where 'L-1' is the number of feasible strategies in state '1'. This set of nodes is considered to represent stage '2'. A feasible set of strategies is again determined at each of the nodes 2, 3, ...,L, and more arcs and nodes are generated for each of these nodes into stage 3. This process is continued until stage 'T' is reached. Arcs emanating from each node in stage 'T' are converged to a single node '$e_i$' which is defined to be the sink node. The arc lengths are calculated from the function '$B_{ijt}(\ldots)$' for the corresponding values of 'j' and 't'. This calculation is possible because each $B_{ijt}(\ldots)$ is a function only of the strategies employed at previous stages on a path from node '1' to a particular node.

It is observed that even if there were only four or five strategies feasible at each stage, the number of nodes and arcs rapidly increase beyond computational limitations. The number of arcs and nodes can be reduced by the following method. Suppose at some node 'n' at stage 't' strategy 'j' is feasible and an arc is emanated from node 'n' to some other node 'q' > 'n'. Node 'q' is at a stage 't+1' by the previously defined procedure, but only if strategy '1' were feasible for node 'q'. If this is true, then the corresponding benefit coefficient '$B_{i,1,t+1}(\ldots)$' would be added to the length of the arc from node 'n' to node 'q', and at this point node 'q' is moved into stage 't+2'. This process is repeated until node 'q' reaches a stage $\hat{t}$ such that a strategy other than just strategy '1' is feasible or node 'q' reaches state (T+1). The procedure described above is always applicable if there are constraints similar to constraints (8) in the problem.

The length of the longest path from source node '1' in stage '1' to node '$e_i$' in stage 'T+1' is the optimal solution to (C.i), and the corresponding solution variables can be obtained from the arcs and nodes on this path. Similar networks are generated for each subproblem (C.i).

## 3.2: Solution of the Network Models

The longest path network problems are solved by using a K-shortest path iterative procedure by Shier (16). Since this procedure solves a shortest path problem, the network arc lengths have to be modified by multiplying them by minus one. This modification is possible because there are no circuits in the network. The K-shortest path algorithm determines the K-shortest paths from a given node S to all the other nodes in a network. If $\underline{A}$ is the arc length matrix for a network G (see Appendix A, section 3), then the K-shortest paths from S to all other nodes are determined in the S-th row of $\underline{A}^*$. Let the S-th row of $\underline{A}^*$ be denoted by

$$\left[ (a_{S1}{}^{*})\ (a_{S2}{}^{*})\ \ldots\ldots\ (a_{Sj}{}^{*})\ \ldots\ldots\ (a_{Sn}{}^{*}) \right],$$

where $(a_{Sj}{}^{*}) = (a_{Sj}{}^{*1}, a_{Sj}{}^{*2}, \ldots, a_{Sj}{}^{*k})$

and

$a_{Sj}{}^{*1}$ is the shortest path length from node S to node j, $a_{Sj}{}^{*2}$ is the next best path, and so on.

Shier uses double sweep method which consists of two phases called the backward and forward passes. The arc length matrix $\underline{A}$ is split into two strictly upper and lower triangular matrices, $\underline{U}$ and $\underline{L}$ respectively, such that

$$\underline{A} = \underline{U} \oplus \underline{L}\ .$$

Let $\underline{X}^o$ be a row vector containing initial estimates of K-shortest path lengths from node S to all the other nodes. i.e.

$$\left[\underline{X}^o = \underline{X}^o_{S1}, X^o_{S2}, \ldots, X^o_{Sn}\right]$$

The double sweep method is defined by a pair of recursive relationships.

$$\underline{X}^{2r-1} = \underline{X}^{2r-1} \otimes \underline{L} \oplus \underline{X}^{2r-2} \qquad \text{(Backward pass)}$$

$$\underline{X}^{2r} = \underline{X}^{2r} \otimes \underline{U} \oplus \underline{X}^{2r-1} \qquad \text{(Forward pass)}$$

Where $r > 0$ is the iteration number. The solution converges to optimum when the vector $\underline{X}$ remains unchanged after two successive application of the passes.

In the network formulation of Problem C, the nodes are numbered such that all arcs lead from a smaller numbered node to a larger numbered node. The elements of $\underline{L}$, will be a K-tuple.

$$\underline{V} = (\infty, \infty, \ldots, \infty)$$

i.e.

$$\underline{L} = \begin{bmatrix} \underline{V} & \underline{V} & \cdots & \underline{V} \\ \underline{V} & \underline{V} & \cdots & \underline{V} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \underline{V} & \underline{V} & \cdots & \underline{V} \end{bmatrix}$$

In such a case, the double sweep method will converge in 2 iterations.

r=1: Backward Pass

$$\underline{X}^1 = \underline{X}^o \oplus \underline{X}^1 \otimes \underline{L} \quad = \quad \underline{X}^o \oplus \underline{V} \; = \; \underline{X}^o$$

Forward Pass

$$\underline{X}^2 \; = \; \underline{X}^1 \oplus \underline{X}^2 \otimes \underline{U} \; = \; \underline{X}^o \oplus X^o \, \underline{U}$$

r=2: Backward Pass.

$$\underline{X}^3 = X^2 \oplus \underline{X}^3 \otimes L \; = \; \underline{X}^2.$$

The algorithm converges.

## 3.3: Synthesis of Solutions to the Network Models

The solutions obtained from network models for the subproblems C.i's are synthesized to generate a good feasible solution. A 0-1 ILP model is used for synthesis and is stated as follows:

Problem D

$$\text{Max } Z = \sum_{i=1}^{N} \sum_{j=1}^{K} G_{ij} \cdot Y_{ij} \tag{20}$$

subject to

$$\sum_{j=1}^{K} Y_{ij} = 1 \qquad \text{for } i = 1, 2, \ldots, N, \tag{21.i}$$

$$\sum_{i}^{N}\sum_{j}^{S} \hat{A}_{tm}(\underline{U}_{ij}) \cdot Y_{ij} \leq b_{tm}, \qquad \text{for } t = 1, 2, \ldots, T \quad m = 1, 2, \ldots, M, \tag{22}$$

$$Y_{ij} = 0, 1, \qquad \text{for } i = 1, 2, \ldots, N \quad j = 1, 2, \ldots, K, \tag{23}$$

where

$G_{ij}$ = is the j-th best solution to (C.i),

$\underline{U}_{ij}$ = is a T-component vector and contains the strategy number used at each stage for the j-th best solution to (C.i),

K = number of best solutions to (C.i),

N = number of subproblems,

T = number of stages

M = number of resources at each stage,

$Y_{ij}$ = is a 0-1 decision variable representing the j-th solution to subproblem i,

$\hat{A}_{tm}(\underline{U}_{ij})$ = amount of resource m consumed in the j-th best solution to subproblem i at stage t.

This problem has 'N' GUB constraints (21.i) and the solution procedure uses a modified effective gradient approach to interchange variables within a GUB constraint to obtain a feasible solution. The initial solution to problem D is obtained by setting

$$Y_{i1} = 1 \text{ for all } i = 1, 2, \ldots, N.$$

Note that this is an upper bound on (B). A modified effective gradient for each variable in the solution is then calculated as follows. Let the indices of variables in the solution be $k_1, k_2, \ldots, k_N$. Define a surplus vector $PS = \{PS_{tm}\}$ as:

$$PS_{tm} = \max\,(0, \sum_i \hat{A}_{tm}(\underline{U}_{i,ki}) - b_{tm})$$

$$\text{for } m = 1, 2, \ldots, M \text{ and } t = 1, 2, \ldots, T.$$

The effective gradient for a variable $Y_{i,k_i}$ is given by

$$E_i = \sum_{t=1}^{T} \sum_{m=1}^{M} PS_{tm} \cdot (\hat{A}_{tm}(\underline{U}_{i,k_i}) - \hat{A}_{tm}(\underline{U}_{i,k_i+1}))\}/\ (G_{i,k_i} - G_{i,k_i+1})$$

$$\text{for } i = 1, 2, \ldots, N.$$

This heuristic selects a GUB constraint for variable exchange such that the exchange results in a maximum movement towards feasibility with a minimum decrease in the objective function value. Note that arbitarary exchanges might move the solution away from feasibility. In this case the effective gradient is redefined as

$$E_i = \sum_{t=1}^{T} \sum_{m=1}^{M} PS_{tm} \cdot (\hat{A}_{tm}(\underline{U}_{i,k_i}) - \hat{\underline{A}}_{tm}(\underline{U}_{i,k_i+1})) \cdot (G_{i,k_i} - G_{i,k_i} + 1)$$

$$\text{for } i = 1, 2, \ldots, N.$$

A variable in the current solution which has the smallest effective gradient is selected for exchange. In other words, if

$$F = \min_{i \in N} (E_i),$$

then the variable $Y_{I,k_I}$, whose effective gradient is equal to F, is selected for exchange. Note that if F is negative, then this exchange results in maximum movement towards feasibility with a minimum decrease in the objective function value. Otherwise, the exchange results in minimum movement away from feasibility with a minimum decrease in the objective function value. Note also that only the next best solution in each GUB constraint is considered for exchange at any iteration, and thus any variable that is deleted from a solution is eliminated from subsequent iterations.

At this point, variable $Y_{I,k_I}$ is deleted from the solution and the corresponding variable $Y_{I,k_I+1}$ is added to the solution. If the new solution is feasible, this step of the algorithm is complete. Otherwise, a new set of effective gradients is calculated and the procedure is repeated. It is possible at any one iteration that a variable $Y_{iK}$ may be considered for exchange, but since there are only K variables in a GUB constraint, this exchange is not feasible. There are two methods to resolve this difficulty: 1) choose K as large as possible, and 2) if a variable $Y_{iK}$ ever enters the solution the corresponding GUB constraint is not considered for further exchange. The latter strategy is forced by setting $E_i$ to a very large number. The first method is preferred over the second one, but the computational limitations restrict the value of K to 25 or less. Hence, the second method will be used if K is greater than 25.

## 3.4: Improvement in the Feasible Solution

The solution obtained in the previous step is improved upon by using a maximum gain heuristic. All feasible one for "one variable" exchanges are considered, and the variable exchange that results in maximum improvement in the objective function value is used for interchange. The new solution obtained after an interchange is again considered for further improvement. This is a 'greedy' heuristic and the resulting solution is at least 1-optimal (44). Hence, no single pairwise exchange can give a better solution than the one already obtained.

The algorithmic procedure for this step is as follows: Let $Y_{i,k_i}$, for all i = 1, 2, ..., N, be the variables in the feasible solution from the previous step. All the variables in a GUB constraint are ranked to obtain a set of indices $p_1$, $p_2$, $p_3$, ..., $p_i$, ..., $P_N$ such that

$$p_i = \min (j \,/ \sum_{\ell \neq i} \hat{A}_{tm}(\underline{U}_{\ell, k_\ell}) + \hat{A}_{tm}(\underline{U}_{ij}) < b_{tm},$$

$$\text{for } m = 1, 2, \ldots, M \text{ and } t = 1, 2, \ldots, T),$$

$$\text{for } i = 1, 2, \ldots, N,$$

and define $\Delta_i = G_{i,p_i} - G_{i,k_i}$ , for each variable $Y_{i,k_i}$, i = 1, 2, ..., N.

Determine

$$H = \max_{i \varepsilon N} (\Delta_i)$$

and obtain the variable, $Y_{I,k_I}$, for exchange such that $\Delta_I = H$. If H is greater than zero, the solution can be improved and a new solution is formed by replacing variable $Y_{I,k_I}$ with $Y_{I,p_I}$. This step is repeated until no further improvement can be obtained. This solution is a near optimal, if not optimal, solution to (B).

## 3.5: THE SOLUTION ALGORITHM

A stepwise algorithm based on the procedure previously described is as follows:

Step 1. Define the solution variables N, S, T, and K, where

N = number of subproblems,

S = number of alternatives,

T = number of time periods,

K = number of solutions evaluated for a subproblem.

Step 2. Set $i = 1$

Step 3. Construct a network model for (C.i) using constraints (14) and (15).

Step 4. Solve the network model using a K-shortest path algorithm and evaluate the K best solutions to subproblem (C.i).

Step 5. Set $i = i+1$. If $i < N$ go to Step 3.

Step 6. Set variables $Y_{ij} = 1$ to form the initial solution. Test the feasibility of this solution, if the solution is feasible an optimal solution to problem B is obtained.

Step 7. Let the indices of the variables in the solution be $k_1, k_2, \ldots, k_N$. Calculate the effective gradient $E_i$ for $i = 1, 2, \ldots, N$.

Step 8. Determine the smallest effective gradient value. Let

$$F = \min_{i \in N} (E_i),$$

then the variable $Y_{I,k_I}$ whose effective gradient is equal to F is selected for exchange.

Step 9. Set $k_I = k_I + 1$, and test the new solution for feasibility. If the new solution is feasible go to Step 11.

Step 10. If $k_I = K$ set $E_I = \infty$ and remove the GUB constraint (11.I) from further consideration. Go to Step 7.

Step 11. Determine a set of indices $p_1, p_2, \ldots, p_i, \ldots, p_N$ such that

$$p_i = \min_{j \in K} (j \;/ \sum_{\ell \neq i} \hat{A}_{tm}(\underline{U}_{\ell,k_\ell}) + \hat{A}_{tm}(\underline{U}_{ij}) < b_{tm},$$

for $m = 1, 2, \ldots, M$ and $t = 1, 2, \ldots, T$),

for $i = 1, 2, \ldots, N$.

Define $\Delta_i = G_{i,p_i} - G_{i,k_i}$, for $i = 1, 2, \ldots, N$.

Step 12. Select a variable for exchange as follows: Determine

$$H = \min_{i \epsilon N} (\Delta_i)$$

The variable $Y_{I,k_I}$ is selected for exchange such that $\Delta_I = H$. If H is equal to zero, stop. A near optimal solution is determined. Otherwise, set $k_I = p_I$, update the resource consumed vector, and go to Step 11.

# CHAPTER 4

## AN EXAMPLE PROBLEM FOR 'RAMS-DTO-1

This example was formulated using real field data from District 17 in the State of Texas. Only a few highway segments (15) are considered for this particular example. Of these fifteen, eleven segments were selected for maintenance or rehabilitation by engineers within the District and the other four are good highway segments.

The segments are classified into two types. The first group consists of 'U.S.' and 'State Highways', whereas the second group consists of 'farm roads'. Both groups of highway have asphalt pavements, but have different thicknesses of base course and surface. The farm roads, because of the lower traffic intensity, have thinner base and surface asphalt layers. Methods for rehabilitating the highway segments are different for the two groups of highways.

There are seven 'type 1' highway segments and eight 'type 2' segments, and the needed highway segment information is listed in Table II.

There are six types of distress conditions used to measure highway segment deterioration in this example. The distress types and the associated maximum gain-of rating are listed in Table III. The last distress type, which is a measure of ride smoothness, is not considered important for the highway segments in the second group, and is not considered in evaluating the corresponding highway pavement condition.

There are a total of nine alternatives to be considered for rehabilitating each highway segment. The maintenance strategies and the associated costs (money spent to rehabilitate an area of one mile by one foot) are given in Table IV. Not all strategies are feasible for both types of highway pavements. For example, strategy eight (light duty reconstruction)

TABLE II

HIGHWAY SEGMENT INFORMATION

| Highway Segment Number | Highway Type | Highway Name | County | Length (Mile) | Width (Feet) |
|---|---|---|---|---|---|
| 1 | 1 | US 79 | Milam 0204-05 | 4.530 | 26.000 |
| 2 | 1 | US 77 | Milam 0209-05 | 12.320 | 28.000 |
| 3 | 1 | US 190 | Milam 0815-12 | 3.620 | 26.000 |
| 4 | 2 | SH OSR | Madison 0475-04 | 7.000 | 20.000 |
| 5 | 2 | SH OSR | Madison 0475-03 | 2.260 | 22.000 |
| 6 | 2 | FM 1696 | Walker 1809-02 | 13.800 | 20.000 |
| 7 | 2 | FM 1791 | Walker 1706-01 | 12.370 | 22.000 |
| 8 | 2 | FM 2821 | Walker 2805-01 | 3.340 | 24.000 |
| 9 | 1 | SH 30 | Walker 0212-02 | 7.390 | 26.000 |
| 10 | 1 | SH 36 | Burleson 0816-03 | 12.010 | 26.000 |
| 11 | 1 | US 290 | Washington 0114-09 | 9.021 | 26.000 |
| 12 | 1 | US 79 | Milam 0204-08 | 5.640 | 26.000 |
| 13 | 1 | SH 36 | Burleson 0186-02 | 9.320 | 26.000 |
| 14 | 2 | SH OSR | Brazos 0475-02 | 6.670 | 20.000 |
| 15 | 2 | FM 908 | Milam 0858-02 | 7.440 | 20.000 |

is applicable only to 'group 2' highway segments (farm roads) whereas strategy nine is applicable only to 'U.S.' and "State Highways". The feasible set of strategies for type 1 and type 2 highway segments i listed in Table V.

There are only seven strategies feasible for each highway segment. The improvement in highway quality level through application of appropriate maintenance strategies are listed in Table VI. Note, there is no column for strategy number '1'. Since strategy '1' is a 'do nothing' strategy, the highway pavement level deteriorates on application of strategy '1', and the amount of deterioration depends on which strategy was previously applied to a particular highway segment and when it was applied. It is assumed that application of strategies 8 and 9 to group 1 and group 2 respectively, results in ideal highway pavements and the ideal highway pavement rating is set equal to the points gained by application of these strategies for the respective groups of highway pavements. The highway pavement quality level resulting from application of any maintenance strategy cannot be greater than the ideal highway pavement quality. If an application of any one strategy causes this to occur, the highway pavement quality rating is fixed at the maximum quality level.

All highway segments have identical pavement deterioration curves, for each type of distress. Pavement deterioration curve fractions for each type of maintenance strategy are listed in Tables VII, VII.B, VII.c and VII.D, by distress type. The pavement deterioration curves are determined as the product of road deterioration fractions and the maximum quality levels.

The initial highway pavement ratings by the distress types are given in Table VIII. The minimum pavement quality requirement (RMIN) and the tolerable pavement quality requirements (RTOL) are defined to be 40% and 80% of the maximum quality level. These are used to determine feasible

## TABLE III

## TYPES OF DISTRESSES

| No. | Distress Type | Max. Gain |
|---|---|---|
| 1 | Rutting | 15.000 |
| 2 | Alligator Cracking | 25.000 |
| 3 | Longtud. Cracking | 25.000 |
| 4 | Transverse Cracking | 20.000 |
| 5 | Failures/Mile | 40.000 |
| 6 | Serviceability Index | 50.000 |

## TABLE IV

## MAINTENANCE STRATEGIES

| No. | Strategy | Unit Cost ($/mile-ft) |
|---|---|---|
| 1 | Do Nothing | 0.000 |
| 2 | Fog Seal | 56.000 |
| 3 | Seal Coat | 214.000 |
| 4 | OGPMS | 950.000 |
| 5 | Thin Overlay | 925.000 |
| 6 | Moderate Overlay | 2000.000 |
| 7 | Heavy Overlay | 3549.000 |
| 8 | Lightduty Reconstruction | 944.000 |
| 9 | Heavyduty Reconstruction | 2600.000 |

## TABLE V

FEASIBLE SET OF STRATEGIES FOR THE TWO HIGHWAY SEGMENT TYPES
(1 = FEASIBLE, 0 = NOT FEASIBLE)

| Highway Segment Type | Strategy Number | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 2 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |

## TABLE VI

IMPROVEMENTS OBTAINED BY IMPLEMENTATION OF STRATEGIES

| Strategy Number | Distress Type | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| 2 | 0.0 | 5.0 | 5.0 | 5.0 | 2.0 | 2.0 |
| 3 | 0.0 | 15.0 | 15.0 | 15.0 | 10.0 | 2.0 |
| 4 | 13.0 | 19.0 | 19.0 | 19.0 | 24.0 | 45.0 |
| 5 | 13.0 | 20.0 | 20.0 | 20.0 | 25.0 | 45.0 |
| 6 | 15.0 | 25.0 | 25.0 | 20.0 | 30.0 | 50.0 |
| 7 | 15.0 | 25.0 | 25.0 | 20.0 | 35.0 | 50.0 |
| 8 | 15.0 | 25.0 | 25.0 | 20.0 | 40.0 | 50.0 |
| 9 | 15.0 | 25.0 | 25.0 | 20.0 | 40.0 | 50.0 |

TABLE VII.A
PAVEMENT DETERIORATION FRACTIONS

| Strategy | Year | Distress Type | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| 2 | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 | 0.900 |
| | 3 | 0.998 | 0.998 | 0.790 | 0.560 | 0.998 | 0.700 |
| | 4 | 0.997 | 0.870 | 0.500 | 0.530 | 0.670 | 0.500 |
| | 5 | 0.690 | 0.620 | 0.500 | 0.390 | 0.670 | 0.400 |
| | 6 | 0.670 | 0.500 | 0.210 | 0.190 | 0.330 | 0.300 |
| | 7 | 0.480 | 0.250 | 0.000 | 0.060 | 0.330 | 0.200 |
| | 8 | 0.260 | 0.080 | 0.000 | 0.050 | 0.000 | 0.100 |
| | 9 | 0.170 | 0.000 | 0.000 | 0.000 | 0.000 | 0.100 |
| | 10 | 0.140 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 3 | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.930 | 0.940 | 0.930 | 0.920 | 0.999 | 0.900 |
| | 3 | 0.910 | 0.890 | 0.880 | 0.860 | 0.998 | 0.700 |
| | 4 | 0.880 | 0.890 | 0.870 | 0.850 | 0.780 | 0.500 |
| | 5 | 0.780 | 0.650 | 0.670 | 0.670 | 0.470 | 0.400 |
| | 6 | 0.310 | 0.280 | 0.370 | 0.380 | 0.220 | 0.300 |
| | 7 | 0.220 | 0.240 | 0.320 | 0.330 | 0.200 | 0.200 |
| | 8 | 0.150 | 0.150 | 0.180 | 0.180 | 0.100 | 0.100 |
| | 9 | 0.070 | 0.090 | 0.090 | 0.090 | 0.040 | 0.100 |
| | 10 | 0.050 | 0.070 | 0.070 | 0.060 | 0.010 | 0.000 |

TABLE VII.B.

PAVEMENT DETERIORATION FRACTIONS

| Strategy | Year | Distress Type | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| 4 | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 |
| | 3 | 0.998 | 0.890 | 0.998 | 0.998 | 0.998 | 0.998 |
| | 4 | 0.997 | 0.820 | 0.997 | 0.997 | 0.997 | 0.900 |
| | 5 | 0.880 | 0.730 | 0.996 | 0.996 | 0.996 | 0.800 |
| | 6 | 0.780 | 0.670 | 0.750 | 0.830 | 0.995 | 0.700 |
| | 7 | 0.460 | 0.670 | 0.500 | 0.670 | 0.994 | 0.600 |
| | 8 | 0.250 | 0.670 | 0.500 | 0.670 | 0.330 | 0.500 |
| | 9 | 0.250 | 0.670 | 0.250 | 0.330 | 0.330 | 0.400 |
| | 10 | 0.250 | 0.360 | 0.000 | 0.000 | 0.330 | 0.300 |
| 5 | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 |
| | 3 | 0.998 | 0.950 | 0.930 | 0.940 | 0.998 | 0.998 |
| | 4 | 0.997 | 0.910 | 0.930 | 0.940 | 0.890 | 0.900 |
| | 5 | 0.790 | 0.900 | 0.400 | 0.430 | 0.530 | 0.800 |
| | 6 | 0.750 | 0.610 | 0.140 | 0.180 | 0.230 | 0.700 |
| | 7 | 0.750 | 0.560 | 0.140 | 0.180 | 0.160 | 0.600 |
| | 8 | 0.750 | 0.550 | 0.120 | 0.140 | 0.150 | 0.500 |
| | 9 | 0.750 | 0.510 | 0.070 | 0.060 | 0.130 | 0.400 |
| | 10 | 0.750 | 0.280 | 0.020 | 0.010 | 0.080 | 0.300 |

## TABLE VII.C
## PAVEMENT DETERIORATION FRACTIONS

| Strategy | Year | Distress Type | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| 6 | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 |
| | 3 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 |
| | 4 | 0.997 | 0.997 | 0.997 | 0.997 | 1.000 | 0.997 |
| | 5 | 0.996 | 0.770 | 0.996 | 0.996 | 0.770 | 0.900 |
| | 6 | 0.830 | 0.640 | 0.330 | 0.630 | 0.510 | 0.800 |
| | 7 | 0.710 | 0.580 | 0.110 | 0.260 | 0.480 | 0.700 |
| | 8 | 0.660 | 0.530 | 0.000 | 0.220 | 0.360 | 0.600 |
| | 9 | 0.620 | 0.510 | 0.000 | 0.110 | 0.330 | 0.500 |
| | 10 | 0.380 | 0.380 | 0.000 | 0.040 | 0.240 | 0.500 |
| 7 | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 |
| | 3 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 |
| | 4 | 0.997 | 0.997 | 0.997 | 0.997 | 0.997 | 0.997 |
| | 5 | 0.996 | 0.996 | 0.996 | 0.996 | 0.996 | 0.996 |
| | 6 | 0.995 | 0.710 | 0.330 | 0.330 | 0.750 | 0.900 |
| | 7 | 0.994 | 0.620 | 0.330 | 0.330 | 0.590 | 0.900 |
| | 8 | 0.993 | 0.440 | 0.280 | 0.280 | 0.500 | 0.800 |
| | 9 | 0.992 | 0.290 | 0.170 | 0.170 | 0.500 | 0.700 |
| | 10 | 0.991 | 0.290 | 0.170 | 0.170 | 0.480 | 0.600 |

TABLE VII.D

PAVEMENT DETERIORATION FRACTIONS

| Strategy | Year | | | Distress Year | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| 8 | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 |
| | 3 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 |
| | 4 | 0.997 | 0.997 | 0.997 | 0.997 | 0.997 | 0.900 |
| | 5 | 0.996 | 0.996 | 0.996 | 0.996 | 0.996 | 0.800 |
| | 6 | 0.720 | 0.490 | 0.995 | 0.995 | 0.470 | 0.700 |
| | 7 | 0.670 | 0.360 | 0.994 | 0.994 | 0.360 | 0.600 |
| | 8 | 0.580 | 0.360 | 0.993 | 0.993 | 0.320 | 0.500 |
| | 9 | 0.500 | 0.360 | 0.650 | 0.650 | 0.270 | 0.400 |
| | 10 | 0.500 | 0.290 | 0.600 | 0.600 | 0.270 | 0.300 |
| 9 | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 | 0.999 |
| | 3 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 |
| | 4 | 0.997 | 0.997 | 0.997 | 0.997 | 0.997 | 0.900 |
| | 5 | 0.996 | 0.996 | 9.996 | 0.996 | 0.996 | 0.800 |
| | 6 | 0.720 | 0.490 | 0.995 | 0.995 | 0.470 | 0.700 |
| | 7 | 0.670 | 0.360 | 0.994 | 0.994 | 0.360 | 0.600 |
| | 8 | 0.580 | 0.360 | 0.993 | 0.993 | 0.320 | 0.500 |
| | 9 | 0.500 | 0.360 | 0.650 | 0.650 | 0.270 | 0.400 |
| | 10 | 0.500 | 0.290 | 0.600 | 0.600 | 0.270 | 0.300 |

## TABLE VIII

## CURRENT RATINGS OF HIGHWAY SEGMENTS BY DISTRESS TYPES

| Highway Segment Number | Distress Type Number | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | 10.0 | 5.0 | 20.0 | 17.0 | 20.0 | 12.0 |
| 2 | 10.0 | 15.0 | 25.0 | 20.0 | 40.0 | 3.0 |
| 3 | 10.0 | 10.0 | 15.0 | 13.0 | 40.0 | 0.0 |
| 4 | 10.0 | 20.0 | 20.0 | 20.0 | 40.0 | 0.0 |
| 5 | 10.0 | 25.0 | 25.0 | 20.0 | 40.0 | 0.0 |
| 6 | 10.0 | 25.0 | 25.0 | 20.0 | 40.0 | 0.0 |
| 7 | 8.0 | 0.0 | 10.0 | 20.0 | 10.0 | 0.0 |
| 8 | 10.0 | 15.0 | 25.0 | 20.0 | 20.0 | 0.0 |
| 9 | 15.0 | 25.0 | 5.0 | 5.0 | 40.0 | 42.0 |
| 10 | 15.0 | 25.0 | 25.0 | 20.0 | 40.0 | 47.0 |
| 11 | 15.0 | 25.0 | 25.0 | 17.0 | 40.0 | 47.0 |
| 12 | 15.0 | 25.0 | 25.0 | 17.0 | 40.0 | 49.0 |
| 13 | 13.0 | 25.0 | 25.0 | 20.0 | 40.0 | 50.0 |
| 14 | 8.0 | 5.0 | 0.0 | 17.0 | 20.0 | 0.0 |
| 15 | 10.0 | 10.0 | 10.0 | 8.0 | 20.0 | 0.0 |

strategies in each time period. For example, if upon application of a strategy during a time period, the resulting pavement quality is less than RMIN, then the strategy is deemed infeasible for that time period. If at some stage, the sum of highway quality rating at that stage and the improvement obtained in highway quality rating on application of a strategy is greater than 140% of RMAX for all distress types, then that particular strategy is considered infeasible at that stage. A pavement segment is also not considered for maintenance scheduling if the highway quality levels are greater than RTOL for all distress types. These constraints, along with the multiple choice constraints and the data in Table V are used to construct the network model for each highway segment.

## Resource Requirements

Four types of resource constraints are considered in this example problem. They are: materials, machinery, men and money. Each resource constraint has two types of inputs: resource requirements and availability. Of all the resources, money (budget) is taken as the most significant one.

The first three resources are listed in Table IX. The resource availability (per mile-ft) is also shown. Hence the total available quantity of a certain resource for the district considered will be equal to the product of total area of pavement (mile-ft) in the district and the unit availability of that resource. In Table IX, the first four are materials, the next eight are machinery and the last eight are manpower.

The resource requirements are listed in Tables X.A, X.B, and X.C.

## Experimentation and Results

The highway maintenance problem was solved for three capital availability data sets. Experimentation was used to compare the schedule

## TABLE IX

## RESOURCES AND AVAILABILITY

| No. | Resource Type | Availability (Qty/mile-ft) |
|---|---|---|
| 1 | Surfacing Aggregate | 9.500 |
| 2 | Asphalt Cement | 4.600 |
| 3 | Aggregate (Item 340) | 87.700 |
| 4 | Aggregate Item 290 | 87.700 |
| 5 | Grader | 0.700 |
| 6 | Pickup | 0.700 |
| 7 | Loader | 0.340 |
| 8 | Truck | 0.840 |
| 9 | Roller | 1.000 |
| 10 | Spreader | 0.340 |
| 11 | Laydown Machine | 0.170 |
| 12 | Asphalt Distributor | 0.340 |
| 13 | Grader Operator | 0.700 |
| 14 | Loader Operator | 0.340 |
| 15 | Truck Operator | 0.840 |
| 16 | Roller Operator | 1.000 |
| 17 | Spreader Operator | 1.340 |
| 18 | Laydown MC. Operator | 0.850 |
| 19 | Asphalt Dis. Operator | 0.670 |
| 20 | General Labor | 1.660 |

TABLE X.A

MATERIAL REQUIREMENTS
(UNIT/MILE-FT)

| Strategy | Material Types | | | |
|---|---|---|---|---|
| | 1 | 2 | 3 | 4 |
| 2 | 0.000 | 0.400 | 0.000 | 0.000 |
| 3 | 9.500 | 0.800 | 0.000 | 0.000 |
| 4 | 0.000 | 3.000 | 20.000 | 0.000 |
| 5 | 0.000 | 1.500 | 29.300 | 0.000 |
| 6 | 0.000 | 4.100 | 80.500 | 0.000 |
| 7 | 0.000 | 8.100 | 29.300 | 132.000 |
| 8 | 10.000 | 1.500 | 0.000 | 0.000 |
| 9 | 0.000 | 1.500 | 29.300 | 143.000 |

TABLE X.B

MACHINERY REQUIREMENTS
(Equipment-days/mile-ft)

| Strategy | Machinery Type | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 2 | 0.000 | 0.008 | 0.000 | 0.017 | 0.000 | 0.000 | 0.000 | 0.008 |
| 3 | 0.000 | 0.012 | 0.012 | 0.060 | 0.024 | 0.012 | 0.000 | 0.012 |
| 4 | 0.000 | 0.111 | 0.000 | 0.278 | 0.111 | 0.000 | 0.056 | 0.056 |
| 5 | 0.000 | 0.111 | 0.000 | 0.278 | 0.111 | 0.000 | 0.056 | 0.056 |
| 6 | 0.000 | 0.222 | 0.000 | 0.556 | 0.222 | 0.000 | 0.111 | 0.111 |
| 7 | 0.000 | 0.333 | 0.000 | 0.834 | 0.333 | 0.000 | 0.168 | 0.168 |
| 8 | 0.667 | 0.667 | 0.333 | 1.667 | 1.000 | 0.333 | 0.000 | 0.333 |
| 9 | 1.000 | 0.778 | 0.333 | 3.611 | 1.111 | 0.000 | 0.056 | 0.000 |

## TABLE X.C

## MANPOWER REQUIREMENTS
(man-days/mile-ft)

| Strategy | Manpower Type | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 2 | 0.000 | 0.000 | 0.017 | 0.000 | 0.000 | 0.000 | 0.016 | 0.008 |
| 3 | 0.000 | 0.012 | 0.060 | 0.024 | 0.048 | 0.000 | 0.024 | 0.012 |
| 4 | 0.000 | 0.000 | 0.278 | 0.111 | 0.000 | 0.280 | 0.056 | 0.168 |
| 5 | 0.000 | 0.000 | 0.278 | 0.111 | 0.000 | 0.280 | 0.056 | 0.168 |
| 6 | 0.000 | 0.000 | 0.556 | 0.222 | 0.000 | 0.560 | 0.111 | 0.336 |
| 7 | 0.000 | 0.000 | 0.834 | 0.333 | 0.000 | 0.840 | 0.168 | 0.504 |
| 8 | 0.667 | 0.333 | 1.667 | 1.000 | 1.332 | 0.000 | 0.666 | 1.650 |
| 9 | 1.000 | 0.333 | 3.611 | 1.111 | 0.000 | 0.280 | 0.056 | 1.818 |

generated by the algorithm with a schedule used by the TSDHPT to study the effect of capital availability on the current maintenance schedule. The latter comparison is a form of parametric analysis that is used to study the sensitivity of the schedules generated by the algorithm to resource variability. The three capital availability data sets and the capital actually used by the TSDHPT are as listed in Table XI.

TABLE XI

CAPITAL AVAILABLE IN EACH TIME PERIOD IN $MILLIONS

| Data Set | | | | | Time Period | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | 1.00 | 1.07 | 1.14 | 1.20 | 1.27 | 1.27 | 1.27 | 1.27 | 1.27 | 1.27 |
| 2 | 1.13 | 1.20 | 1.27 | 1.34 | 1.34 | 1.34 | 1.34 | 1.34 | 1.34 | 1.34 |
| 3 | 1.20 | 1.07 | 1.14 | 1.20 | 1.27 | 1.27 | 1.27 | 1.27 | 1.27 | 1.27 |
| TSDHPT | 1.13 | | | | | | | | | |

The current maintenance schedule generated by the algorithm for data set 2 is compared with the schedule used by TSDHPT, and the results for the three data sets are compared to observe the effect of capital availability on the maintenance schedules. Data sets 1 and 2 are compared to note the sensitivity of the solution to an uniform increase of $.13 million in each time period, and the results for data sets 1 and 3 are used to observe the effect of current resource availability on the maintenance schedules. The capital used in each time period and the current maintenance schedules generated by the algorithm for the three data sets, along with the capital and schedule used by TSDHPT, are tabulated in Tables XII and XIII respectively

Data set 1 led to an infeasible solution. There is an insignificant difference in the objective function values for the data sets 2 and 3, due to the abundance of capital in time periods other than time period 1. The highway segments used in the problem are only a part of the total number of highway segments in a District. These highway segments were selected by district supervisors for maintenance in a particular year, and the sets of highway segments scheduled for maintenance in other time periods are different. Thus, to generate resource effective maintenance schedules all the highway segments in a district should be included. The abundance of resources in time periods other than period 1 also results in the best highway conditions possible and hence explains the negligible difference in objective function values (benefits). The upper bound on benefits for the problem is 2,391,862. Highway segments 10 through 13 have been added to the problem to demonstrate the inherent capacity of the algorithm to ignore good highway pavements when no maintenance is required.

The current maintenance schedules generated by the algorithm for data sets 1, 2, and 3 and the schedule proposed by TSDHPT are shown in Table XIII. Data set 1 generated an infeasible solution due to insufficient funds in the early years of the planning period. The overall benefits for data sets 2 and 3 are 2,217,113 and 2,227,004. It is observed that having increased capital in later time periods does not necessarily increase the overall effectiveness of the maintenance schedules.

For data set 2, the algorithm proposed inferior strategies to be applied to segments 1, 4, 6, and 15 compared to the strategies proposed by TSDHPT. At the same time, for segments 7, 8, and 9 it proposed superior strategies.

This has been done to maximize the overall quality of highway segments for the length of the planning period.

The maintenance schedules generated by the algorithm for the data set 2, and the percentage of resource utilized are presented in Tables C.1 and C.2 respectively (see Appendix C). For segment 14, the algorithm and TSDHPT, proposed strategy 8 to be implemented in period 1. According to the schedule generated by the algorithm, any kind of maintenance will be applied to segment 14 only at the end of period 5. For segment 15, the algorithm proposed strategy 2 and TSDHPT proposed strategy 8 to be implemented currently. From Table C.1 it can be seen that segment 15 must be maintained in almost all the years of the planning period. The program has selected the best alternative (maintenance policy) of all the large number of possible alternatives.

## TABLE XII

## CAPITAL CONSUMED IN EACH TIME PERIOD

| Data Set | Time Period | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | ------------- INFEASIBLE SOLUTION --------------------- | | | | | | | | | |
| 2 | 1.13 | 0.98 | 0.97 | 0.66 | 1.29 | 0.91 | 1.21 | 1.31 | 1.08 | 0.92 |
| 3 | 1.18 | 0.90 | 0.98 | 0.61 | 1.18 | 1.03 | 1.08 | 1.18 | 1.24 | 1.01 |
| TSDHPT | 1.13 | | | | | | | | | |

## TABLE XIII

## CURRENT MAINTENANCE SCHEDULES

| Data Set | Highway Segment Number | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 1 | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * |
| 2 | 4 | 5 | 5 | 1 | 1 | 1 | 8 | 1 | 4 | 1 | 1 | 1 | 1 | 8 | 3 |
| 3 | 4 | 5 | 5 | 2 | 3 | 1 | 8 | 8 | 4 | 1 | 1 | 1 | 1 | 8 | 2 |
| TSDHPT | 5 | 5 | 6 | 3 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 8 | 8 |

# CHAPTER 5

## SUMMARY AND CONCLUSIONS

A mathematical model of a multi-period highway maintenance problem was developed. The formulation results in a large scale 0-1 INLP problem, which because of its large size and nonlinearity is beyond the scope of existing solution techniques. The nonlinearity in the problem is due to the dependence of the objective function coefficients upon time and pavement conditions in the problem.

A heuristic solution methodology was used to solve the 0-1 INLP problems. The solution methodology uses the concepts of relaxation, decomposition and network modeling to convert the 0-1 INLP problem to an equivalent 0-1 ILP problem of manageable size. Relaxation of resource constraints and the separable nature of the relaxed problem enables further decomposition of the 0-1 INLP problem into smaller independent subproblems. These subproblems are modeled as longest path network problems, and a combination of best and worst can be evaluated for each subproblem. The solutions to all subproblems are synthesized by using a 0-1 ILP formulation and a good feasible solution is determined. The solutions evaluated for a subproblem are a subset of the total solutions to each subproblem, and hence the region of investigation for the 0-1 ILP problem is a subset of the region for the 0-1 INLP problem. In other words, the 0-1 ILP problem is a restriction of the 0-1 INLP problem. Thus, the optimality of the solution depends upon the region of investigation, and only 'near' optimality can be guaranteed for this solution methodology.

The solution methodology was applied to a multi-period highway rehabilitation and maintenance problem. The data for the problem was obtained from a real world data base and a form of parametric analysis was used to study the effect of capital availability. Highway segments actually used in the problem were only a part of the highway segments in a district. The budget allocation for a district is based upon requirements of all the highway segments in the district, but the data used in the problem is a small subset of the highway segments in a district and resulted in an abundance of resources in most time periods. Thus, the maximum benefits calculated in all the problems were very close to the upper bound generated by the algorithm. The maximum benefits in all cases were within 3% of the upper bound and were relatively insensitive to changes in capital allocations in the first time period.

An efficient computer program based on the solution methodology was written and is presented in Appendix B. The program uses input data similar to the ones used by the other software packages of RAMS family of computer programs. This makes the usage of the program very easy for the user of RAMS computer programs.

A systematic procedure that can be used by engineers within the Districts to optimally schedule rehabiliation and maintenance of highway segments, within constraints imposed by resource availability in the planning period and the specified minimum highway pavement quality requirements over the planning period is presented. The computer program that has been developed will also aid these engineers in determining the funds required for every year of the planning period which in turn will help the state to assess the needs and requirements in planning the rehabilitation and maintenance of highways in the State in future years.

# REFERENCES

1. Ahmed, N.U., "A Code for 0-1 Integer Linear Programming Problems with Multiple Choice Constraints," an unpublished Ph.D. Dissertation, Dept. of Industrial Engineering, Texas A&M University, College Station, Texas (Jan. 1978).

2. Ahmed, N.U., Lu, D.Y., Lytton, R.L., Mahoney, J.P., and Phillips, D.T., "The Rehabilitation and Maintenance District Optimization System," Research Report 207-3, Texas Transportation Institute, Texas A&M University (Nov. 1978).

3. Brooks, R., and Geoffrion, A.M., "Finding Everett's Lagrange Multipliers by Linear Programming," Operations Research, Vol. 14, No. 6(1966), pp. 1149-1153.

4. Dreyfus, S.E, "An Appraisal of Some Shortest Path Algorithm," Operations Research, Vol. 17(1969), pp. 395-412.

5. Geoffrion, A.M., "Lagrangian Relaxation for Integer Programming," Mathematical Programming Study, 291974), pp. 82-114.

6. ______________ and Marsten, R.E., "Integer Programming Algorithms: A Framework and State of Art Survey," Management Science, Vol. 18 (1972), pp. 465-491.

7. Lasdon, Leon S., Optimization Theory for Large Systems, Macmillan, New York, 1970.

8. Lu, D.L., and Lytton, R.L., "Strategic Planning for Pavement Rehabilitation and Management System," Transportation Research Record, No. 598(1976), pp. 29-35.

9. Marsten, R.E., and Morin, T.L., "A Hybrid Approach to Discrete Mathematical Programming," Mathematical Programming, Vol. 14 (1978), pp. 21-40.

10. Nauss, R.M., "An Efficient Algorithm for the 0-1 Knapsack Problem," Management Science, Vol. 23, No. 1(1976), pp. 27-311.

11. Pearman, A.D., "Heuristic Approaches to Road Network Optimization," Engineering Optimization, Vol. 1(1974), pp. 37-49.

12. Phillips, D.T., and Garcia, A., Fundamentals of Network Analysis, Prentice-Hall Inc., New Jersey (1980).

13. Phillips, D.T., and R.L. Lytton, A Resource Constrained Capital Budgeting Model for State Highway Rehabilitation and Maintenance, AIIE Spring Annual Conference, Dallas, Texas, November 1977.

14. Scott, A.J., "The Optimal Network Problem: Some Computational Experience," Transportation Research, 3(1968).

15. Senju, S., and Toyoda, Y., "An Approach to Linear Programming with 0-1 Variables," Management Science, Vol. 15(1968), pp. 196-207.

16. Shier, A., "Iterative Methods for Determining the K Shortest Path," Networks, Vol 6(1976), pp. 206-229.

17. ________, "Computational Experiences with an Algorithm for Finding the K Shortest Paths in a Network," J. Research of the National Bureau of Standards-B. Mathematical Science, Vol. 788 (1974), pp. 139-165.

18. Shapiro, J.F., "Generalized Lagrange Multipliers in Integer Programming," Operations Research, Vol. 19, 1(1971), pp. 68-76.

19. Toyoda, Y., "A Simplified Algorithm for Obtaining Approximate Solutions to 0-1 Programming Problems", Management Science, Vol 21(1975), pp. 1417-1426.

# APPENDIX A

# SOLUTION OF THE MATHEMATICAL MODEL

## APPENDIX A

## SOLUTION OF THE MATHEMATICAL MODEL

The size of INLP problem (A) is beyond the limitations of existing solution techniques. Current solution techniques are limited to around 1000 variable for exact 0-1 ILP and to approximately 2000 variables for a heuristic ILP. The hybrid dynamic programming and branch and bound technique of Marsten and Morin (9) for separable INLP is limited to around 500 variables. A solution methodology for Problem (A) is developed in this appendix. This methodology is heuristic and uses the concepts of relaxation and decomposition, network modeling, and appropriate heuristics. The concepts of relaxation and decomposition are briefly reviewed and an iterative K-shortest path algorithm is described.

### A.1: Relaxation and Decomposition

Relaxation is a fundamental concept which is inherent in most integer programming techniques (6). A problem (PR) is said to be a relaxation of a problem (P) if all the solutions to (P) are a subset of the solutions to (PR). Relaxation of integrality constraints is commonly used to generate bounds and fathom feasible regions without explicit investigation (branch and bound techniques). If one has a prior knowledge of constraints that are not binding at optimality, then these constraints can be relaxed and may result in simpler problem formulation. Define an INLP problem as follows:

Problem P.1

$$\text{Max } Z = f(\underline{X}), \tag{A.1}$$

subject to $g(\underline{X}) \leq 0,$ (A.2)

$$\underline{X} \subseteq Z^n \subseteq R^n, \text{ and } \underline{X} \text{ is integer.} \tag{A.3}$$

Let problem (PR.1) be some relaxation of (P.1). Then (PR.1) has the following properties:

1) If problem (PR.1) has no solution, then (P.1) has no solution.
2) Let $Z^{*R}$ and $Z^{*}$ be optimal solutions to (PR.1) and P.1 respectively; then $Z^{*R} = Z^{*}$, and $Z^{*R}$ is not feasible to (P.1) except at strict equality.
3) If $\underline{X}^{*R}$, an optimal solution to (PR.1), is also feasible for (P.1) then $\underline{X}^{*R} = \underline{X}^{*}$, and $Z^{*R} = Z^{*}$.

Geoffrion has suggested an iterative procedure using relaxation for solving NLP (7). This procedure successively adds and deletes constraints according to specified rules until an optimal solution is obtained. A complicating constraint is a constraint in a problem whose relaxation or elimination results in a much simpler problem. Another form of relaxation is to identify complicating constraints, weight these constraints with multipliers and place them in the objective funtion. This relaxation is called <u>Lagrangian relaxation</u>. Some basic theorems for Lagrangian relaxation have been developed by Everett (7). Shapiro (18) has used this concept for group theoretic approaches to integer programming and Geoffrion (5) has used Lagrangian relaxation for LP based branch and bound techniques. A separable INLP problem can be stated as

Problem P.2.

$$\text{Max } Z = \sum_{j=1}^{N} f_j(X_j), \qquad \text{(A.4)}$$

subject to

$$\sum_{j=1}^{N} A_{jm}(X_j) \leq b_m, \qquad \text{for } m = 1, 2, \ldots, M \qquad (A.5)$$

$$\underline{D}_j(X_j) \leq \underline{c}\ , \qquad \text{for } j = 1, 2\ \ \ldots, N \qquad (A.6)$$

$$X_j = 1, 2, \ldots, \qquad \text{for } j = 1; 2, \ldots, N. \qquad (A.7)$$

A Lagrangian relaxation of (P.2) is given by

Problem P.3

$$L(\underline{U},\underline{X}) = \sum_{j=1}^{N} f_j(X_j) - \sum_{m=1}^{M} \mu_m \cdot (\sum_{j=1}^{N} A_m(X_j) - b_m) \qquad (A.8)$$

subject to constraints (3.6), (3.7), and

$$\mu_m \leq 0 \text{ for } m = 1, 2, \ldots, M. \qquad (A.9)$$

Everett (7) and Brooks and Geoffrion (3) have suggested iterative procedures for solving ILP and determining an optimal set of multipliers $\underline{U}$. These search procedures are usually not viable for INLP problems. Geoffrion (5) has given a lucid exposition of the theory and practical uses of Lagrangian relaxation. Nauss (10) has used Lagrangian relaxation for solving 0-1 ILP problems with multiple choice constraints, where relaxation is used to obtain bounds and penalties for a LP based branch technique.

A result of relaxing constraint (A.5) in (P.2) is that the problem can be decomposed into N independent subproblems, because of the angular nature of constraints (A.6). If the optimal Lagrangian multipliers are known, an optimal solution can be obtained by solving the independent subproblems. This kind of decomposition has been used by Dantzig-Wolfe

(7) (large scale LP), Bender (7) (mixed IP) and in NLP partitioning techniques (7). Lagrangian multipliers or dual variables are iteratively determined in these methods.

## A.2: The Effective Gradient

The concept of an effective gradient was initially used by Senju and Toyoda (15) to heuristically solve large scale 0-1 ILP capital budgeting type problems where feasible projects have to be selected from a larger set of projects so as to maximize the net return within constraints imposed by resources. The effective gradient is a heuristic measure that is used for selecting or deleting projects, and is defined as the ratio of an objective function coefficient of a project to the projection of the project's resource vector at some stage in the solution procedure.

Suppose the initial available resource vector is $\underline{P}$ and let the resources consumed vector after 'n' projects have been selected be $\underline{P}_n$. A unit vector in the direction of $\underline{P}_n$ is $\underline{PU} = \underline{P}_n / |\underline{P}_n|$, where $|\underline{P}_n|$ is the Euclidean norm of the vector. Suppose a project 'j' which has not yet been selected has a resource vector $\underline{R}_j$ and objective function coefficient $C_j$. Then, the effective gradient for project 'j' is calculated as:

$$E_j = C_j/(\underline{R}_j \cdot \underline{P}_n / |\underline{P}_n|), \text{ for } j = 1, 2, \ldots$$

This heuristic selects a project at each iteration that gives a maximum return with a minimum consumption of critical resources, and a similar measure can be used to delete projects in a dual algorithm. This heuristic has been successfully implemented by Senju and Toyoda (15), Toyoda (19), and Ahmed (1). The first algorithm uses a primal approach, whereas the other two are dual algorithms. The first algorithm uses the effective gradient for selecting projects until no other project can be selected, whereas the latter two algorithms use the heuristic to delete projects until a feasible solution is obtained.

## A.3: An Iterative Procedure for Solving K-Shortest Path Problems

Problems involving network analysis subproblems, principally the determination of the shortest path, occur quite often in the analysis and synthesis of transportation and communication networks. However, one is often interested in the 'K' best paths rather than just the best path from one node to another node. Several techniques for determining the k-shortest paths in a network have been developed over the last two decades, but most of them are incorrect and/or inefficient (See Dreyfus (4), and Phillips and Garcia (12) for discussion). The most efficient algorithm to date has been developed by Shier (16). Shier used a special algebra for determining the K-shortest paths network.

Let $R_\infty$ be the set of real numbers to which infinity, ∞, has been appended. Consider for a fixed integer n, the set $S^n$ defined by

$$S^n = \{\underline{a} = (a_1, a_2, \ldots, a_n) : a_i \quad R \quad , a_1 < a_2 < \ldots < a_n\} .$$

Thus $S^n$ consists of n-tuples of numbers from $R_\infty$ arranged in strictly increasing order. Two abstract binary operation ÷ and X on the elements of $S^n$ are defined as follows:

$$\underline{a} + \underline{b} = \underline{c} \longleftrightarrow c_j = \min_j \left[\{a_1, \ldots, a_n\} \text{ U } \{b_1, \ldots, b_n\}\right] ,$$

$$\text{for } j = a1, 2, \ldots, n,$$

$$\underline{a} \text{ x } \underline{b} = \underline{c} \longleftrightarrow c_j = \min_j \{a_i + b_\ell : i, \ell = 1, 2, \ldots, n\},$$

$$\text{for } j = 1, 2, \ldots, n,$$

where $\text{Min}_j[W]$ denotes the $j^{th}$ smallest distinct element of the set W. For example, let $\underline{a} = \{1,2,3,4\}$ and $\underline{b} = \{2,3,5,7\}$, then $\underline{a} + \underline{b} = \{1,2,3,4,\}$ and $\underline{a} \times \underline{b} = \{3,4,5,6\}$. Also define $\underline{e} = (0, \infty, \infty, \ldots, \infty)$ and $\underline{v} = (\infty, \infty, \ldots, \infty)$. These operations also hold for conformable matrices, where each element of a matrix represents an n-tuple.

This basic algebra is used in Shier's K-shortest path algorithm. Suppose $G(N,K)$ is a finite directed network over $R_\infty$, where $N$ = set of nodes and $K \leq N \cdot$ arcs. Let $\ell_{ij} \varepsilon R_\infty$ be the length of an arc from node i to node j. A path of size 'm' between nodes $i_1$ to $i_m$ is an ordered sequence of arcs $(i_1,i_2), (i_2,i_3), \ldots, (i_{m-1}, i_m)$ in the network. A path is elementary if all nodes are distinct; a circuit is a path such that $i_1 = i_m$, and a circuit is elementary if all nodes $i_1$ to $i_{m-1}$ are distinct. The length $(\lambda)$ of a path is defined as the arithmetic sum of arc lengths $\ell_{ij}$ along the path. If there is no arc from node 'i' to node 'j' then define $\ell_{ij} = \infty$. $\underline{A} = (a_{ij}) \varepsilon M^K$, an NxN matrix, is called the arc length matrix, where $a_{ij} = \{\ell_{ij}, \infty, \ldots, \infty\}$.

The K-shortest path lengths from node i to node j are given by $a^*_{ij}$ where $\underline{A}^* = (a^*_{ij})$ is defined as follows:

$$\underline{A}^* = \sum_{j=1}^{\infty} \underline{A}^j = \underline{E} + \underline{A}^1 + A^2 + \ldots,$$

where $\underline{E}$ is a square matrix whose diagonal elements are 'e' and all other elements are 'v', $\underline{A}^0 = \underline{E}$, and $\underline{A}^{j-1} \times \underline{A} = \underline{A}^j$, for $j$ 1. Thus, the problem of determining the K-shortest path length in a network is solved by calculating $\underline{A}^*$. The following will be stated without proof (for proof see Shier (16):

Lemma: Suppose $\underline{A}$ is the arc length matrix for an N-node network G. Then, there exists $w > 0$ such that $\underline{A}^* = \underline{E} + \underline{A} + \ldots + \underline{A}^h$ for all $h \geq w$.

This lemma states that, $\underline{A}^*$ the matrix of K-shortest paths between every pair of nodes in a network, can be obtained by adding the first w terms in the equation defining $\underline{A}$.

Shier (16) compared three methods for computing the K-shortest path lengths in an N-node network G from a given node 'S' to all other nodes in the network. The three methods were: the Jacobi method, the Gauss-Seidel method and the double sweep method. The study required the determination of the S-th row of matrix $\underline{A}^*$, where $\underline{A}$ is the arc length matrix for G. The last method was found to require the least number of computations and thus the double sweep method will be used in this research.

The double sweep method is an iterative procedure and consists of two phases called the backward and forward passes. The arc length matrix is split into two strictly upper and lower triangular matrices, $\underline{V}$ and $\underline{L}$ respectively, such that $\underline{A} = \underline{V} + \underline{L}$. During the forward pass, only the lower triangular matrix $\underline{L}$ is used. Explicitly, an iteration r is as follows: Let $\underline{X}^0$ be a given row vector. Successive computations of $\underline{X}^{2r-1}$ and $\underline{X}^{2r}$ are

$$\underline{X}^{2r-1} = \underline{X}^{2r-1} \otimes \underline{L} \oplus \underline{X}^{2r-2} \qquad \text{(backward pass)}$$

$$\underline{X}^{2r} = \underline{X}^{2r} \otimes \underline{V} \oplus \underline{X}^{2r-1} \qquad \text{(forward pass)}$$

where $r > 0$ is the iteration number.

The iterative procedure is stopped when $\underline{X}^{2r-1}$ is equal to $\underline{X}^{2r}$ for some iteration 'r', and $\underline{X}^{2r} = \underline{X}^{2r-1}$ = Sth row of $\underline{A}^*$ where the K-shortest paths from 'S' to all other nodes are desired (for proof see Shier (16)).

Let $a^*_{S3} = a^{*1}_{S3}, a^{*2}_{S3}, \ldots, a^{*K}_{S3}$, then $a^{*1}_{S3}$ is the shortest path length, $a^{*2}_{S3}$ is the next best path length, and so on, from node 'S' to node 3.

## A.4: Solution Methodology for Problem (A)

Problem (A) can be stated in its general form as follows:

Problem P.4

$$\text{Max } Z = \sum_{i}^{N} \sum_{\ell}^{L} \sum_{q}^{Q} f_{i\ell q}(\underline{X}_i) \qquad (A.20)$$

subject to

$$\sum_{i=1}^{N} A_m(\underline{X}_i) \leq b_m \qquad \text{for } m = 1, 2, \ldots, M \qquad (A.21)$$

$$\underline{C}_i(\underline{X}_i) \leq \underline{d}_i \qquad \text{for } i = 1, 2, \ldots, N \qquad (A.22)$$

$$\underline{X} = \{x_{i11}, x_{i12}, \ldots, x_{i1Q}, x_{i21}, \ldots, x_{i\ell q}, \ldots, x_{iLQ}\}$$

$$\text{for } i = 1, 2, \ldots, N$$

$$x_{i\ell q} = 0,1 \qquad \text{for } i = 1, 2, \ldots, N \qquad (A.23)$$

$$\ell = 1, 2, \ldots, L$$

$$q = 1, 2, \ldots, Q$$

where constraints (A.22) consist of multiple choice constraints, precedence relationships, and other alternative feasibility constraints. Constraints (A.21) are the resource constraints.

This problem has an angular structure and constraints (A.21), are coupling constraints that link the $X_i$ decision variables. If the coupling constraints (A.21) were relaxed, then the problem could be decomposed into 'N' independent subproblems. A Lagrangian relaxation of (P.4) results in the following dual problem:

Problem P.5

$$\text{Max } Z = \underset{\underline{U}}{\text{Min}} \ (\underset{\underline{X}}{\text{Max}} \ L \ (\underline{U},\underline{X}) = \sum_i \sum_\ell \sum_q f_{i\ell q}(\underline{X}_i) - \sum_{m=1}^{M} u_m(\sum_{i=1}^{N} A_{im}(\underline{X}_i) - b_m)) \qquad (A.24)$$

subject to

$$u_m \geq 0, \text{ for } m = 1,2,\ldots,M \qquad (A.25)$$

and constraints (A.22), and (A.23).

This problem is easy to solve if the optimal set of Lagrangian multipliers '$\underline{U}^*$' were known. If the '$u_m$' are fixed to some value, an optimal solution relative to the values of the Lagrangian multipliers can be obtained by decomposing (P.5) into N independent subproblems and solving each of them. An iterative procedure similar to Everett's procedure (7) for determining an optimal set of multipliers $u_m$'s could be used, but this would require solving N subproblems at each iteration. This is by no means an easy task because of the size of each subproblem, and an iterative procedure would be generally computationally infeasible. There might also exist a gap between dual and primal solution such that:

$$\text{Max } A < \underset{\underline{U}}{\text{Min}} \ (\underset{\underline{X}}{\text{Max}} \ L(\underline{U},\underline{X})) \qquad (A.26)$$

Thus, this approach to determine the solution iteratively is abandoned.

Another approach would be to fix all $u_m$'s to a value of '0', which results in a complete relaxation of constraints (A.21), solve the resulting subproblems and then integrate constraints (A.21) into the solution. Integration of constraints (A.21) is achieved by finding the K-best solutions to each of the subproblems, and determining an optimal solution to problem (P.5) from amongst these solutions. Relaxation of constraint (A.21) results in the following N independent subproblems.

Problem P.5.i

$$\text{Max} \sum_{\ell=1}^{L} \sum_{q=1}^{Q} f_{i\ell q}(\underline{X}_i) \tag{A.27}$$

subject to

$$\underline{C}_i(X_i) \leq \underline{d}_i,$$

$$x_{i\ell q} = 0, 1, \qquad \text{for } i = 1, 2, \ldots, N$$
$$\ell = 1, 2, \ldots, L$$
$$q = 1, 2, \ldots, Q$$

Since constraints (A.22), and (A.23) consist only of multiple choice constraints, integrality constraints, and precedence and feasibility constraints, it is easy to model this problem as a longest path network problem. A network formulation of problem (P.5.i) is shown in Figure A.1. Source node '$s_i$' represents a starting point. A set of feasible variables 'L1', with respect to constraints (A.22), is determined from among the variables $x_{i\ell 1}$ where $\ell = 1, 2, \ldots, L$. Let the number of variables in L1 be equal to $m < L$. Arcs originate from node '$s_i$' leading to nodes 1, 2, ..., m. A new set of feasible variables is determined from among variables $x_{i\ell 2}$ for each node 1, 2, ... m, and more acrs are added from these nodes m+1,...,n corresponding to the feasible variables. This process is repeated and new arcs and nodes are added, until nodes r, r+1...,t are generated for variables $x_{i\ell Q-1}$. Arcs from these nodes r, r+1, ... are connected to a sink node '$e_i$'. Arc lengths correspond to variables $x_{i\ell q}$ and are calculated from the objective function $f_{1\ell q}(\underline{X}_i)$.

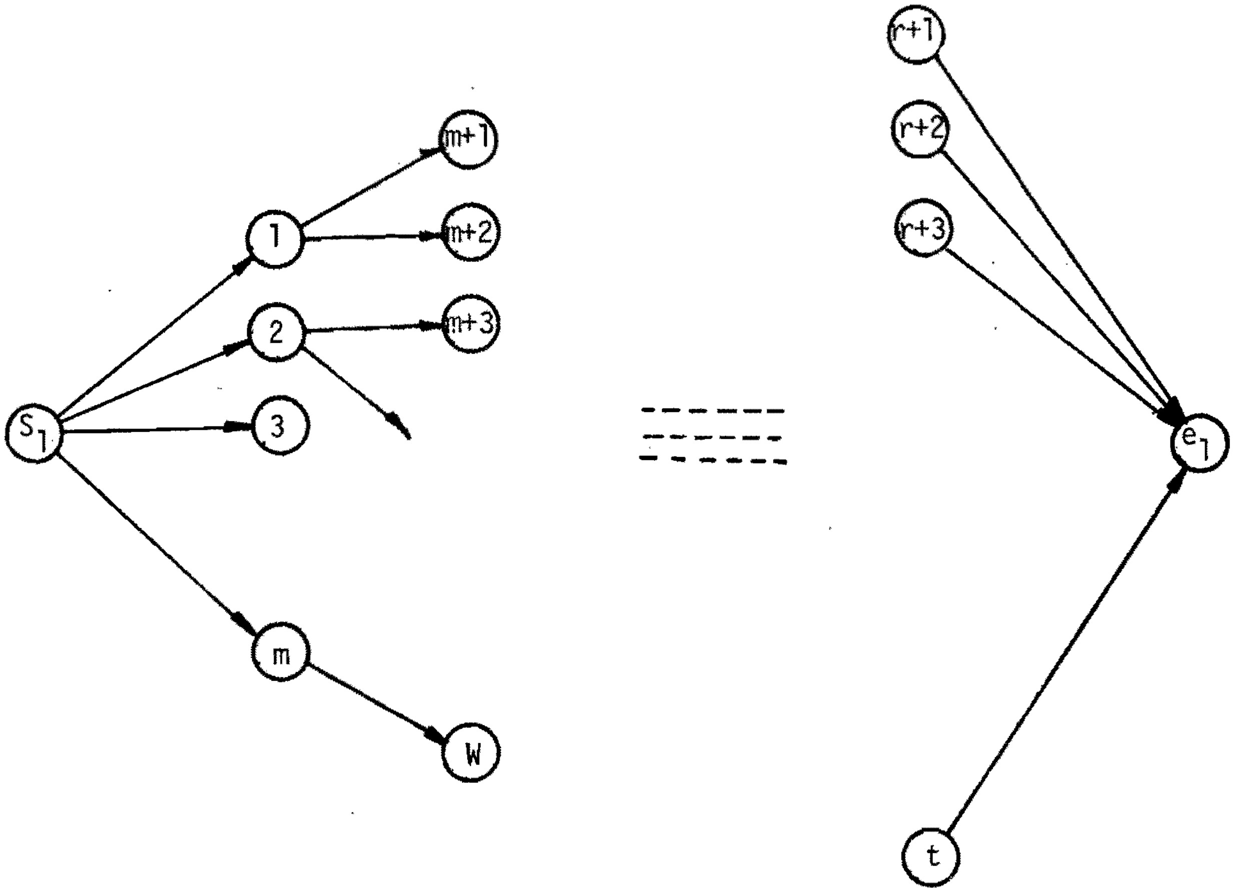


Figure A.1 - A Network Model for Problem (P.5.1)

A path from source node '$s_i$' to sink node '$e_i$' represents a feasible solution to problem (P.5.i), and the objective function value is equal to the sum of individual arc lengths along the path. The longest path from source node '$s_i$' to sink node '$e_i$' is an optimal solution to problem (P.5.i). If the networks for each problem (P.5.i) were linked sequentially, (see Figure A.2) such that $e_i$ is connected to $s_{i+1}$ for $i=1,2,\ldots,N-1$, then the longest path from source node $s_1$ to sink node $e_N$ is an optimal solution to a relaxation of problem (P.4) (constraints (A.21) are relaxed). This path length is the sum of the longest paths in the network models for (P.5.i). This solution would also be an optimal solution to (P.4) if it satisfied constraints (A.21), but this would rarely occur. Suppose one determined the K-best paths (longest paths) corresponding to the network formulation in Figure A.2, and determined the best path which is feasible with respect to constraints (A.21). If this could be done, an optimal solution to (P.4) could be obtained. However, the number of best paths, K, that would have to be evaluated may be large. Note that the number of possible solutions to the entire network of Figure (A.2) is equal to $K^N$, where K is the number of solutions evaluated for each subproblem, and N is the number of subproblems. Even for small values of K and N this number would be very large, hence the network model of Figure (A.2) is computationally infeasible.

The longest path network formulation of (P.5.i) is solved by using an iterative K-shortest path algorithm (11). The network formulation is very effective for the algorithm, and at most two iterations are required to obtain the required solutions to (P.5.i). Note that if the nodes are numbered such that all the arcs lead from a smaller numbered node to a larger numbered node, then the lower triangular part of the arc length matrix only has infinity as its elements. Hence, the iterative procedure

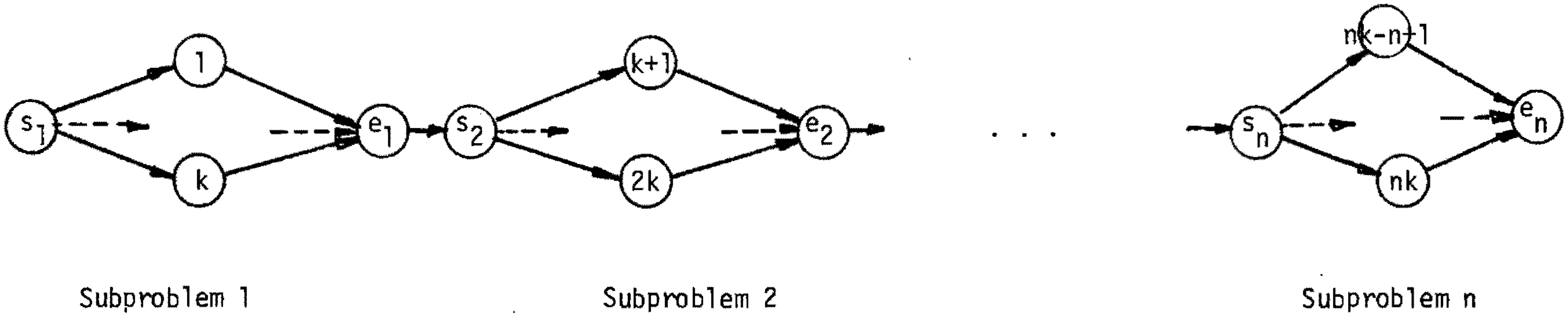


Figure A.2. A Network Model Formulation for a Relaxed Problem (P.5)

will converge after 2 iterations.

Another method for solving (P.5) is to formulate a 0-1 ILP model using the K-best solutions to each subproblem. Let K be the number of best solutions to each of the problems (P.5.i), and let the 'K' best solutions to (P.5.i) be $G_{i1}$, $G_{i2}$, ..., $G_{iK}$ such that $G_{i1} > G_{i2} > \ldots > Gi_k$. Define a 0-1 variable $Y_{ij}$ for the j-th best solution to (P.5.i) and a Q-component vector $\underline{U}_{ij}$ such that $U_{ij}q$ contains the variable $x_{i\ell q}$ in the j-th best solution to problem (P.5.i). If $U_{ij}{}^q = p$, then $x_{ipq} = 1$ in (P.4). Since only the 'K' best solutions to each of the problems (P.5.i) are synthesized, the resulting problem is a restriction of (P.4). This restricted 0-1 formulation of (P.4) is as follows:

Problem P.6

$$\text{Max } Z = \sum_{i=1}^{N} \sum_{j=1}^{K} G_{ij} \cdot Y_{ij} \qquad \text{(A.28)}$$

subject to

$$\sum_{j=1}^{K} Y_{ij} = 1, \qquad \text{for } i = 1, 2, \ldots, N \qquad \text{(A.29)}$$

$$\sum_{i} \sum_{j} \hat{A}_m(\underline{U}_{ij}) \cdot Y_{ij} = b_m, \qquad \text{for } m = 1, 2, \ldots, M \qquad \text{(A.30)}$$

$$Y_{ij} = 0, 1, \qquad \text{for } i = 1, 2, \ldots, N$$
$$j = 1, 2, \ldots, K$$

where $\hat{A}_m(\underline{U}_{ij})$ is the amount of resource m required for the j-th best solution to subproblem i.

This restriction of (P.4) is not only linear but the relaxation, decomposition, and subsequent synthesis has reduced the number of variables from approximately 15000 ($N \cdot L \cdot Q$; $N=100$, $L = 15$, and $Q = 10$) to about 2000 ($N \cdot K$, $N = 100$, and $K = 20$). The solution to (P.6) may not be optimal for (P.4) because of the restricted region of (P.6). The quality of the solution to (P.6) depends upon the value of K and the nature of constraints (A.30). Problem (P.6) is a large scale 0-1 ILP and a procedure based on the concept of an effective gradient (15,1) will be used for its solution. This solution technique will be heuristic and dual in nature.

A combination of the best solutions to each subproblem (P.5.i) is used as the initial solution. Hence each $Y_{i1}$ are set equal to 1 in the starting solution. Variables in the generalized upper bounding (GUB) constraints (A.29) are then exchanged such that at each iteration the variable exchange moves the solution towards feasibility with a minimum reduction in the objective function value. The effective gradient is defined as follows:
Let the surplus vector be $PS = \{PS_m\}$ such that

$$PS_m = \text{Max}\ (0, \sum_{i=1}^{N} \sum_{j=1}^{K} \hat{A}_m(\underline{U}_{ij}) \cdot Y_{ij} - b_m) \text{ for } m = 1, 2, \ldots, M$$

If $Y_{ik}$ is in the current solution, then the corresponding effective gradient $E_i$ is

$$E_i = \sum_{m=1}^{M} PS_m \cdot (\hat{A}_m(\underline{U}_{ik}) - \hat{A}_m(\underline{U}_{i,k+1}))/(G_{ik} - G_{i,k+1})$$

$$\text{for } i = 1, 2, \ldots, N.$$

There are other heuristics that can be used for this interchange process. For example, a GUB set to be considered for exchange is such that deletion of a variable in the GUB set from the current solution results in maximum movement towards feasibility with a minimum loss in objective funtion value. This process of variable interchange within GUB sets is continued till a feasible solution is obtained. This feasible solution is improved upon by using a greedy heuristic such that the variable with a maximum gain in objective function value is added and that variable from the corresponding GUB set is deleted from solution without destroying feasibility.

# APPENDIX B

# DOCUMENTATION OF RAMS-DTO-1

# A MULTIPERIOD DISTRICT OPTIMIZATION PROGRAM

## REHABILITATION AND MAINTENANCE SYSTEMS

## (DISTRICT TIME OPTIMIZATION)

### B.1: PROGRAM INFORMATION

| | |
|---|---|
| Authors: | Shashikant Sathaye<br>C.V. Shanmugham |
| Installation: | Amdahl 470V/6, Data Processing Center, Texas A&M University, College Station, Texas |
| Language: | Fortran IV |
| Date Written: | Spring 1980 |

### SUBROUTINE SETUP

The program contains the following subroutines: NETINP, NETGEN, FEASBL, DSHPTH, DSWP, XMULT, TRACE, RESINP, EFGRAD, SORTI, NIMGRD, FCNSRC and RESULT. The structural relationships between subroutines are shown in Figure B.1. The program runs in two phases; a network modeling phase and a resource synthesis phase. These 2 phases operate independently, but the output from the first phase is used as input to the second. The proper sequence of execution is controlled by the MAIN program. The subroutine NETINP reads in resources requirements and availability (except capital) and all the other data required for network formulation.

The subproblems are independent and can be solved separately. Subroutine NETINP is structured such that each subproblem data is separately read into the program and at any one time, data for only one subproblem resides in the computer core. This aids in reducing the computer core storage requirements.

After data for a subproblem is read into the program, subroutine NETGEN is executed. This subroutine is used to construct longest path network models of the subproblems. The longest path network models of the

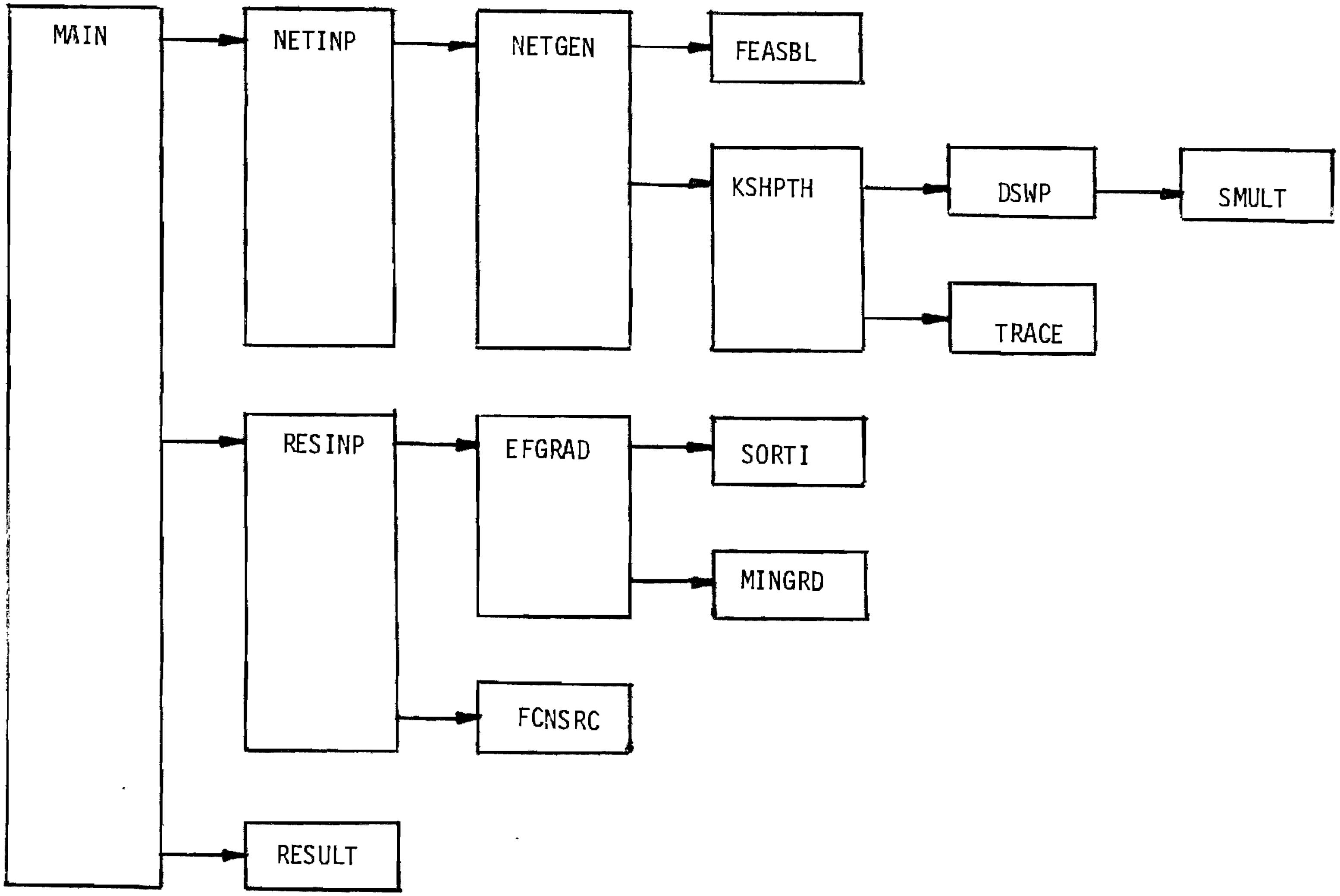


Figure B.1 - Structural Setup of Subroutines

subproblems are solved using a K-shortest path algorithm and subroutine KSHPTH is used for this purpose. The network modelling section operates on each subproblem data and the desired solutions to the subproblems are generated. At the completion of network modelling phase, control is transferred back to the MAIN program. Then the MAIN program calls the synthesis section. Subroutine RESINP is used for budget data input to the synthesis section. The solutions to the subproblems are transferred to the synthesis section through a COMMON block. Resource data is read into the program in subroutine RESINP. Then subroutine EFGRAD is executed and a good initial solution is generated. This solution is improved in subroutine FNSRC to generate a near optimal, if not optimal, solution. Finally, the subrouting RESULT prints out the optimal solution, if it exists.

### Subroutine NETINP

NETINP reads the resource requirements and availability. It also reads in the gain-of-rating matrix, maximum rating available, pavement survival fractions. It calls on NETGEN to generate the feasible network for a highway segment.

### Subroutine NETGEN

This subroutine is used to generate a network for each highway segment such that each path in the network from the source to the sink (node 1 to the largest node) represents a feasible set of variables with respect to the constraint matrix. After a network has been generated, NETGEN calls on KSHPTH (a K-shortest path algorithm) to determine the K-best paths.

Subroutine FEASBL

Called from NETGEN, this subroutine generates a set of feasible alternatives at each vertex in the network.

Subroutine KSHPTH

The set of subroutines KSHPTH, DSWP, XMULT and TRACE runs Shier's (46) algorithm to find the K-shortest paths.

Subroutine RESINP

This subroutine is used to read in the budget available for the years in the planning period. It also sets up the resource requirements and availability matrix for all the years.

Subroutine EFGRAD

The set of subroutines EFGRAD, SORTI, and MINGRD generates an initial solution to the problem.

Subroutine FCNSRC

From the initial solution generated earlier, a subroutine FCNSRC determines an optimal, or near optimal solution, to the multiperiod highway maintenance problem.

Subroutine RESULT

RESULT prints out the optimal decisions, i.e. the maintenance strategies to be used in the planning period for each highway segment in the district. It also prints out the resource utilization for each and everyone of the resources in the planning period.

## B.2: DESCRIPTION OF INPUT DATA

The following instructions must be followed for the proper execution of the program:

The value of an entry classified as an integer must be entered right-justified in the designated columns. The value of a real variable should be entered within the designated columns, with a decimal period. All alpha numeric variables are entered, left justified

CARD A (one card only)

The problem parameters are entered in this card.

| Column | Variable | Description | Type |
|---|---|---|---|
| 6 - 10 | NH | Number of highway segments | Integer |
| 11-15 | NS | Number of maintenance strategies excluding the "Do Nothing" alternative | Integer |
| 16 - 20 | NDIST | No. of distress types | Integer |
| 21 - 25 | NT | No. of years in the planning period | Integer |
| 26 - 30 | NM | No. of material resources | Integer |
| 31 - 35 | NE | Number of equipment Resources | Integer |
| 36 - 40 | MN | No. of manpower types | Integer |
| 41 - 45 | NTYP | No. of Highway types - (Currently set as 2) | Integer |
| 46 - 50 | KPATH | No. of best solutions generated for each segment (Subproblem) | Integer |

CARD B (NS number of cards)

The names of maintenance strategies and the cost of applying a strategy on a segment of one mile long and one foot wide are entered in the set B. One card for each strategy J.

| Column | Variable | Description | Type |
|---|---|---|---|
| 6-25 | -- | Strategy Name | Alpha-Numeric |
| 26-40 | A (J,1,1) | Unit-Cost/mile-foot | real |

CARD C (NDIST Number of Cards)

The names of distress types are entered in this set. One card per distress type.

| Column | Variable | Description | Type |
|---|---|---|---|
| 6-25 | -- | Distress Type | Alpha-Numeric |

CARD D [(NM+NE+MN) Number of Cards]

The resource types (excluding budget) and their availability per mile-foot of segment in the district are entered in the set D one card per resource, J.

| Column | Variable | Description | Type |
|---|---|---|---|
| 6-25 | -- | Resource Type | Alpha-Numeric |
| 26-40 | BRES (J,1) | Availability (Qty/ Mile-foot) | Real |

CARD E (NS Number of Cards)

The material requirements associated with each strategy are input within data set. If the materials types are less than 10, use one card per strategy, J. Otherwise, use $|\frac{NM}{10}| + 1$ number of cards. In the second case, the total number of cards will be NS * ($|\frac{NM}{10}| + 1$).

| Column | Variable | Description | Type |
|---|---|---|---|
| 11-17 | A(J,2,1) | Material 1 requirement for strategy J | Real |
| 18-24 | A(J,3,1) | Material 2 requirement for strategy J | Real |
| . | . | . | . |
| . | . | . | . |
| 74-80 | A(J,11,1) | Material 10 requirement for strategy J | Real |

CARD F (NS Number of Cards)

The machinery requirements are input in this data set. In the number of cards needed see section CARD E and replace NM by NE.

| Column | Variable | Description | Type |
|---|---|---|---|
| 11-17 | A(J,NM+2,1) | Equipment 1 requirement for strategy J | Real |
| 18-24 | A(J,NM+3,1) | Equipment 2 requirement for strategy J | Real |
| . | . | . | . |
| . | . | . | . |
| 74-80 | A(J,NM+11,1) | Equipment 10 requirement for strategy J | Real |

CARD G (NS Number of Cards)

The various manpower requirements are input in this data set. In the number of cards needed, see section CARD E and replace NM by MN.

| Column | Variable | Description | Type |
|---|---|---|---|
| 11-17 | A(J,NM+NE+2,1) | Manpower 1 requirement for strategy J | Real |
| 18-24 | A(J,NM+NE+3,1) | Manpower 2 requirement for strategy J | Real |
| . | . | . | . |
| . | . | . | . |
| 74-80 | A(J,NM+NE+11,1) | Manpower 10 requirement for strategy J | Real |

CARD H (NS Number of Cards)

The gain of rating matrix (strategy vs distress type) is entered in this set. If NDIST is greater then 10, see section CARD E, to find the number of cards required and replace NM by NDIST. A new card is used for each strategy.

| Column | Variable | Description | Type |
|---|---|---|---|
| 11-17 | RIMP(J,1) | Gain of rating for distress 1, when strategy J is applied | Real |
| 18-24 | RIMP(J,2) | Gain of rating for distress 2, when strategy J is applied | Real |
| . | . | . | . |
| . | . | . | . |
| 74-80 | RIMP(J,10) | Gain of rating for distress 10, when strategy J is applied | Real |

CARD I (NH Number of Cards)

This data set describes the current pavement quality, with respect to the distress types. If NDIST is greater than 10, see section CARD H, for the number of cards required. Use a new card for each segment I.

| Column | Variable | Description | Type |
|---|---|---|---|
| 11-17 | RC(I,1) | Current pavement rating of segment I for distress type 1 | Real |
| 18-24 | RC(I,2) | Current pavement rating of segment I for distress type 2 | Real |
| 74-80 | RC(I,10) | Current pavement rating of segment I for distress type 10 | Real |

CARD J ((NS*NT) Number of Cards)

The pavement deterioration fractions are entered in this data set. The data is grouped into NS number of sections, each section having NT number cards. If NDIST is greater than 10, each section will have $|\frac{NT}{10}| + 1$ number of cards. A new card will be used for each strategy J and a year L.

| Column | Variable | Description | Type |
|---|---|---|---|
| 11-17 | RDET(J,1,L) | Pavement deterioration fraction of distress type 1, for strategy J and year L | Real |
| 18-24 | RDET(J,2,L) | Pavement deterioration fraction of distress type 2, for strategy J and year L | Real |
| . | . | . | . |
| . | . | . | . |
| 74-80 | RDET(J,10,L) | Pavement deterioration fraction of distress type 10, for strategy J and year L | Real |

CARD K (NTYP(=2) Number of Cards)

This data contains the feasible set of maintenance strategies for the two highway types. 1 implies feasibility and 0 implies infeasible strategy. One card for each type I.

| Column | Variable | Description | Type |
|---|---|---|---|
| 6 - 10 | IY(I,1) | Indication if strategy 1 is feasible for higway type I | Integer |
| 11 - 15 | IY(I,2) | Indicator if strategy 2 is feasible for highway type I | Integer |
| . | . | . | . |
| . | . | . | . |
| 76-80 | IY(I,15) | Indicator of strategy 15 is feasible for highway type I | Integer |

CARD L (NH Number of Cards)

The highway segment information are entered within data set.* One card for each segment I.

| Column | Variable | Description | Type |
|---|---|---|---|
| 6-8 | - | Segment Number | Integer |
| 9-10 | ITYP(I) | Highway type of segment 1 | Integer |
| 11-35 | -- | Identification of Segment I | Alpha-numeric |
| 39-45 | L1(I) | Length of segment I, (miles) | Real |
| 46-51 | L2(I) | Width of segment I, (feet) | Real |

CARD M ($|\frac{NT}{7}|$ + 1 Number of Cards)

The budgets for each and every year of the planning period are entered in this set.

| Column | Variable | Description | Type |
|---|---|---|---|
| 6-15 | BRES (1,1) | Budget for year 1 | Real |
| 16-25 | BRES (1,2) | Budget for year 2 | Real |
| . | . | . | . |
| . | . | . | . |
| 66-75 | BRES (1,7) | Budget for year 7 | Real |

# APPENDIX C

# INPUT AND OUTPUT OF THE COMPUTER PROGRAM

INPUT DATA FOR EXAMPLE PROBLEM

```
000000000111111111122222222223333333333444444444455555555556666666666777777
123456789012345678901234567890123456789012345678901234567890123456789012345

A      15     8     6    10     4     8     8     2    15
B     FOG SEAL                     56.000
B     SEAL COAT                   214.000
B     OGPMS                       950.000
B     THIN OVERLAY                925.000
B     MODERATE OVERLAY           2000.000
B     HEAVY OVERLAY              3549.000
B     LIGHTDUTY RECONSTRUC        944.000
B     HEAVY DUTY RECONSTRT       2600.000
C     RUTTING                      15.000
C     ALLIGATOR CRACKING           25.000
C     LONGTUD. CRACKING            25.000
C     TRANSVERSE CRACKING          20.000
C     FAILURES/MILE                40.000
C     SERVICEABILITY INDEX         50.000
D     SURFACING AGGREGATE           9.500
D     ASPHALT CEMENT                4.600
D     AGGREGATE(ITEM 340:)         87.700
D     AGGREGATE ITEM 290           87.700
D     GRADER                        0.700
D     PICKJP                        0.700
D     LOADER                        0.340
D     TRUCK                         0.840
D     ROLLER                        1.000
D     SPREADER                      0.340
D     LAYDOWN MACHINE               0.170
D     ASPHALT DISTRIBUTOR           0.340
D     GRADER OPERATOR               0.700
D     LOADER OPERATOR               0.340
D     TRUCK OPERATOR                0.840
D     ROLLER OPERATOR               1.000
D     SPREADER OPERATOR             1.340
D     LAYDOWN MC. OPERATOR          0.850
D     ASPHALTDIS. OPERATOR          0.670
D     GENERAL LABOR                 1.660
E         1  0.000  0.400  0.000  0.000
E         2  9.500  0.800  0.000  0.000
E         3  0.000  3.000 20.000  0.000
E         4  0.000  1.500 29.300  0.000
E         5  0.000  4.100 80.500  0.000
E         6  0.000  8.100 29.300132.000
E         7 10.000  1.500  0.000  0.000
E         8  0.000  1.500 29.300143.000
F         1  0.000  0.008  0.000  0.017  0.000  0.000  0.000  0.008
F         2  0.000  0.012  0.012  0.060  0.024  0.012  0.000  0.012
F         3  0.000  0.111  0.000  0.278  0.111  0.000  0.056  0.056
F         4  0.000  0.111  0.000  0.278  0.111  0.000  0.056  0.056
F         5  0.000  0.222  0.000  0.556  0.222  0.000  0.111  0.111
F         6  0.000  0.333  0.000  0.834  0.333  0.000  0.168  0.168
F         7  0.667  0.667  0.333  1.667  1.000  0.333  0.000  0.333
F         8  1.000  0.778  0.333  3.611  1.111  0.000  0.056  0.000
```

```
000000000111111111122222222223333333333444444444455555555556666666666777777
123456789012345678901234567890123456789012345678901234567890123456789012345

G        1  0.000  0.000  0.017  0.000  0.000  0.000  0.016  0.008
G        2  0.000  0.012  0.060  0.024  0.048  0.000  0.024  0.012
G        3  0.000  0.000  0.278  0.111  0.000  0.280  0.056  0.168
G        4  0.000  0.000  0.278  0.111  0.000  0.280  0.056  0.168
G        5  0.000  0.000  0.556  0.222  0.000  0.560  0.111  0.336
G        6  0.000  0.000  0.834  0.333  0.000  0.840  0.168  0.504
G        7  0.657  0.333  1.667  1.000  1.332  0.000  0.666  1.650
G        8  1.000  0.333  3.611  1.111  0.000  0.280  0.056  1.818
H        1  0.000  5.000  5.000  5.000  2.000  2.000
H        2  0.000 15.000 15.000 15.000 10.000  2.000
H        3 13.000 19.000 19.000 19.000 24.000 45.000
H        4 13.000 20.000 20.000 20.000 25.000 45.000
H        5 15.000 25.000 25.000 20.000 30.000 50.000
H        6 15.000 25.000 25.000 20.000 35.000 50.000
H        7 15.000 25.000 25.000 20.000 40.000 50.000
H        8 15.000 25.000 25.000 20.000 40.000 50.000
I        1 10.000  5.000 20.000 17.000 20.000 12.000
I        2 10.000 15.000 25.000 20.000 40.000  3.000
I        3 10.000 10.000 15.000 13.000 40.000  0.000
I        4 10.000 20.000 20.000 20.000 40.000  0.000
I        5 10.000 25.000 25.000 20.000 40.000  0.000
I        6 10.000 25.000 25.000 20.000 40.000  0.000
I        7  8.000  0.000 10.000 20.000 10.000  0.000
I        8 10.000 15.000 25.000 20.000 20.000  0.000
I        9 15.000 25.000  5.000  5.000 40.000 42.000
I       10 15.000 25.000 25.000 20.000 40.000 47.000
I       11 15.000 25.000 25.000 17.000 40.000 47.000
I       12 15.000 25.000 25.000 17.000 40.000 49.000
I       13 13.000 25.000 25.000 20.000 40.000 50.000
I       14  8.000  5.000  0.000 17.000 20.000  0.000
I       15 10.000 10.000 10.000  8.000 20.000  0.000
J           1.000  1.000  1.000  1.000  1.000  1.000
J           0.999  0.999  0.999  0.999  0.999  0.900
J           0.998  0.998  0.790  0.560  0.998  0.700
J           0.997  0.870  0.500  0.530  0.670  0.500
J           0.690  0.620  0.500  0.390  0.670  0.400
J           0.570  0.500  0.210  0.190  0.330  0.300
J           0.480  0.250  0.000  0.060  0.330  0.200
J           0.260  0.080  0.000  0.050  0.000  0.100
J           0.170  0.000  0.000  0.000  0.000  0.100
J           0.140  0.000  0.000  0.000  0.000  0.000
J           1.000  1.000  1.000  1.000  1.000  1.000
J           0.930  0.940  0.930  0.920  0.999  0.900
J           0.910  0.890  0.880  0.860  0.998  0.700
J           0.880  0.890  0.870  0.850  0.780  0.500
J           0.780  0.650  0.670  0.670  0.470  0.400
J           0.310  0.280  0.370  0.380  0.220  0.300
J           0.220  0.240  0.320  0.330  0.200  0.200
J           0.150  0.150  0.180  0.180  0.100  0.100
J           0.070  0.090  0.090  0.090  0.040  0.100
J           0.050  0.070  0.070  0.060  0.010  0.000
```

```
000000000111111111122222222223333333333444444444455555555556666666666777777
123456789012345678901234567890123456789012345678901234567890123456789012345


J                    1.000   1.000   1.000   1.000   1.000   1.000
J                    0.999   0.999   0.999   0.999   0.999   0.999
J                    0.998   0.890   0.998   0.998   0.998   0.998
J                    0.997   0.820   0.997   0.997   0.997   0.900
J                    0.830   0.730   0.996   0.996   0.996   0.800
J                    0.780   0.670   0.750   0.830   0.995   0.700
J                    0.450   0.670   0.500   0.670   0.994   0.600
J                    0.250   0.670   0.500   0.670   0.330   0.500
J                    0.250   0.670   0.250   0.330   0.330   0.400
J                    0.250   0.360   0.000   0.000   0.330   0.300
J                    1.000   1.000   1.000   1.000   1.000   1.000
J                    0.999   0.999   0.999   0.999   0.999   0.999
J                    0.998   0.950   0.930   0.940   0.998   0.998
J                    0.997   0.910   0.930   0.940   0.890   0.900
J                    0.790   0.900   0.400   0.430   0.530   0.800
J                    0.750   0.610   0.140   0.180   0.230   0.700
J                    0.750   0.560   0.140   0.180   0.160   0.600
J                    0.750   0.550   0.120   0.140   0.150   0.500
J                    0.750   0.510   0.070   0.060   0.130   0.400
J                    0.750   0.280   0.020   0.010   0.080   0.300
J                    1.000   1.000   1.000   1.000   1.000   1.000
J                    0.999   0.999   0.999   0.999   0.999   0.999
J                    0.998   0.998   0.998   0.998   0.998   0.998
J                    0.997   0.997   0.997   0.997   1.000   0.997
J                    0.996   0.770   0.996   0.996   0.770   0.900
J                    0.830   0.640   0.330   0.630   0.510   0.800
J                    0.710   0.580   0.110   0.260   0.480   0.700
J                    0.660   0.530   0.000   0.220   0.360   0.600
J                    0.520   0.510   0.000   0.110   0.330   0.500
J                    0.380   0.380   0.000   0.040   0.240   0.500
J                    1.000   1.000   1.000   1.000   1.000   1.000
J                    0.999   0.999   0.999   0.999   0.999   0.999
J                    0.998   0.998   0.998   0.998   0.998   0.998
J                    0.997   0.997   0.997   0.997   0.997   0.997
J                    0.996   0.996   0.996   0.996   0.996   0.996
J                    0.995   0.710   0.330   0.330   0.750   0.900
J                    0.994   0.620   0.330   0.330   0.590   0.900
J                    0.993   0.440   0.280   0.280   0.500   0.800
J                    0.992   0.290   0.170   0.170   0.500   0.700
J                    0.991   0.290   0.170   0.170   0.480   0.600
J                    1.000   1.000   1.000   1.000   1.000   1.000
J                    0.999   0.999   0.999   0.999   0.999   0.999
J                    0.998   0.998   0.998   0.998   0.998   0.998
J                    0.997   0.997   0.997   0.997   0.997   0.900
J                    0.996   0.996   0.996   0.996   0.996   0.800
J                    0.720   0.490   0.995   0.995   0.470   0.700
J                    0.670   0.360   0.994   0.994   0.360   0.600
J                    0.580   0.360   0.993   0.993   0.320   0.500
J                    0.500   0.360   0.650   0.650   0.270   0.400
J                    0.500   0.290   0.600   0.600   0.270   0.300
```

```
000000000111111111122222222223333333333444444444455555555556666666666777777
123456789012345678901234567890123456789012345678901234567890123456789012345

J           1.000  1.000  1.000  1.000  1.000  1.000
J           0.999  0.999  0.999  0.999  0.999  0.999
J           0.998  0.998  0.998  0.998  0.998  0.998
J           0.997  0.997  0.997  0.997  0.997  0.900
J           0.996  0.996  0.996  0.996  0.996  0.800
J           0.720  0.490  0.995  0.995  0.470  0.700
J           0.670  0.360  0.994  0.994  0.360  0.600
J           0.580  0.360  0.993  0.993  0.320  0.500
J           0.500  0.360  0.650  0.650  0.270  0.400
J           0.500  0.290  0.600  0.600  0.270  0.300
K        1    0    1    1    1    1    1    0    1
K        1    1    1    0    1    1    1    1    0
L      1 1US 79   MILAM .      0204-05  4.530 26.000
L      2 1US 77   MILAM        0209-05 12.320 28.000
L      3 1US 190  MILAM        0815-02  3.620 26.000
L      4 2SH OSR  MADISON      0475-04  7.000 20.000
L      5 2SH OSR  MADISON      0475-03  2.260 22.000
L      6 2FM1696  WA_KER       1809-02 13.800 20.000
L      7 2FM 1791 WALKER       1706-01 12.370 22.000
L      8 2FM2821  WA_KER       2805-01  3.340 24.000
L      9 1SH 30   WALKER       0212-02  7.390 26.000
L     10 1SH 36   BURLESON     0816-03 12.010 26.000
L     11 1US 290  WASHINGTON   0114-09  9.021 26.000
L     12 1US 79   MILAM        0204-08  5.640 26.000
L     13 1SH 36   BURLESON     0186-02  9.320 26.000
L     14 2SH OSR  BRAZOS       0475-02  6.670 20.000
      15 2FM 908  MILAM        0858-02  7.440 20.000
        1202000   1070000   1140000   1200000   1270000   1270000   1270000
        1270000   1270000   1270000
```

OUTPUT DATA OF EXAMPLE PROBLEM

3LE C.1

## REHABILITATION AND MAINTENANCE SYSTEM

## (DISTRICT TIME 3PTIMIZATION)

TEXAS TRANSPORTATION INSTITUTE
TEXAS A&M UNIVERSITY
COLLEGE STATION, TEXAS 77843

| HIGHWAY | TYPE | LENGTH | WIDTH | STRATEGY USED AT TIME PERIOD | | | | | | | | | | BENEFIT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SEGMENT | | (MILE) | (FEET) | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| 1 | 01 | 4.53 | 26.00 | 4 | 0 | 0 | 3 | 0 | 1 | 3 | 1 | 3 | 0 | 105057. |
| 2 | 01 | 12.32 | 28.00 | 5 | 0 | 0 | 1 | 3 | 3 | 1 | 3 | 1 | 0 | 293805. |
| 3 | 01 | 3.62 | 26.00 | 5 | 0 | 0 | 1 | 3 | 3 | 1 | 3 | 1 | 0 | 80869. |
| 4 | 02 | 7.00 | 20.00 | 2 | 2 | 7 | 0 | 0 | 0 | 0 | 5 | 0 | 0 | 95055. |
| 5 | 02 | 2.26 | 22.00 | 3 | 7 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 34148. |
| 6 | 02 | 13.80 | 20.00 | 1 | 1 | 1 | 1 | 1 | 3 | 2 | 3 | 1 | 0 | 154632. |
| 7 | 02 | 12.37 | 22.00 | 8 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 0 | 205312. |
| 8 | 02 | 3.34 | 24.00 | 8 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 0 | 54063. |
| 9 | 01 | 7.39 | 26.00 | 4 | 3 | 0 | 1 | 3 | 3 | 1 | 3 | 1 | 0 | 159995. |
| 10 | 01 | 12.01 | 26.00 | 1 | 1 | 1 | 4 | 0 | 0 | 0 | 5 | 0 | 0 | 285163. |
| 11 | 01 | 9.02 | 26.00 | 1 | 4 | 0 | 0 | 0 | 3 | 5 | 0 | 0 | 0 | 215346. |
| 12 | 01 | 5.64 | 26.00 | 1 | 3 | 0 | 4 | 0 | 0 | 0 | 5 | 0 | 0 | 135313. |
| 13 | 01 | 9.32 | 26.00 | 1 | 3 | 6 | 0 | 0 | 0 | 6 | 0 | 0 | 0 | 228602. |
| 14 | 02 | 6.67 | 20.00 | 8 | 0 | 0 | 0 | 0 | 6 | 0 | 0 | 0 | 0 | 99983. |
| 15 | 02 | 7.44 | 20.00 | 2 | 5 | 0 | 2 | 3 | 3 | 2 | 3 | 1 | 0 | 79664. |
| | | | | | | | | | | | | | | 2227004. |

## TABLE C.2

### PERCENTAGE UTILIZATION OF RESOURCES IN THE PLANNING PERIOD

| RESOURCE | AVAILABLE | UNIT | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | BUDGET | DOLLARS | 1202000. | 1070000. | 1140000. | 1200000. | 1270000. | 1270000. | 1270000. | 1270000. | 1270000. | 1270000. |
| | | | 98.66 | 66.91 | 98.57 | 47.93 | 17.23 | 96.69 | 94.24 | 98.62 | 3.41 | 0.00 |
| 2 | 26457. | TONS | 20.14 | 22.33 | 0.00 | 5.18 | 36.71 | 83.67 | 9.17 | 60.89 | 7.27 | 0.00 |
| 3 | 12811. | TONS | 19.30 | 15.46 | 19.01 | 14.63 | 6.38 | 23.08 | 19.73 | 21.85 | 1.26 | 0.00 |
| 4 | 244243. | TONS | 7.81 | 4.60 | 11.07 | 4.60 | 0.00 | 6.17 | 16.21 | 11.54 | 0.00 | 0.00 |
| 5 | 244243. | TONS | 0.00 | 2.88 | 8.65 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 6 | 1949. | EQ.-DAYS | 16.62 | 0.00 | 0.00 | 0.00 | 0.00 | 16.91 | 2.55 | 0.00 | 0.00 | 0.00 |
| 7 | 1949. | EQ.-DAYS | 21.03 | 3.69 | 5.90 | 3.36 | 0.63 | 20.15 | 9.07 | 6.52 | 0.12 | 0.00 |
| 8 | 947. | EQ.-DAYS | 17.14 | 0.79 | 0.00 | 0.18 | 1.30 | 19.67 | 2.85 | 2.15 | 0.25 | 0.00 |
| 9 | 2339. | EQ.-DAYS | 43.85 | 8.47 | 12.31 | 7.18 | 2.62 | 44.30 | 19.06 | 15.78 | 0.52 | 0.00 |
| 10 | 2785. | EQ.-DAYS | 20.47 | 2.81 | 4.13 | 2.36 | 0.88 | 20.79 | 7.13 | 5.29 | 0.17 | 0.00 |
| 11 | 947. | EQ.-DAYS | 17.14 | 0.79 | 0.00 | 0.18 | 1.30 | 19.67 | 2.85 | 2.15 | 0.25 | 0.00 |
| 12 | 473. | EQ.-DAYS | 8.86 | 6.74 | 12.19 | 6.65 | 0.00 | 4.39 | 12.69 | 11.38 | 0.00 | 0.00 |
| 13 | 947. | EQ.-DAYS | 21.82 | 4.28 | 6.10 | 3.66 | 1.30 | 21.86 | 9.73 | 7.84 | 0.25 | 0.00 |
| 14 | 1949. | MAN-DAYS | 16.62 | 0.00 | 0.00 | 0.00 | 0.00 | 16.91 | 2.55 | 0.00 | 0.00 | 0.00 |
| 15 | 947. | MAN-DAYS | 17.14 | 0.79 | 0.00 | 0.18 | 1.30 | 19.67 | 2.85 | 2.15 | 0.26 | 0.00 |
| 16 | 2339. | MAN-DAYS | 43.85 | 8.47 | 12.31 | 7.18 | 2.62 | 44.30 | 19.06 | 15.78 | 0.52 | 0.00 |
| 17 | 2785. | MAN-DAYS | 20.47 | 2.81 | 4.13 | 2.36 | 0.88 | 20.79 | 7.13 | 5.29 | 0.17 | 0.00 |
| 18 | 3732. | MAN-DAYS | 17.40 | 0.80 | 0.00 | 0.19 | 1.32 | 19.96 | 2.89 | 2.18 | 0.26 | 0.00 |
| 19 | 2367. | MAN-DAYS | 8.86 | 6.74 | 12.25 | 6.65 | 0.00 | 4.43 | 12.77 | 11.38 | 0.00 | 0.00 |
| 20 | 1866. | MAN-DAYS | 19.90 | 2.64 | 3.09 | 2.03 | 1.32 | 21.08 | 6.66 | 5.07 | 0.25 | 0.00 |
| 21 | 4623. | MAN-DAYS | 20.12 | 2.26 | 3.76 | 2.11 | 0.27 | 19.46 | 6.74 | 3.94 | 0.05 | 0.00 |

# APPENDIX D

# LISTING OF THE COMPUTER PROGRAM

```
C     ******************************************************************
C
C     REHABILITATION AND MAINTENANCE SYSTEM
C
C     DISTRICT TIME OPTIMIZATION - PROGRAM I
C
C     TEXAS TRANSPORTATION INSTITUTE
C     TEXAS A&M UNIVERSITY
C     COLLEGE STATION, TEXAS 77843
C
C
C     AUTHORS         : SHASHIKANT SATHAYE
C                       CHIYYARATH V. SHANMUGHAM
C
C     INSTALLATION    : AMDAHL 470V/6
C                       DATA PROCESSING CENTER
C                       TEXAS A&M UNIVERSITY
C
C
C     LANGUAGE        : FORTRAN IV
C
C     DATE WRITTEN    : SEPTEMBER 1980
C
C
C     ******************************************************************
      CALL NETINP
      CALL RESINP
      CALL EFGRAD
      CALL FCNSRC
      CALL RESULT
      STOP
      END




      SUBROUTINE  NETINP
C
C     SUBROUTINE TO INPUT NETWORK INFORMATION
C
C     READ IN RESOURCE AVAILABILITY, RESOURCE REQUIREMENTS,
C     GAIN OF RATING MATRIX, CURRENT RATING OF HIGHWAY SEGMENTS
C
      REAL        L1, L2
      COMMON /A1/ NH, NS, NT, NDIST, NRES, NTYP, KPATH,
     1            L1(20), L2(20), ITYP(20)
      COMMON /A2/ B(25,10), R(15), RTOL(15,10), RIMP(15,25),
     1            IXF(15), RMIN(15,10), RDET(15,15,10)
      COMMON /A3/ NM, NE, MN, X
      COMMON /B5/ BRES(25,10), A(20,25,10)
      DIMENSION   RC(20,6), IY(2,10)
  500 FORMAT ( 5X, 15I5 )
  510 FORMAT ( 25X, F15.3 )
  520 FORMAT ( 10X, 10F7.3 )
  530 FORMAT ( 8X, I2, 28X, 2F7.3, 14X, 2F7.3 )
```

```
      READ  (1,500) NH, NS, NDIST, NT, NM, NE, MN, NTYP, KPATH
C     WRITE (6,500) NH, NS, NDIST, NT, NM, NE, MN, NTYP, KPATH
      NS          = NS + 1
      NRES=NM+NE+MN+1
      DO 2100 J = 2, NS
      READ  (1,510) A(J,1,1)
C     WRITE (6,510) A(J,1,1)
 2100 CONTINUE
      DO 2200 K = 1, NDIST
      READ  (1,510)
C     WRITE (6,510)
 2200 CONTINUE
      DO 2300 J = 2, NRES
      READ  (1,510) BRES(J,1)
C     WRITE (6,510) BRES(J,1)
 2300 CONTINUE
      NR          = NM + 1
      DO 2400 J = 2, NS
      READ  (1,520) ( A(J,K,1), K = 2, NR )
C     WRITE (6,520) ( A(J,K,1), K = 2, NR )
 2400 CONTINUE
      NK          = NR + 1
      NR          = NR + NE
      DO 2500 J = 2, NS
      READ  (1,520) ( A(J,K,1), K = NK, NR )
C     WRITE (6,520) ( A(J,K,1), K = NK, NR )
 2500 CONTINUE
      NK          = NR + 1
      DO 2600 J = 2, NS
      READ  (1,520) ( A(J,K,1), K = NK, NRES )
C     WRITE (6,520) ( A(J,K,1), K = NK, NRES )
 2600 CONTINUE
      DO 2700 J = 2, NS
      READ  (1,520) ( RIMP(J,K), K = 1, NDIST )
C     WRITE (6,520) ( RIMP(J,K), K = 1, NDIST )
 2700 CONTINUE
      DO 2800 I = 1, NH
      READ  (1,520) ( RC(I,K), K = 1, NDIST )
C     WRITE (6,520) ( RC(I,K), K = 1, NDIST )
 2800 CONTINUE
      DO 2900 J = 2, NS
      DO 2900 L = 1, NT
      READ  (1,520) ( RDET(J,K,L), K = 1, NDIST )
C     WRITE (6,520) ( RDET(J,K,L), K = 1, NDIST )
 2900 CONTINUE
      DO 3000 I = 1, NTYP
      READ  (1,500) ( IY(I,J), J = 1, NS )
C     WRITE (6,500) ( IY(I,J), J = 1, NS )
 3000 CONTINUE
      DO 20 J=2,NS
      DO 20 K=1,NDIST
      DO 25 L=1,NT
   25 RDET(J,K,L)=RIMP(NS,K)*RDET(J,K,L)
   20 CONTINUE
      DO 30 K=1,NT
      DO 30 J=1,NDIST
      RMIN(J,K)=0.4*RIMP(NS,J)
   30 RTOL(J,K)=0.8*RIMP(NS,J)
      X=0
      DO 3500 I = 1, NH
      READ  (1,530) IT, L1(I), L2(I), TRAF, ENVR
      WRITE (6,530) IT, L1(I), L2(I), TRAF, ENVR
      ITYP(I)    = IT
      NDIST      = 6
      IF ( IT. EQ. 2 ) NDIST = 5
      DO 3200 J = 1, NS
      IXF(J)     = IY(IT,J)
 3200 CONTINUE
      X          = X + L1(I) * L2(I)
      DO 3300 K = 1, NDIST
      R(K)       = RC(I,K)
 3300 CONTINUE
C
C     GENERATE A FEASIBLE NETWORK FOR HIGHWAY SEGMENT I
C
      CALL NETGEN (I)
 3500 CONTINUE
      RETURN
      END




      SUBROUTINE  RESINP
C
C     READ IN BUDGET AVAILABILITY AND SET UP THE RESOURCE
C     REQUIREMENT AND AVAILABLITY MATRICES FOR ALL YEARS
C
      REAL          L1, L2
      COMMON /A1/ NH, NS, NT, NDIST, NRES, NTYP, KPATH,
     1              L1(20), L2(20), ITYP(20)
      COMMON /A3/ NM, NE, MN, X
      COMMON /B5/ BRES(25,10), A(20,25,10)
  500 FORMAT ( 5X, 7F10.0 )
      READ  (1,500) ( BRES(1,L), L = 1, NT )
C     WRITE (6,500) ( BRES(1,L), L = 1, NT )
      DO 135 J=2,NRES
      BRES(J,1)=X*BRES(J,1)
      DO 135 K=1,NT
  135 BRES(J,K)=BRES(J,1)
      DO 150 I = 2, NS
      DO 145 J=1,NRES
      DO 145 K=2,NT
  145 A(I,J,K)=1.07*A(I,J,K-1)
  150 CONTINUE
      DO 160 J=1,NRES
      DO 160 K=1,NT
  160 A(1,J,K)=0.
      RETURN
      END
      SUBROUTINE  FEASBL (ITIME.INODE)
C
C     GENERATE A SET OF FEASIBLE ALTERNATIVES AT EACH VERTEX OF_
C     THE NETWORK FOR HIGHWAY SEGMENT I
C
      REAL         L1. L2
      DIMENSION    RR(20). NTIME(20)
      COMMON /A1/  NH. NS. NT. NDIST. NRES. NTYP. KPATH.
     1             L1(20). L2(20). ITYP(20)
      COMMON /A2/  B(25.10). R(15). RTOL(15.10). RIMP(15.25).
     1             IXF(15). RMIN(15.10). RDET(15.15.10)
      COMMON /B3/  RATNOD(2500.10). IX(25). NLIF(25).
     1             RATIMP(25,50). NSTAT(2500)
      INDEX=0
      NSTA=NSTAT(INODE)
      DO 4 K=1.NDIST
      RTTL=RTOL(K.ITIME)
      IF(RATNOD(INODE.K).LT.RTTL) GO TO 4
      INDEX=INDEX+1
    4 CONTINUE
      DO 6 I=1.NS
      B(I.ITIME)=0.
  6   IX(I)=0
      IX(1)=1
      IF(INDEX.GE.NDIST) GO TO 200
      IX(1)=0
      DO 21 J=1.NDIST
      NTIME(J)=NT
      DO 20 K=1.NT
      IK=K
      IF(RATNOD(INODE.J).GT.RDET(NSTA.J.K)) GO TO 22
  20  CONTINUE
  22  IF(NTIME(J).GT.K)NTIME(J)=K
  21  CONTINUE
      IK=NT
  25  DO 30 K=1.NDIST
      IF(RDET(NSTA.K.NTIME(K)).LT.RMIN(K.ITIME)) GO TO 8
      IF(RATNOD(INODE.K).LT.RMIN(K.ITIME)) GOTO 8
      RATIMP(1.K)=RDET(NSTA.K.NTIME(K))-RATNOD(INODE.K)
      IF(RDET(NSTA.K.IK).GT.RATNOD(INODE.K)) RATIMP(1.K)=0.
      B(1.ITIME)=B(1.ITIME)+RATNOD(INODE.K)-RMIN(K.ITIME)
  30  CONTINUE
      IX(1)=1
      NLIF(1)=1+ITIME
    8 DO 100 I=2.NS
      IF(IXF(I).EQ.0) GO TO 70
      NLIF(I)=NT
      IX(I)=1
      NTT=ITIME+1
      INDEX=0
      DO 5 K=1.NDIST
      RMAX=1.6*RIMP(NS,K)
      RR(K)=RATNOD(INODE,K)+RIMP(I,K)
      IF(RR(K).LT.RMIN(K,NTT)) GO TO 70
      IF(RR(K).GE.RMAX) INDEX=INDEX+1
    5 CONTINUE
      IF(INDEX.GE.NDIST) GO TO 70
      DO 300 K=1,NDIST
      NTIME(K)=NT
      DO 250 JT=1,NT
      IF(RR(K).GT.RDET(I,K,JT))GO TO 255
  250 CONTINUE
  255 IF(NTIME(K).GT.JT)NTIME(K)=JT
  300 CONTINUE
      DO 60 IK=1,NDIST
      IF(RR(IK).GT.RIMP(NS,IK))RR(IK)=RIMP(NS,IK)
      RRM=RR(IK)
      IF(NTIME(IK).NE.1) NTIME(IK)=NTIME(IK)-1
      DO 40 J=NTT,NT
      NLIFF=J
      IJ=J-NTT+NTIME(IK)
      IF(IJ.GE.NT) GO TO 60
      RRM=RDET(I,IK,IJ+1)
      IF(RRM.LT.RTOL(IK,J)) GO TO 45
   40 CONTINUE
      GO TO 60
   45 IF(NLIF(I).GT.NLIFF) NLIF(I)=NLIFF
   60 CONTINUE
      NLIFF=NLIF(I)-ITIME
      DO 80 J=1,NDIST
      B(I,ITIME)=B(I,ITIME)+1./2.*(RR(J)+RDET(I,J,NTIME(J)))-RMIN(J,NTT)
      IF(NLIFF.EQ.1) GO TO 85
      NLIF1=NLIF(I)-1
      DO 90 IT=NTT,NLIF1
      JT=IT-NTT+1+NTIME(J)
   90 B(I,ITIME)=B(I,ITIME)+1./2.*(RDET(I,J,JT-1)+RDET(I,J,JT))-
     1RMIN(J,IT+1)
   85 IF(RATNOD(INODE,J).LT.RMIN(J,NTT))B(I,ITIME)=
     1B(I,ITIME)+NLIFF*(RMIN(J,NTT)-RATNOD(INODE,J))
   80 CONTINUE
      DO 75 K=1,NDIST
      IJ=NLIF(I)-ITIME+NTIME(K)
      IF(IJ.GT.NT) IJ=NT
   75 RATIMP(I,K)=RDET(I,K,IJ)-RATNOD(INODE,K)
      GO TO 100
   70 IX(I)=0
  100 CONTINUE
      RETURN
  200 DO 10 J=1,NDIST
      DO 15 K=1,NT
      IK=K
      IF(RATNOD(INODE,J).GT.RDET(NSTA,J,K))GO TO 210
   15 CONTINUE
      IK=NT
  210 RATIMP(1,J)=RDET(NSTA,J,IK)-RATNOD(INODE,J)
      B(1,ITIME)=B(1,ITIME)+1./2.*(RATNOD(INODE,J)+RDET(NSTA,J,IK))-
     1RMIN(J,ITIME)
   10 CONTINUE
      NLIF(1)=ITIME+1
      RETURN
      END




      SUBROUTINE  NETGEN (NHIGH)
C
C     GENERATE THE NETWORK FOR HIGHWAY SEGMENT NHIGH
C
C     EACH PATH FROM SOURCE TO SINK (NODE 1 TO NODE M)
C     REPRESENTS A FEASIBLE SET OF VALUES WITH RESPECT
C     TO THE CONSTRAINT MATRIX
C
      REAL          L1, L2
      INTEGER*2     STNODE, ENNODE
      COMMON /A1/ NH, NS, NT, NDIST, NRES, NTYP, KPATH,
     1            L1(20), L2(20), ITYP(20)
      COMMON /A2/ B(25,10), R(15), RTOL(15,10), RIMP(15,25),
     1            IXF(15), RMIN(15,10), RDET(15,15,10)
      COMMON /B3/ RATNOD(2500,10), IX(25), NLIF(25),
     1            RATIMP(25,50), NSTAT(2500)
      COMMON /C1/ RSREQ(20,15,10), RESREQ(3000), CO(20,15)
      COMMON /C2/ STNODE(3000), ENNODE(3000), BENARC(3000), NODE(2500)
      IARC=0
      INODE=0
      JNODE=1
      NSTAT(JNODE)=NS
      DO 1 I=1,NDIST
    1 RATNOD(1,I)=R(I)
      NODE(1)=1
    2 INODE=INODE+1
      ITIME=NODE(INODE)
      IF(IARC.GT.2995) GO TO 260
      IF(ITIME.NE.NT) GO TO 3
      IARC=IARC+1
      STNODE(IARC)=INODE
      ENNODE(IARC)=-1
      BENARC(IARC)=1
      RESREQ(IARC)=0
      GO TO 101
    3 CALL FEASBL(ITIME,INODE)
      NCOUNT=0
      DO 100 I=1,NS
      IF(IX(I).EQ.0) GO TO 100
      JNODE=JNODE+1
      IARC=IARC+1
      STNODE(IARC)=INODE
      ENNODE(IARC)=JNODE
      NSTAT(JNODE)=I
      IF(NSTAT(JNODE).EQ.1) NSTAT(JNODE)=NSTAT(INODE)
      NODE(JNODE)=NLIF(I)
      IF(JNODE.EQ.2000) GO TO 260
      IF(NODE(JNODE).LT.NT) GO TO 60
      NCOUNT=NCOUNT+1
      IF(NCOUNT.GT.1) GO TO 60
      ENNODE(IARC)=-1
      JNODE=JNODE-1
   60 BENARC(IARC)=B(I,NODE(INODE))
      RESREQ(IARC)=I
      DO 11 J=1.NDIST
      RATNOD(JNODE.J)=RATNOD(INODE.J)+RATIMP(I.J)
      IF(RATNOD(JNODE.J).GE.100.) RATNOD(JNODE.J)=100.
   11 CONTINUE
  100 CONTINUE
  101 IF(JNODE.GT.INODE) GO TO 2
  260 IARC=IARC+1
      STNODE(IARC)=JNODE+1
      NODE(JNODE+1)=NT
      RESREQ(IARC)=0
      ENNODE(IARC)=JNODE
      BENARC(IARC)=-1
      CALL KSHPTH(IARC.NHIGH.KPATH.JNODE)
      RETURN
  300 FORMAT(' '.5X,30I4)
      RETURN
      END
```

```
      SUBROUTINE  KSHPTH (IARC.NHIGH.KPATH.JNODE)
C
C     USES SHIER'S ALGORITHM TO DETERMIN THE K-SHORTEST
C     PATHS THROUGH THE NETWORK FOR HIGHWAY SEGMENT NHIGH
C
      INTEGER      LLEN(2500). LINC(3000). LVAL(3000). START
      INTEGER      ULEN(2500). UINC(3000). UVAL(3000). VAL
      INTEGER*2    STNODE. ENNODE
      COMMON /B1/ N. MU. ML. LLEN. LINC. LVAL. ULEN. UINC. UVAL
      COMMON /B2/ INF. K
      COMMON /B4/ START(2501). INC(5000). VAL(5000)
      COMMON /C2/ STNODE(3000). ENNODE(3000). BENARC(3000). NODE(2500)
      DO 500 NPN=1,2
      INF=99999999
      J=0
      MU=0
      ML=0
      NPREV=0
      N=0
      IR=0
    1 IR=IR+1
      IF(IR.GT.IARC) GO TO 30
      NA=STNODE(IR)
      IF(ENNODE(IR).EQ.-1)ENNODE(IR)=JNODE+1
      NB=ENNODE(IR)
      LEN=IFIX(-BENARC(IR)*100.)
      IF(NPN.EQ.2) LEN=-LEN
      IF(NA.GT.N) N=NA
      IF(NB.GT.N) N=NB
      IF(NA.EQ.NPREV)  GO TO 10
      IF(NA.EQ.NPREV+1) GO TO 3
      IF(NA.EQ.0) GO TO 30
      L1=NPREV+1
      L2=NA-1
      DO 2 L=L1.L2
      START(L)=0
      ULEN(L)=0
    2 LLEN(L)=0
    3 IF(J.EQ.0) GO TO 5
      ULEN(NPREV)=JU
      LLEN(NPREV)=JL
    5 START(NA)=J+1
      JU=0
      JL=0
      NPREV=NA
   10 J=J+1
      INC(J)=NB
      VAL(J)=LEN
      IF(NB.GT.NA) GO TO 20
      MU=MU+1
      UINC(MU)=NB
      UVAL(MU)=LEN
      JU=JU+1
      GO TO 1
   20 ML=ML+1
      LINC(ML)=NB
      LVAL(ML)=LEN
      JL=JL+1
      GO TO 1
   30 START(NPREV+1)=J+1
      ULEN(NPREV)=JU
      LLEN(NPREV)=JL
   40 DO 300 I=1.2
      IF( I.EQ.2) GO TO 101
      K=KPATH
      NS=JNODE+1
      I3=100
      CALL DSWP(NS.I3)
      GO TO 300
  101 I1=1
      I2=KPATH
      IF(NPN.EQ.1) I2=I2-1
   50 CALL TRACE(NS.I1.I2.IARC.NHIGH.NPN)
  300 CONTINUE
  500 CONTINUE
  100 RETURN
      END
      SUBROUTINE  DSWP (NS,IMAX)
      INTEGER     LLEN(2500), LINC(3000), LVAL(3000)
      INTEGER     ULEN(2500), UINC(3000), UVAL(3000), X
      COMMON /B1/ N, MU, ML, LLEN, LINC, LVAL, ULEN, UINC, UVAL
      COMMON /B2/ INF, K
      COMMON /B3/ X(2500,15)
      N1=N-1
      DO 20 I=1,N
      DO 20 J=1,K
   20 X(I,J)=INF
      X(NS,1)=0
      ITNS=1
   30 IFIN=ML
      INDX=1
      DO 40 III=1,N1
      I=-III+N1+1
      IF(LLEN(I).EQ.0) GO TO 40
      IS=IFIN-LLEN(I)+1
      CALL XMULT(I,IS,IFIN,LINC,LVAL,INDX)
      IFIN=IS-1
   40 CONTINUE
      IF(ITNS.EQ.1) GO TO 50
      IF(INDX.EQ.1) GO TO 100
   50 ITNS=ITNS+1
      IS=1
      INDX=1
      DO 60 I=2,N
      IF(ULEN(I).EQ.0) GO TO 60
      IFIN=IS+ULEN(I)-1
      CALL XMULT(I,IS,IFIN,UINC,UVAL,INDX)
      IS=IFIN+1
   60 CONTINUE
      IF(INDX.EQ.1) GO TO 100
      ITNS=ITNS+1
      IF(ITNS.LT.IMAX) GO TO 30
      WRITE(6,900) IMAX
  900 FORMAT('NUMBER OF ITERATIONS EXCEEDS',I5)
      GO TO 200
  100 CONTINUE
  200 RETURN
      END
```

```
      SUBROUTINE  XMULT (I,IS,IFIN,INC,VAL,INDX)
      INTEGER     INC(3000), VAL(3000), A(15), X
      COMMON /B2/ INF, K
      COMMON /B3/ X(2500,15)
      DO 10 J=1,K
   10 A(J)=X(I,J)
      MAX=A(K)
      DO 100  L=IS,IFIN
      II=INC(L)
      IV=VAL(L)
      DO 90 M=1,K
      IX=X(II,M)
      IF(IX.GE.INF) GO TO 100
      IXV=IX+IV
      IF(IXV.GE.MAX) GO TO 100
      DO 30 JJJ=2,K
      J=-JJJ+K+2
      IF(IXV-A(J-1)) 30,90,50
   30 CONTINUE
      J=1
   50 JJ=K
   70 IF(JJ.LE.J) GO TO 80
      A(JJ)=A(JJ-1)
      JJ=JJ-1
      GO TO 70
   80 A(J)=IXV
      INDX=0
      MAX=A(K)
   90 CONTINUE
  100 CONTINUE
      IF(INDX.EQ.1) GO TO 120
      DO 110 J=1,K
  110 X(I,J)=A(J)
  120 RETURN
      END




      SUBROUTINE  TRACE (NS,NF,PMAX,IARC,NHIGH,NPN)
      INTEGER     P(3000), Q(3000), PV(3000), START, VAL, X, PMAX
      INTEGER*2   STNODE, ENNODE
      COMMON /B2/ INF, K
      COMMON /B3/ X(2500,15)
      COMMON /B4/ START(2501), INC(5000), VAL(5000)
      COMMON /C1/ RSREQ(20,15,10), RESREQ(3000), CO(20,15)
      COMMON /C2/ STNODE(3000), ENNODE(3000), BENARC(3000), NODE(2500)
      DO 10 I=1,3000
      P(I)=0
      Q(I)=0
   10 PV(I)=0
      JJ=1
      IF(NS.EQ.NF)JJ=2
      NP=0
      LAB=X(NF,JJ)
      IF(LAB.LT.INF) GO TO 15
      WRITE (6,909) NS,NF
  909 FORMAT(1H1,'THERE ARE NO PATHS FROM NODE',I4,'TO NODE',I4)
      GO TO 200
   15 CONTINUE
   20 KK=1
      LAB=X(NF,JJ)
      IF(LAB.EQ.INF) GO TO 200
      LL=LAB
      P(1)=NF
  30 LAST=0
  40 NT=P(KK)
     IS=START(NT)
     DO 45 ND=NT,2500
     IF(START(ND+1).NE.0) GO TO 48
  45 CONTINUE
  48 IF=START(ND+1)-1
     II=IS+LAST
  50 IF(II.GT.IF) GO TO 90
     NI=INC(II)
     NV=VAL(II)
     LT=LAB-NV
     DO 60 J=1,K
     IF(X(NI,J)-LT)60,80,70
  60 CONTINUE
  70 II=II+1
     GO TO 50
  80 KK=KK+1
     IF(KK.GT.50) GO TO 190
     P(KK)=NI
     Q(KK)=II-IS+1
     PV(KK)=NV
     LAB=LT
     IF(LAB.NE.0) GO TO 30
     IF(NI.NE.NS) GO TO 30
     NP=NP+1
     IF(NPN.EQ.2) NP=PMAX
     CO(NHIGH,NP)=0
     DO 150 J=2,KK
     IT=NODE(P(J-1))
     DO 140 I=1,IARC
     IF(STNODE(I).NE.P(J-1).OR.ENNODE(I).NE.P(J)) GO TO 140
     CO(NHIGH,NP)=CO(NHIGH,NP)+BENARC(I)
     RSREQ(NHIGH,NP,IT)=RESREQ(I)
 140 CONTINUE
 150 CONTINUE
 500 CONTINUE
 902 FORMAT(1X,I4,I8,5X,(20I5))
 990 FORMAT(1X,2I4,15F8.3)
555  CONTINUE
     IF(NP.GE.PMAX) GO TO 2J0
  90 LAST=Q(KK)
     P(KK)=0
     LAB=LAB+PV(KK)
     KK=KK-1
     IF(KK.GT.0) GO TO 40
     JJ=JJ+1
     IF(JJ.GT.K) GO TO 200
     GO TO 20
 190 WRITE(6,903)
 903 FORMAT(1H0,'NUMBER OF ARCS IN PATH EXCEEDS 50')
 200 IF(NP.GE.PMAX) GO TO 210
     IF(NP.EQ.0) GO TO 240
     DO 220 J=NP,PMAX
     CO(NHIGH,J)=CO(NHIGH,NP)
```

```
      DO 230 IT=1,NT
  230 RSREQ(NHIGH,J,IT)=RSREQ(NHIGH,NP,IT)
  220 CONTINUE
  210 RETURN
  240 WRITE(6,995) NHIGH
  995 FORMAT('0',15X,'THERE ARE NO FEASIBLE MAINTENANCE POLICIES FOR',
     1' HIGHWAY SEGMENT NUMBER',I5)
      XXLL=-1.
      XXLL=ALOG(XXLL)
      RETURN
      END
```

```
      SUBROUTINE  MINGRD (RX,NH,IJ)
      DIMENSION   RX(NH)
      TIN=RX(1)
      IJ=1
      DO 100 I=1,NH
      IF(RX(I).GE.0.) GOTO 100
      IF(RX(I).GT.TIN) GO TO 100
      TIN=RX(I)
      IJ=I
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  EFGRAD
C
C     GENERATES THE INITIAL SOLUTION
C
      REAL        L1, L2
      COMMON /A1/ NH, NS, NT, NDIST, NRES, NTYP, KPATH,
     1            L1(20), L2(20), ITYP(20)
      COMMON /B5/ BRES(25,10), A(20,25,10)
      COMMON /C1/ RSREQ(20,15,10), RESREQ(3000), CO(20,15)
      COMMON /D1/ X(20), INF(20), JFLAG(20), RX(15), XL(20),
     1            JPAT(20,15), SRES(50,10)
      XSUM=0.
      INP=0
      DO 1 I=1,NH
      XL(I)=L1(I)*L2(I)
      X(I)=1
      INF(I)=0
    1 CONTINUE
      DO 100 I=1,NH
      DO 90 J=1,KPATH
      CO(I,J)=CO(I,J)*XL(I)
      JPAT(I,J)=J
      DO 80 K=1,NT
      IF(RSREQ(I,J,K).EQ.0) GO TO 80
      IR=RSREQ(I.J.K)
      DO 40 L=1.NRES
   40 RX(J)=RX(J)+A(IR.L.K)*XL(I)
   80 CONTINUE
      IF(RX(J).LE.0) GO TO 90
      RX(J)=CO(I.J)/RX(J)
   90 CONTINUE
      CALL SORT1 (RX.KPATH.JPAT.I. NH)
      JFLAG(I)=1
  100 CONTINUE
  105 INDEX=0
      DO 110 J=1.NRES
      DO 111 K=1.NT
      SUM=0.
      SRES(J.K)=-BRES(J.K)
      DO 112 NI=1.NH
      IF(RSREQ(NI.JPAT(NI.JFLAG(NI)).K).EQ.0) GO TO112
      I=RSREQ(NI.JPAT(NI.JFLAG(NI)).K)
      SUM=SUM+A(I.J.K)*XL(NI)
  112 CONTINUE
      SRES(J.K)=SUM-BRES(J.K)
      IF(SUM.LT.BRES(J.K)) GO TO 111
      INDEX=1
  111 CONTINUE
  110 CONTINUE
  115 IF(INDEX.EQ.0) GO TO 300
      IF(INP.GE.NH) GO TO 300
      DO 200 NI=1.NH
      RX(NI)=299.
      IF(INF(NI).LT.0) GO TO 200
      RX(NI)=0
      RR=0.
      SRR=0.
      DO 120 J=1.NRES
      DO 120 K=1.NT
      IF(SRES(J.K).LE.0.) GO TO 120
      IF(RSREQ(NI.JPAT(NI.JFLAG(NI)).K).EQ.0) GO TO 120
      I=RSREQ(NI.JPAT(NI.JFLAG(NI)).K)
      RR=RR+SRES(J.K)*A(I.J.K)*XL(NI)
      SRR=SRR+SRES(J.K)*SRES(J.K)
  120 CONTINUE
      RX(NI)=-RR/CO(NI.JPAT(NI.JFLAG(NI)))
  200 CONTINUE
      CALL MINGRD(RX.NH.IMIN)
      IF(RX(IMIN).GE.0.) GO TO 400
      IJ=JFLAG(IMIN)
      INDEX=0
      JFLAG(IMIN)=IJ+1
      IF(JFLAG(IMIN).LT.KPATH) GO TO 202
      INF(IMIN)=-1
      WRITE(6,3334)IMIN.JFLAG(IMIN).JPAT(IMIN.JFLAG(IMIN))
 3334 FORMAT(' '.'IMIN'.I5.' JFLAG=',I5.'JPAT=',I5)
      INP=INP+1
 202  DO 220 J=1.NT
      SRK=-9999999.
      IF(RSREQ(IMIN,JPAT(IMIN,JFLAG(IMIN)),J).EQ.0) GO TO 210
      IR=RSREQ(IMIN,JPAT(IMIN,JFLAG(IMIN)),J)
      DO 205 I=1,NRES
      SRES(I,J)=SRES(I,J)+A(IR,I,J)*XL(IMIN)
      IF(SRES(I,J).GT.SRK) SRK=SRES(I,J)
  205 CONTINUE
  210 IF(RSREQ(IMIN,JPAT(IMIN,IJ),J).EQ.0) GO TO 217
      IR=RSREQ(IMIN,JPAT(IMIN,IJ),J)
      DO 215 I=1,NRES
      SRES(I,J)=SRES(I,J)-A(IR,I,J)*XL(IMIN)
      IF(SRES(I,J).GT.0.) INDEX=1
  215 CONTINUE
      GO TO 220
  217 IF (SRK.GT.0.) INDEX=1
  220 CONTINUE
      IF(INDEX.GT.0) GO TO 115
      DO 230 I=1,NRES
      DO 230 J=1,NT
      IF(SRES(I,J).GT.0.) INDEX=1
  230 CONTINUE
      GO TO 115
  400 WRITE(6,450)
  450 FORMAT('0',5X,'THE PROBLEM HAS NO FEASIBLE SOLUTION')
  300 CONTINUE
      RETURN
      END




      SUBROUTINE  SORT1 (RX,KP,JP,I,NH)
      DIMENSION   JP(20,15), RX(15)
      DO 100 IIJ=1,KP
      DO 100 IYJ=IIJ,KP
      IF(RX(IIJ).GT.RX(IYJ)) GO TO 100
      TEMP=RX(IYJ)
      RX(IYJ)=RX(IIJ)
      RX(IIJ)=TEMP
      TEMP=JP(I,IYJ)
      JP(I,IYJ)=JP(I,IIJ)
      JP(I,IIJ)=TEMP
  100 CONTINUE
      RETURN
      END




      SUBROUTINE  FCNSRC
C
C     DETERMINES THE OPTIMAL OR NEAR-OPTIMAL SOLUTION
C     TO THE PROBLEM
C
      DIMENSION   KFLAG(20), LFLAG(20)
      REAL        L1, L2
      COMMON /A1/ NH, NS, NT, NDIST, NRES, NTYP, KPATH,
     1            L1(20), L2(20), ITYP(20)
      COMMON /B5/ BRES(25,10), A(20,25,10)
      COMMON /C1/ RSREQ(20,15,10), RESREQ(3000), CO(20,15)
      COMMON /D1/ X(20), INF(20), JFLAG(20), RX(15), XL(20),
     1            JPAT(20,15), SRES(50,10)
      DO 10 NN=1,NH
      JFLAG(NN)=JPAT(NN,JFLAG(NN))
      KFLAG(NN)=0
   10 CONTINUE
      IM=0
      XXSUM=0.
    1 DO 20 NN=1,NH
      LFLAG(NN)=0
      DO 30 I=1,KPATH
      IF(I.GE.JFLAG(NN)) GO TO 20
      DO 40 K=1,NT
      IR1=RSREQ(NN,JFLAG(NN),K)
      IR2=RSREQ(NN,I,K)
      IF(IR2.LE.0) GO TO 40
      IF(IR1.LE.0) GO TO60
      DO 50 L=1,NRES
      SSRES=SRES(L,K)+(A(IR2,L,K)-A(IR1,L,K))*XL(NN)
      IF(SSRES.GT.0) GO TO 30
   50 CONTINUE
      GO TO 40
   60 DO 70 L=1,NRES
      SSRES=SRES(L,K)+(A(IR2,L,K))*XL(NN)
      IF(SSRES.GT.0.) GO TO 30
   70 CONTINUE
   40 CONTINUE
      LFLAG(NN)=I
      IM=1
      GO TO 20
   30 CONTINUE
   20 CONTINUE
      IF(IM.EQ.0) GOTO 300
      XMAX=0.
      DO 100 I=1,NH
      IF(LFLAG(I).LE.0) GO TO 100
      XMAXI=CO(I,LFLAG(I))-CO(I,JFLAG(I))
      IF(XMAXI.LT.XMAX) GO TO 100
      IM=I
      XMAX=XMAXI
  100 CONTINUE
      KFLAG(IM)=-1
      DO 110 K=1,NT
      IR1=RSREQ(IM,JFLAG(IM),K)
      IR2=RSREQ(IM,LFLAG(IM),K)
      IF(IR1.LE.0) IR1=1
      IF(IR2.LE.0) IR2=1
      DO 120 L=1,NRES
  120 SRES(L,K)=SRES(L,K)+(A(IR2,L,K)-A(IR1,L,K))*XL(IM)
  110 CONTINUE
      JFLAG(IM)=LFLAG(IM)
      IM=0
      GO TO 1
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  RESULT
C
C     PRINTS OUT THE FINAL RESULTS TO THE PROBLEM
C
      REAL          L1.L2
      COMMON /A1/ NH. NS. NT. NDIST. NRES. NTYP. KPATH.
     1            L1(20). L2(20). ITYP(20)
      COMMON /A3/ NM. NE. NMN. XX
      COMMON /B5/ BRES(25.10). A(20.25.10)
      COMMON /C1/ RSREQ(20.15.10). RESREQ(3000). CO(20.15)
      COMMON /D1/ X(20). INF(20). JFLAG(20 ). RX(15). XL(20).
     1            JPAT(20.15). SRES(50.10)
  600 FORMAT ( 1H1. // )
  601 FORMAT ( / )
  610 FORMAT ( 47X. 37HREHABILITATION AND MAINTENANCE SYSTEM. //.
     1         52X. 28H(DISTRICT TIME OPTIMIZATION). ///.
     2         51X. 30HTEXAS TRANSPORTATION INSTITUTE. /.
     3         56X. 20HTEXAS A&M UNIVERSITY. /.
     4         52X. 28HCOLLEGE STATION. TEXAS 77843. /// )
  620 FORMAT ( 30X. 28HHIGHWAY  TYPE  LENGTH  WIDTH.
     1          5X. 28HSTRATEGY USED AT TIME PERIOD. 5X. 7HBENEFIT. /.
     2         30X. 7HSEGMENT. 8X. 13H(MILE) (FEET). 3X. 10I3. / )
  630 FORMAT ( 47X. 35HPERCENTAGE UTILIZATION OF RESOURCES. /.
     1         54X. 22HIN THE PLANNING PERIOD. //.
     2         05X. 27HRESOURCE  AVAILABLE  UNIT  . 10I9. / )
  650 FORMAT ( 30X. I4. 6X. 1H0. I1. 2X. 2F7.2. 3X. 10I3. F12.0 )
  660 FORMAT ( //. 91X. F12.0 )
  670 FORMAT ( 9X. 1H1. 5X. 20HBUDGET     DOLLARS  . 10F9.0 )
  671 FORMAT ( 8X. I2. 5X. F9.0. 2X. 9HTONS     . 10F9.2 )
  672 FORMAT ( 8X. I2. 5X. F9.0. 2X. 9HEQ.-DAYS . 10F9.2 )
  673 FORMAT ( 8X. I2. 5X. F9.0. 2X. 9HMAN-DAYS . 10F9.2 )
  674 FORMAT ( 35X. 10F9.2 )
      WRITE (6.600)
      WRITE (6.610)
      WRITE (6.620) ( L. L = 1. NT )
      XSUM      = 0.0
      DO 2000 I = 1. NH
      J         = JFLAG(I)
      DO 1800 K = 1. NT
      INF(K)    = RSREQ(I.J.K)
 1800 CONTINUE
      WRITE (6.650) I.ITYP(I).L1(I).L2(I).(INF(K).K=1.NT).CO(I.J)
      XSUM      = XSUM + CO(I.J)
 2000 CONTINUE
      WRITE (6.660) XSUM
      WRITE (6.600)
      WRITE (6,630) ( L, L = 1, NT )
      WRITE (6,601)
      WRITE (6,670) ( BRES(1,L), L = 1, NT )
      DO 2200 L = 1, NT
      XX          = BRES(1,L)
      X(L)        = 100.0 * ( XX + SRES(1,L) ) / XX  + 0.00001
 2200 CONTINUE
      WRITE (6,674) ( X(L), L = 1, NT )
      WRITE (6,601)
      NK          = 2
      NR          = NM + 1
      DO 2400 K = NK, NR
      XX          = BRES(K,1)
      DO 2300 L = 1, NT
      X(L)        = 100.0 * ( XX + SRES(K,L) ) / XX + 0.00001
 2300 CONTINUE
      WRITE (6,671) K, XX, ( X(L), L = 1, NT )
 2400 CONTINUE
      WRITE (6,601)
      NK          = NR + 1
      NR          = NR + NE
      DO 2600 K = NK, NR
      XX          = BRES(K,1)
      DO 2500 L = 1, NT
      X(L)        = 100.0 * ( XX + SRES(K,L) ) / XX + 0.00001
 2500 CONTINUE
      WRITE (6,672) K, XX, ( X(L), L = 1, NT )
 2600 CONTINUE
      WRITE (6,601)
      NK          = NR + 1
      DO 2800 K = NK, NRES
      XX          = BRES(K,1)
      DO 2700 L = 1, NT
      X(L)        = 100.0 * ( XX + SRES(K,L) ) / XX + 0.00001
 2700 CONTINUE
      WRITE (6,673) K, XX, ( X(L), L = 1, NT )
 2800 CONTINUE
      RETURN
      END
```