TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
TX-81 +239-4		

4. Title and Subtitle
Rehabilitation and Maintenance System State Optimal Fund Allocation - Program I

5. Report Date
February 1981

6. Performing Organization Code

7. Author(s)
Don T. Phillips, Chiyyarath V. Shanmugham, Robert L. Lytton, and Ghasemi-Tari, Farhad

8. Performing Organization Report No.
Research Report 239-4

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 February, 1981

14. Sponsoring Agency Code

15. Supplementary Notes

Research Study Title: Pavement Rehabilitation Fund Allocation

16. Abstract

The State Optimal Fund Allocation problem is presented. The problem is modeled as a Nonlinear Knapsack Problem, and the solution methodology uses the concepts of dynamic programming techniques.

An example problem with five Highway Districts is formulated and solved using a computer program developed for that purpose. There are 11, 12, 17, 14, and 15 different budget levels for Districts 1, 2, 3, 4, and 5 respectively. The problem was solved in approximabley 0.21 seconds of execution time on the AMDAHL 470V/6 computer at Texas A&M University.

It was concluded that the proposed mathematical model and the solution algorithm is a simple, but powerful tool in solving the State Optimal Fund Allocation Problem.

17. Key Words
Pavements, Rehabilitation and Maintenance, Mathematical Model, Solution Methodology, Computer Program

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	41	

REHABILITATION AND MAINTENANCE SYSTEM STATE OPTIMAL FUND ALLOCATION - PROGRAM I

(RAMS-SOFA-1)

By

Don T. Phillips

Chiyyarath V. Shanmugham

Robert L. Lytton

Ghasemi-Tari, Farhad

Research Report Number 239-4

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
College Station, Texas

February 1981

ABSTRACT

The State Optimal Fund Allocation problem is presented. The problem is modeled as a Nonlinear Knapsack Problem, and the solution methodology uses the concepts of dynamic programming techniques.

An example problem with five Highway Districts is formulated and solved using a computer program developed for that purpose. There are 11, 12, 17, 14, and 15 different budget levels for Districts 1, 2, 3, 4, and 5 respectively. The problem was solved in approximabely 0.21 seconds of execution time on the AMDAHL 470V/6 computer at Texas A&M University.

It was concluded that the proposed mathematical model and the solution algorithm is a simple, but powerful tool in solving the State Optimal Fund Allocation Problem.

SUMMARY

This report describes in detail the State Optimal Fund Allocation (RAMS-SOFA-1) Model of the Rehabilitation And Maintenance System family of computer programs. The Texas Transportation Institute (TTI) developed this model, the solution methodology and the computer programs to assist the Texas State Department of Highways and Public Transportation to determine optimally the rehabilitation and maintenance funds to allocate to the various Highway Districts.

The RAMS-DO-1 Model, which was documented in TTI Research Report 207-3, and the associated computer programs will enable the Districts to determine the benefits obtained at various budget levels. This information is transferred to the central office of the Texas State Department of Highways and Public Transportation. Utilizing RAMS-SOFA-1, the central office can determine optimally the funds to be allocated to the various districts, so that the rehabilitation and maintenance benefits are maximized on a statewide basis.

This report contains a description of the mathematical model, solution methodology and a computer program based on dynamic programming technique. An example problem with 5 districts is solved and presented. A user's guide to the RAMS-SOFA-1 program is provided in Appendix A. Appendix A also contains a listing of input data, output (solution) of example problem and a listing of the computer program.

IMPLEMENTATION STATEMENT

RAMS-SOFA-1 is a computer program which has been developed by the Texas Transportation Institute for use by the Texas State Department of Highways and Public Transportation to determine optimally the rehabilitation and maintenance funds to be allocated the various Highway Districts in the state so that the overall rehabilitation and maintenance benefits in the state are maximized. This report describes in detail the mathematical model, the solution technique, and the documentation on the computer program.

DISCLAIMER

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

A Rehabilitation And Maintenance System (RAMS) has been developed by the Texas Transportation Institute to aid the Texas State Department of Highways and Public Transportation to make better decisions in rehabilitation and maintenance of the Texas state highway network. The System contains a set of mathematical models and a number of computer programs.

RAMS-DO-1 is one of the major programs of the RAMS family; the objective of this district optimization model is to maximize the overall effectiveness of the maintenance activities, subject to constraints such as limited resources and minimum requirements on pavement quality and service life. The mathematical model and the computer program are presented in Texas Transportation Institute Research Report 207-3 (2). The problem of determining the best rehabilitation and maintenance strategy for the various highway segments in a highway district has been analyzed by Mahoney, Ahmed and Lytton (7). Their approach is based on a mathematical model developed for optimization of the district rehabilitation and maintenance problem by Lu and Lytton (6). The District optimization problem is formulated as a 0-1 integer linear programming problem (ILP) and is solved by an algorithm developed by Ahmed (1) and Phillips. This algorithm is based on an efficient algorithm by Toyoda (12) to solve large 0-1 integer linear programming problems, but modified suitably to handle multiple choice constraints using the RAMS-DO-1 computer program. Using this program, each district can determine the optimal set of rehabilitation and maintenance strategies for the entire District network for one year. The program may also be used to estimate the benefits that will be realized for various budget levels between the lower and upper limits specified by the state. The benefits for the various budget levels from each district can be used by the central office to allocate the annual available state rehabilitation and maintenance budget to the districts.

The process of allocating funds optimally among the districts, without use of a systematic approach usually leads to an inefficient solution when different combinations of the decisions are involved. Therefore, there is a need for developing a systematic approach for determining the amount of funds to be allocated to each District, in order to obtain the maximum summation of the benefits to the entire state. This can be done through the development of an appropriate mathematical model and its computerized solution.

This report presents a mathematical model capable of selecting an optimal set of budget levels for the districts under the condition of a fixed annual state rehabilitation and maintenance budget. Two conceptual models are presented, a 0-1 integer linear programming (ILP) model and a nonlinear knapsack problem (NKP). A brief description of both the models with historical computational experience is presented in Chapter II. Based on the computational experience the most appropriate model is selected and a computer program for the selected model is presented in the Appendix. In Chapter III, a hypothetical case study is presented. The summary and conclusions are presented in Chapter IV. A brief description of the computer program, the user's guide and input and output of a sample problem are given in the Appendix.

CHAPTER II

DEVELOPMENT OF THE MATHEMATICAL MODEL

The problem of allocating rehabilitation and maintenance funds to the different districts can be modeled as an integer linear programming problem (10, 11) or a nonlinear knapsack problem (8).

(1) Integer Linear Programming Model

The ILP model is as follows:

Maximize the total benefit,

$$\text{Max} \quad \sum_{j=1}^{N} B_{ij} \; X_{ij} \tag{1}$$

Subject to:

Limitation of total available budget,

$$\sum_{j=1}^{N} \sum_{i=1}^{K_j} C_{ij} \; X_{ij} \leq C \tag{2}$$

Only one budget level must be selected in each district,

$$\sum_{i=1}^{K_j} X_{ij} = 1, \qquad (j = 1, 2, \ldots, N) \tag{3}$$

Upper bounds of available budget in each district,

$$\sum_{i=1}^{K_j} C_{ij} \; X_{in} \leq U_j \qquad (j = 1, 2, \ldots, N) \tag{4}$$

Lower bounds of available budget in each district,

$$\sum_{i=1}^{K_j} C_{ij} \; X_{in} \geq L_j \qquad (j = 1, 2, \ldots, N) \tag{5}$$

Where

N = the number of districts,

X_{ij} = 1, if budget level i selected for district j
0, otherwise

B_{ij} = the benefit obtained by using budget level i in district j,

C_{ij} = the amount of budget at level i, for district j,

C = annual rehabilitation and maintenance budget for the state,

K_j = the number of district budget levels for district j,

U_j = the upper level of available funds for district j,

and

L_j = the lower level of available funds for district j.

If we consider 25 districts (N=25) and 25 different budget levels for each district (K_j=25, j=1, 2, . . ., N), the ILP problem will have 625 major 0-1 variables and 51 inequality constraints. Even though there exists an efficient algorithm such as that of Bales (3) the achievement of an exact optimal solution is computationally expensive.

(2) Nonlinear Knapsack Model

The alternative approach to ILP is to define a Nonlinear Knapsack Model for the state optimal fund allocation problem. This approach reduces the number of decision variables to N (districts), and by employing dynamic programming techniques the exact optimal solution can be obtained at a smaller computational cost and effort than the ILP model. The model is as follows:

$$\text{Max} \sum_{j=1}^{N} B_j(d_j) \qquad (6)$$

Subject to:

$$\sum_{j=1}^{N} C_j\ (d_j) \leq C \qquad (7)$$

$$d_j \ \varepsilon\ D_j \ (d_j \text{ is contained in } D_j) \qquad (8)$$

$$D_j = \{1, 2, \ldots, K_j\}$$

$$L_j \leq C_j\ (d_j) \leq U_j \qquad (9)$$

The above problem can be solved by considering it in the form of a decision tree as shown in Figure 1. The nodes indicate the alternative budget levels (K_j) in each district (j) and the arcs represent decisions. The method is to enumerate the possible combinations of budget levels exhaustively and then to select the best combination(s) which will generate the largest total benefit, while remaining within the total state budget. There are

$$\prod_{j=1}^{N} K_j = K_1 K_2 \ldots K_N$$

possible decisions that can be made. Some of these decisions may be infeasible, i.e., it will violate the constraint (7). For a problem with 25 districts and 25 budget levels in each district, there will be

$$\prod_{j=1}^{25} (25) = 25^{25}$$

number of enumerations. This number is too large for the exhaustive enumeration technique to be considered as a viable solution method.

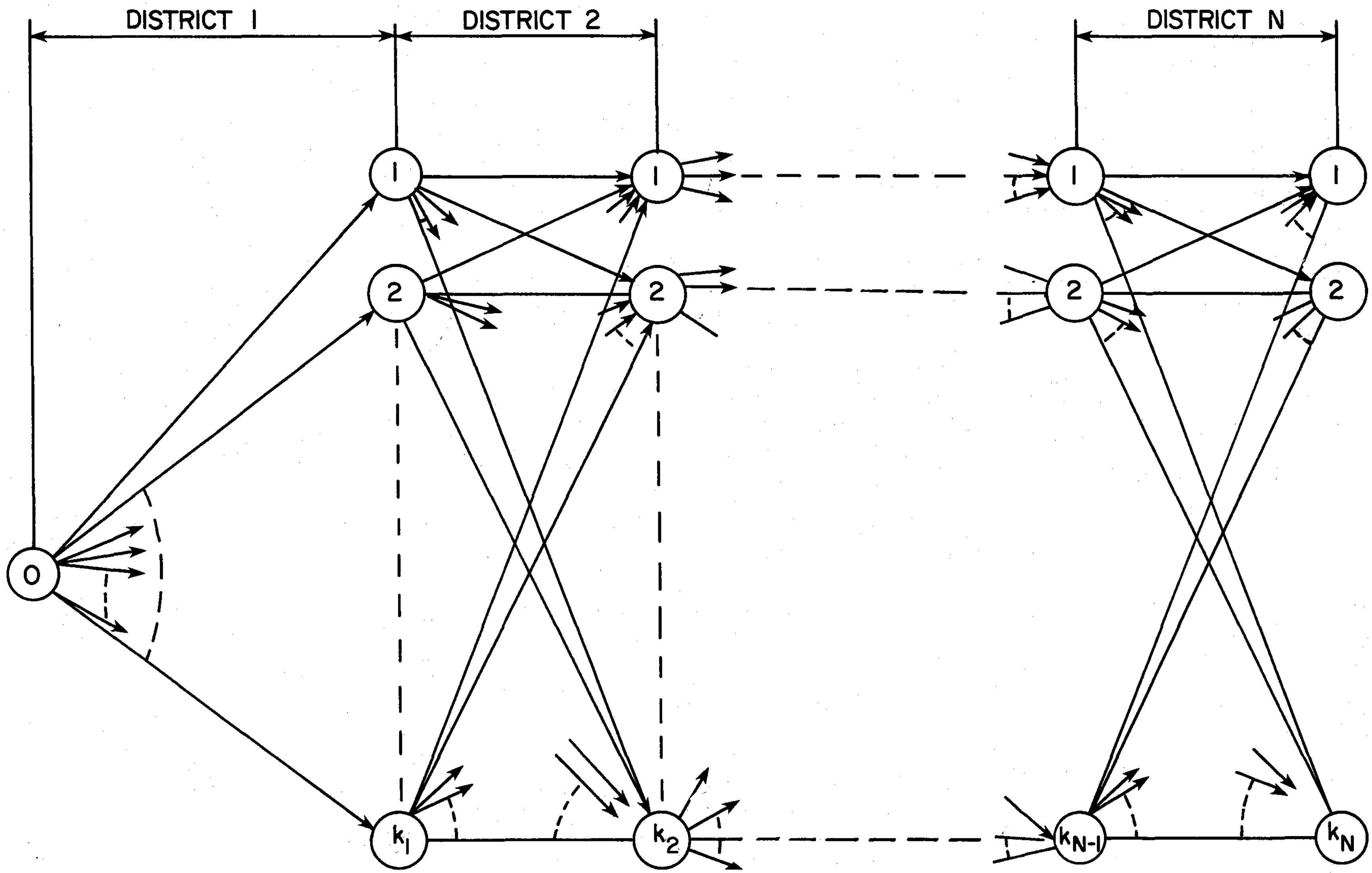

FIGURE 1. DECISION TREE

The sequential structure of the problem (Figure 1) allows the problem to be formulated and solved using the dynamic programming technique. (4, 5, 9)

(3) Optimization of NKP Model by Dynamic Programming

The sequential structure of the state optimal fund allocation problem allows the transformation of the N-decision problem into N one-decision problems. The decomposition of the large problem into N small problems (stages) is accomplished by the dynamic programming procedure (4, 5, 9). The schematic representation of the decomposition procedure is shown in Figure 2.

Each state (j) in this case is considered as District (j) in which the decision (d_j) of a different funding level d_j results in a benefit of B_j (d_j). Let S_j be the capital available for stages (Districts) j through 1 i.e.,

$$S_j = C - \sum_{k=j+1}^{N} C_j (d_j)$$

Let f_j (S_j) be the total benefit obtained for stages j through 1, for a given value of S_j. The maximum total benefit for all stages is

$$f_N^* (S_N = C) = \max_{d_1, d_2, \ldots d_N} \{f_N (S_N)\}$$

The optimization process starts at stage (District) 1. For each possible value of S_1 (budget available for District 1), the best budget level which will generate the maximum benefit is selected. Mathematically,

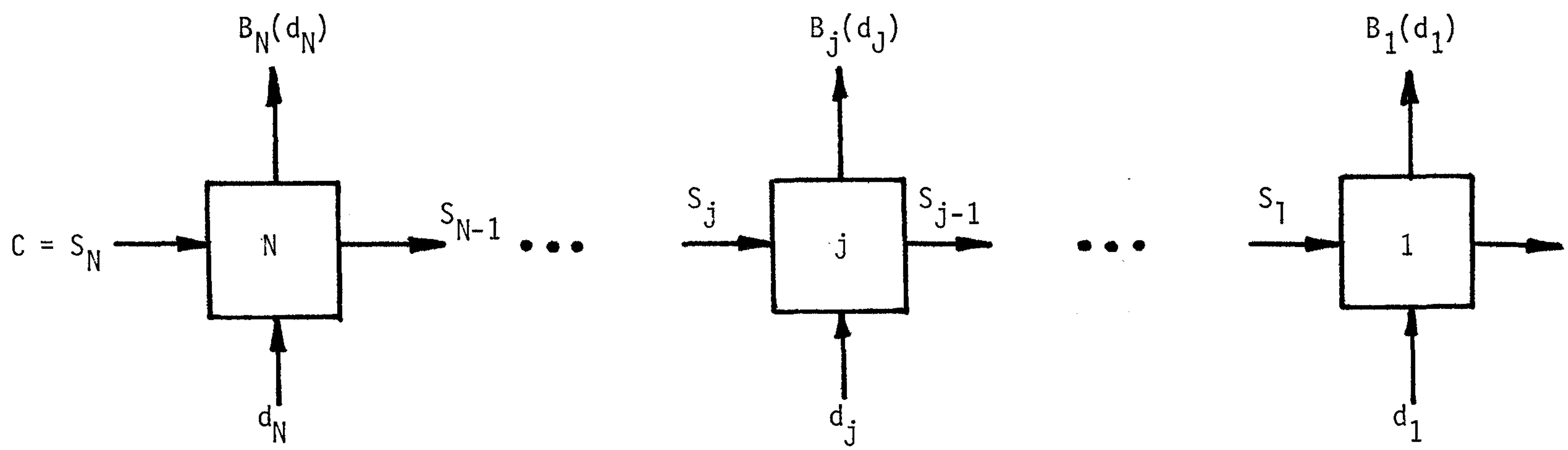

FIGURE 2 - DYNAMIC PROGRAMMING FORMULATION

$$f_1^* (S_1) = \underset{d_1}{\text{Max}} \quad \{f_1(S_1)\}$$

$$= \text{Max} \quad \{B_1(S_1, d_1)\}$$

where

$$L_1 \leq C_1(d_1) \leq \min \quad \{U_1, S_1\}$$

At stage 2, the maximum benefits for Districts 2 and 1, for a given value of budget available (S_2) for Districts 2 and 1, is computed. The benefit at stage 2 is the sum of benefits for District 2 for a particular (feasible) decision d_2, and the best benefit for District 1 for the available budget $S_2 - C_2(d_2)$. i.e.,

$$f_2^* (S_2) = \underset{d_2}{\text{Max}} \quad \{f_2(S_2)\}$$

$$= \max \quad \{B_2(S_2, d_2) + f_1^*(S_1)\}$$

where

$$S_1 = S_2 - C_2(d_2)$$

and

$$L_2 < C_2(d_2) < \min \quad \{S_2, \sum_{j=1}^{2} U_j\}$$

The process is continued for stages 3 through N and $F_N^* (S_N=C)$ is obtained as the optimal value of the benefits.

In order to find the optimal value of budgets for each district, we will start at stage N and trace back the computations to stage 1. At stage N, the capital available is $S_N=C$, and the optimum budget level for district N is $C_N(d_N)$. The budget available at stage N-1 is:

$$S_{N-1} = S_N - C_N(d_N).$$

The corresponding optimum budget level for district N-1 is selected as C_{N-1} (d_{N-1}). The process is continued until the optimum budget level at each district is obtained.

For the problem with 25 districts and 25 budget levels at each district, there are 25^{25} possible solution. By using the dynamic programming technique, the number of solutions needed to be enumerated will be reduced to 15,000. To reduce the number od solutions generated further, two tests are performaed at each stage. The first test is the feasibility test which eliminates those decisions leading to an infeasible solution. The second test is the dominancy test to eliminate those decisions which return a lower benefit at a higher cost.

CHAPTER III

AN EXAMPLE PROBLEM FOR THE STATE OPTIMAL FUND ALLOCATION PROGRAM

An example problem is presented to illustrate the mathematical model and the solution methodology. The problem considered has 5 districts and has an annual rehabilitation and maintenance budget of 52 million dollars. The central office has received information such as number of district rehabilitation and maintenance plans, the budgets needed and the benefits obtainable for each plan from the five highway district offices. The information received are shown in Tables I-V.

TABLE I

R&M PLANS, BUDGET LEVELS, AND BENEFITS FOR DISTRICT 1

R&M PLAN	BUDGET ($x10^6$ Dollars)	BENEFIT
1	4	6.800
2	5	7.900
3	6	8.900
4	7	10.700
5	8	11.900
6	9	13.000
7	10	14.800
8	11	16.000
9	12	17.600
10	13	18.500
11	14	19.900

TABLE II

R&M PLANS, BUDGET LEVELS, AND BENEFITS FOR DISTRICT 2

R&M PLAN	BUDGET ($x10^6$ Dollars)	BENEFIT
1	8	9.900
2	9	11.000
3	10	13.000
4	11	15.600
5	12	17.000
6	13	18.700
7	14	19.900
8	15	21.400
9	16	23.000
10	17	24.600
11	18	26.000
12	19	27.700

TABLE III

R&M PLANS, BUDGET LEVELS, AND BENEFITS
FOR DISTRICT 3

R&M PLAN	BENEFIT ($x10^6$ Dollars)	BENEFIT
1	6	7.000
2	7	8.900
3	8	9.900
4	9	11.000
5	10	13.000
6	11	14.800
7	12	17.000
8	13	19.100
9	14	22.000
10	15	23.900
11	16	24.900
12	17	26.080
13	18	28.008
14	19	30.760
15	20	32.198
16	21	34.988
17	22	37.089

TABLE IV

R&M PLANS, BUDGET LEVELS, AND BENEFITS
FOR DISTRICT 4

R&M PLANS	BUDGET ($x10^6$ Dollars)	BENEFIT
1	5	4.300
2	6	6.900
3	7	9.900
4	8	11.000
5	9	12.640
6	10	14.582
7	11	16.872
8	12	18.089
9	13	20.480
10	14	23.098
11	15	26.751
12	16	29.793
13	17	31.001
14	19	33.999

TABLE V

R&M PLANS, BUDGET LEVELS, AND BENEFITS
FOR DISTRICT 5

R&M PLANS	BUDGET ($\times 10^6$ Dollars)	BENEFIT
1	9	8.992
2	10	11.098
3	11	13.000
4	12	14.938
5	13	16.900
6	14	18.000
7	15	20.035
8	16	22.671
9	17	25.018
10	18	27.700
11	19	29.900
12	20	32.000
13	21	35.075
14	22	39.999
15	23	44.783

Referring to Table I, the budget level 4 million dollars in District 1 is assumed to provide 6.8 units of benefit, the budget level 5 million dollars provides 7.9 units of benefit, and so forth. Tables II-V include the same type of information for Districts 2, 3, 4, and 5 respectively. The objective of the decision problem is to select a budget level for each district which will maximize the total benefit. The total budget levels selected for districts should not exceed the total annual state rehabilitation and maintenance fund level of 52 million dollars. Besides it is required that one and only one budget level must be selected for each district.

The problem was solved using a computer program based on the dynamic programming technique. The results are summarized in Table VI. In Table VI, the budget levels (minimum, maximum, and optimum) for the various districts are listed. The benefits obtained at the optimum budget levels are shown in the last column. In District 1, the optimal level is selected as the minimum budget level resulting in a benefit of 6.8. Districts 2 and 3 are also at the minimum budget levels while in District 4, the optimum budget level is between the lower and upper limits of budgets specified. In District 5, the optimum is at the maximum budget level. The total benefit for all the 5 districts is 85.983, at a cost of 52 million dollars.

The exhaustive enumeration of all of the possible solutions to this problem will require

$$11 \times 12 \times 17 \times 14 \times 15 = 471,240$$

different combinations to be generated. The dynamic programming technique has generated at most

$$11 \times 12 + 12 \times 17 + 17 \times 14 + 14 \times 15 = 784$$

budget level combinations. The feasibility and the dominancy test would have reduced the number of combinations below 784.

TABLE VI

OPTIMAL POLICIES FOR EXAMPLE PROBLEM

DISTRICT	BUDGET LEVELS MINIMUM	MAXIMUM	OPTIMUM	BENEFIT
1	4000000	14000000	4000000	6.800
2	8000000	19000000	8000000	9.900
3	6000000	22000000	6000000	7.000
4	5000000	18000000	11000000	17.500
5	9000000	23000000	23000000	44.783
		TOTAL	52000000	85.983

CHAPTER IV

SUMMARY AND CONCLUSIONS

A mathematical model capable of selecting an optimal set of budget levels for the districts under the condition of a fixed annual state rehabilitation and maintenance budget is presented. Two models are discussed in this report. The first is a 0-1 integer linear programming model and the second is a nonlinear knapsack model.

For a state rehabilitation and maintenance problem with 25 districts and 25 different budget levels at each district, the integer linear programming model will generate 625 zero-one decision variables and 51 inequality constraints. Even though the problem is not classified as a large scale problem, an exact optimal solution using an integer programming algorithm will be expensive.

The alternative model considered is the nonlinear knapsack model. An exhaustive enumeration technique employed to solve the problem considering it as a decision tree in Figure 1, will generate

$$8,8817842 \times 10^{34}$$

possible solutions (district budget levels) and it is virtually impossible to scan through all these solutions to determine the optimal solution.

By applying the concepts of dynamic programming, the above problem can be solved by enumerating at most 15,000 solutions. The feasibility and the dominancy tests will reduce the number of solutions evaluated still further. It is shown that dynamic programming is a simple, but a very powerful tool in solving the state optimal fund allocation problem.

A computer program based on the described method is written and is presented in Appendix A. A sample problem with 5 Districts was generated with varying budget levels and was solved using the computer program. The sample problem was solved in 0.21 seconds of execution time on the AMDAHL 470V/6 computer.

A simple but a powerful procedure is presented that can be used by the central office of the State Department of Highways and Public Transportation to estimate the optimal rehabilitation and maintenance funds to be allocated to the districts. Given the various budget levels and the corresponding benefits estimated by the Districts, the state can estimate the optimal funds to be allocated to the Districts to attain the maximum benefit at the state level.

REFERENCES

1. Ahmed, N. V., "Optimization of Large Scale 0-1 Integer Linear Programming Problems with Multiple Choice Constraints," Ph.D. dissertation, Texas A&M University, College Station, Texas 1978.

2. Ahmed, N. V., Lu, D. Y., Lytton, R. L., Mahoney, J. P. and Phillips, D. T., "The Texas Rehabilitation and Maintenance District Optimization System," Texas Transportation Institute Research Report No. 207-3 (1978).

3. Balas, E., "An Additive Algorithm for Solving Linear Programs With Zero-One Variables," Operations Research, 13, pp. 517-546, 1965.

4. Beightler, C. S., Phillips, D. T. and Wilde, D. J., "Foundations of Optimization", Second Edition, Prentice-Hall, N.J. (1979).

5. Hadley, C., Nonlinear and Dynamic Programming, Addison-Wesley Publishing Company, Inc., 1964.

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

7. Mahoney, J. P., Ahmed, N. V., and Lytton, R. L., "Optimization of Pavement Rehabilitation and Maintenance Using Integer Programming," A Technical Paper for Presentation at the Transportation Research Board Meeting, January 1978.

8. Morin, T. L. and Marsten, R. E., "An Algorithm for Nonlinear Knapsack Problems," Management Science, Vol. 22, No. 10 (1976), pp. 1147-1158.

9. Phillips, D. T., Ravindran, A., and Sulberg, J. J., "Operations Research: Principles and Practice," John Wiley and Sons, Inc. New York, 1976.

10. Plane, D. R. and McMillian C., "Discrete Optimization: Integer Programming and Network Analysis for Management Decisions," Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1971.

11. Taha, H. A., "Integer Programming Theory, Applications, and Computations," Academic Press, New York, San Francisco, London, 1975.

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

APPENDIX A

REHABILITATION AND MAINTENANCE SYSTEM

STATE OPTIMAL FUND ALLOCATION

(PROGRAM I)

A.1. Program Information

A.2. Description of Input Data

A.3. Input Data for Example Problem

A.4. Output - Result of Example Problem

A.5. Program Listing

REHABILITATION AND MAINTENANCE SYSTEMS

STATE OPTIMAL FUND ALLOCATION

(PROGRAM I)

A.1: PROGRAM INFORMATION

Authors: Chiyyarath V. Shanmugham
Ghasemi-Tari, Farhad

Installation: Amdahl 470V/6
Data Processing Center
Texas A&M University

Date: Fall 1980

This is a general purpose program which solves the problem concerned with the selection of the best project in different invest segments under the restriction of the total limited budget. The solution technique used is Dynamic Programming Approach as described in Beightler et.al. (4), Hadley (5) and Phillips et.al. (9).

Program Set up

The program contains a MAIN routine and two subroutines: RETRNS and SEARCH.

MAIN routine reads in the input data and generates the table of optimal policies. RETRNS, called from MAIN, determines the cumulative returns (Benefits) for each stage of dynamic programming formulation. Subroutine SEARCH performs the backtracking operation, to determine the optimal policy decisions and the corresponding benefits, starting with the last stage.

A.2: DESCRIPTION OF INPUT DATA

The input data must be coded according to the following instructions for the proper execution of the program.

The value of an entry classified as INTEGER must be entered right justified in the designated columns. The value of a real variable must be entered within the designated columns, with a decimal period.

CARD A (One Card Only)

Column	Variable	Description	Type
1	'A'	Card Type	
6-10	NDIS	Number of Districts in the State	Integer
11-20	CAPT	Annual State Budget	Integer
21-30	UNIT	Monetary Unit to recode the large money value to a smaller unit. Recommended Values are 10,100, 1,000, 10,000, 100,000, 1,000,000, etc.	Integer

CARD B (NDIS Cards Only)

Column	Variable	Description	Type
1	'B'	Card Type	
6-10	NALT(I)	Number of R&M plans (budget levels) for District I	Integer
11-20	MIN(I)	Minimum Budget level for District I.	Integer
21-30	MAX(I)	Maximum Budget level for District I.	Integer

CARD C $\left(\sum_{i=1}^{NDIS} NALT(I)\right.$ - Number of Cards)

Column	Variable	Description	Type
1	'C'	Card Type	
6-10	K	Budget level Number	Integer
11-20	C(I,K)	Budget Required for District I at level K	Integer
21-30	B(I,K)	Benefit Obtained for District I, at budget level K	Real

```
00000000011111111112222222222333333333344444444445
12345678901234567890123456789012345678901234567890

A        5  52000000   1000000
B       11   4000000  14000000
B       12   8000000  19000000
B       17   6000000  22000000
B       14   5000000  18000000
B       15   9000000  23000000
C        1   4000000     6.800
C        2   5000000     7.900
C        3   6000000     8.900
C        4   7000000    10.700
C        5   8000000    11.900
C        6   9000000    13.000
C        7  10000000    14.800
C        8  11000000    16.000
C        9  12000000    17.600
C       10  13000000    18.500
C       11  14000000    19.900
C        1   8000000     9.900
C        2   9000000    11.000
C        3  10000000    13.000
C        4  11000000    15.600
C        5  12000000    17.000
C        6  13000000    18.700
C        7  14000000    19.900
C        8  15000000    21.400
C        9  16000000    23.000
C       10  17000000    24.600
C       11  18000000    26.000
C       12  19000000    27.700
C        1   6000000     7.000
C        2   7000000     8.900
C        3   8000000     9.900
C        4   9000000    11.000
C        5  10000000    13.000
C        6  11000000    14.800
C        7  12000000    17.000
C        8  13000000    19.100
C        9  14000000    22.000
C       10  15000000    23.900
C       11  16000000    24.900
C       12  17000000    26.080
C       13  18000000    28.008
C       14  19000000    30.760
C       15  20000000    32.189
C       16  21000000    34.998
C       17  22000000    37.089
C        1   5000000     4.330
C        2   6000000     6.900
C        3   7000000     9.900
C        4   8000000    11.000
```

A.3: INPUT DATA FOR EXAMPLE PROBLEM

```
00000000011111111112222222222333333333344444444445
12345678901234567890123456789012345678901234567890

C          5   9000000      12.640
C          6  10000000      14.582
C          7  11000000      16.872
C          8  12000000      18.089
C          9  13000000      20.480
C         10  14000000      23.098
C         11  15000000      26.751
C         12  16000000      29.793
C         13  17000000      31.001
C         14  18000000      33.999
C          1   9000000       8.992
C          2  10000000      11.098
C          3  11000000      13.000
C          4  12000000      14.935
C          5  13000000      16.900
C          6  14000000      18.000
C          7  15000000      20.035
C          8  16000000      22.671
C          9  17000000      25.018
C         10  18000000      27.700
C         11  19000000      29.900
C         12  20000000      32.000
C         13  21000000      35.075
C         14  22000000      39.999
C         15  23000000      44.783
```

A.4: OUTPUT - OPTIMAL POLICY TABLE FOR EXAMPLE PROBLEM

REHABILITATION AND MAINTENANCE SYSTEM

(STATE OPTIMAL FUND ALLOCATION)

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

DISTRICT	BUDGET LEVELS MINIMUM	MAXIMUM	OPTIMUM	BENEFIT
1	4000000	14000000	4000000	6.800
2	8000000	19000000	8000000	9.900
3	6000000	22000000	6000000	7.000
4	5000000	18000000	11000000	17.500
5	9000000	23000000	23000000	44.783
		TOTAL	52000000	85.983

A.5: LISTING OF COMPUTER PROGRAM

```
C
C     ******************************************************************
C
C      REHABILITATION AND MAINTENANCE SYSTEM
C
C      STATE OPTIMAL FUND ALLOCATION - PROGRAM I
C
C      TEXAS TRANSPORTATION INSTITUTE
C      TEXAS A&M UNIVERSITY
C      COLLEGE STATION, TEXAS 77843
C
C      AUTHORS              : C. V. SHANMUGHAM
C                             GHASEMI-TARI, FARHAD
C
C      INSTALLATION         : AMDAHL 470V/6
C                             DATA PROCESSING CENTER
C                             TEXAS A&M UNIVERSITY
C
C      DATE                 : FALL 1980
C
C     ******************************************************************
C
      COMMON /AA/ NDIS, CAPT, INFN, KK, LL
      COMMON /BB/ MIN(30), ILOW(30), NALT(30)
      COMMON /CC/ MAX(30), IHIG(30), ID(30), FS(30)
      COMMON /DD/ C(30,30), B(30,30), BI(30,30)
      COMMON /EE/ D(30,600), F(30,600)
      INTEGER     C, CAPT, D, TOTC, UNIT
  500 FORMAT ( 5X, I5, 2I10 )
  510 FORMAT ( 5X, I5, I10, F10.0 )
  600 FORMAT ( 30X, 37HREHABILITATION AND MAINTENANCE SYSTEM, ///,
     1         33X, 31H(STATE OPTIMAL FUND ALLOCATION), //,
     2         34X, 30HTEXAS TRANSPORTATION INSTITUTE, /,
     3         39X, 20HTEXAS A&M UNIVERSITY, /,
     4         35X, 28HCOLLEGE STATION, TEXAS 77843, // )
  610 FORMAT ( 22X, 52(1H-), /, 22X, 52(1H-), 22X, 8HDISTRICT,
     1         12X, 13HBUDGET LEVELS, 12X, 7HBENEFIT, /,
     2         34X, 29HMINIMUM    MAXIMUM    OPTIMUM, /,
     3         22X, 52(1H-) )
  620 FORMAT ( /, 25X, I2, 3X, 3I11, F11.3 )
  630 FORMAT ( /, 22X, 52(1H-), //, 47X, 7HTOTAL, I11, F11.3,
     1         //, 22X, 52(1H-), /, 22X, 52(1H-) )
  666 FORMAT ( 1H1, /// )
      KK         = 0
      LL         = 0
      INFN       = -999999
C
C     READ INPUT DATA
C
C     NDIS       : NUMBER OF DISTRICTS IN STATE
C     CAPT       : ANNUAL BUDGET AVAILABLE FOR THE YEAR
C     UNIT       : ($) - MEASURE OF BUDGET
C
C     NALT(I)    : NUMBER OF R&M PLANS FOR DISTRICT I
C     MIN (I)    : MINIMUM BUDGET FOR DISTRICT I
```

```
C      MAX (I)    : MAXIMUM BUDGET FOR DISTRICT I
C
C      KALT       : R&M PLAN NUMBER FOR DISTRICT I
C      C(I,J)     : BUDGET REQUIRED FOR PLAN J IN DISTRICT I
C      B(I,J)     : BENEFIT OBTAINED FOR PLAN J IN DISTRICT I
C
       READ  (5,500) NDIS, CAPT, UNIT
       DO 1100 I = 1, NDIS
       READ  (5,500) NALT(I), MIN(I), MAX(I)
 1100  CONTINUE
       DO 1300 I = 1, NDIS
       NOAL        = NALT(I)
       DO 1200 J = 1, NOAL
       READ  (5,510) KALT, C(I,J), B(I,J)
 1200  CONTINUE
 1300  CONTINUE
       DO 1500 I = 1, NDIS
       MIN(I)      = MIN(I) / UNIT
       MAX(I)      = MAX(I) / UNIT
       NOAL        = NALT(I)
       DO 1400 J = 1, NOAL
       C(I,J)      = C(I,J) / UNIT
 1400  CONTINUE
       C(I,NOAL+1)=C(I,NOAL)
 1500  CONTINUE
       CAPT        = CAPT / UNIT
C
C      FIND CUMULATIVE RETURNS (BENEFITS) FOR ALL DISTRICTS
C
       DO 1600 I = 1, NDIS
       CALL RETRNS(I)
 1600  CONTINUE
C
C      BACKTRACKING OPERATION
C
C      FIND THE OPTIMAL POLICY FOR EACH DISTRICT AND THE
C      CORRESPONDING BENEFIT
C
       CALL SEARCH
C
C      GENERATE THE TABLES OF OPTIMAL POLICIES
C
       WRITE (6,666)
       WRITE (6,600)
       WRITE (6,610)
       TOTC        = 0
       DO 2000 I = 1, NDIS
       MIN(I)      = MIN(I) * UNIT
       MAX(I)      = MAX(I) * UNIT
       J           = ID(I)
       C(I,J)      = C(I,J) * UNIT
       TOTC        = TOTC + C(I,J)
       IF ( I .EQ.1 ) GO TO 1700
       IF ( I .EQ. 1 ) GO TO 1700
       II          = I - 1
```

```
      RIGH       = FS(II)
      BI(I,J)    = FS(I) - RIGH
      GO TO 1800
 1700 BI(I,J)    = FS(I)
 1800 CONTINUE
      WRITE (6,620) I, MIN(I), MAX(I), C(I,J), BI(I,J)
 2000 CONTINUE
      TOTR       = FS(NDIS)
      WRITE (6,630) TOTC, TOTR
      WRITE (6,666)
      STOP
      END




      SUBROUTINE RETRNS(I)
C
C     SUBROUTINE RETRNS DETERMINES THE CUMULATIVE RETURNS (BENEFITS)
C     FOR EACH STAGE (DISTRICT) OF THE DYNAMIC PROGRAMMING
C     FORMULATION
C
C     D(I,K) : OPTIMAL DECISION FOR STAGE (DISTRICT) I FOR THE
C              STATE VARIABLE VALUE K
C     F(I,K) : CUMULATIVE RETURNS (BENEFIT) FOR STAGE (DISTRICT) I
C              FOR STATE VARIABLE VALUE K
C
      COMMON /AA/ NDIS, CAPT, INFN, KK, LL
      COMMON /BB/ MIN(30), ILOW(30), NALT(30)
      COMMON /CC/ MAX(30), IHIG(30), ID(30), FS(30)
      COMMON /DD/ C(30,30), B(30,30), BI(30,30)
      COMMON /EE/ D(30,600), F(30,600)
      INTEGER      C, CAPT, D, TOTC, UNIT
      I1         = I - 1
      NOAL       = NALT(I)
      IF ( I .GT. 1 ) GO TO 1300
C
C     COMPUTATIONS FOR STAGE 1
C
      KK         = MAX(I)
      LL         = MIN(I)
      K          = LL
      DO 1200 J = 1, NOAL
 1100 IF ( K .GE. C(I,J+1) ) GO TO 1200
      F(I,K)     = B(I,J)
      D(I,K)     = J
      K          = K + 1
      GO TO 1100
 1200 CONTINUE
      F(I,KK)    = B(I,NOAL)
      D(I,KK)    = NOAL
      GO TO 2200
 1300 IF ( I .EQ. NDIS ) GO TO 1750
C
C     COMPUTATIONS FOR STAGES 2 THROUGH NDIS-1
```

```
C
      KK          = KK + MAX(I)
      LL          = LL + MIN(I)
      IF ( KK .GT. CAPT ) KK = CAPT
      DO 1700 K = LL, KK
      F(I,K)      = INFN
      D(I,K)      = -1
      DO 1650 J = 1, NOAL
      IK          = K - C(I,J)
      MAXCH       = 0
      MINCH       = 0
      DO 1400 L = 1, I1
      MAXCH       = MAXCH + MAX(L)
      MINCH       = MINCH + MIN(L)
 1400 CONTINUE
      IF ( MAXCH .GT. CAPT ) MAXCH = CAPT
      IF ( IK .LT. MINCH ) GO TO 1700
      IF ( IK .GT. MAXCH ) GO TO 1500
      FSTR        = B( I,J) + F( I1,IK)
      IF ( FSTR .LT. F(I,K) ) GO TO 1600
      F(I,K)      = FSTR
      GO TO 1600
 1500 F(I,K)      = B( I,J) + F( I1,MAXCH)
 1600 D(I,K)      = J
 1650 CONTINUE
 1700 CONTINUE
      GO TO 2200
C
C     COMPUTATIONS FOR LAST (NDIS) STAGE
C
 1750 DO 2150 J = 1, NOAL
      K           = CAPT
      F(I,K)      = INFN
      D(I,K)      = -1
      IK          = K - C(I,J)
      MAXCH       = 0
      MINCH       = 0
      DO 1900 L = 1, I1
      MAXCH       = MAXCH + MAX(L)
      MINCH       = MINCH + MIN(L)
 1900 CONTINUE
      IF ( MAXCH .GT. CAPT ) MAXCH = CAPT
      IF ( IK .LT. MINCH ) GO TO 2100
      IF ( IK .GT. MAXCH ) GO TO 2000
      FSTR        = B( I,J) + F( I1,IK)
      IF ( FSTR .LT. F(I,K) ) GO TO 2100
      F(I,K)      = FSTR
      GO TO 2100
 2000 F(I,K)      = B( I,J) + F( I1,MAXCH)
 2100 D(I,K)      = J
 2150 CONTINUE
 2200 ILOW(I)     = LL
      IHIG(I)     = KK
      RETURN
      END
```

```
      SUBROUTINE SEARCH
C
C     SUBROUTINE SEARCH PERFORMS THE BACKTRACKING OPERATION
C     IN THE DYNAMIC PROGRAMMING METHODLOGY, TO DETERMINE
C     THE OPTIMAL POLICY DECISIONS AND THE CORRESPONDING
C     BENEFITS, STARTING WITH THE LAST STAGE.
C
C     FS(I)      : CUMULATIVE RETURN (BENEFIT) FOR STAGES
C                  1 THROUGH I
C     ID(I)      : OPTIMAL POLICY FOR STAGE I
C
      COMMON /AA/ NDIS, CAPT, INFN, KK, LL
      COMMON /BB/ MIN(30), ILOW(30), NALT(30)
      COMMON /CC/ MAX(30), IHIG(30), ID(30), FS(30)
      COMMON /DD/ C(30,30), B(30,30), BI(30,30)
      COMMON /EE/ D(30,600), F(30,600)
      INTEGER     C, CAPT, D, TOTC, UNIT
      I         = NDIS
      FS(I)     = F(I,CAPT)
      J         = D(I,CAPT)
      ID(I)     = J
      I1        = I - 1
      IS        = CAPT - C(I,J)
      DO 1200 I = 1, I1
      II          = NDIS - I
      K           = IS
      IF ( IS .GT. IHIG(II) ) K = IHIG(II)
      FS(II)      = F(II,K)
      J           = D(II,K)
      ID(II)      = J
      IS          = IS - C(II,J)
 1200 CONTINUE
      RETURN
      END
```